AGENCY FOR INTERNATIONAL DEVELOPMENT
WASHINGTON, D. C. 20523
BIBLIOGRAPHIC INPUT SHEET

FOR AID USE ONLY

1. SUBJECT CLASSIFICATION	A. PRIMARY Food production and nutrition	AE70-0000-G704
	B. SECONDARY Distribution and marketing--Korea	

2. TITLE AND SUBTITLE
Demand-price-trade model of MASM3: technical documentation

3. AUTHOR(S)
Abkin, M. H.

4. DOCUMENT DATE	5. NUMBER OF PAGES	6. ARC NUMBER
1977	222p.	ARC KS380.141.A151

7. REFERENCE ORGANIZATION NAME AND ADDRESS
Mich.State

8. SUPPLEMENTARY NOTES (Sponsoring Organization, Publishers, Availability)
(Supported under contracts AID/csd-2975 and AID/ta-C-1322)
(In KASS special report 17)

9. ABSTRACT

10. CONTROL NUMBER
PN-AAG-559

11. PRICE OF DOCUMENT

12. DESCRIPTORS
Models
Sector analysis
Agricultural products
Computer programs
Simulation
Demand
Prices
International trade
Korea Rep.

13. PROJECT NUMBER
931053600: 931024400

14. CONTRACT NUMBER
AID/csd-2975 Res.

15. TYPE OF DOCUMENT

AID 590-1 (4-74)

KOREAN AGRICULTURAL SECTOR STUDY 23

Special Report 17

DEMAND-PRICE-TRADE MODEL OF KASM3: TECHNICAL DOCUMENTATION

Michael H. Abkin

National Agricultural Economics
Research Institute
Ministry of
Agriculture and Fisheries
Seoul, Korea

Department of
Agricultural Economics
Michigan State
University
East Lansing, Michigan

DEMAND-PRICE-TRADE MODEL OF KASM3:

TECHNICAL DOCUMENTATION

Michael H. Abkin

KOREAN AGRICULTURAL SECTOR SIMULATION PROJECTS

KASS Special Report 17

National Agricultural Economics Research Institute
Ministry of Agriculture and Fisheries
Seoul, Korea

Department of Agricultural Economics
Michigan State University
East Lansing, Michigan

June, 1977

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

FOREWORD

The Korean Agricultural Sector Model (KASM3) is a system of simulation models designed for use in the analysis of the agricultural sector and subsectors of the Republic of Korea. The primary objective of this system of models is to serve as an aid to national-level, public decision makers in the formulation and analysis of agricultural and rural development policies, programs and projects. However, KASM3 can also be useful to researchers and others interested in Korean agricultural sector analysis.

While the models have been developed specifically for Korea, the basic structures are general enough to have wider applicability, singly or in combination, in other countries as well. For this reason, and to emphasize the modularity of KASM3 and enhance the utility of its technical documentation to assist in implementing one or more component models in Korea or elsewhere, KASM3 is documented as a series of KASS Special Reports as follows:

Special Report 13:	Korean Agricultural Sector Model, Version KASM3: System Technical Documentation
Special Report 14:	Population and Migration Model of KASM3: Technical Documentation
Special Report 15:	Crop Technology Change Model of KASM3: Technical Documentation
Special Report 16:	Farm Resource Allocation and Production Model of KASM3: Technical Documentation
Special Report 17:	Demand-Price-Trade Model of KASM3: Technical Documentation
Special Report 18:	National Economy Model of KASM3: Technical Documentation

Work on the Korean Agricultural Sector Model, Version KASM3, has been supported by the U.S. Agency for International Development under contracts AID/csd-2975 and AID/ta-C-1322, and by the Ministry of Agriculture and Fisheries, Republic of Korea.

George E. Rossmiller
Director
Agriculture Sector Analysis and Simulation Projects
Michigan State University
East Lansing, Michigan

Sung Ho Kim
Director
National Agricultural Economics Research Institute
Ministry of Agriculture and Fisheries
Seoul, Korea

CHAPTER 1

INTRODUCTION AND SUMMARY

As a whole, the Korean agricultural sector system of models (KASM) is designed to address a wide range of policy problems facing public agricultural decision makers in Korea. Each model of the system represents a major component of integrated agricultural sector analysis -- population, crop technology change, farm resource allocation, demand and national economy. This report documents the demand-price-foreign trade model (DEMAND) of the KASM system.

Mathematical Model

Chapter 2 documents in detail the mathematical model of DEMAND -- its rationale, its linkages with other KASM models and its internal structure.

A key component of agricultural sector analysis in the context of a controlled market economy such as Korea's is projections of agricultural commodity prices and demands. Farm prices are important determinants of farm-level investment and production decisions, and farm and consumer prices are measures of rural and urban welfare. Finally, per capita consumption measures nutritional well-being, and total consumption levels when compared with domestic production, indicate likely import requirements and export potentials. DEMAND is designed to make these demand-price-trade projections within the Korean policy context for each of the agricultural commodity groups considered in KASM.

Within the KASM system, DEMAND is closely linked with the production models, from which DEMAND receives domestic supply and farm income and for which DEMAND determines producer prices. Also, DEMAND uses nonfarm income and the nonfood price index from the national economy model (NECON) and computes food and nonfood consumption, agricultural trade and food prices for use in NECON. Finally, farm and nonfarm population levels are determined in the population and migration model.

In addition to these inputs to DEMAND from other KASM models, also needed are projections of world food prices and a set of policy inputs. Food price and consumption policies in Korea have multiple and sometimes conflicting operational objectives. These include: 1) increasing farm income, 2) increasing domestic production, 3) reducing food imports and, hence, foreign exchange requirements, and 4) reducing inflationary pressures.

Since DEMAND is intended for use in policy analysis, several relevant price and quantity control policy instruments are built into DEMAND. For each commodity:

1. the consumption level over time may be specified to observe the consequent impacts on imports/exports and prices;

2. imports/exports over time may be specified to investigate the implications for prices and consumption;

3. the consumer price or bounds on consumer price movements over time may be specified to observe the resulting nonfarm consumption and imports/exports;

4. producer price projections may be specified to investigate the effects on farm consumption and, when linked to a production model, production; and

5. carryover stock levels may be regulated to reflect the building of government reserves and to investigate the impacts on consumption, prices and imports/exports.

DEMAND is composed of three components: farm demand, price and nonfarm demand, and outputs of trade accounting and DEMAND performance measures.

The farm demand component projects per capita farm consumption of each of the food commodities of KASM based on lagged producer prices and income using linearized Cobb-Douglas consumption functions. Each commodity's demand equation is solved independently, and aggregate nonfood consumption is computed as a residual. Lagged producer prices are used in the consumption functions instead of concurrent consumer prices, in accordance with the implicit assumption that farm food consumption is basically subsistence.

The price and nonfarm demand component of DEMAND solves, in each time period, a simultaneous linear system of equations for 1) per capita nonfarm consumption of each of the food commodities of KASM, 2) per capita nonfarm consumption expenditures for the aggregate nonfood commodity, and 3) consumer prices for each food commodity. For each commodity, either the consumer price is specified exogenously and DEMAND solves for consumption, or consumption is predetermined by domestic supply and policy-specified imports/exports and DEMAND solves for the market-clearing consumer price. Bounds on the resulting consumer price may also be specified, in which case the solution algorithm iterates until all such bounds are satisfied. The solution obtained is also consistent with a total consumption expenditure constraint. This component also computes producer prices, either as specified by policy or related to consumer prices with constant marketing margins.

Both farm and nonfarm demand depend on price and income elasticity values as well as on price and income. In order to maintain a reasonable and mutually consistent set of direct-price and cross-price elasticities over time, DEMAND computes cross-price elasticities of each time period as functions of relative consumption levels and the direct-price elasticities. Appendix D discusses the theoretical and mathematical derivation of this computation.

Income elasticities also vary over time in the model in order to keep consumption 1) within reasonable limits as income increases in the long run and 2) consistent with reasonable expectations of long-run nutrition levels. Consumption targets are specified for each commodity, and, as actual consumption of that commodity approaches the target, the income elasticity goes to zero.

Finally, based on consumptions and prices determined in the other DEMAND components, foreign trade accounts and a number of other variables are computed which can be used as measures of how the DEMAND system responds to alternative policy assumptions. These include nutrition levels, self-sufficiency, price indices and consumption expenditure accounts.

Data Base and Computer Program

The data for the demand-price-trade model, as for any model, are classified into four categories: 1) initial conditions; 2) system parameters, i.e., constants which determine the strength of relationships among variables; 3) policy parameters; and 4) input variables. DEMAND's requirements and sources for these types of data are discussed in Appendix E. A listing of the current data file appears in Appendix B.

The computer program of DEMAND is also listed in Appendix B and a sample output in Appendix C. Variable definitions appear in Appendix A. Chapter 5 describes the structure of the computer program, its requirements in terms of machine core storage and execution time, and how to implement it. Also included is a sample worksheet to simplify the specification of policy inputs to DEMAND.

Model Testing and Areas for Further Research

Three types of tests have been conducted on DEMAND: internal consistency checks, time-series tracking tests and sensitivity analyses. Results of these tests are presented in Chapter 3, and their implications for further research and model development are discussed in Chapter 4.

The internal consistency checks revealed problems primarily in two areas. First, the linearization of the basic Cobb-Douglas demand-price model tends to be unstable when inverted to solve for price as a function of supply, particularly for price-inelastic commodities. It is not uncommon for market-clearing prices determined in this way to range from unreasonably large to even negative values. Therefore, it is advisable to use DEMAND only with bounded or exogenously specified consumer prices. An important area for further research and model development, then, would be to consider alternatives to the linearized Cobb-Douglas consumption functions. The problem is attributed to this functional form because 1) the linearization can be very inaccurate for relatively large changes, 2) it may be invalid to use the inverse of demand elasticities as price flexibilities, and 3) elasticities do not allow for asymmetric consumption responses depending on the direction, magnitude and duration of price or income changes.

The second internal consistency problem in DEMAND is that the all-commodities consumer price index CPI does not remain 1.0 over time. All prices computed in DEMAND are real prices, with CPI as the implicit deflator. The aggregate index of all real prices, then, should be 1.0, essentially because it is deflated by itself. However, in DEMAND's standard run CPI drifts upwards from 1.0 to a value of 1.129 by 1985. It is suggested that an additional constraint equation be added to the simultaneous equations system which would determine the nonfood price index such that the aggregate CPI has unit value.

Time-series tracking tests compare DEMAND's projections of per capita consumption of each commodity and of aggregations of grains, meats and fruits/vegetables with the historical values of those variables for the period 1971-1975. Normalized sums of squared errors are used as measures of goodness of fit. For most commodities, particularly the major ones and the three aggregations, DEMAND tracks consumption reasonably well. In general, trends are captured much better than oscillations about the trends. Overall, the total of the 19 commodity-specific goodness-of-fit measures is 3.8763, with individual ones ranging from a low of .0143 for rice to a high of 1.1174 for industrial crops. For reference purposes in evaluating this fit, zero would be a perfect fit, while 87.3313 would be the result if the model projected only zeroes for these series.

Finally, sensitivity analyses can be very useful in identifying those data requirements of the model which should receive relatively higher priority in allocating scarce resources for data collection and analysis. They can also indicate the policy instruments which are likely to be relatively more effective in influencing the system's behavior.

On the whole, elasticities do not appear too sensitive except for the commodities most directly affected. Therefore, it would not seem to be worthwhile to spend much more effort re-estimating demand elasticities, particularly if alternative consumption models are to be considered. The nonfarm average propensity to consume is by far the most sensitive model parameter of those tested. It may be desirable, therefore, to consider projecting this as a time-varying parameter rather than keeping it as a constant. Finally, the aggregate response to changes in the price bounds is relatively small, the sensitivity of these parameters primarily being confined to the commodities affected most directly.

CHAPTER 2

TECHNICAL DESCRIPTION

This chapter, divided in two sections, is a detailed technical description of the demand-price-foreign trade model (DEMAND) of the Korean Agricultural Sector system of Models (KASM). The first section, system definition, describes DEMAND from an external point of view. That is, we will look at the rationale for DEMAND (why it exists), its linkages with other models of KASM, and its inputs and outputs. In the second section, model description, we will go into the internal structure of DEMAND in some detail, including the major assumptions and equations and the policy options built into the model. The next chapter of this documentation discusses the results of time-series tracking and sensitivity tests of DEMAND.

System Definition

Rationale

A key component of agricultural sector analysis in the context of a controlled market economy such as Korea's is projections of agricultural commodity prices and demands. Farm prices, for instance, are important determinants of farm-level investment and production decisions. In addition, as essential elements of farm income, they are measures of rural social welfare insofar as they contribute to the ability of farm households to purchase such nonagricultural goods and services as bicycles, radios, electricity, health care, education, etc. Furthermore, when compared with prices on the world market, they are indications of Korea's relative competitive position in the production of the various agricultural commodities. Therefore, agricultural sector policy analysis should provide information on the likely response of farm prices to various supply-demand conditions resulting from alternative production and consumption policies.

Similarly, consumer food prices affect nonfarm social welfare through their impact on the proportion of income spent of food consumption. In addition, they can indicate inflationary pressures arising from demand-pull factors.

Finally, for both farm and nonfarm sectors, levels of per capita consumption translate into measures of nutritional well-being, and changes in the relative consumption of various groups of commodities (e.g., grains compared with meats and vegetables) are an indicator of the progress of economic and social development. Furthermore, total consumption demand, when compared with domestic production, provides information on likely import requirements and export potentials for the various agricultural commodities.

DEMAND is designed to make these demand-price-trade projections within the Korean policy context for each of the 19 agricultural commodity groups considered in KASM (Table 2.1). However, its structure is general enough to have wider applicability in other countries. For sector analyses, DEMAND can be used as a component model of the Korean Agricultural Sector system of Models (KASM), which includes models of agricultural production, technology changes, population and the national economy. It may also be used independently of KASM for consumption analyses which do not require all the information provided by the total sector model.

KASM Interaction

Figure 2.1 shows the linkages between DEMAND and other models of the KASM system. In this figure, the technology change and resource allocation models are shown together as "production."

Farm and nonfarm population projections, provided by the population and migration model (POPMIG) [1], are used by DEMAND to determine total consumption demand for food commodities. There is not now any feedback from DEMAND to POPMIG. However, DEMAND does provide information on nutrition levels, and at some later date POPMIG could be extended to, for example, make birth and death rates functions of nutrition.

The farm income accounting component [4] combines producer prices from DEMAND with output and production costs from the production models to project farm income. Per capita farm income is lagged and fed back to DEMAND for use in the consumption functions.

The production models are driven, in part, by lagged producer prices. The production functions in the technology change model (CHANGE) [3] use producer prices to partially determine input application rates and, hence, yields. In addition, producer prices are used in the diffusion of innovations component of CHANGE. Producer prices are used in the resource allocation and production model (RAP) [4] for crop land allocation and livestock production decisions. In return, information on domestic supply is used by DEMAND in determining prices and imports/exports.

TABLE 2.1

Commodity Groups of the
Korean Agricultural Sector System of Models

Crops

1. Rice
2. Barley
3. Wheat
4. Other grains
5. Fruit
6. Pulses
7. Vegetables
8. Potatoes
9. Tobacco
10. Forage
11. Mulberry (silk)
12. Industrial crops

Livestock

13. Beef
14. Milk
15. Pork
16. Chicken
17. Eggs

Fisheries

18. Fish

Residual

19. Agricultural residual

Nonagricultural

20. Nonagricultural aggregate

The national economy model (NECON) links the agricultural sector with the rest of the Korean economy [5]. The aggregate price index of nonfood consumer goods and services, used in DEMAND as a price deflator, is determined in NECON. Also, nonfarm income, resulting from performance of the nonagricultural sectors in NECON, is lagged for use in DEMAND as a driver in the nonfarm consumption functions and as a constraint on total nonfarm per capita consumption expenditures. In return, the value of agricultural exports and food consumption expenditures used in NECON are components of final demand for agriculture's output. Nonfood consumption is disaggregated in NECON by sector. Agricultural imports are used in NECON to compute import coefficients for agriculture, and, finally, the food price index from DEMAND serves as the price index for the agricultural sector in NECON.

If DEMAND is to be run independently of the rest of KASM, then the inputs described above and shown in Figure 2.1 must be projected and supplied by the user. Next, we discuss the policy and other inputs and outputs of DEMAND and the behavioral parameters which, together with the structure of DEMAND, determine the model's response to its inputs.

<u>Other Inputs and Outputs</u>

Figure 2.2 is a simple input-output ("black-box") diagram of DEMAND. In this figure, inputs and outputs from and to other models of KASM, discussed above, are listed with other variables needed by DEMAND and computed by it.

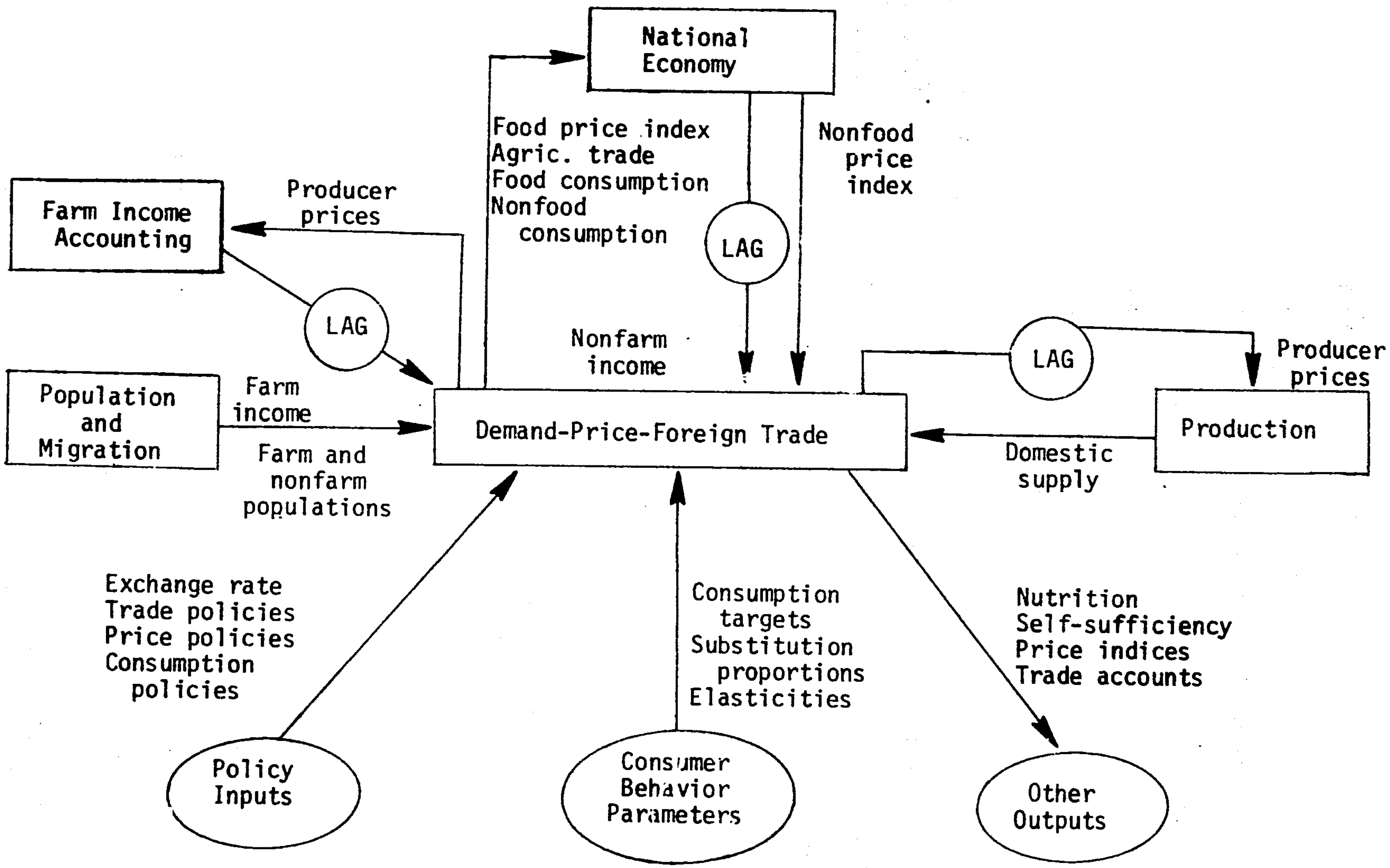

Figure 2.1

Major Linkages between the Demand-Price-Foreign Trade Model and Other Components and Models of the Korean Agricultural Sector System of Models

Consumer Behavior Parameters

Consumption targets
Initial income elasticities
Own-price elasticities
Substitution proportions

KASM and Other Inputs

Farm population
Nonfarm population
Farm income
Nonfarm income
Domestic supply
Nonfood price index
World food prices

DEMAND

Outputs

Farm food consumption
Nonfarm food consumption
Producer prices
Consumer prices
Agricultural imports/exports
Nonfarm consumption expenditures
Self-sufficiency
Food price indices

Policy Inputs

Specification of consumption levels
Specification of imports/exports
Specification of consumer prices or bounds
on consumer prices
Specification of producer prices
Regulation of commodity stock levels
Foreign exchange rate

Figure 2.2

Input-Output Diagram of the
Demand-Price-Foreign Trade Model

An additional variable needed by DEMAND and not supplied by other KASM models is world food prices. Since these are not computed in KASM, the user must project them independently[1] and supply the data to DEMAND. Projections are needed on import prices and export prices for each of the 19 agricultural commodity groups considered in KASM (Table 2.1). Import and export prices constrain domestic prices in DEMAND (depending on policy specifications--see below) and are used to compute agricultural trade balances.

In addition to variables computed for use by other models of KASM, DEMAND outputs information on nutrition, food price indices, self-sufficiency requirements and self-sufficiency levels. Per capita daily protein and calorie intake is computed for farm and nonfarm populations and for the national average. Total intake of each nutrient by each group is further broken down by plant and animal sources.

Food consumer price indices are projected by DEMAND for all foods, for rice, for other grains and for meats. For eocnomic and national security reasons, self-sufficiency, particularly in the major food grains, is a major objective of Korean policymakers. Therefore, the relative impact of various policy options on self-sufficiency levels is essential information for policy analysis in Korea. Thus, DEMAND calculates for each commodity group domestic production requirements and, for crop commodities, land allocation requirements for 10% self-sufficiency. These calculations are based on consumption demand, crop

---

[1]See Appendix E for a discussion of how these projections have been made for the current data in DEMAND. New projections should be made from time to time as new information warrants.

yields and assumed loss and seeding rates. If the production models of KASM are operating with DEMAND to supply information on domestic supply, then this is compared with total demand to determine actual self-sufficiency levels.

Consumer Behavior Parameters

Although behavioral parameters appear in Figure 2.2 to be inputs to DEMAND, they are instead part of the system itself. Along with the initial condition data,[2] they combine with the generalized structure of DEMAND described in the next section to form a model of Korean food demand and price. This model, composed of structure and data, responds to agricultural sector and policy inputs to generate output projections as indicated in Figure 2.2. There are four sets of behavioral parameters: consumption targets, initial income elasticities, own-price elasticities, and substitution proportions. Each of these sets is disaggregated by farm and nonfarm consumers and by the 20 commodities shown in Table 2.1.

Consumption targets in DEMAND serve to contrain long-run consumption (roughly the year 2000 and beyond) from increasing without bound as income increases and to keep consumption consistent with reasonable expectations of long-run calorie and protein intake. The model accomplishes this by assuming the income elasticity of each commodity goes to zero over time as consumption approaches the target. For stability, targets, initial consumption levels and initial income

[2]See the data documentation of the demand-price-foriegn trade model in Appendix E.

elasticities must all be mutually consistent in that commodities whose targets are higher (lower) than initial consumption levels must have positive (negative) initial income elasticities.

Price elasticities determine the changes in consumption resulting from price changes. In addition to own-price elasticities, cross-price elasticities are needed to determine the response of consumption to prices of substitute commodities. DEMAND uses substitution proportions along with own-price elasticities to determine cross-price elasticities consistent with changes in relative consumption levels over time.

Policy Inputs

Food price and consumption policies in Korea have multiple and sometimes conflicting operational objectives. These include: 1) increasing farm income through high producer prices, particularly for grains; 2) increasing domestic production, also with high producer prices as incentives (in addition to crop improvement research, land and water development programs and credit and subsidy programs); 3) reducing food import and, hence, foreign exchange requirements through import controls, production programs and nonprice measures (such as riceless days, release of mixed grains, etc.) to reduce consumption; and 4) reducing inflationary pressures by controlling consumer prices.

One major example of a conflict arising from the above policy goals is the fact that keeping consumer prices low not only encourages high demands, thus increasing pressures for imports (as has happened in Korea for beef), but also in some cases fuels inflation by creating inflationary government deficits between high producer prices and low consumer prices for government-marketed commodities (as has happened for rice and barley).

Since DEMAND is intended for use in policy analysis, several relevant policy options, indicated in Figure 2.2, are built into DEMAND for each commodity:

1. the consumption level over time may be specified to observe the consequent impacts on imports/exports and prices;

2. imports/exports over time may be specified to investigate the implications for prices and consumption;

3. the consumer price or bounds on consumer price movements over time may be specified to observe the resulting nonfarm consumption and imports/exports;

4. producer price projections may be specified to investigate the effects on farm consumption, and when linked to a production model, production; and

5. carryover stock levels may be regulated to reflect the building of government reserves and to investigate the impacts on consumption, prices and imports/exports.

In addition, alternative foreign exchange rates may be specified to investigate the effects on the dollar value of foreign trade in food and on domestic consumer prices and consumption.

These policy options[3] can be classified in two dimensions (Table 2.2). One dimension categorizes them as quantity or price control policies. The other categorizes them according to how they are implemented in the model, i.e., optional independent policies and mandatory mutually exclusive policies.

---

[3]Except the foreign exchange rate. This affects DEMAND only indirectly in the current model formulation. Furthermore, it has broader implications for the national economy model and is, therefore, specified in that model as a policy assumption [5] and passed on to DEMAND.

Carryover Stock Level

Desired stock levels, in terms of the number of months' consumption in stock to be carried over each year, may be specified for each of the commodity groups considered. Following government statistics, stock levels are defined as quantities held at the start of the rice year (November) in farm and nonfarm households, by the government and at ports awaiting customs clearance [6]. An increase (decrease) in the desired stock level leads to a decrease (increase) in consumer supply for a given level of production. Thus, stock level changes affect prices and trade requirements. An important question which may be investigated with this policy is the building up of emergency reserves, apart from the government's ongoing price stabilization reserves, of the major food grains.

Producer Price

The producer price of a commodity (in ₩/MT, in constant 1970 won) may be projected over time by policy specification or may be determined by the model. In the latter case, the producer price is related to the consumer price with a constant marketing margin. If the producer price is to be set by policy, a real producer price (₩/MT in 1970 won) must be supplied by the user for specified years. The model will then linearly interpolate producer prices between those years. As an example, both the producer and consumer prices of barley may be specified by policy to maintain the dual price system--that is, where the producer price is higher than the consumer price, representing a government subsidy.

TABLE 2.2

Policy Instruments in DEMAND[1]

Mandatory Mutually Exclusive[3] \ Optional Independent[2]		None	Quantity: Regulation of carry-over stock levels[4]	Price: Producer price set by policy[5]
Quantity	Per capita consumption set by policy	X	X	X
Quantity	Imports/exports set by policy	X	X	X
Quantity	Imports/exports set by policy as long as specified price bounds not violated	X	X	X
Price	Bounds on consumer prices set by policy	X	X	X
Price	Consumer price set by policy	X	X	X

Notes:

[1]These instruments are commodity specific; that is, a different policy may be specified for each commodity group. Each X in the table is a possible policy option.

[2]These are independent in that their selection is not conditional on the use of any of the other policy instruments. They are optional in that either one or both or neither of the two policy instruments may be specified. That is, for each commodity, any row in the table may have an X in the first column or in either or both of the second and third columns.

[3]These are mandatory in that one of them must be chosen and mutually exclusive in that not more than one may be chosen. That is, no column may have more than one X in it.

[4]If this is not specified, the model uses a default value which represents recent past behavior.

[5]If the producer price of a commodity is not specified by policy, DEMAND computes it as a function of the consumer price and assumed marketing margin.

Per Capita Consumption

National per capita consumption of a commodity may be projected exogenously over time by specifying national average consumption (MT/ person-year) for specified years. The model will then linearly interpolate consumption for years between these points. Given domestic supply and population, this policy in effect determines the level of imports/exports such that the specified consumption level is achieved, and DEMAND projects the consumer prices resulting from this policy.

For example, it may be desired to explore the consequences of increasing rice consumption to a certain level by a certain year as a substitute for imported wheat, under the assumption that Korea has achieved a stable self-sufficiency in rice. In reality, such a target would be implemented through a combination of price controls and quantity restrictions such as mixing wheat with barley in bread and/or no wheat in alcoholic beverages. The model, however, would assume achievement of the target and would project the implied import/export requirements and the consequent consumer prices.

Imports/Exports

Imports/exports (or deficits/surpluses) of a commodity may be projected exogenously by providing values for imports/exports (MT/year) for specified years. The model will then linearly interpolate between these years. Given domestic supply from the production models, this policy determines total supply. DEMAND, then, determines the resulting market-clearing consumer prices.

For example, exports of pork may be projected independently of the model by tying them to projections of Japanese demand (essentially, Korea's only market for pork). The model would then project the domestic consumer prices of pork and of substitute meat commodities resulting from these exports.

In general, DEMAND's trade accounting treats deficits and surpluses as imports and exports, respectively. When DEMAND is run in conjunction with the resource allocation and trade model, exception is made for rice, barley and potatoes. Surpluses of barley and potatoes are first assumed to be fed to livestock to reduce feed grain import requirements, any remaining surpluses then being treated as exports (in the case of potatoes) or additions to stock (in the case of barley). All rice surpluses are treated as additions to stock.

Imports/Exports and Bounded Consumer Price

This option is implemented in two stages in DEMAND. First, projections of imports/exports for the commodity are specified by the user. At each point in time, then, DEMAND computes the consumer price consistent with clearing the given supply, after allowing for stock changes. This price is compared with its upper and lower bounds, also specified by the user (discussed below), and, if one or the other bound is violated, the second stage fixes the consumer price at the nearest bound and solves the model again, this time for consumption instead of price. In this case, imports/exports are a residual, overriding the value specified by policy in the first stage.

As a practical matter, it is advisable to follow this combination policy rather than imports/exports or per capita consumption alone. Otherwise, it may easily happen that unbounded prices will take on unrealistically high or low (even negative) values. In general, the more price inelastic a commodity is the more widely will its price respond to exogenously specified changes in per capita consumption or imports/exports.

Bounded Consumer Price

To bound the consumer price of a commodity, the user specifies upper and lower bounds relative to world prices and/or relative to the domestic consumer price in the previous time period. At any time t, given domestic supply and assuming deficits/surpluses do not change from the previous time period, the model determines market-clearing prices. If any price so determined violates its bounds, it is set to the nearest bound and the process is iterated to determine the deficits/surpluses consistent with the now-fixed price.

Either or both of two kinds of bounds may be specified--foreign trade bounds or price stabilization bounds. If only one type is desired, the upper and lower bounds of the other should be set very high and very low respectively. If both are to be considered, the more constraining one will, of course, be effective.

For example, a price stabilization policy may require that real beef prices neither increases by more than 10% nor decrease by more than 5% per year. In this case, parameters placing bounds on consumer prices relative to world prices are set very high and low respectively, to make those constraints ineffective. As another example, fruit can be subjected to a combination trade and price stabilization policy where fruit prices will be bounded above by both 110% of the import price and 115% of its previous domestic price, and below by the export price and 90% of its previous domestic price.

Consumer Price

The consumer price of a commodity (in ₩/MT, in constant 1970 won) may be exogenously projected over time by specifying its values for specified years. The model then linearly interpolates consumer prices between those years. With this price, consumption is determined, and, given domestic supply, deficits/surpluses are computed.

For example, the Korean government presently tries to control rice, barley and wheat flour prices. Alternative policy projections of these prices may be fed into DEMAND to examine the long-term consequences of achieving those target prices in terms of consumption, nutrition, deficits/surpluses and, when linked with the KASM production models, production, income, food grain self-sufficiency, etc.

Model Description

This section discusses in detail the equations of the demand-price-trade model (DEMAND) and their underlying assumptions and theoretical bases. A list of variable definitions for DEMAND appears in Appendix A, and a computer program list is included in Appendix B.

DEMAND is composed of three components: 1) farm demand, 2) price and nonfarm demand, and 3) outputs of trade accounting and performance measures of DEMAND. The structure of these components is shown in Figure 2.3, and each will be described in turn following a comparison of the farm and nonfarm demand models.

Farm and Nonfarm Demand Models Compared

Both farm and nonfarm demand for the food commodities and the aggregate nonfood commodity (see Table 2.1) are linearized Cobb-Douglas functions of price and income. Although the form of the consumption functions is the same, there are two basic differences between their implementation for farm and nonfarm demand.

The first difference is the implicit assumption that farm consumption is basically subsistence. This assumption is implied by the use in the consumption functions of exponential averages of past producer prices rather than concurrent consumer prices. It is also implicit in that farm and nonfarm consumers do not compete, in DEMAND, for the available supply. That is, farm consumption is computed first and then subtracted from domestic supply to determine the supply to nonfarm consumers (Figure 2.3).

In the case of staples, this is not an unreasonable assumption, as evidenced in Table 2.3. Grains, pulses and potatoes have accounted for about 75% of the value of farm food consumption, only about 5% of which have been purchased, the rest coming from own production, wages in kind and gifts. Of the remaining 25%, however, including fruits and vegetables and livestock products, about 45% to 50% has been purchased in the market. For these commodities, therefore, concurrent

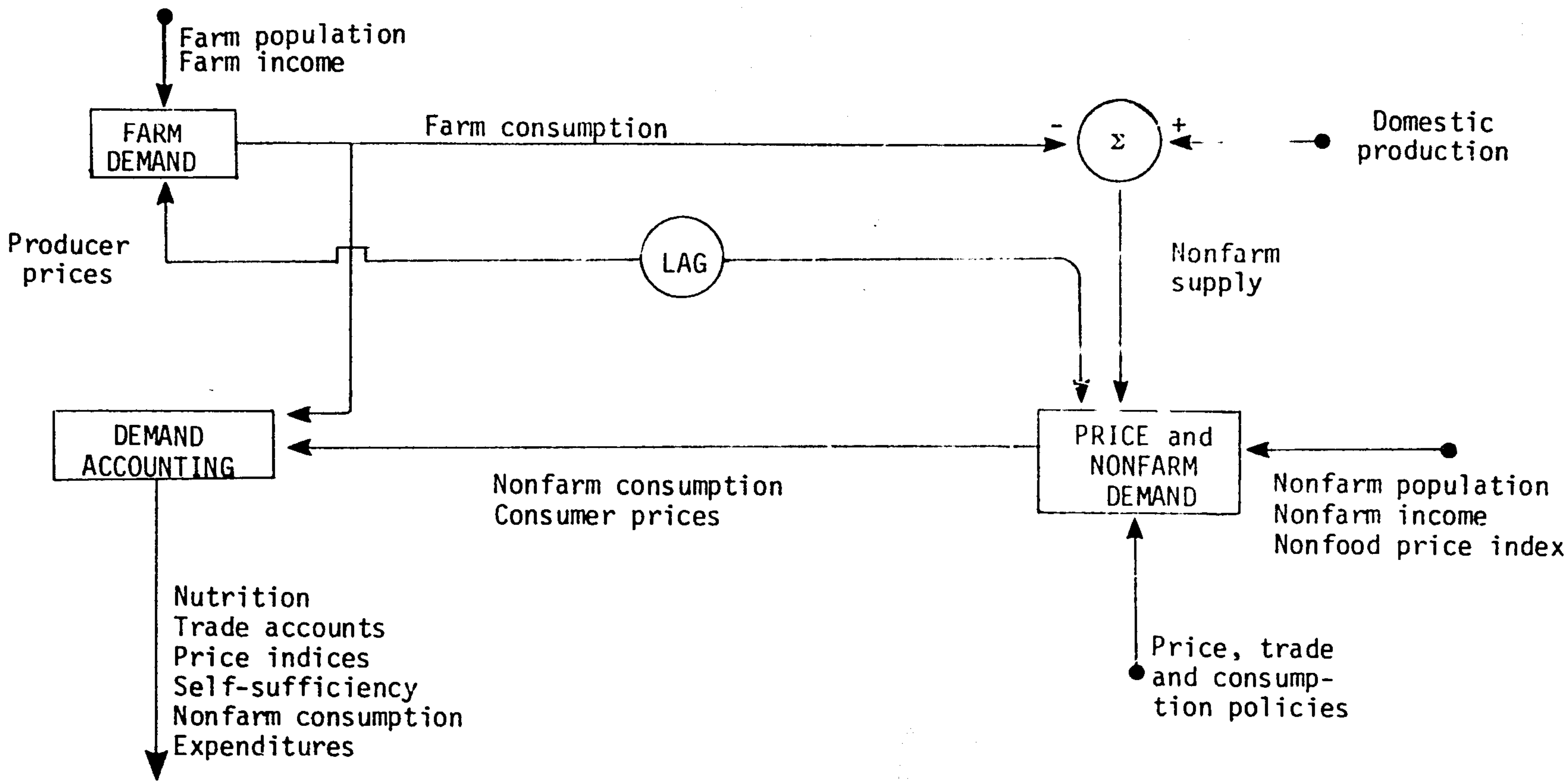

Figure 2.3

Internal Structure of DEMAND

TABLE 2.3

Farm Household Expenditures on Food Consumption, Percent Cash and Percent to Total

Item	Percent to Total		Percent Cash	
	1970	1974	1970	1974
Food	100.0	100.0	16.2	15.8
Major Food	73.2	76.1	6.1	4.3
Rice	51.0	56.4	4.8	3.6
Barley and wheat	17.5	15.8	10.6	7.4
Misc. grains	1.1	0.6	4.5	5.9
Pulses	0.9	1.1	6.2	6.8
Potatoes	2.7	2.3	1.3	1.2
Side Food	9.7	9.0	55.3	65.4
Fruits and vegetables	4.1	3.0	10.4	18.2
Meats	1.7	2.2	75.4	84.0
Milk and eggs	0.2	0.4	57.2	58.1
Marine food	3.7	3.5	95.9	95.1
Other*	17.1	14.9	37.3	44.3

*Primarily alcoholic beverages, confectioneries, condiments and prepared foods.

Source: Ministry of Agriculture and Fisheries, Report on the Results of Farm Household Economy Survey, 1971 and 1975.

consumer prices rather than past producer prices may be a better determiner of farm consumption behavior. Nevertheless, since the commodities purchased largely represent a relatively minor portion of total farm food consumption--indeed, only about 16% of the total has been purchased in recent years--in the interests of model and computational simplicity, DEMAND maintains the implicit assumption of farm subsistence consumption for all commodities. Therefore, farm consumption is independent of nonfarm consumption and concurrent prices.

This brings us to the second basic difference between the farm and nonfarm demand models. The system of nonfarm demand equations--one for each commodity, including the aggregate nonfood commodity--is solved simultaneously with a total consumption expenditure constraint for a set of per capita consumptions and a set of consumer prices. For any individual commodity either consumption or price must be prespecified, leaving the other to be solved for.

The farm demand equations, on the other hand, are computed singly and independently of one another, where all prices are prespecified in that they are lagged prices. After concurrent producer prices are computed in the price and nonfarm demand component, the farm expenditure constraint comes into effect when farm consumption of the aggregate nonfood commodity is computed as a residual. Figure 2.4 indicates the sequence of computations flowing from farm food consumption, to simultaneous determination of nonfarm food and nonfood consumption and consumer and producer prices, to calculation of farm nonfood consumption as a residual.

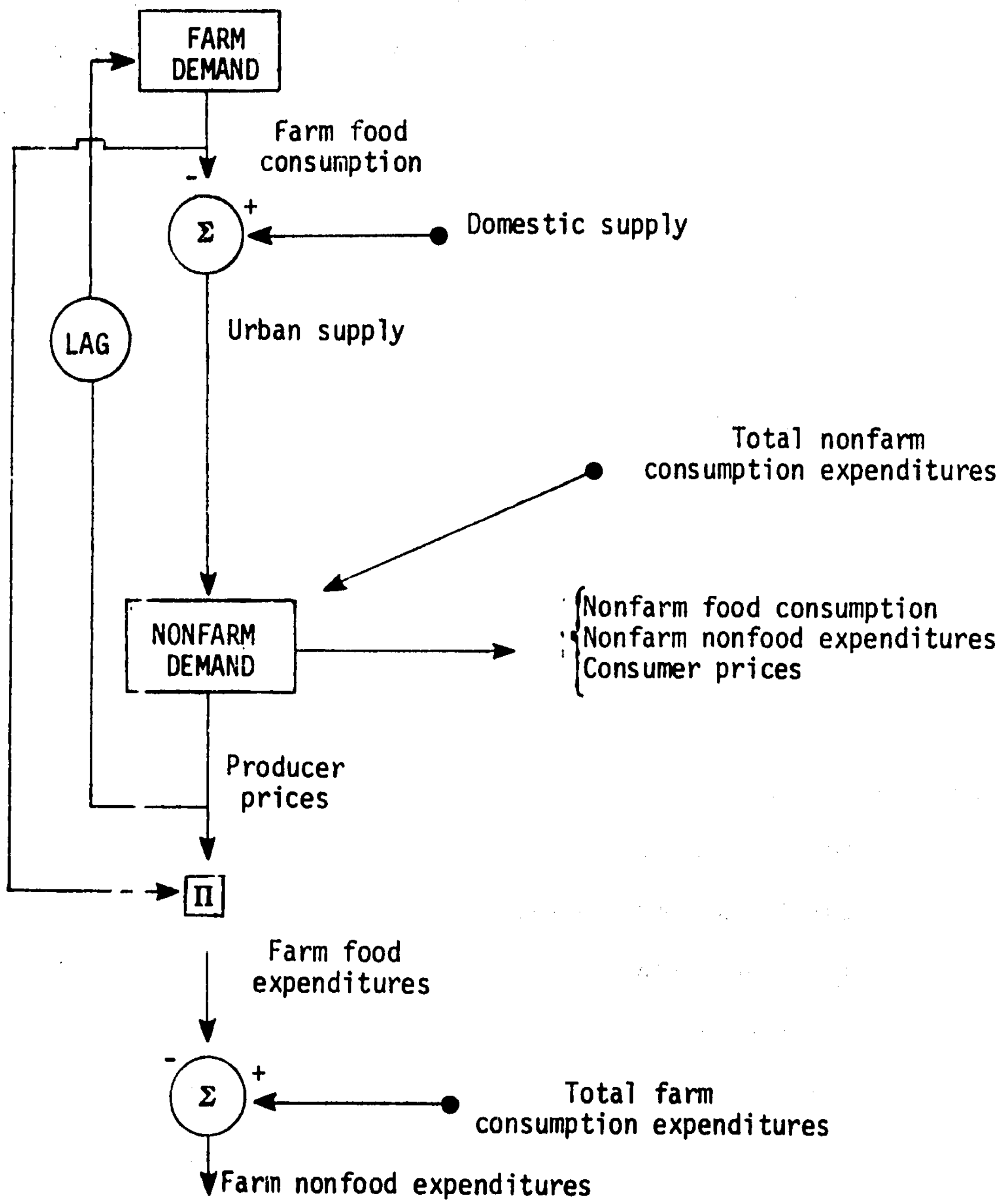

Figure 2.4

Computational Sequence of
Farm and Nonfarm Demand

Farm Demand

The farm demand component of DEMAND projects per capita farm consumption of each of the food commodities of KASM based on lagged producer prices and income. This subsection 1) derives the linear form of the demand equation, and 2) describes the equations generating outputs of the farm demand component. In addition, there is a detailed discussion of the computation of independent variables and elasticities. This detailed discussion may be skipped, if the reader wishes, without loss of continuity.

Consumption Functions

The linearized consumption function for each commodity is derived from the Cobb-Douglas form:

$$(1) \quad PCCON_{ir}(t) = \alpha_i \left(\frac{RFRMY_r(t)}{CPUNF(t)}\right)^{ELASIR_i(t)} \prod_{j=1}^{n} \left(\frac{EPAVG_{jr}(t)}{CPUNF(t)}\right)^{ELASPR_{ij}(t)}$$

where[4] for commodity i in region r:

PCCON = farm consumption (MT/person-year)
RFRMY = exponential average of regional farm income (W/person-year) (equation (9))
EPAVG = exponential average of producer prices (W/MT) (equation (7))
CPUNF = index of nonfood price
ELASIR = income elasticity (equation (12))
ELASPR = price elasticities (equation (11))
n = NCOMAG = number of agricultural commodities

and where consumption is initialized ($PCCON_{ir}(t_0)$) in equation (34). The linearization is accomplished by first taking logarithms

---

[4]See Appendix A for a complete list of variable definitions for DEMAND.

$$(2)\quad \ln PCCON_{ir}(t) = \ln \alpha_i + ELASIR_i(t)\,[\ln RFRMY_r(t) - \ln CPUNF(t)] + \sum_{j=1}^{NCOMAG} ELASPR_{ij}(t)\,[\ln EPAVG_{jr}(t) - \ln CPUNF(t)]$$

and then taking derivatives (assuming, as a simplifying approximation, that elasticities are constant)

$$(3)\quad \frac{dPCCON_{ir}}{PCCON_{ir}} = ELASIR_i \left(\frac{dRFRMY_r}{RFRMY_r} - \frac{dCPUNF}{CPUNF}\right) + \sum_{j=1}^{NCOMAG} ELASPR_{ij} \left(\frac{dEPAVG_{jr}}{EPAVG_{jr}} - \frac{dCPUNF}{CPUNF}\right)$$

and finally approximating the differentials with first differences, i.e., $dx \approx x(t) - x(t\text{-}DT)$, and rearranging terms to get the form used in the model

$$(4)\quad PCCON_{ir}(t) = PCCON_{ir}(t\text{-}DT) \Bigg\{ 1 + ELASIR_i(t\text{-}DT) \left(\frac{RFRMY_r(t) - RFRMY_r(t\text{-}DT)}{RFRMY_r(t\text{-}DT)} - \frac{CPUNF(t) - CPUNF(t\text{-}DT)}{CPUNF(t\text{-}DT)}\right) + \sum_{j=1}^{NCOMAG} ELASPR_{ij}(t\text{-}DT) \left(\frac{EPAVG_{jr}(t) - EPAVG_{jr}(t\text{-}DT)}{EPAVG_{jr}(t\text{-}DT)} - \frac{CPUNF(t) - CPUNF(t\text{-}DT)}{CPUNF(t\text{-}DT)}\right) \Bigg\}$$

Price and income are deflated by the nonfood price index in order to maintain the homogeneity condition in the demand equations. The homogeneity condition requires that the sum of all income, own-price and cross-price elasticities for each commodity be zero. Deflating by the nonfood price index is equivalent to computing the cross-price elasticity of demand for each commodity with respect to the nonfood price index as a residual in order for the sum to be zero.[5]

Farm Demand Outputs

Based on regional per capita farm consumption from equation (4), DEMAND computes for each commodity i regional and national consumption.

$$(5)\quad CON_{ir}(t) = PCCON_{ir}(t)\ POPR_r(t)$$

$$(6)\quad RDEM_i(t) = \sum_{r=1}^{NREGN} CON_{ir}(t)$$

where:

CON = total regional farm consumption of a commodity (MT/year)
RDEM = total national farm consumption of a commodity (MT/year)
NREGN = number of regions

Notice that consumption is determined only for agricultural commodities (i = 1, 2, ..., NCOMAG). Farm consumption of the aggregate nonfood commodity is assumed to be the residual of total consumption expenditures over expenditures on the agricultural commodities.

---

[5]This can be seen if we write equation (1) as

$$C_i = \alpha_i \left(\frac{Y_i}{P_n}\right)^{\varepsilon_i^Y} \prod_{j=1}^{n-1} \left(\frac{P_j}{P_n}\right)^{\varepsilon_{ij}^P} \equiv \alpha_i\, Y_i^{\varepsilon_i^Y} \prod_{j=1}^{n} P_j^{\varepsilon_{ij}^P}$$ where the

equivalence holds if

$$\varepsilon_{in}^P = -\left(\varepsilon_i^Y + \sum_{j=1}^{n-1} \varepsilon_{ij}^P\right) \text{ so that } \varepsilon_i^Y + \sum_{j=1}^{n} \varepsilon_{ij}^P = 0.$$

Equations (4) - (6) are essentially the farm demand component of DEMAND. Prior to computing (4), however, DEMAND computes the price and income variables and elasticities appearing on the right-hand side of (4). The computation of price and income elasticities and exponential averages of recent past producer prices and farm income will be described below. The reader not interested in this detail may skip this discussion and proceed directly to the next subsection on the price and nonfarm demand component.

Independent Variables and Elasticities

Independent variables in the farm consumption functions (equation (4)) for each commodity in each region are exponential averages of recent past farm prices (EPAVG) and per capita farm household disposable income (RFRMY).

$$(7)\quad \frac{d}{dt}\left(EPAVG_{ir}(t)\right) = \frac{1}{DELP}\left(PAVG_{ir}(t)\ VFPIX(t) - EPAVG_{ir}(t)\right)$$

$$EPAVG_{ir}(t_0) = PAVG_{ir}(t_0-1)$$

where:

EPAVG = exponential average of recent producer prices (W/MT)
PAVG = producer price (W/MT) (equation (30))
VFPIX = ratio of the urban consumer price index to the index of prices paid by farmers
DELP = exponential averaging time constant (years)[6]

---

[6] DELP being a time constant means (ignoring VFPIX) that, if the actual price $PAVG(t) = PAVG(t_1)$ for all $t > t_1$, i.e., it remains constant after time t_1, then the average price DELP years later $EPAVG(t_1 + DELP) = .37(PAVG(t_1) - EPAVG(t_1))$, i.e., it takes DELP years for the average price to close 63% of the gap between it and the actual price.

Prices PAVG are adjusted by VFPIX for purposes of farm consumption response. For consistency, all prices in KASM are real prices implicitly deflated by the consumer price index for all cities and all commodities. For farm demand, however, prices should be deflated by the index of prices paid by farm households, not that of urban households. If the two change at the same rate, it doesn't make any difference, but farm consumer prices have been rising faster than urban consumer prices. Therefore, the factor VFPIX has been included in equation (7).

National average per capita farm income is also exponentially averaged and disaggregated by region for the farm demand equations. In addition, total per capita farm consumption expenditures are computed based on the previous period's income to be used later to determine nonfood consumption as a residual (equation (85)).

(8) $FRMY_r(t) = FPCI(t)\ TRPOP(t)\ PRFRMY_r(t)/POPR_r(t)$

(9) $\frac{d}{dt}(RFRMY_r(t)) = \frac{1}{YDEL}(FRMY_r(t) - RFRMY_r(t))$

(10) $EFRMY(t) = APCF \cdot FPCI(t\text{-}DT)\ TRPOP(t\text{-}DT)/TRPOP(t)$

where:

FPCI = per capita farm income (W/person-year) (from the farm income accounting component of KASM)
RFRMY = exponential average of recent past regional farm income (W/person-year)
YDEL = exponential averaging time constant (years) (see discussion in footnote 6 above)
FRMY = regional farm income (W/person-year)
TRPOP = total farm population (persons) (from the population model of KASM)
POPR = farm population, by region (persons) (from the population model of KASM)

PRFRMY = proportion of total farm income earned in a region (from the farm income accounting component of KASM)
EFRMY = farm consumption expenditures (W/person-year)
APCF = average farm propensity to consume (proportion)
r = indexes regions, r = 1, 2, ..., NREGN
DT = simulation time increment (years)

Farm consumption behavior depends not only on the independent variables price and income but also on the price and income elasticities. In order to maintain a reasonable and mutually consistent set of direct-price and cross-price elasticities, equation (11) computes cross-price elasticities as functions of relative consumption levels and the direct-price elasticities. Appendix D discusses the theoretical and mathematical derivation of equation (11). Equation (11) is used for the set of grain commodities. An identical equation computes cross-price elasticities for meat commodites (beef, pork, chicken and fish).

$$(11)\quad ELASPR_{ij}(t) = -PSUBCG_{ij1}\ PSUBTG_{j1}\ ELASPR_{jj}\ \frac{PCONVG_j PCCON_{jr}(t-DT)}{PCONVG_i PCCON_{ir}(t-DT)}$$

where:

$ELASPR_{ij}$ = cross-elasticity of farm demand for commodity i with respect to the price of commodity j
$ELASPR_{jj}$ = own-price elasticity of farm demand for commodity j
$PSUBCG_{ij}$ = proportion of substitution for grain commodity j coming from grain commodity i
$PSUBTG_j$ = proportion of change in demand for commodity j which is substituted
PCONVG = conversion factor for grains (e.g., wheat to flour) (proportion)
PCCON = per capita consumption (MT/person-year) (equation (4))
i,j = index grain commodities (rice, barley, wheat, miscellaneous grains and potatoes)
r = indexes regions

Notice that equation (11) implies that the larger consumption of commodity j (e.g., rice) is relative to that of commodity i (e.g., miscellaneous grains), the greater will be the proportional response of consumption of commodity i to changes in the price of j, and conversely.

Income elasticities also vary over time in the model in order to keep consumption 1) within reasonable limits as income increases in the long run and 2) consistent with reasonable expectations of long-run nutrition levels. Consumption targets are specified for each commodity, and, as actual consumption of a commodity approaches the target, the income elasticity goes to zero. Essentially this is saying that, if relative prices remain constant, consumption will increase (or decrease) asymptotically toward the target as income increases. If, in any year, the price response component of consumption causes the target to be passed, equation (12) will tend to bring consumption back toward the target in later years.

For example, assume current consumption is increasing over time below the target for a given commodity. Then, if price responses cause consumption to exceed the target level in some year, the sign on the income elasticity will change and cause consumption to decrease toward the target in succeeding years.

$$(12) \quad ELASIR_i(t) = ELASIR_i(t_0) \frac{PCT_i(t) - PCCON_{ir}(t)}{PCT_i(t_0) - PCCON_{ir}(t_0)}$$

where:

ELASIR = income elasticity of consumption of farm consumers
PCT = consumption target (MT/person-year)
PCCON = farm consumption (MT/person-year) (equation (4))

For all commodities except rice, the target PCT_i is assumed to be constant over time (see Appendix 5 for data values). For rice, however, we assume Korea will ultimately follow Japan's pattern of declining rice consumption, although income elasticities will remain positive at least until the late 1970's [6, Chapter 5]. The target for rice, then, is assumed to decline linearly from its initial value to its final value over the twenty-year period 1980-2000.

Price and Nonfarm Demand

The price and nonfarm demand component of DEMAND solves, in each time period, a simultaneous linear system of equations for 1) per capita nonfarm consumption of each of the food commodities of KASM, 2) per capita nonfarm consumption expenditures for the aggregate nonfood commodity, and 3) consumer prices for each food commodity. The solution obtained is consistent with a total consumption expenditure constraint. This component of DEMAND also computes producer prices, either as specified by policy or related to consumer prices with constant marketing margins.

The remainder of this subsection describes 1) the consumption functions and the constraints which make up the simultaneous equations system, and 2) the equations which generate other outputs of the price and nonfarm demand component. In addition, detailed discussions are given of 3) the initialization of DEMAND, 4) policy inputs and the calculation of independent variables and elasticities, and 5) the derivation of the reduced form of the system of equations and the iterative solution alogrithm. These detailed discussions may be skipped, if the reader wishes, with no loss in continuity.

Consumption Functions and Constraints

There are 21 linear equations in the simultaneous system: 20 consumption functions, one for each KASM commodity currently considered (NCOM=20; see Table 2.1); and one for the expenditure constraint. In vector-matrix form, the equations are:

$$\text{(13)} \quad PCCONU(t) = AINTER(t) + AE(t)CPU(t) + BDEM(t)S1(t)$$

$$\text{(14)} \quad CPU(t)^{T}PCCONU(t) = ALPHA \cdot EINCOM(t)$$

where the superscript "T" signifies the transpose of the vector or matrix, and where:

PCCONU	= a 20x1 vector of per capita nonfarm consumption (MT/person-year for the 19 agricultural commodities, and ₩/person-year for the nonfood commodity)
CPU	= a 20x1 vector of consumer prices in constant 1970 won (₩/MT for the 19 agricultural commodities; and index for the nonfood commodity, with $CPU_{20}(1970)$ = 1.0)
AINTER	= a 20x1 vector of intercepts
AE	= a 20x20 matrix of price slopes
BDEM	= a 20x1 vector of slopes with respect to S1
S1	= a scalar elasticity expansion parameter computed to insure the expenditure constraint (discussed below)
ALPHA	= a constant scalar parameter computed to insure consistency among the initial conditions (equation (27) discussed below)
EINCOM	= total per capita nonfarm consumption expenditures (₩/person-year) (equation (22)).

Each of these equations will be discussed in turn, beginning with the demand equations (13).

The consumption functions are linearized from the Cobb-Douglas form for each commodity i:

$$(15)\quad PCCONU_i(t) = \alpha_i \left[INCOM(t)^{ELASI_i(t)} \prod_{j=1}^{n} CPU_j(t)^{ELASP_{ij}(t)} \right]^{S1(t)}$$

where:

INCOM = exponentially averaged recent past nonfarm income (₩/person-year) (equation (34))
ELASI = income elasticity
ELASP = price elasticities
i = indexes commodities, i=1, 2, ..., NCOM
n = number of commodities = NCOM = 20

by first taking logarithms

$$(16)\quad \ln PCCONU_i(t) = \ln \alpha_i + S1(t)\left[ELASI_i(t) \ln INCOM(t) + \sum_{j=1}^{n} ELASP_{ij}(t) \ln CPU_j(t) \right]$$

and then differentiating, assuming as a simplifying approximation that the elasticities are constant,

$$(17)\quad \frac{dPCCONU_i}{PCCONU_i} = S1 \left[ELASI_i \frac{dINCOM}{INCOM} + \sum_{j=1}^{n} ELASP_{ij} \frac{dCPU_j}{CPU_j} \right]$$

$$\left[ELASI_i \ln INCOM + \sum_{j=1}^{n} ELASP_{ij} \ln CPU_j \right]$$

Approximating the differential by $dx \approx x(t) - x(t-DT)$, rearranging terms, and using the homogeneity condition (equation (31)), guaranteed by equations (32) and (33) below, we get equation (13) in vector-matrix form, where the coefficients

$$\text{(18)}\quad AINTER_i(t) = PCCONU_i(t-DT)\left\{1 + S1(t-DT)\left[ELASI_i(t-DT)\,\frac{INCOM(t)}{INCOM(t-DT)} - X(t)\right]\right\}$$

$$\text{(19)}\quad AE_{ij}(t) = PCCONU_i(t-DT)\;S1(t-DT)\;\frac{ELASP_{ij}(t-DT)}{CPU_j(t-DT)}$$

$$\text{(20)}\quad BDEM_i(t) = PCCONU_i(t-DT)\;X(t)$$

$$\text{(21)}\quad X(t) = ELASI_i(t-DT)\ln INCOM(t-DT) + \sum_{j=1}^{n} ELASP_{ij}(t-DT)\ln CPU_j(t-DT).$$

S1 is an elasticity expansion parameter which is computed as part of the solution to (13)-(14). Looking at (15), (16) or (17) it becomes clear why S1 is called an elasticity expansion parameter: it multiplies all the elasticities for all commodities (i.e., expanding or contracting them) in such a way that the resulting consumptions and prices are consistent with the expenditure constraint. Nominally, S1 has unit value if no consumption adjustment is necessary to meet the constraint. If an adjustment is necessary, the use of S1 may be interpreted theoretically as saying those commodities which are more price and income responsive will be affected more by the adjustment.

Equation (14) says that the sum of expenditures on each commodity (price times consumption) must equal total consumption expenditures EINCOM times a constant adjustment parameter ALPHA. ALPHA is computed during the initialization phase of the model's execution in order to insure consistency among the initial conditions. The justification for ALPHA and its interpretation are discussed more fully below. Total consumption expenditures are

(22) $EINCOM(t) = APCN \cdot UPCI\ (t-DT)POP_2(t-DT)/POP_2(t)$

where:

APCN = average nonfarm propensity to consume out of the previous period's income
UPCI = nonfarm per capita income (₩/person-year) (from the national economy model of KASM).

Price and Nonfarm Demand Outputs

In addition to per capita consumption and consumer prices, this component also computes total nonfarm consumption of each commodity and producer price. Total consumption is

(23) $Q_i(t) = PCCONU_i(t)POP_2(t)$

where:

Q = nonfarm consumption (MT/year for agricultural commodities; W/year for the nonfood commodity)
POP_2 = nonfarm population (persons) (from the population model of KASM)
= indexes commodities, i=1, 2, ..., NCOM

Actual time series of producer prices are used in DEMAND for the years 1970 to the most recent year recorded, called BASEYR in the model. The time series stored in the model are updated regularly with the latest empirical values, and BASEYR is incremented one. The projection period in the model is for $t \geq$ BASEYR + 1. The period 1970 $\leq t <$ BASEYR + 1 is considered the tracking period, where actual stock changes and consumer, producer and world prices are input to DEMAND and the model results in terms of consumption and imports/exports can be compared with recorded empirical values. This is discussed further in the next chapter of this documentation representing test results.

For the projection period, $t \geq BASEYR + 1$, producer prices of each commodity are projected in either of two ways: 1) outside the model by policy specification (see Table II.2), or 2) related to the consumer price in the model by a constant marketing margin. In the first case, the user provides data to the model on the value of the commodity's producer price at certain years in the future. DEMAND then linearly interpolates between those years. In the second case, the model computes at each point in time

(24) $TPAVG_i(t) = CPU_i(t)/(1+MM_i)$

where:

TPAVG = national average producer price (W/MT in 1970 constant won)
MM = marketing margin as a proportion of producer price
i = indexes agricultural commodities, i = 1, 2, ..., NCOMAG, where NCOMAG = 19

Conceptually, producer prices may be different in each farming region. As a simplification, however, KASM currently assumes prices in each region are the same as the national average, i.e.

(25) $PAVG_{ir}(t) = TPAVG_i(t)$

for each region r = 1, 2, ..., NREGN

The remainder of this subsection goes into some detail on the initialization of DEMAND, policy inputs and the computation of independent variables and elasticities used in the consumption functions, and derivation of the reduced form and the iterative solution algorithm. If the reader wishes, he may skip these discussions and go directly, without loss of continuity, to the next subsection on trade accounting and performance measures of DEMAND.

Initialization of DEMAND

The constant coefficient ALPHA is computed at time zero in DEMAND to insure consistency among the initial conditions for prices, consumptions, total consumption expenditure and the proportion of total consumption which is food, especially since information on these various initial conditions comes from a variety of sources. In particular, household income and consumption expenditure information comes from urban household surveys for <u>all cities</u>, while consumer prices used initially and in the tracking period in KASM are for the city of Seoul, and per capita consumption in quantity terms is derived from food balance sheet data for 1970 [6, Chapter 3]. ALPHA could be interpreted as the ratio between Seoul prices and averages of prices for all cities, as can be seen if we rewrite (14) equivalently as

$$(26)\quad \sum_{i=1}^{NCOM} \frac{1}{ALPHA}\; CPU_i(t)\; PCCONU_i(t) = EINCOM(t)$$

ALPHA, then, is computed at time zero as

$$(27)\quad \sum_{i=1}^{NCOMAG} PCCONU_i(t_0) \big/ (.405\; EINCOM(t_0)) = ALPHA$$

where 40.5% of consumption expenditures in all cities in 1970 was for food.

Consumption of each commodity is initialized (and used in (27)) for both farm and nonfarm consumers consistent with total supply available for human consumption at t_0 and with assumed ratios of farm-to-nonfarm consumption at t_0 [6, Chapter 4]. Supply is computed at time zero based on other 1970 data in KASM for yields, production and marketing losses, imports and exports, stock changes, feed use and industrial consumption and livestock production.

$$(28)\quad PCCONU_i(t_0) = \frac{SUPPLY_i(t_0)}{FNR_i POP_1(t_0) + POP_2(t_0)}$$

$$(29)\quad PCCON_{ir}(t_0) = FNR_i \; PCCONU_i(t_0)$$

$$(30)\quad PCCONU_{NCOM}(t_0) = ALPHA \cdot EINCOM(t_0) - \sum_{i=1}^{NCOMAG} CPU_i(t_0) \; PCCONU_i(t_0)$$

where:

SUPPLY = total supply available for domestic human consumption (MT/year)
POP_1 = farm population (persons)
POP_2 = nonfarm population (persons)
FNR = ratio of farm consumption (in MT/person-year) to nonfarm consumption at t_0
i = indexes agricultural commodities, i = 1, 2, ..., NCOMAG
r = indexes regions, r = 1, 2, ..., NREGN

Independent and Policy Variables and Elasticities

Cross-price and income elasticities for nonfarm demand are computed in the same way as those for farm demand. These are explained above so will not be repeated here. See equations (11) and (12) and the associated discussions there and in Appendix D.

The homogeneity condition states that, if all prices and income change in the same proportion, there will be no change in consumption. This condition is satisfied for each commodity i when the sum of all price and income elasticities for that commodity is zero. That is, we must have

$$(31)\quad ELASI_i(t) + \sum_{j=1}^{n} ELASP_{ij}(t) = 0$$

for all i and all t, where ELASI is the nonfarm income elasticity of demand, ELASP is the nonfarm price elasticity of demand, and n = NCOM is the number of commodities.

For each agricultural commodity (i = 1, 2, ..., NCOMAG), this is accomplished by assuming its cross-elasticity with respect to the nonfood price index is a residual, i.e.,

$$(32)\quad ELASP_{INCOM}(t) = -ELASI_i(t) - \sum_{j=1}^{NCOMAG} ELASP_{ij}(t)$$

For the nonfood commodity, given the income and own-price elasticities, the cross-price elasticities with food prices are apportioned according to consumption expenditures so that (31) holds.

$$(33)\quad ELASP_{nj}(t) = -\left[ELASP_{nn}(t) + ELASI_n(t)\right] \cdot \frac{PCCONU_j(t)\ CPU_j(t)}{\sum_{i=1}^{m} PCCONU_i(t)\ CPU_i(t)}$$

where n = NCOM indexes the nonfood commodity; j indexes the agricultural commodities; and m = NCOMAG, the number of agricultural commodities.

As in the farm demand component, consumption responds to an exponential average of recent past income levels (equation (16)). This average is the solution to the differential equation

$$(34)\quad \frac{d}{dt}\left(INCOM(t)\right) = \frac{1}{YDEL}\left(UPCI(t) - INCOM(t)\right)$$

$$INCOM(t_0) = UPCI(t_0)$$

where:

INCOM = exponentially averaged income (W/person-year)
UPCI = per capita income (W/person-year) (from the national economy model of KASM)
YDEL = exponential averaging time constant (see discussion in footnote (6) above)

DEMAND cannot determine both price and consumption of a given commodity simultaneously. One or the other for each commodity must be specified, and the system of equations (13)-(15) solves for the rest and for S1. In addition to income, therefore, other independent variables in the simultaneous system of equations are a subset of the prices and a subset of the consumptions. If NP of the prices are independent variables, then NCOM-NP of the consumptions (those associated with the remaining prices) are also independent variables. The solution gives the remaining NP consumptions and NCOM-NP prices, as well as S1.

In the tracking period, $1970 \leq t < BASEYR + 1$, all prices are specified as the recorded empirical time series (i.e., NP = NCOM), and all the consumptions are determined by the model. In the projection period, $t \geq BASEYR + 1$, there are two ways the price of a commodity may be specified as an independent variable, both involving the policy instruments discussed earlier (Table 2.2).

First, the user supplies data to DEMAND on values the price of that commodity is to have at certain points in time. The model, then, linearly interpolates between those points each time period in the simulation and solves for per capita consumption.

The policy instruments specifying bounded prices for a commodity (Table 2.2) are implemented each time period in two stages. The first stage treats consumption of the commodity as the independent variable.

If in any time period the resulting consumer price lies outside the bounds, DEMAND sets it at the nearest bound for that time period. In this case, the second stage treats the price as the independent variable and solves the system again, this time for consumption.

The price bounds require that

(35) $CPUMIN_i(t) \leq CPU_i(t) \leq CPUMAX_i(t)$

where:

CPUMIN = lower bound on price (W/MT) (equation (36))
CPUMAX = upper bound on price (W/MT) (equation (37))
i = indexes those commodities for which the policy instrument bounding prices is used

The upper and lower bounds in equation (35) can be related either to world prices as a foreign trade policy, to previous domestic prices as a price stabilization policy, or both. If it is both, the more binding constraint in each time period is used in (35). That is,

(36) $CPUMIN_i(t) = \max\ [PWPMIN_i PWLDEX_i(t), (1\text{-}DT\cdot PCPMIN_i)CPU_i(t\text{-}DT)]$

(37) $CPUMAX_i(t) = \min\ [PWPMAX_i PWLDIM_i(t), (1\text{+}DT\cdot PCPMAX_i)CPU_i(t\text{-}DT)]$

where:

PWLDEX = export price (W/MT)
PWLDIM = import price (W/MT)
PWPMIN = lower bound proportion of export price
PWPMAX = upper bound proportion of import price
PCPMIN = maximum proportional rate of decrease in domestic consumer price
PCPMAX = maximum proportional rate of increase in domestic consumer price
max = takes the maximum of the terms in brackets
min = takes the minimum of the terms in brackets

The constant parameters PWPMIN, PWPMAX, PCPMIN and PCPMAX are specified by policy assumption. The import and export prices are not projected by DEMAND. The user provides the model with projected values at certain points in time, and the model linearly interpolates between those points.

There are four ways in which per capita consumption of a commodity may be specified as the independent variable (Table 2.2), in which cases the simultaneous system solves for the consumer price of that commodity. In all cases, consumption is assumed to equal the total supply available to the nonfarm market, and DEMAND finds the consumer price which clears that market. Total nonfarm supply is computed as domestic production plus net imports minus stock increases, losses and farm consumption.

(38) $SUPPLY_i(t) = TDSUP_i(t) + DEFCIT_i(t)$

where:

TDSUP = domestic supply (MT/year)
DEFCIT= net imports (imports minus exports) (MT/year)
i = indexes commodities whose consumptions are specified as independent variables

and where

(39) $TDSUP_i(t) = TOUTPT_i(t)\ [1-PFLOSS_i] - RDEM_i(t) - CHGSTK_i(t) - FEED_i(t)$

where:

TOUTPT = total domestic production (MT/year) (from the production model of KASM)
PFLOSS = farm and marketing losses (proportion)
RDEM = farm consumption (MT/year) (equation (6))
CHGSTK = increases in carryover stock (MT/year) (equation 43))
FEED = animal consumption (MT/year) (from the production model of KASM)

In addition, industrial consumption of wheat and inedible industrial crops are excluded from TDSUP.

With domestic production, feed use and farm consumption taken as given to the nonfarm demand component, total nonfarm supply and, hence, consumption are determined by the stock changes and imports/exports.

Changes in stocks carried over at the end of the rice year (see the discussion above in the subsection on policy inputs to DEMAND) are a function of the difference between desired and actual stock levels. The desired stock level of a commodity i is:

(40) $DSTK_i(t) = STKRT_i TDEML_i(t)/12$

where:

DSTK = desired carryover stock level (MT)
STKRT = months of consumption desired to be carried over, a policy parameter
TDEML = an exponential average of recent consumption requirements (MT/year)

The average TDEML is computed by

(41) $$\frac{dTDEML_i(t)}{dt} = \frac{1}{3}\left(TDEM_i(t) - TDEML_i(t)\right)$$

$$TDEML_i(t_0) = TDEM_i(t_0)$$

where TDEM is human consumption requirements (equation (69)), and the time constant 3 years is assumed. (See footnote (6) above for a discussion of the meaning of "time constant.") The actual stock level is

(42) $$\frac{dSTK_i(t)}{dt} = CHGSTK_i(t)$$

$$STK_i(t_0) = STKO_i$$

where:

STK = actual stock level (MT)
CHGSTK = change in stock level (MT/year)

and where

(43) $$CHGSTK_i(t) = \begin{cases} \frac{1}{2}(DSTK_i(t) - STK_i(t)) & \text{for } t \geq BASEYR + 1 \\ VCSTK_i(t) & \text{for } < BASEYR + 1 \end{cases}$$

For $t_0 \leq t < BASEYR + 1$, recorded empirical stock changes VCSTK are used, while an exponential lag with time constant 2 years (this could be generalized) is assumed for $t \geq BASEYR + 1$.

Given stock changes, now, there are three ways net imports may be specified so that consumption of a commodity is an independent variable. (See the discussion above on policy inputs to DEMAND.) First, the user may project target per capita consumption levels of the commodity for the nation at various points in time in the future. DEMAND will linearly interpolate between those points and, given farm consumption, stock change, feed use, production and losses, compute net imports necessary to meet the targeted national average per capita consumption of the commodity.

(44) $DEFCIT_i(t) = TPOP(t)PCSUP_i(t) - TDSUP_i(t) - RDEM_i(t)$

where:

TPOP = total population (persons) (from the population model of KASM)

PCSUP = target national average per capita consumption (MT/person-year) (a policy input)

Secondly, net imports may be projected directly by the user and input to DEMAND, which linearly interpolates between the specified points in time and determines the market-clearing consumer price in each time period. This specification of imports may be combined with the specification of price bounds as in equations (35)-(37) above. In this case, if the price bounds are violated in the first solution iteration, the price is fixed at the bound and net imports are computed as a residual of the resulting consumption, essentially overriding the targeted net import level.

Finally, nonfarm consumption of a commodity may be determined as an independent variable by assuming net imports do not change from the previous time period. That is,

(45) $DEFCIT_i(t) = DEFCIT_i(t\text{-}DT)$

This specification is combined in DEMAND with the price bounds so that net imports will change over time only if necessary to keep prices within the bounds.

Reduced Form and Iterative Solution Algorithm

Let's begin by reordering the vector of consumer prices CPU, putting all prices which are independent variables first in a subvector called P_1, followed by those prices which are to be market-determined in a subvector P_2. So now we have, instead of CPU, a reordered, partitioned vector $\begin{pmatrix} P_1 \\ P_2 \end{pmatrix}$, where P_1 is of dimension NP x 1 and P_2 is of dimension (NCOM-NP) x 1. Similarly, let's reorder the vector of per capita consumptions PCCONU into a corresponding $\begin{pmatrix} C_1 \\ C_2 \end{pmatrix}$, where C_1 is the reordered subvector of NP dependent variable consumptions, and C_2 is the reordered subvector of NCOM-NP consumptions treated as independent variables. Similarly, we can reorder and partition the coefficient vectors and matrices in equations (13)-(14) to correspond to P_1, P_2, C_1, and C_2. Thus, we can rewrite (13)-(14) as dropping the time subscripts)

$$(13') \quad \begin{pmatrix} C_1 \\ C_2 \end{pmatrix} = \begin{pmatrix} A0_1 \\ A0_2 \end{pmatrix} + \begin{pmatrix} A1_{11} & A1_{12} \\ A1_{21} & A1_{22} \end{pmatrix} \begin{pmatrix} P_1 \\ P_2 \end{pmatrix} + \begin{pmatrix} A2_1 \\ A2_2 \end{pmatrix} S1$$

where:

A0 = the reordered and partitioned intercept vector AINTER
A1 = the reordered and partitioned matrix of price slopes AE
A2 = the reordered and partitioned vector of S1 slopes BDEM

$$(14') \quad P_1^T C_1 + C_2^T P_2 = ALPHA \cdot EINCOM$$

The problem is to solve for the unknowns C_1, P_2 and S1 in terms of the knowns P_1 and C_2 (and income, which has been included in the intercept terms--see equation (18)).

Before putting (13')-(14') into reduced form, the solution is simplified one step by substituting the equations for C_1 in (13') into (14') giving

$$(14'') \quad P_1^T \left\{AO_1 + [A1_{11} A1_{12}] \begin{bmatrix} P_1 \\ P_2 \end{bmatrix} + A2_1 S1\right\} + C_2^T P_2 = ALPHA \cdot EINCOM$$

Finally, the reduced form can be written--putting all the unknowns on the left and the knowns on the right and combining (13') and (14')--as

$$(46) \quad AY = X$$

where:

$$Y = \begin{bmatrix} P_2 \\ S1 \end{bmatrix}_{N \times 1}$$

$$X = \begin{bmatrix} C_2 - AO_2 - A1_{21} P_1 \\ ALPHA \cdot EINCOM - P_1^T (AO_1 + A1_{11} P_1) \end{bmatrix}_{N \times 1}$$

$$A = \begin{bmatrix} A1_{22} & A2_2 \\ C_2{}^T + P_1{}^T A1_{12} & P_1 A2_1 \end{bmatrix}_{N \times N}$$

and where the dimension N = NCOM - NP + 2. Now the system (46) can be solved by

(47) $Y = A^{-1} X$

If one or more prices are bounded, the solution is iterative. In the first iteration, market-clearing prices P_2 are determined consistent with the given consumption C_2. If any price in P_2 lies outside its bounds, the price furthest outside is set to its nearest bound, that is, it is removed from P_2 and placed in P_1, and its corresponding consumption is moved from C_2 to C_1. A new reduced form is computed and the system solved again. This iteration continues until all bounded prices lie within their respective bounds.

Trade Accounting and Performance Measures

Based on consumptions and prices determined in the other DEMAND components, foreign trade accounts and a number of other variables are computed which can be used as measures of how the DEMAND system responds to alteranative policy assumptions. These include nutrition levels, self-sufficiency levels and requirements for self-sufficiency, price indices, consumption expenditure accounts, and government subsidies implied by the price policies. In addition, food consumption is apportioned to processed and unprocessed foods as outputs to the national economy model.

Trade Accounting

For accounting purposes, DEMAND treats all deficits and surpluses, except for rice, barley and potatoes, as imports and exports respectively. In reality, of course, stocks, for example, may be built up or depleted beyond the policy specifications in equations (40)-(43); or, in the case of surplus grains, feed grain import requirements can be at least partially offset by domestic food grain surpluses, as KASM does do for barley and potatoes [illegible] the production model is running. Furthermore, surpluses or deficits of rice and barley are added to or subtracted from stocks instead of accounted as exports or imports, subject to stock level constraints.

Deficits for all commodities (negative deficits are surpluses) are computed by (48). Once a solution to (47) has been obtained for consumer prices and per capita nonfarm consumption and total nonfarm consumption Q has been computed in (23), deficits computed in (48) are consistent with those computed in (44) or (45) or specified by policy directly.

(48) $DEFCIT_i(t) = Q_i(t) - TDSUP_i(t)$

where (48) is essentially an inversion of equation (38), i.e., $Q_i = SUPPLY_i$ and the market is cleared.

For tobacco exports (i=9), equation (48) is adjusted to account for the different forms of tobacco as sold by farmers, domestically consumed and foreign traded [6, Chapter 3]. Also, for all imports

($DEFCIT_i(t) > 0$), the quantity imported is increased over the deficits computed in (48) by a proportional shrinkage loss, PILOSS.

For exported commodities ($DEFCIT_i(t) < 0$), the value of exports and export profits are

(49) $VALDEF_i(t) = DEFCIT_i(t)\ PWLDEX_i(t)$

(50) $PRFEXP_i(t) = -DEFCIT_i(t)\ [PWLDEX_i(t) - CPU_i(t)]$

where:

VALDEF = value of exports (if negative)(W/year)
PWLDEX = export price (W/MT)
PRFEXP = profits on exports (W/year)
CPU = domestic consumer price (W/MT)

In addition, total value of exports, VALDF1, is the negative sum of VALDEF for exported commodities.

For imported commodities ($DEFCIT_i(t) > 0$),

(51) $VALDEF_i(t) = DEFCIT_i(t)\ PWLDIM_i(t)/(1+TARIFF_i)$

(52) $TARREV_i(t) = VALDEF_i(t) TARIFF_i$

(53) $PRFIMP_i(t) = DEFICIT_i(t)\ [CPU_i(t) - PWLDIM_i(t)]$

where:

VALDEF = value of imports (if positive) (W/year)
TARIFF = import tariff rate
PWLDIM = import price (W/MT)
TARREV = tariff revenues (W/year)
PRFIMP = profit on imports (W/year)

Total value of imports, VALDF2, is also computed.

Finally, DEMAND computes the agricultural commodity trade balance and total trade profits.

(54) $TVALDF(t) = VALDF1(t) - VALDF2(t)$

(55) $DVALDF(t) = TVALDF(t)/WOND(t)$

(56) $$TRDPRF(t) = \sum_{i=1} PRFIMP_i(t) + PRFEXP_i(t)$$

where:

TVALDF = agricultural commodities trade balance (W/year)
DVALDF = agricultural commodities trade balance ($/year)
WOND = foreign exchange rate (W/$)
TRDPRF = total trade profits (W/year)

Self-Sufficiency

In Korea, 100% self-sufficiency in rice, barley, soybeans and livestock products is a major policy objective for economic and political reasons. Given yields, total production and feed requirements from the production models of KASM, DEMAND computes what would be required in the way of production and, for crops, land allocations for 100% self-sufficiency. In addition, actual self-sufficiency levels are computed for comparison.

Three types of self-sufficiency requirements are calculated: 1) requirements for human consumption only; 2) requirements for human consumption and feed use; and 3) requirements for human consumption, feed use and stock changes. The second type is the one generally considered in Korea. That is, if imports (exports) equal carryover stock increases (decreases) for a commodity, then Korea considers itself 100% self-sufficient in that commodity.

For each crop, land area requirements depend on effective yield, defined as actual yields adjusted for farm and marketing losses and seed requirements.

(57) $$EFFYLD_i(t) = YLD_i(t)\ [1 - PFLOSS_i] - SDPHA_i$$

where:

EFFYLD = effective yield (MT/ha-year)
YLD = actual yield (MT/ha-year) (from the technology change model of KASM)
PFLOSS = loss rate (proportion)
SDPHA = seeding rate (MT/ha-year)
i = indexes crop commodities, i = 1, 2, ..., NCROP

Then, area planted to that commodity to meet the three types of self-sufficiency requirements are, respectively,

$$(58)\quad AREQH_i(t) = TDEM_i(t)/EFFYLD_i(t)$$

$$(59)\quad AREQHF_i(t) = AREQH_i(t) + FEED_i(t)/EFFYLD_i(t)$$

$$(60)\quad ARQHFS_i(t) = AREQHF_i(t) + CHGSTK_i(t)/EFFYLD_i(t)$$

where:

AREQH = crop area required to meet human consumption demand (ha)
TDEM = total human consumption demand (MT/year) (equation (74))
AREQHF = crop area required to meet human and animal consumption demands (ha)
FEED = feed use (MT/year) (from the production model of KASM)
ARQHFS = crop area required to meet human and animal consumption demands and stock changes (ha)
CHGSTK = stock change (MT/year) (equation (48))

Total land utilization requirements for each type of self-sufficiency are, respectively,

$$(61)\quad TARH(t) = \sum_{i=1}^{NCROP} AREQH_i(t)$$

$$(62)\quad TARHF(t) = \sum_{i=1}^{NCROP} AREQHF_i(t)$$

$$(63)\quad TARHFS(t) = \sum_{i=1}^{NCROP} ARQHFS_i(t)$$

Total production in MT/year, required for each kind of self-sufficiency depends on the above land area requirements and the actual yields:

(64) $QREQH_i(t) = YLD_i(t)\ AREQH_i(t)$

(65) $QREQHF_i(t) = YLD_i(t)\ AREQHF_i(t)$

(66) $QRQHFS_i(t) = YLD_i(t)\ ARQHFS_i(t)$

Finally, production requirements for each livestock commodity to meet human consumption demands are

(67) $QRQHLV_i(t) = TDEM_i(t)/(1\text{-}PFLOSS_i)$

where i indexes livestock commodities.

For each commodity, the actual self-sufficiency level is the ratio of domestic production to effective demand, where effective demand is exclusive of stock changes. That is, the second type of self-sufficiency is computed:

(68) $$SELFSU_i(t) = \frac{TOUTPT_i(t)}{TOUTPT_i(t) + DEFCIT_i(t) - CHGSTK_i(t)}$$

where:

SELFSU = self-sufficiency level (proportion)
TOUTPT = domestic production (MT/year) (from the production model of KASM)

National Consumption and Nutrition

Farm and nonfarm consumption are combined to form total and per capita national consumption.

(69) $TDEM_i(t) = RDEM_i(t) + Q_i(t)$

(70) $PCNCON_i(t) = TDEM_i(t)/TPOP(t)$

where:

TDEM = human consumption of a commodity (MT/year)
RDEM = farm consumption (MT/year) (equation (6))
Q = nonfarm consumption (MT/year) (equation (23))
PCNCON= national average per capita consumption (MT/person-year)
TPOP = national population (persons) (from the population model of KASM)

Nutrition accounting is done for daily calorie and protein consumption of the farm and nonfarm populations as well as the national average. The nutrient intake is also broken down by source, plant sources, animal sources and total. Farm consumption of calories from plant sources is

(71) $CAL_{11}(t) = \sum_k CALPU_k(t)\ PCCON_k(t)/365$

where:

$CALL_{11}$ = farm consumption of calories from plant sources (k cal/person-day)
CALPU = calorie content of foods (k cal/MT)
k = indexes crop commodities

Similarly, nonfarm and national average plant calorie consumption, CAL_{12} and CAL_{13}, respectively, are computed using PCCONU and PCNCON, respectively. In the same way, calorie consumption from animal sources CAL_{21}, CAL_{22} and CAL_{23} are computed for farm and nonfarm consumers and for the national average, respectively, where the index k in equation (71) represents livestock commodities. Total calorie consumption, i.e., from both plant and animal sources, is

(72) $CAL_{3j}(t) = CAL_{1j}(t) + CAL_{2j}(t)$

for j = 1, 2, 3, i.e., farm, nonfarm and national average. The percent of total calorie consumption coming from each source is also computed for each consumer group.

Finally, similar equations compute protein consumption PRO (grams/person-day) from each source for each consumer group based on the protein content of foods PROTPU (grams/MT).

Price Indices and Government Subsidies

If the marketing margin MM, used in equation (24) to compute producer prices, is assumed to represent normal marketing costs including returns to the entrepreneur and other value added, then, for those commodities whose producer prices are set by policy assumption, we can compute an implied government subsidy. Negative subsidies could be interpreted as additional taxes.

(73) $GOVSUB_i(t) = TPAVG_i(t)[1 + MM_i] - CPU_i(t)$

where GOVSUB is the implied subsidy in ₩/MT.

Six price indices are computed by DEMAND: an all commodities consumer price index; consumer price indices for all foods, for rice, for other grains (barley, wheat, pulses, potatoes and miscellaneous grains) and for livestock products (beef, milk, pork, chicken, eggs and fish); and a world price index for food commodities. Since all prices in KASM are in real terms (1970 won), price indices computed are deflated price indices. Therefore, the consumer price index for all commodities CPI, which is the implied deflator, should always be unity (i.e., it is deflated by itself--see discussion in Chapter 5).

(74) $CPI(t) = \sum_{i=1}^{NCOM} W_i \frac{CPU_i(t)}{CPU_i(t_0)} \Big/ \sum_{i=1}^{NCOM} W_i$

where:

CPI = all commodities consumer price index, deflated
CPU = real consumer prices (W/MT)
W_i = price index weights

Whereas (74) is summed over all commodities, the food, rice, other grains and livestock price indices (CPFI, CRPI, COGPI, and CMPI, respectively) are summed over only the appropriate commodities. The world food price index for Korea (WPFI) is based on import and export prices and weighted by trade volumes in 1970.

Consumption Expenditure Accounting

For nonfarm consumers, total, food and nonfood consumption expenditures and savings are computed by DEMAND. Total consumption expenditures, which are constrained to equal EINCOM in equation (14), are

$$(75)\quad TEXPT(t) = \sum_{i=1}^{n} CPU_i(t)\ Q_i(t)/ALPHA$$

where TEXPT is in W/year and n = NCOM, the number of commodities. Nonfood expenditures TEXNF, in W/year, are

$$(76)\quad TEXNF(t) = CPU_{NCOM}(t)Q_{NCOM}(t)/ALPHA$$

and food expenditures by nonfarm consumers TEXPF, in W/year, are

$$(77)\quad TEXPF(t) = TEXPT(t) - TEXNF(t)$$

The food proportion of nonfarm consumption expenditures is

$$(78)\quad RFTEX(t) = TEXPF(t)/TEXNF(t)$$

and the nonfarm savings rate is

$$(79)\quad XDIFF(t) = POP_2(t-DT)UPCI(t-DT) - TEXPT(t)$$

where it is assumed consumption and savings in the current period are out of income from the previous period. Per capita values of the above variables, except RFTEX, are also computed.

For farm consumers, per capita nonfood consumption expenditures (needed by the national economy model of KASM) are computed as a residual:

$$(80)\quad FEXNFC(t) = EFRNY(t) - \sum_{i=1}^{n} TPAVG_i(t)RDEM_i(t)/POP_1(t)$$

where EFRMY is total per capita farm consumption expenditures (equation (10)) and n = NCOMAG, the number of food commodities. The residual FEXNFC is constrained not to fall below a minimum level of nonfood consumption, where savings resources are used if necessary to stay above the minimum.

Processed and Unprocessed Food Consumption

Finally, the national economy model of KASM needs projection of farm and nonfarm food consumption expenditures broken down into processed and unprocessed foods. Constant proportions, based on 1972 survey data, are used to compute the breakdown. This is a gross simplification, especially for long run analyses, but it can serve as a first approximation for the purposes of KASM. For farm consumers,

$$(81)\quad FCONU(t) = \sum_{i=1}^{n} TPAVG_i(t)RDEM_i(t)PFU_i \Big/ POP_1(t)$$

$$(82)\quad FCONP(t) = \sum_{i=1}^{n} TPAVG_i(t)RDEM_i(t)(1-PFU_i) \Big/ POP_1(t)$$

where:

FCONU = farm expenditures on unprocessed foods (W/person-year)
FCONP = farm expenditures on processed foods (W/person-year)
POP_1 = farm population
PFU = proportion of total expenditures of a commodity in unprocessed form
n = NCOMAG, the number of agricultural commodities

Similar equations compute UCONU and UCONP for nonfarm consumers.

CHAPTER 3

MODEL TESTING AND CREDIBILITY

There are four major tests a model must pass to be considered credible in the eyes of the ultimate users of its product [18, Chapter 1]. This is as true of intuitive mental models as it is of computerized mathematical models. These tests are:

1. coherence--a test of internal logical consistency;
2. correspondence--a test of comparing the model's behavior with that of its real world referrent;
3. clarity--a test of comprehensibility of the model and its results; and
4. workability--a test of efficiency, i.e., the usefulness (value) of the model's results relative to the difficulty (cost) of obtaining them.

These credibility test of a model are continually applied not only during model development but every time it is used. The clarity and workability tests are of particular relevance and importance in model utilization. This chapter limits its discussion to the coherence and correspondence testing conducted during model development. These include internal consistency checks, time-series tracking tests, and sensitivity analysis.

Internal Consistency

The internal logical consistency of DEMAND, as for any computer simulation model, is tested every time the model is run. This involves verifying, for ... le, that upper and lower bounds on variables are not violated, and generally that model behavior is consistent with basic assumptions. This process is frequently associated with debugging the computer program but also relates to verifying the mathematical model and its set of parameter values.

In early stages of model development and testing [9] it became apparent that the linearization of the basic Cobb-Douglas demand-price model (equation (15) in Chapter 2) tended to be unstable when inverted to solve for price as a function of supply, particularly for price-inelastic commodities. It was not uncommon for market-clearing prices determined in this way to range from unreasonably large to even negative values. At first, reserve prices were used as lower bounds to prevent low or negative prices, and the iterative solution process described in Chapter 2 was implemented [9]. Later, DEMAND was generalized to provide both upper and lower bounds related to foreign trade policies and/or domestic price stabilization policies.

Therefore, it is advisable to use DEMAND only with bounded or exogenously specified consumer prices. That is, the switch KFT_i (described in Table 5.2 in Chapter 5) should have value 3, 4 or 5 for each commodity i; values 1 and 2 specify unbounded price policy options.

In DEMAND's standard run (Appendix C), eight commodities, including the major grains rice, barley and wheat, have consumer prices projected exogenously (KFT=5) in the projection period beginning in 1976, and the remaining 12 are to be determined by the market subject to bounds (KFT=3 or 4). In 1976, the first year of the projection period, 11 iterations are required, with 10 commodities' prices hitting the bounds and only two being completely market determined. For the next four years, 1977-1980, only six of the 12 are bounded, and Runs 7-9 are bounded during the last five years of the simulation. These results are, of course, dependent on the price bounds and exogenous production projections specified for the standard run; less restrictive bounds would tend to reduce the number of iterations (bounded prices) in the solution.

An important area for further model development, therefore, would be to consider alternatives to the linearized Cobb-Douglas consumption functions. The state-space modeling approach could be a useful point of departure.

Another, more minor, internal consistency problem encountered relates to the assumption that all prices in KASM are real prices, where the implicit deflator is Korea's all-cities, all-commodities consumer price index. This deflator does not appear explicitly in the model; however, it is used to deflate past prices input to the model. These time series of deflated prices are used in the model as a basis for projections into the future. Therefore, the projected prices implicitly have the same deflator. For consistency with this implicit assumption, then, the index formed from all the consumer prices, including that of the nonfood aggregate commodity, must be unity. That is, it is implicitly deflated by itself.

We can see this if we assume we can write each real consumer price CPU as

(83) $CPU_i(t) = P_i(t)/I(t)$

where:

P_i = the nominal consumer price (₩/MT)
I = the all commodities consumer price index

and where the index I is defined as

$$(84)\quad I(t) = \sum_{i=1}^{n} W_i \frac{P_i(t)}{P_i(0)}$$

with $\Sigma W_i=1$. To get an index CPI of the real prices CPU, then (assuming $CPU_i(0)=P_i(0)$),

$$(85)\quad CPI(t) = \sum_{i=1}^{n} W_i \frac{CPU_i(t)}{CPU_i(0)} = \sum_{i=1}^{n} W_i \frac{CPU_i(t)}{P_i(0)} = \sum_{i=1}^{n} W_i \frac{P_i(t)/I(t)}{P_i0}$$

$$= \frac{1}{I(t)} \sum_{i=1}^{n} W_i \frac{P_i(t)}{P_i(0)} = \frac{1}{I(t)} I(t) = 1$$

where we have substituted (83) and (84) into (85).

However, in DEMAND's standard run, CPI drifts upwards from 1.0 to a value of 1.129 by 1985. This happens because the nonfood price index is projected outside of DEMAND, by the user or by KASM's national economy model (NECON), independently of the food prices. Furthermore, the major food prices, especially rice with a price index weight of 169.7 out of 1000.0, are also projected exogenously in the standard run by policy assumption. While nominal prices could be independently projected in this way, real prices are not independent of one another since they are all part of the deflator. Therefore, a possible remedy we can suggest would be to add another equation to the simultaneous system (equations (13) and (14) in Chapter II) which would determine the nonfood price index as a dependent variable of DEMAND, rather than of NECON, so that CPI = 1.0.

Time-Series Tracking

Necessary, but not sufficient, for passing the correspondence test of credibility is the model's ability to track historical time series to an acceptable degree. What is "acceptable" depends on how the model is to be used. In addition to time-series tracking, decision makers typically require the model's behavior to correspond to their own conception of "how things work." This applies both 1) to simulating past behavior which may not be measured in recorded statistics but which the decision maker has some "feelings" about, and 2) to projecting future behavior under alternative assumptions.

This is not to say that a model's role is limited to confirming the preconceptions of the decision makers using it, although simply quantifying those preconceptions may be not without value in some circumstances. Rather, a model can also serve an educational function, leading to changes in those preconceptions, to the extent that unexpected or surprising behavior of the model can be satisfactorily explained.

National average per capita consumptions of the 17 food commodities (PCNCON -- equation (70) in Chapter 2) were selected for comparison of DEMAND's behavior with historical data. In addition, consumptions of three commodity groups -- grains, fruits and vegetables, and meats -- were observed to examine the model's tracking ability at a more aggregate level. The model was run during the tracking period (1970-1975) using historical values of consumer and producer prices, nonfarm income and domestic production. Therefore, consumption was the only variable, other than trade and other performance variables, left for DEMAND to determine. Essentially, then, given the form of the consumption functions, the time-series tracking tests boil down to a test of the elasticity values used. Although the tracking period available is relatively short -- 1971-1975 -- it is long enough to give a rough idea of DEMAND's tracking ability. (The year 1970 is excluded from the results since DEMAND is initialized to 1970 consumption data.)

A normalized sum of squared errors is used for each of the 20 variables to measure how well the model fits the historical series. In addition, since these measures are normalized, they can be added together as an overall measure of goodness of fit. The measures are:

$$SS_i = \sum_{t=1971}^{1975} \left(\frac{Y_i(t)-Y_i(t)}{\overline{Y}_i} \right)^2$$

$$TSS = \sum_{i=1}^{17} SS_i$$

where SS_i is the goodness-of-fit measure of variable i; $Y_i(t)$ is the simulated value of variable i in year t; $\hat{Y}_i(t)$

Obviously, SS_i=0 would mean that DEMAND exactly reproduced the recorded series of variable i. In order to evaluate the goodness-of-fit, we need some idea of how large a value of SS_i would be considered "bad," or, conversely, how small a value is necessary to say that the model tracks well. A reasonable reference value is the bad case where the model produces only zero values for variable i; for example, zero rice consumption over the tracking period. This would not be a worse case, however, since negative values or simulated values more than twice the recorded values would be worse than zero. If DEMAND produced nothing but zeroes, then, calling $SS_i(0)$ the resulting goodness-of-fit measure, we would have:

$$SS_i(0) = \left(\sum_{t=1971}^{1975} \frac{\hat{Y}_i(t)}{\overline{Y}_i} \right)^2$$

which can be shown to be equivalent to

$$SS_i(0) = 4\left(\frac{S_i}{\overline{Y}_i} \right)^2 +5$$

where S_i is the sample standard deviation of the five historical data points of variable i and $\overline{Y}_i$ is their mean.

The historical and simulated values of per capita consumption of each commodity and commodity group are shown in Table 3.1. Only the groups and rice consumption are graphed in Figure 3.1. Table 3.1 also shows the computed goodness-of-fit measures for each, their sum TSS, and the values these measures would have if DEMAND produced only zeroes for the 20 series.

For most commodities, particularly the major ones and the three aggregations, DEMAND tracks consumption reasonably well. This is particularly remarkable for rice and barley, considering the Korean government has been heavily involved in quantity control programs for these commodities while DEMAND relies only on price and income to explain consumption. In the case of rice, for example, Figure 3.1 indicates that DEMAND seems to be able to pick up the trend but not the rather large oscillations historically observed (at least, as derived from official statistics) about that trend.

Table 3.1

DEMAND Time-Series Tracking Results of
Per Capita Consumption (kg/person-year)

Commodity		1971	1972	1973	1974	1975	SS[1]	SS(0)[2]
1. Rice	H[3]	135.4	120.6	116.6	124.1	115.5	.0143	5.0171
	S[4]	124.0	120.9	123.1	124.0	122.0		
2. Barley	H	40.4	43.3	45.1	45.1	41.8	.0182	5.0091
	S	39.6	41.2	41.1	41.9	43.5		
3. Wheat	H	50.9	56.5	56.7	40.1	46.6	.1162	5.0047
	S	43.5	46.9	45.3	40.7	42.5		
4. O. grains	H	3.5	3.6	4.7	6.7	3.8	.3755	5.3606
	S	4.4	4.3	4.9	5.9	6.1		
5. Fruits	H	11.2	13.1	11.5	15.2	16.4	.0451	5.1142
	S	11.9	13.4	12.9	14.0	14.3		
6. Pulses	H	7.9	7.2	7.5	7.9	9.6	.0549	5.0539
	S	8.2	8.1	8.0	8.0	8.1		
7. Vegetables	H	74.9	68.6	64.4	72.0	69.3	.0269	5.0126
	S	68.1	70.2	72.5	73.6	73.0		
8. Potatoes	H	56.0	51.9	42.1	36.8	57.8	.2129	5.1384
	S	52.6	46.1	43.2	38.5	36.3		
9. Tobacco	H	1.0	1.4	2.3	1.5	1.2	.8380	5.4511
	S	.9	1.0	1.1	1.2	1.2		
10. Ind. crops	H	1.3	1.6	1.6	1.4	2.2	1.1174	5.1859
	S	1.0	1.1	1.1	1.1	.7		
11. Beef	H	1.2	1.2	1.3	1.5	2.0	.0275	5.2180
	S	1.3	1.2	1.4	1.6	1.9		
12. Milk	H	2.4	4.0	3.2	3.9	4.7	.5469	5.2303
	S	4.1	3.8	4.7	4.0	3.3		
13. Pork	H	2.4	2.5	2.5	2.6	2.7	.0508	5.0081
	S	2.3	2.5	2.6	2.7	2.1		
14. Chicken	H	1.5	1.6	1.5	1.5	1.5	.0154	5.0035
	S	1.6	1.6	1.6	1.4	1.6		
15. Eggs	H	4.1	4.4	3.9	4.3	4.4	.0573	5.0106
	S	4.4	4.7	4.7	4.7	4.8		
16. Fish	H	24.9	30.0	36.0	42.3	42.0	.0454	5.1876
	S	30.3	32.8	39.2	40.1	43.9		
17. Residual[5]	H	7.0	12.7	16.6	15.5	13.5	.3135	5.3256
	S	12.3	12.4	12.5	12.6	12.7		

Table 3.1 (continued)

Commodity		1971	1972	1973	1974	1975	SS[1]	SS(0)[2]
Grain[5]	H	294.1	283.1	272.7	260.7	275.1	.0136	5.0080
	S	272.3	267.5	265.6	259.0	258.5		
Frt/veg[6]	H	86.1	81.7	75.9	87.2	85.7	.0193	5.0124
	S	80.0	83.6	85.4	87.6	87.3		
Meats[7]	H	30.0	35.3	41.3	47.9	48.2	.0343	5.1533
	S	35.5	38.1	44.8	45.8	49.5		
All commodities							3.8763[8]	87.3313[9]

Notes:

1. Goodness-of-fit measure, normalized sum of squared errors.
2. Goodness-of-fit measure if simulated series were zero.
3. Historical time series from Alan R. Thodey, <u>Food and Nutrition in Korea, 1965-1974</u>, KASS Special Report 11, 1976.
4. Simulated time series, from standard run of DEMAND.
5. Rice, barley, wheat, other grains, pulses, potatoes.
6. Fruits, vegetables.
7. Beef, pork, chicken, fish.
8. Sum of the 17 SS values.
9. Sum of the 17 SS(0) values.

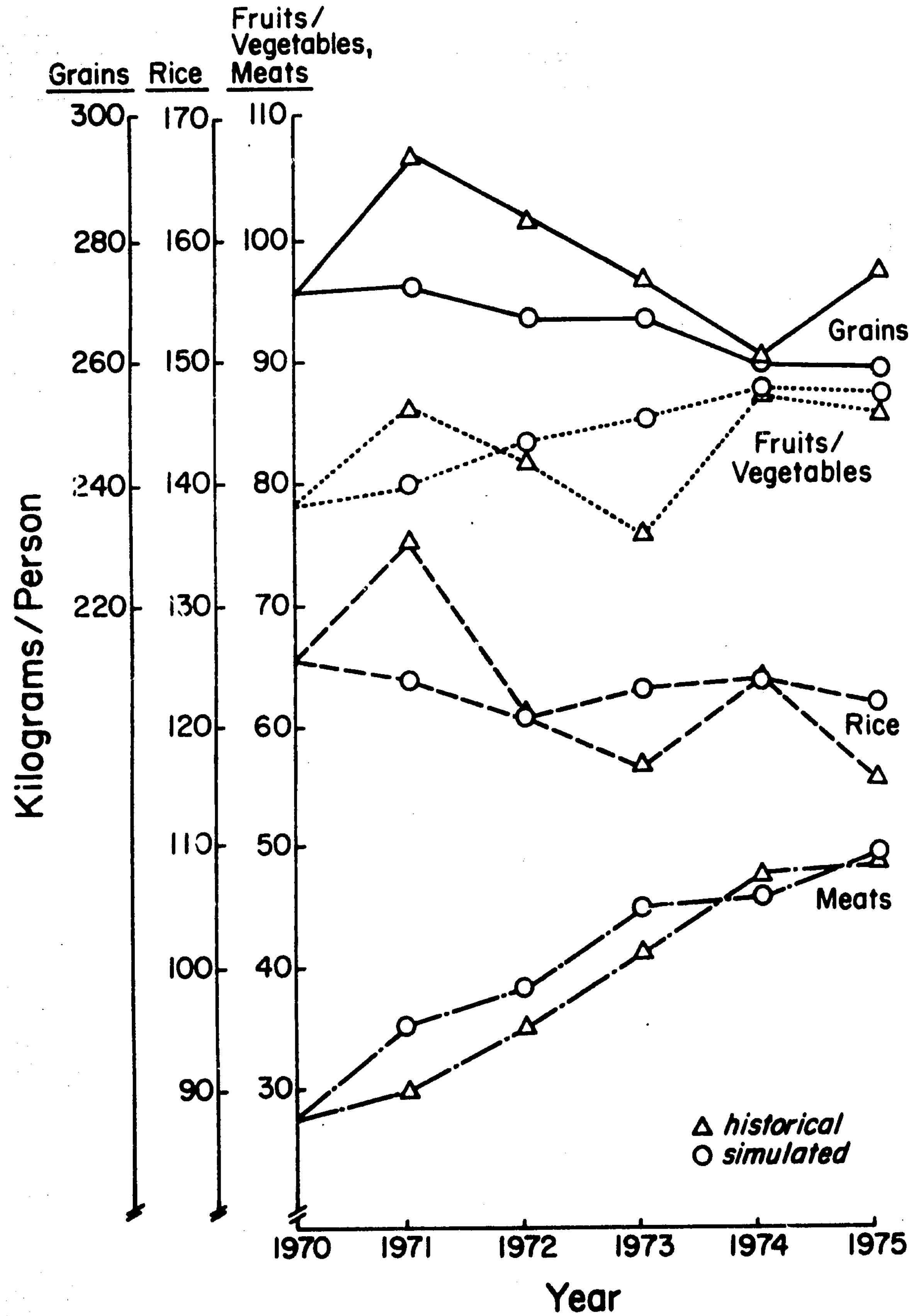

Figure 3.1

DEMAND Time-Series Tracking Results

A similar observation can be made in general, that is, DEMAND fits the trends reasonably well. Indeed, this is not surprising since the elasticity values were "tuned" specifically to achieve these tracking results. Further tuning may improve the results, but the marginal improvements which may be obtained should be weighed against the efforts which would be required to obtain them. Furthermore, one should perhaps not expect or require a price-income demand model to capture more than the basic trends of a complex, dynamic process involving changing tastes, storage behavior and other economic and psychological factors.

A final observation we make here illustrates an important point to keep in mind when evaluating the time-series tracking performance of a model. That is, the historical series used for comparison are also subject to error. For example, the data indicate that potato consumption increased about 60% in 1975 alone. DEMAND, however, projects a continuation of the downward trend of recent years. Thus, it is not always clear, particularly when the simulated and historical series are relatively close, which one is closer to "reality."

Sensitivity Analysis

In sensitivity analysis we are interested in testing the responsiveness of a model to changes in parameter values. Usually, the value of a single parameter is changed by an appropriate amount, and the effect of this change on the behavior of the model variables is observed.

Sensitivity analysis provides important information for both coherence and correspondence tests of credibility. That is, indications of the model's internal logical consistency as well as its consistency with its real-world referrent can be gotten by observing and explaining the behavioral changes arising from parameter value changes. Furthermore, sensitivity analysis provides insights into the workings of the model as a system and, by implication, of the real-world system as well, insights which are valuable in designing policy strategies. Finally, sensitivity analysis results indicate those parameters whose values are relatively more crucial to the model's behavior, information which can profitably be used in allocating scarce resources for data gathering and parameter estimation.

This section reports on a series of 23 sensitivity runs of the DEMAND model. First, we discuss the selection of parameters and response variables to be tested and of the sensitivity measure used. This discussion is then followed by a presentation of the results.

Design of the Analysis

In each of the 23 runs, a single parameter or set of interdependent parameters was changed by a standard 20% from its standard value or by an amount reflecting the subjective degree of confidence in the standard value.

The first twelve runs test the commodity-specific consumer behavior parameters -- own-price and income elasticities, and substitution proportions -- for farm and nonfarm consumers separately. Rice and fish were selected for these parameters since they account for the largest part of grains and meats consumption, respectively. Changes in their parameter values can, therefore, be expected, partly through substitution behavior, to be relatively more sensitive than those of other commodities.

Own-price and income elasticities of rice and fish were increased 20% for farm and nonfarm consumers, except the fish income elasticities which were decreased 20% since their standard values were already high. Also tested were the total substitution proportions of rice and fish. These parameters indicate the proportion of a decrease (increase) in rice or fish consumption which is compensated by an increase (decrease) in the consumption of substitute commodities (see Appendix D). For rice, the values were increased from the standard .95 to 1.00, while for fish they were increased 20% from the standard value of .33.

The next four runs test the effect of changing the values of parameters which cut across all commodities. These include farm and nonfarm average propensities to consume and the time constants used in exponentially averaging producer prices and income. The average propensities to consume were raised from .85 to .90 and .80 to .90 for nonfarm and farm consumers, respectively. The exponential average delay for producer prices was raised 50% from 2 years to 3 years, which would tend to further smooth the price response of farm consumption to reflect more subsistence behavior. Conversely, the income averaging delay was decreased 50% from 2 years to 1 year for both farm and nonfarm consumers to provide a quicker consumption response to income changes.

Runs 17-19 test the model's response to changes in the exogenous consumer price projections of rice, barley and wheat, respectively. Producer price projections remained unchanged. Remember that the standard run makes the policy assumption that prices of rice, barley and wheat are projected exogenously rather than by the model, i.e., KFT_i=5 and $KPAVG_i$=1 for i=1, 2 and 3 (see Table 5.2 in Chapter 5).

Finally, the last four runs test the sensitivity of the consumer price bounds of pulses, vegetables, potatoes and beef, respectively. The set of bounds for each commodity was increased 50%. That is, the maximum allowable percentage increases and decreases in domestic consumer prices for these commodities were increased 50%.

Also part of the analytical design is selection of the performance variables to be observed and the sensitivity measure to be used. In accordance with the testing of rice and fish elasticities and substitution proportions, per capita consumption of rice and fish is included among the response variables observed. Also included are consumptions of the main substitutes of these commodities, barley and beef, respectively. These four variables are those likely to be the most affected by these parameter changes. The remaining response variables all represent aggregate behavior to reflect the sensitivity of the tested parameters in the system as a whole rather than just on the variables most directly affected. These include per capita daily calorie and protein consumption, total per capita food consumption expenditures of nonfarm consumers, the agricultural trade balance and the consumer price index of livestock products, including meats, milk and eggs. The price index of this commodity group was selected since all of these prices are market-determined, subject to bounds, of course. Finally, the total of the 17 time-series tracking measures, TSS, is included to reflect the impact of parameter value changes on the model's tracking ability.

The measure used to quantify the sensitivity is the elasticity, i.e., the percentage change in the performance variable per percentage change in the parameter value. The respective changes are measured relative to the standard run solution in 1985.

Results of the Analysis

The sensitivity elasticities of each of the 10 response variables with respect to each of the 23 test parameters are given in Table 3.2. For reference purposes, base run values for 1985 are shown in the first line of the table.

Of the elasticity parameters tested, by far the most sensitive is the nonfarm own-price elasticity of rice (Run 1). Its effect on own consumption (rice) and cross consumption (barley) is larger than other rice or fish parameters. It is noteworthy that, because of the way substitution behavior is modeled (Appendix D), the cross effects of changes in the own-price elasticities of rice and fish (Runs 1 and 7) on consumption of barley and beef, respectively, are stronger than the direct effects on rice and fish consumption.

Also notice that the farm elasticities tend to be much less sensitive than the nonfarm elasticities. Two factors explain this result. First, national average consumptions and the aggregate variables that depend on them are more heavily weighted toward nonfarm consumers, where the nonfarm population increases to 71% of the total by 1985.

Table 3.2
Results of DEMAND Sensitivity Tests

Test Parameters: Definition	Symbolic Name	Percent Change	Rice Consump. PCNCON(1)	Barley Consump. PCNCON(2)	Beef Consump. PCNCON(13)	Fish Consump. PCNCON(18)	Calorie Consump. CAL(3,3)	Protein Consump. PRO(3,3)	Nonfarm Food Consump. TEXPFC	Agric. Trade Balance TVALDF	Meat Consumer Price Index CMPI	Time Series Tracking Measure TSS
			Response Variables (Values for 1985)									
			kg/cap-year	kg/cap-year	kg/cap-year	kg/cap-year	Cal/cap-day	g/cap-day	th.won/cap-yr	billion won/yr	1970=100	-
Base run with standard data values	-	-	127.9	44.3	3.0	59.7	2707	84.7	56.2	1290	143.3	3.8661
			Sensitivity Elasticity (% Change in Output/% Change in Data)									
1. Nonfarm own-price elasticity rice	ELASP(1,1)	+20	-.070	.248	0	0	.015	.018	-.009	-.039	.003	.026
2. Farm own-price elasticity rice	ELASPR(1,1)	+20	-.020	.023	0	0	-.004	0	0	-.004	0	-.001
3. Nonfarm rice substitution proportion	PSUBTG(1,2)	+ 5.25	0	.301	0	0	.049	.045	0	-.030	0	.008
4. Farm rice substitution proportion	PSUBTG(1,1)	+ 5.25	0	.043	0	0	.007	0	0	0	0	0
5. Nonfarm rice initial income elasticity	ELISAV(1)	+20	.023	.011	0	0	.013	.006	0	.004	-.003	-.004
6. Farm rice initial income elasticity	ELISVR(1)	+20	.020	0	0	0	.009	.006	0	0	0	.001
7. Nonfarm own-price elasticity fish	ELASP(18,18)	+20	0	-.011	.167	-.017	0	0	.027	-.019	.091	.004
8. Farm own-price elasticity fish	ELASPR(18,18)	+20	0	0	0	-.008	0	0	0	0	.014	.001
9. Nonfarm fish substitution proportion	PSUBTM(4,2)	+20	0	-.011	.167	0	0	0	.036	-.031	.091	0
10. Farm fish substitution proportion	PSUBTM(4,1)	+20	0	0	0	0	0	0	0	-.004	.014	.002
11. Nonfarm fish initial income elasticity	ELISAV(18)	-20	0	0	0	.017	-.002	0	.018	-.008	.010	-.056
12. Farm fish initial income elasticity	ELISVR(18)	-20	0	0	0	.067	.002	.012	0	-.050	.003	.007
13. Nonfarm average propensity to consume	APCN	+ 5.88	.013	.307	.567	.028	.069	.060	.484	-.330	.522	2.129
14. Farm average propensity to consume	APCF	+12.5	0	0	0	0	0	0	0	0	0	0
15. Producer price exponential average delay	DELP	+50	-.002	.005	0	0	-.001	0	0	.005	-.001	-.034
16. Income exponential average delay	YDEL	-50	-.003	.041	0	-.003	.004	.005	.025	-.003	.004	.095
17. Projected rice consumer price	VCPUT(1)	- 5	-.031	.181	0	0	.015	.024	.214	-.016	-.014	0
18. Projected barley consumer price	VCPUT(2)	-10	0	-.068	0	0	-.007	-.012	.036	.008	-.007	0
19. Projected wheat consumer price	VCPUT(3)	+ 5	0	.045	0	0	0	0	0	0	0	0
20. Consumer price bounds, pulses	PCPMAX/MIN(6)	+50	0	0	0	0	.004	.014	-.004	-.002	0	0
21. Consumer price bounds, vegetables	PCPMAX/MIN(7)	+50	0	0	0	0	0	0	.004	0	0	0
22. Consumer price bounds, potatoes	PCPMAX/MIN(8)	+50	-.002	-.009	0	0	.006	.002	-.007	-.022	0	0
23. Consumer price bounds, beef	PCPMAX/MIN(13)	+50	-.002	.009	.194	-.003	.002	.005	-.039	-.045	-.188	0

Secondly, the price responsiveness of farm consumers is muted by the subsistence consumption assumption where exponentially averaged past prices are used instead of current prices.

It is interesting that there is a noticeable effect on barley consumption of changes in the fish own-price elasticity and substitution proportion for nonfarm consumers (Runs 7 and 9). This is attributable to the income effect, whereby the income constraint forces barley consumption (and possibly others, as well) down to allow for the increased beef consumption.

The most sensitive parameter tested is the nonfarm average propensity to consume APCN (Run 13). Changes in this parameter essentially shift all consumptions up or down by loosening or tightening the income constraint. Notice that tuning this one parameter can be extremely effective in improving the model's historical tracking ability -- the sensitivity elasticity of TSS is 2.129, i.e., a 10% decrease in APCN would achieve a better than 20% improvement in TSS (assuming, of course, linearity in the relationship between APCN and TSS). Looking at a more disaggregated level, most of that sensitivity is in livestock products, whose tracking measure has a sensitivity elasticity of 13.395, and most of that is in fish consumption tracking, with an elasticity of 66.414. By comparison, the tracking measures of grains and rice have sensitivity elasticities of .232 and 1.181, respectively.

The farm average propensity to consume APCF, by contrast, has virtually no effect on the response variables observed (Run 14). This is due to the fact that no income constraint is built into the farm demand model. The only variable affected by a change in APCF is farm nonfood consumption expenditures, which is computed as a residual. This in turn would affect behavior in the national economy model (NECON) when DEMAND is linked to NECON.

The last seven runs test changes in parameters relating to the projection period only, i.e., after 1975. Therefore, these parameters have no effect on the model's tracking measure. Of the consumer prices projected, that of rice, naturally, is the most sensitive; rice alone accounted for 18% of nonfarm per capita food consumption expenditures in Korea in 1970.

Somewhat surprisingly, the price bounds appear relatively insensitive, those of beef showing the most sensitivity (Runs 20-23). Looking at the commodities directly involved, however, pulses, vegetables, potatoes and beef consumption have sensitivity elasticities of .105, -.002, .276 and .194, respectively, in Runs 20-23 (only beef's is shown in Table 3.2). The only aggregate variable examined in Table 3.2 which shows any appreciable sensitivity to the price bounds is the food trade balance TVALDF. This variable responds directly to the individual consumption changes (which are substantially sensitive as noted above) since domestic production is projected exogenously in these runs.

CHAPTER 4

FURTHER RESEARCH AND MODEL DEVELOPMENT

No model intended for practical use can ever be said to be complete. It must evolve with the evolving needs of its users. To accomplish this, the researchers and analysts responsible for developing, maintaining and applying DEMAND must work in close interaction with the decision makers who are the ultimate users of the product. Therefore, we can no more anticipate all future directions for model development than we can anticipate all future concerns of decision makers. Nevertheless, as a result of the development and tests conducted on DEMAND to this stage, we can identify some potentially fruitful areas for research and model development. These are discussed in terms of the model's structure and data base.

Structural Development

We noted earlier that the logical implication of DEMAND's assumption that all prices are real prices requires the all-commodities consumer price index computed by the model, which is the implicit deflator, to be identically unity, i.e., deflated by itself. (See the discussion in Chapter III deriving this conclusion.) In the standard run of DEMAND, however, its value rises to 1.129 by 1985.

This result is attributed to the fact that DEMAND projects real food prices independently of the nonfood price index. While this independence may be logically valid for nominal prices (although not necessarily economically valid), it is not valid for real prices since all prices are included in the deflator used to convert from nominal to real prices. We can see this if we write mathematically,

$$RFP = FP/PI = FP/f(FP, NP) = g(FP, NP)$$

$$RNP = NP/PI = NP/f(FP, NP) = h(FP, NP)$$

where RFP and RNP are real food and nonfood prices, respectively, FP and NP are the corresponding nominal prices, and PI = f(FP, NP) is the aggregate price index.

Therefore, one improvement that could be made in the model's structure would be to relate food and nonfood prices to each other so the aggregate price index (equation (74) in Chapter 2) remains unity. One suggested approach is to add a constraint equation to the simultaneous system (equations (13) and (14) in Chapter 2) which would insure that the aggregate price index would have a value of one, at the same time treating the nonfood price index as a dependent rather than independent variable.

We also discussed in Chapter 3 the instability of the price-nonfarm demand simultaneous equations model in the absence of exogenously imposed consumer price bounds. Unconstrained solutions tend to include unreasonably high or low (or even negative) prices, particularly for commodities with low price elasticities. Therefore, two of the policy options built into the model cannot be used since they do not include bounds on consumer prices.

While the model is quite usable and even useful in its present form as long as price bounds are imposed, a worthwhile line of development would be, first, to investigate the causes of this instability and, then, to redesign the model to remedy the situation.

Preliminary observations suggest the problem may lie in the use of linearized Cobb-Douglas consumption functions. Such linearizations can be reasonable approximations only when the perturbations are small. Thus, when solving for consumption in terms of price, as DEMAND's standard run does for rice, for example, relatively large changes in price would result in substantial inaccuracies. Indeed, the rice price in Korea has shown large annual increases in real terms -- 10.9% in 1971, 12.2% in 1972 and 16.7% in 1975. Conversely, similar inaccuracies would result when solving for price in terms of consumption, as DEMAND's standard run does for potatoes, for example, if large changes in consumption (per capita supply to be cleared) occur. Again, per capita supply of potatoes declined 18.9% and 12.6% in 1973 and 1974, respectively, and increased 57.1% in 1975.

A more basic problem, however, may lie in the Cobb-Douglas functions themselves to represent consumption behavior. The primary shortcoming of this functional form is that the price and income elasticities are generally assumed to remain constant with respect to price, income and quantity demanded, although DEMAND does not assume they are necessarily constant over time (see Chapter 2 and Appendix D). In particular, worthwhile benefits can be gained from further research into and modeling of asymmetrical consumer behavior. That is, the strength (or elasticity) of the response of consumption to price and income changes may depend on the direction, magnitude and duration of those changes.

Another shortcoming which may contribute to the instability of the price-demand simultaneous equations system is the use of inverted consumption functions as price functions. That is, it may not be realistic to solve for price by simply inverting the consumption functions and using the inverse of demand elasticities as price flexibilities.

In sum, then, a good case can be made for investigating alternatives to the Cobb-Douglas consumption functions. A state-space modeling approach may be a useful point of departure, where rates of change of unknown consumptions and prices (C_1 and P_2 in equation (13') in Chapter 2) are expressed as functions of C_1 and P_2 and of known consumptions and prices C_2 and P_1. That is, for example,

$$\frac{d}{dt} Y(t) = f(Y(t), X(t))$$

where $Y(t) = \begin{pmatrix} C_1(t) \\ P_2(t) \end{pmatrix}$ is the state vector, and $X(t) = \begin{pmatrix} C_2(t) \\ P_1(t) \\ I\ (t) \end{pmatrix}$

is the input vector with income I.

Finally, the farm demand component follows the basic assumption that the consumption behavior of farm households is essentially subsistence in character. As discussed in Chapter 2, this assumption is implied by the use in the consumption functions of exponential averages of past producer prices rather than concurrent consumer prices and also by the independence of farm and nonfarm consumption rather than their competition for available supply. While these assumptions are reasonable for the staple commodities, farm consumption of other commodities, particularly meats, milk and eggs and marine products, is decidedly not subsistence (see Table 2.3 in Chapter 2). Improvements in the farm demand component could be achieved by developing a more market-oriented model, at least for these commodities.

Data Development

The data base of any model is always subject to revision as more information becomes available for re-estimation of parameters and exogenous projections. Here, let's consider briefly three classes of DEMAND data.

The sensitivity analysis reported in Chapter 3 indicates that the demand elasticities and substitution proportions are relatively insensitive except for the commodities directly affected and major substitutes. A great deal of effort was expended in trying to estimate these parameters using time-series and cross-section data [7]. At best, the results were useful as a starting point for further tuning of the model to track consumption over the period 1970-1975. In view of the above, therefore, it would probably not be worthwhile to devote much more time to further estimation of elasticities. This would be particularly so if alternative model structures are to be considered, as suggested in the last section.

The extreme sensitivity of the nonfarm average propensity to consumer, however (see Table 3.2 in Chapter 3), does suggest the value of updating the estimation of this parameter as new data become available. It may even be worthwhile to consider making this a time-varying parameter rather than a constant.

The second set of data we discuss here is the historical time series stored in the model. These are used during the tracking period in the model in place of model solutions (in the case of prices), to compare model solutions with historical performance (in the case of per capita consumptions), and for exogenous model inputs (such as the nonfood price index and world food prices). Therefore, it is important that these series be updated each year as new data become available. Values currently stored for 1970-1975 are shown in Table E.2 in Appendix E.

Finally, DEMAND requires, as a major input, projections of import and export prices of food commodities. Periodic situation and outlook analyses should be conducted, bringing together information from IBRD, FAO, USDA, etc., to keep the projections as current as possible.

In general, we can say that sensitivity tests give guidelines on allocating scarce resources for data gathering and analysis. Those parameters which make more of a difference in model behavior should be those which receive more priority for refinement. We have seen this in regard to the sensitivity tests reported here, but it also applies to others which may (should) be conducted in the future as part of the continual process of model development and utilization.

CHAPTER 5

COMPUTER PROGRAM DOCUMENTATION

This chapter describes the computer program of DEMAND and how to implement it. DEMAND is one model of the KASM system of models. Job control input, plotting and changing data are managed by the KASM3 executive program. These operations will be referred to here as they relate to DEMAND. However, the reader is referred to KASS Special Report 13 [2] for detailed specifications of job control data input cards, the plot package and the data change routine. Variable definitions, computer program lists and sample output appear in Appendices A-C.

Machine Specifications

The following data pertain to the CDC CYBER 72 computer at the Korean Institute of Science and Technology (KIST). KASM3 is segmented for CDC loading and execution, and the data below are for the DEMAND segments only.

Machine: CDC CYBER 72

Operating System: KIST SCOPE 3.4.3 PSR 406, dated 11/17/76

Compiler: CDC FTN 4.5 + R406

Core Requirements

--DEMAND segment programs: 4454 decimal words
--DEMAND segment commons: 3986 decimal words
--KASM3 root segment commons: 2075 decimal words

CP Compile time: 21.60 seconds

CP Execution Time

-- base run (16 time cycles): 6.66 seconds
-- average per time cycle: .42 seconds

Program Structure

The KASM3 executive program calls DEMI before the time loop begins to initialize DEMAND, and it calls DEMAND each simulation time cycle, both calls taking place when the model switch KDEM > 0. Both DEMI and DEMAND call CHDATA to implement data changes when ICHDAT ≠ 0. The call sequence is shown in Figure 5.1 and subprogram definitions in Table 5.1.

All communication within DEMAND and between DEMAND and other KASM programs (except CHDATA) is through common blocks. A parameter list is used for communication with CHDATA. DEMAND writes the values of its output variables on logical unit LUPLTI each time period if plotting or special livestock tables are desired.

Program Implementation

The DEMAND model is called by PROGRAM KASM3 when KDEM > 0. KDEM is a model indicator switch read during job control data input in SUBROUTINE CONTR of KASM3. KASM3 also manages the run loop, the time loop and print and plot control switches. See KASS Special Report 13 [2] for complete descriptions of these operations and of job control input requirements.

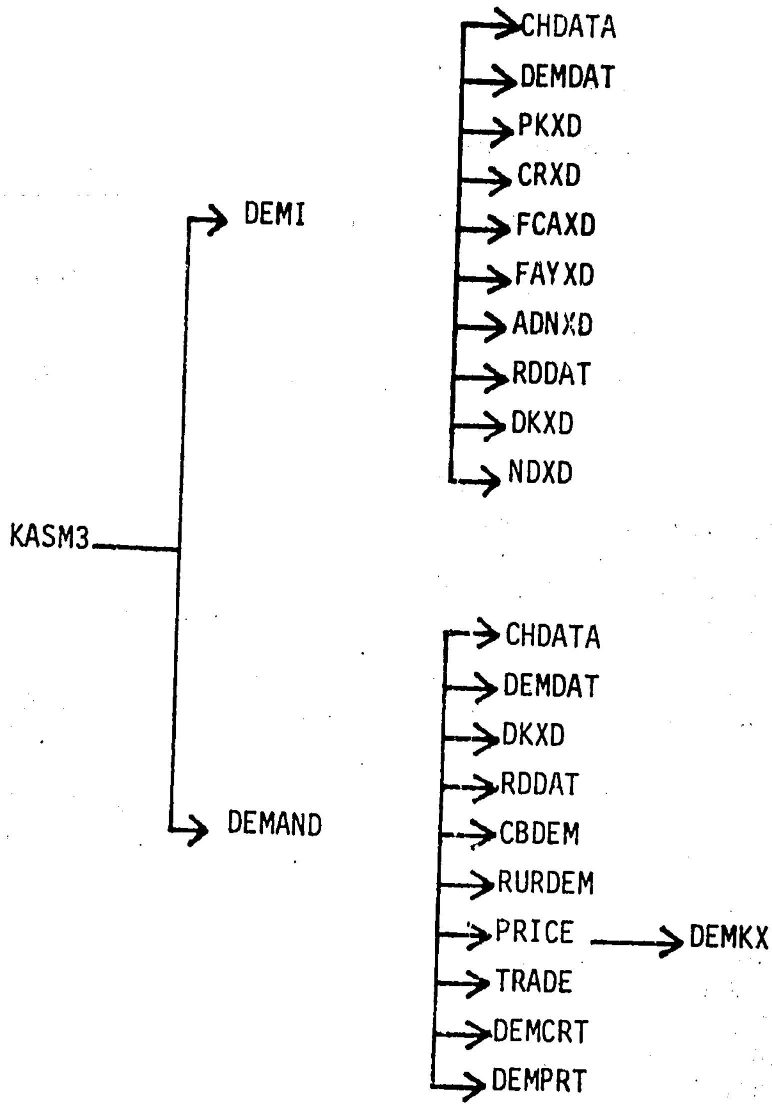

Figure 5.1

DEMAND Call Sequence

TABLE 5.1

DEMAND Subprograms

KASM3	--	executive PROGRAM of KASM, Version 3.
DEMI	--	SUBROUTINE which initializes the DEMAND model, called each run prior to time loop.
DEMAND	--	executive SUBROUTINE of the DEMAND model, called each time period.
CHDATA	--	a utility SUBROUTINE of KASM used to change data, if desired, from standard values.
DEMDAT	--	BLOCKDATA which defines standard values of data internal to DEMAND.
DKXD	--	BLOCKDATA which defines standard values of data for past time series and exogenous projections of producer prices and imports/exports.
PKXD	--	BLOCKDATA which defines standard values of data for past time series and exogenous projections of population.
CRXD	--	BLOCKDATA which defines standard values of data for past time series and exogenous projections of crop yields and input application rates.
FCAXD	--	BLOCKDATA which defines standard values of data for past time series and exogenous projections of crop areas.
FAYXD	--	BLOCKDATA which defines standard values of data for past time series and exogenous projections of livestock production and fruit tree plantings.
ADNXD	--	BLOCKDATA which defines standard values of data for past time series and exogenous projections of fisheries production and feed use of crop commodities.
RDDAT	--	BLOCKDATA which defines standard values of data for commodity-specific farm and market loss proportions and seed application rates.
NDXD	--	BLOCKDATA which defines standard values of data for part-time series and exogenous projections of nonfood price indices, farm and nonfarm per capita incomes, and foreign exchange rates.

TABLE 5.1 (Continued)

CBDEM	--	BLOCKDATA which orders DEMAND common blocks in proper sequence for plotting and changing data.
RURDEM	--	SUBROUTINE which computes farm consumption.
PRICE	--	SUBROUTINE which computes prices and nonfarm consumption.
DEMKX	--	SUBROUTINE which exogenously projects producer prices and imports/exports.
TRADE	--	SUBROUTINE which computes imports, exports and other foreign trade accounts.
DEMCRT	--	SUBROUTINE which computes DEMAND performance criteria.
DEMPRT	--	SUBROUTINE which prints DEMAND annual output tables.

Data for DEMAND may be changed permanently by updating the appropriate BLOCKDATA subprogram, DEMDAT or DKXD. This should be done annually at least to update the historical time series stored in the model. See Appendix E, Table E.2, for a list of these time series data and their locations in the program. Temporary data changes, such as for sensitivity tests or to represent alternative policy options, may be made by specifying them on data cards read by SUBROUTINE CHDATA (see [21]).

The types of policy options built into DEMAND are described in detail in Chapter 2. The present chapter provides information on how to implement those options in the computer program in the sample worksheet form and accompanying notes appearing in Table 5.2, which shows the standard values assumed for policy parameters in DEMAND.

Annual output tables for DEMAND are printed by SUBROUTINE DEMPRT whenever ITABPR $\neq$ 0 and individual table switches $KDPRT_i \neq 0$. KDPRT is dimensioned 15, so up to 15 tables can be accommodated in DEMPRT. Currently 10 tables are programmed for DEMAND:

1. Price Indices
2. Supply and Disappearance of Agricultural Commodities
3. Per Capita Consumption and Consumption Targets
4. Urban Income and Expenditures
5. Calorie and Protein Consumption Per Capita
6. Marketing Prices and Costs
7. Own-Price and Income Elasticities
8. Cross-Price Elasticities
9. Agricultural Export-Import Balance
10. Agricultural Imports and Exports

In addition, print switches 14 and 15 control detailed printing, for debugging purposes, of intermediate computations in the simultaneous equations price adjustment loop in SUBROUTINE PRICE. Switch 13 controls printing of the time-series tracking measures of goodness-of-fit.

Special note should be made here concerning the operation of DEMAND when production is projected neither endogenously (by the farm resource allocation model when the model switch KRAP > 0) nor exogenously (by the production accounting component when KRAP < 0). That is, when DEMAND is executed independently of production (KDEM > 0 and KRAP = 0), then DEMAND automatically determines all nonfarm consumption based only on exogenous projections of consumer prices; the model cannot project prices itself without production information. Therefore, data for projecting consumer prices (VCPUT in Table 5.2) must be given for <u>all</u> commoditites just as if KFT_i = 5 for all commodities i, i = 1, 2, ..., NCOM; i.e., any other values specified for KFT_i are overridden.

Finally, time-series data for DEMAND is stored in the program (see Table E.2 in Appendix E), giving the user the option of starting the simulation at any time during the tracking period. SUBROUTINE DEMI automatically initializes DEMAND at the specified initial year. The tracking period is defined as the period for which time-series data are stored, TTABI≤TIMEI≤BASEYR, where TTABI is currently 1970, TIMEI is the desired starting year, and BASEYR is the latest year for which data are stored, currently 1975. DEMAND uses actual prices for TIMEI≤T≤BASEYR, projecting the resulting consumption. If time-series tracking tests are to be conducted, the job control parameter KTRACK should be ≠ 0. Otherwise, if KTRACK = 0, DEMAND automatically reinitializes consumption to recorded values in BASEYR before starting the projection period (T>BASEYR).

TABLE 5.2

Demand-Price-Trade Model Policy Inputs Worksheet[1]:

Base Run
(run description)

Commodity	TARIFF[2] STKRT[3] KPAVG[4] KFT[5]	PWPMAX[6] PWPMIN[6] PCPMAX[7] PCPMIN[7]	Projected Variables[8]	Projection Period[9] BASEYR + 1	1981	1986	1991	2001
1. Rice	.16		VPP	120,000	120,000	120,000	120,000	120,000
	1.5		VCPUT	123,022	123,022	125,000	125,000	125,000
	1		VPCSUP					
	5		VDFCT					
2. Barley	.20		VPP	60,000	60,000	60,000	60,000	60,000
	7.0		VCPUT	65,000	65,000	65,000	65,000	65,000
	1		VPCSUP					
	5.		VDFCT					
3. Wheat[10]	.20		VPP	45,000	45,000	45,000	45,000	45,000
	2.0		VCPUT	43,881	43,881	45,000	45,000	45,000
			VPCSUP					
			VDFCT					
4. Other grains	0	5.0	VPP					
	0	0	VCPUT					
	0	.1	VPCSUP					
	3.0	.05	VDFCT	500,000	400,000	400,000	400,000	400,000
5. Fruit	0	3.0	VPP					
	0	1.65	VCPUT					
	0	4.0	VPCSUP					
	4.0	1.0	VDFCT					

Commodity	TARIFF[2] STKRT[3] KPAVG[4] KFT[5]	PWPMAX[6] PWPMIN[6] PCPMAX[7] PCPMIN[7]	Projected Variables[8]	Projection Period[9] BASEYR + 1	 1981	 1986	 1991	 2001
6. Pulses	0 0 0 3	2.8 1.0 .05 .05	VPP VCPUT VPCSUP VDFCT	50,000				
7. Vegetables	0 0 0 4	5.0 0 [illegible] .05	VPP VCPUT VPCSUP VDFCT					
8. Potatoes[11]	0 0 0 4	5.0 0 .15 .1	VPP VCPUT VPCSUP VDFCT					
9. Tobacco	0 0 1 5		VPP VCPUT VPCSUP VDFCT	700,000	700,000	700,000	700,000	700,000
10. Forage[12]	0 0 0 5		VPP VCPUT VPCSUP VDFCT	1	1	1	1	1
11. Silk[13]	0 0 1 5		VPP VCPUT VPCSUP VDFCT	1,200,000	1,200,000	1,200,000	1,200,000	1,200,000
12. Industrial crops[14]	0 0 0 4	5.0 0 .3 .1	VPP VCPUT VPCSUP VDFCT					

Table 5.2--continued

Commodity	TARIFF[2] STKRT[3] KPAVG[4] KFT[5]	PWPMAX[6] PWPMIN[6] PCPMAX[7] PCPMIN[7]	Projected Variables[8]	Projection Period[9] BASEYR + 1	1981	1986	1991	2001
13. Beef	0	1.0	VPP					
	0	.7	VCPUT					
	0	.1	VPCSUP					
	3	.05	VDFCT	0	0	0	0	0
14. Milk	0	1.0	VPP					
	0	.3	VCPUT					
	0	.1	VPCSUP					
	3	.05	VDFCT	0	0	0	0	0
15. Pork	0	5.0	VPP					
	0	.7	VCPUT					
	0	.1	VPCSUP					
	3	.1	VDFCT	-4,000	-4,500	-5,000	-5,000	-5,000
16. Chicken	0	5.0	VPP					
	0	0	VCPUT					
	0	.15	VPCSUP					
	3	.1	VDFCT	0	0	0	0	0
17. Eggs	0	5.0	VPP					
	0	0	VCPUT					
	0	.15	VPCSUP					
	3	.1	VDFCT	0	0	0	0	0
18. Fish	0	1.0	VPP					
	0	1.0	VCPUT					
	0	4.0	VPCSUP					
	4	1.0	VDFCT					
19. Residual[15]	0		VPP					
	0		VCPUT	100,000	100,000	100,000	100,000	100,000
	0		VPCSUP					
	5		VDFCT					

Table 5.2.--continued

Notes for DEMAND-Price-Trade Model: Policy Inputs Worksheet

1. All data are set in DATA statements. All are in BLOCKDATA DEMDAT, except VPP which is in BLOCKDATA DKXDC. Attached is the set of policy input data for the standard, or base, run of KASM, representing Korea's Fourth Five-Year Plan.

2. TARRIFF -- import duties as a proportion of input price.

3. STKRT -- months of consumption in stock (carry out) at the end of the rice year, October 31.

4. KPAVG -- policy switch indicating whether the producer price is specified by policy assumption (KPAVG ≠ 0) or determined by a marketing margin (KPAVG = 0). If KPAVG ≠ 0, also give values for VPP (see note 8 below).

5. KFT -- policy switch indicating the manner in which consumer prices and nonfarm consumption are determined.

KFT = 1 -- National average per capita consumption is specified by policy assumption, and the market-clearing consumer price is determined by the model. Values for VPCSUP must also be given (see note 8 below).

KFT = 2 -- Imports/exports are specified by policy assumption, and the market-clearing consumer price is determined by the model. Values for VDFCT must also be given (see note 8 below).

KFT = 3 -- Imports/exports and bounds on the consumer price are specified by policy assumption. If the market-clearing price determined by the model violates a bound, the price is set to the nearest bound and the model is resolved for consumption with imports/exports computed as a residual, ignoring the policy-specified imports/exports. Values for PWPMAX, PWPMIN, PCPMAX, PCPMIN and VDFCT must also be given (see notes 6-8 below).

KFT = 4 -- Bounds on the consumer price are specified by policy assumption, and the market-clearing price is determined by the model. If it violates a bound, the model is resolved with the consumer price set at the nearest bound to determine consumption. Values must also be given for PWPMAX, PWPMIN, PCPMAX and PCPMIN (see notes 6 and 7 below).

Table 5.2.--continued

KFT = 5 -- The consumer price is specified by policy assumption, and consumption is determined by the model. Values for VCPUT must also be given (see note 8 below).

6. PWPMAX -- the proportion of the import price above which the domestic consumer price will not be allowed to rise. A large value, such as 5.0, is used when the price policy is unrelated to the world market, i.e., when this constraint is to be ineffective.

PWPMIN -- the proportion of the export price below which the domestic consumer price will not be allowed to fall. A small value, such as 0.0, is used when the price policy is unrelated to the world market, i.e., when this constraint is to be ineffective.

Values must be given to PWPMAX and PWPMIN for each commodity for which KFT is 3 or 4 (see note 5 above). Otherwise, no value need be specified, as indicated by the dashes in the attached base run data. In each time period, the product of PWPMAX and the import price must be greater than the product of PWPMIN and the export price.

7. PCPMAX -- the maximum proportional yearly increase allowed for the domestic consumer price. A large value, such as 5.0, is used when the price policy is unrelated to domestic price movements, i.e., when this constraint is to be ineffective.

PCPMIN -- the maximum proportional yearly decrease allowed for the domestic consumer price. A large value, such as 1.0 (a 100% decrease), is used when the price policy is unrelated to domestic price movements, i.e., when this constraint is to be ineffective.

Values must be given to PCPMAX and PCPMIN for each commodity for which KFT is 3 or 4 (see note 5 above). Otherwise, no value need be specified, as indicated by the dashes in the attached base run data. PCPMAX must be greater than PCPMIN.

8. These are variables which must be given values by policy assumption for each year shown, depending on values given the policy switches KPAVG and KFT (see notes 4 and 5 above).

VPP -- producer price in 1970 constant won/MT. It must be specified for each commodity for which KPAVG $\neq$ 0.

VCPUT -- consumer price in 1970 constant won/MT. It must be specified for each commodity for which KFT = 5.

Table 5.2.--continued

VPCSUP -- national average per capita consumption in MT/person-year. It must be specified for each commodity for which KFT = 1.

VDFCT -- imports/exports (exports are negative imports) in MT/year. It must be specified for each commodity for which KFT = 2 or 3.

9. Values of the projected policy variables (see note 8 above) must be given for each year shown in the projection period. The model interpolates values on a straight line for years in between. The first year of the projection period, BASEYR + 1, is the first year following the latest year of recorded empirical values, BASEYR. The remaining years of the projection period for which values are required -- 1981, 1986, 1991 and 2001 -- are specified in the array ARG (except for producer prices VPP and imports/exports VDFCT) in a DATA statement in BLOCKDATA DEMDAT. For VPP and VDFCT, these dates are specified in the array ARGPP in BLOCKDATA DKXD. These dates may be changed if desired.

10. Wheat: producer and consumer prices are in won/MT of whole grain equivalents, not flour; consumption and imports/exports are also in MT of whole grain equivalents, not flour.

11. Potatoes: prices and quantities are for fresh weight equivalents, not grain equivalents.

12. Forage: a KASM commodity for the production models of KASM, but no human food consumption is assumed.

13. Silk: no human food consumption is assumed. Only a producer price is needed for production decisions in the production models of KASM. Therefore, KPAVG≠0 and producer prices VPP must be projected by policy assumption.

14. Industrial crops: only edible oils are considered for human consumption and, hence, consumer prices, imports/exports and per capita consumption. Producer prices are an average for all industrial crops for production decisions in the production models of KASM.

15. Residual: on the production side (i.e., for producer prices), this is taken to be edible offal from livestock products; on the consumption side, it includes, in addition to edible offal, imported sugar, coffee/tea/cocoa and vegetable oils. See Alan R. Thodey, Food and Nutrition in Korea 1965-1974, KASS Special Report II, Chapter 4.

APPENDIX A

Variable Definitions

This appendix defines the variables used in the demand price-foreign trade model of KASM. The variables are listed alphabetically, and the definitions have the following format:

```
NAME     A, B.
         Dimensioned by (C).  Located in /D/.  Defined in E.
           Referenced in F.
```

where

NAME	=	symbolic name of the variable
A	=	definition of the variable
B	=	unit of measure
C	=	meaning of dimensions, if any
D	=	name of labeled COMMON where the variable is located, or LOCAL if it is a local variable, or F.P. if it is a formal parameter
E	=	list of subroutines where the variable is defined, i.e., where it is given a value
F	=	list of subroutines where the variable is referenced, i.e., where it is used.

For some variables, some of the subroutines listed in E and F may not appear in the computer program list in Appendix B. Such subroutines belong to other KASM models or components, whereas Appendix B lists only DEMAND programs.

AE Matrix of coefficients of the price vector in the nonfarm consumption functions, MT/person-year per won/MT.
Dimensioned by (commodity, commodity). Located in (LOCAL). Defined in PRICE. Referenced in PRICE.

AINTER Intercept term vector in nonfarm consumption functions, MT/person-year.
Dimensioned by (commodity). Located in (LOCAL). Defined in PRICE. Referenced in PRICE.

AINV Linear system coefficient matrix for demand-price function (also its inverse).
Located in (LOCAL). Defined in PRICE. Referenced in PRICE.

ALPHA Income adjustment factor for initial condition consistency, no units.
Located in /DEMSVC/. Defined in DEMI, PRICE. Referenced in DEMCRT, DEMI, PRICE.

ANIMAL Index of animal sources of nutrients, no units.
Located in /CONTRC/. Defined in CONTRD. Referenced in DEMCRT, DEMPRT.

APCF Average propensity to consume for farm consumers, proportion.
Located in /DEMKC/. Defined in DEMKX. Referenced in RURDEM, FRMAC.

APCFD Data for APCF.
Located in /DKXDC/. Defined in DKXD. Referenced in DEMKX.

APCN Average propensity to consume for nonfarm consumers, proportion.
Located in /CPDEMC/. Defined in DEMDAT. Referenced in DEMAND, DEMCRT, DEMI, DEMPRT, PRICE.

AREQH Crop area required to meet human consumption from domestic production, ha.
Dimensioned by (commodity). Located in /CVDEMC/. Defined in DEMCRT. Referenced in DEMCRT.

AREQHF Crop area required to meet human and feed demand from domestic production, ha.
Dimensioned by (commodity). Located in /CVDEMC/. Defined in DEMCRT. Referenced in DEMCRT.

ARG Table function independent variable, year.
Dimensioned by (year). Located in /PPDEMC/. Defined in DEMDAT. Referenced in PRICE, TRADE.

ARGPP Table function independent variable values, year.
Dimensioned by (year). Located in /DKXDC/. Defined in DKXD. Referenced in DEMKX.

ARQHFS Crop area required to meet human and feed demand and stock changes from domestic production, ha.
Dimensioned by (commodity). Located in /CVDEMC/. Defined in DEMCRT. Referenced in DEMCRT.

BASEYR Year at which tracking period ends, year.
Located in /CONTRC/. Defined in CONTRD. Referenced in DEMCRT, DEMKX, PRICE, TRADE.

BDEM Vector of coefficients of the income constraint term of the nonfarm consumption functions, MT/person-year.
Located in (LOCAL). Defined in PRICE. Referenced in PRICE.

CAL Calorie intake, calories/person-day.
Dimensioned by (source, population group). Located in /CVDEMC/. Defined in DEMCRT. Referenced in DEMCRT, DEMPRT.

CALPC Percentage calorie intake, percent.
Dimensioned by (source, population group). Located in /CVDEMC/. Defined in DEMCRT. Referenced in DEMPRT.

CALPU Calorie content of foods, calories/MT.
Dimensioned by (commodity). Located in /CPDEMC/. Defined in DEMDAT. Referenced in DEMCRT.

CALRPC Per capita calorie requirements, calories/person-year.
Dimensioned by (population group). Located in /POPKC/. Defined in POPMIG. Referenced in DEMCRT, POPPR1.

CHGSTK Stock change, MT/year.
Dimensioned by (commodity). Located in /VDEMC/. Defined in TRADE. Referenced in DEMAND, DEMCRT, TRADE.

CMARKT Marketing margin, won/MT.
Dimensioned by (commodity). Located in /CVDEMC/. Defined in DEMCRT. Referenced in DEMCRT, DEMPRT.

CMIN Minimum farm nonfood consumption expenditures, won/person-year.
Located in (LOCAL). Defined in DEMCRT. Referenced in DEMCRT.

CMPI Consumer price index for livestock products, no units.
Located in /CVDEMC/. Defined in DEMCRT. Referenced in DEMCRT, DEMPRT.

COGPI Consumer price index for grains except rice, no units.
Located in /CVDEMC/. Defined in DEMCRT. Referenced in DEMCRT, DEMPRT.

CON Farm consumption, MT/year
Dimensioned by (region, commodity). Located in /DEMYC/. Defined in RURDEM. Referenced in FRMAC, RURDEM.

CPFI Consumer price index for food, no units.
Located in /DEMNC/. Defined in DEMCRT. Referenced in ASNEIF, DEMCRT, DEMPRT, NECPRT.

CPI Consumer price index for all commodities, no units.
Located in /CVDEMC/, Defined in DEMCRT. Referenced in DEMCRT, DEMPRT.

CPU Urban consumer price, won/MT.
Dimensioned by (commodity). Located in /DEMSVC/. Defined in PRICE. Referenced in DEMCRT, PRICE, RURDEM.

CPUL Lagged urban consumer price, won/MT.
Dimensioned by (commodity). Located in (LOCAL). Defined in PRICE. Referenced in PRICE.

CPUMAX Maximum allowable urban consumer price, won/MT.
Dimensioned by (commodity). Located in (LOCAL). Defined in PRICE. Referenced in PRICE.

CPUMIN Minimum allowable urban consumer price, won/MT.
Dimensioned by (commodity). Located in (LOCAL). Defined in PRICE. Referenced in PRICE.

CPUNF Consumer price index of nonfood commodities, no units.
Located in /NECDYC/. Defined in NECDYX, PRICN. Referenced in ASNEIF, PRICE, PRICN, RURDEM, NECPRT.

CPUO Urban consumer price in base year of price index, won/MT.
Dimensioned by (commodity). Located in /DEMSVC/. Defined in DEMI. Referenced in DEMCRT.

CRPI Consumer price index for rice, no units.
Located in /CVDEMC/. Defined in DEMCRT. Referenced in DEMPRT.

CVALD New value specified for DEMAND data variables, various units.
Dimensioned by (variable). Located in /DCHDC/. Defined in CHDATA. Referenced in CHDATA, DEMAND, DEMI.

DCPUNF Proportional change in nonfood consumer price index, proportion.
Located in (LOCAL). Defined in RURDEM. Referenced in RURDEM.

DEFCIT Surplus or deficit, MT/year.
Dimensioned by (commodity). Located in /DEMKC/. Defined in DEMKX, DEMPRT, TRADE. Referenced in DEMCRT, DEMPRT, PRICE, TRADE, YSET.

DELP Lag time for exponentially averaging producer prices, years.
Located in /CPDEMC/. Defined in DEMDAT. Referenced in RURDEM.

DEMC Coefficient relating nonfarm income elasticity to consumption level and target, no units.
Dimensioned by (commodity). Located in /DEMSVC/. Defined in DEMI. Referenced in DEMI, PRICE.

DEMI Coefficient relating farm income elasticity to consumption level and target, no units.
Dimensioned by (region, commodity). Located in /DEMSVC/. Defined in DEMI. Referenced in RURDEM.

DP Population table function interval, years.
Located in /PKXDC/. Defined in PKXD. Referenced in DEMI, POPKX, POPMIG.

DSAVD Use of savings for additional consumption won/year.
Dimensioned by (region). Located in /DEMYC/. Defined in DEMCRT, FRMAC. Referenced in FRMAC.

DSTK Desired stock level, MT.
Dimensioned by (commodity). Located in /VDEMC/. Defined in TRADE. Referenced in TRADE.

DT Simulation time increment, years.
Located in /CONTRC/. Defined in CONTRD. Referenced in PRICE, RURDEM, TRADE.

DVALDF Value of agricultural deficit, dollars/year.
Located in /VDEMC/. Defined in TRADE. Referenced in DEMPRT.

E[illegible]V Effective livestock production, proportion.
Dimensioned by (commodity). Located in /PRDDC/. Defined in CLOUT. Referenced in DEMCRT.

EFFYLD Yield less losses and seed requirements, MT/ha-year.
Dimensioned by (crop commodity). Located in /PRDDC/. Defined in CLOUT. Referenced in DEMCRT, PRDPRT.

EFRMY Farm consumption expenditures, won/person-year.
Located in /VDEMC/. Defined in RURDEM. Referenced in DEMCRT.

EINCML Lagged value of EINCOM, won/person-year.
Located in (LOCAL). Defined in PRICE. Referenced in PRICE.

EINCOM Nonfarm consumption expenditures, won/person-year.
Located in /DEMSVC/. Defined in DEMI, PRICE. Referenced in DEMCRT, DEMI, DEMPRT.

ELASI Nonfarm income elasticity, no units.
Dimensioned by (commodity). Located in /VDEMC/. Defined in PRICE. Referenced in DEMPRT, PRICE.

ELASIR Farm income elasticity, no units.
Dimensioned by (commodity). Located in /VDEMC/. Defined in RURDEM. Referenced in DEMPRT, RURDEM.

ELASP Nonfarm own-price and cross-price elasticities, no units.
Dimensioned by (commodity, commodity). Located in /VDEMC/. Defined in DEMDAT, PRICE. Referenced in DEMPRT, PRICE.

ELASPR Farm own-price and cross-price elasticities, no units.
Dimensioned by (commodity, commodity). Located in /VDEMC/. Defined in DEMDAT, RURDEM. Referenced in DEMPRT, RURDEM.

ELISAV Initial nonfarm income elasticity, no units.
Dimensioned by (commodity). Located in /ICDEMC/. Defined in DEMDAT. Referenced in DEMI.

ELISVR Initial farm income elasticity, no units.
Dimensioned by (commodity). Located in /ICDEMC/. Defined in DEMDAT. Referenced in DEMI.

EPAVG Exponential average of producer price, won/MT.
Dimensioned by (region, commodity). Located in /DEMSVC/. Defined in RURDEM. Referenced in OREDY, RURDEM.

EPAVGL Lagged value of exponential average of producer price, won/MT.
Dimensioned by (region, commodity). Located in (LOCAL). Defined in RURDEM. Referenced in RURDEM.

EPAVGO Initial exponentially averaged farm prices, won/MT.
Dimensioned by (commodity). Located in /ICDEMC/. Defined in DEMDAT. Referenced in DEMI.

FARM Index of farm population group, no units.
Located in /CONTRC/. Defined in CONTRD. Referenced in DEMCRT, DEMPRT.

FCONP Farm consumption of processed foods, won/person-year. Located in /DEMNC/. Defined in DEMCRT. Referenced in ASNEIF, DEMCRT, NECPRT.

FCONU Farm consumption of unprocessed foods, won/person-year. Located in /DEMNC/. Defined in DEMCRT. Referenced in ASNEIF, DEMCRT, NECPRT.

FDFSHD Exogenous data of fish product used for feed, MT/year. Located in /ADNXDC/. Defined in ADNXD. References in CLOUT, DEMI.

FEED Total amount of a commodity used for feed, MT/year. Dimensioned by (commodity). Located in /PRDDC/. Defined in CLOUT. Referenced in CLOUT, DEMCRT.

FEXNFC Farm nonfood consumption, won/person-year. Located in /DEMNC/. Defined in DEMCRT. Referenced in ASNEIF, DEMCRT, NECPRT.

FGIM Feed grain imports, MT/year. Located in /PRDDC/. Defined in CLOUT, TRADE. Referenced in TRADE.

FGIVAL Value of feed grain imports, won/year. Located in /VDEMC/. Defined in TRADE.

FNR Farm consumption as a proportion of nonfarm consumption, proportion. Dimensioned by (commodity). Located in /ICDEMC/. Defined in DEMDAT. Referenced in DEMI.

FPCI Farm household per capita disposable income, won/person-year. Located in /NECDYC/. Defined in ACCTG, FRMAC, NECDYX. Reference in ASNEIF, FRMAC, FYAPRT, NECPRT, RURDEM.

FRMY Farm disposable income, won/person-year. Dimensioned by (region). Located in /VDEMC/. Defined in RURDEM. Referenced in RURDEM.

FRMYA Adjusted regional farm disposable income, won/person-year. Located in (LOCAL). Defined in RURDEM. Referenced in RURDEM.

GOVSUB Government market subsidy, won/MT. Dimensioned by (commodity). Located in /CVDEMC/. Defined in DEMCRT. Referenced in DEMPRT.

ICHDAT Switch indicating whether data values are to be changed, no units.
Located in /CONTRC/. Defined in CONTR. Referenced in DEMAND, DEMI.

IGRAIN Array of indices of grain substitutes, no units.
Dimensioned by (commodity). Located in /CPDEMC/. Defined in DEMDAT. Referenced in PRICE, RURDEM.

IMAX Index in price determination algorithm denoting commodity which exceeds price bounds by largest proportion, no units.
Located in (LOCAL). Defined in PRICE. Referenced in PRICE.

IMEAT Array of indices of meat substitutes, no units.
Dimensioned by (commodity). Located in /CPDEMC/. Defined in DEMDAT. Referenced in PRICE, RURDEM.

INAPA Index array denoting method of price determination, no units.
Dimensioned by (commodity). Located in /VDEMC/. Defined in PRICE. Referenced in PRICE, RURDEM.

INCOM Exponential average of nonfarm disposable income, won/person-year.
Located in /DEMSVC/. Defined in PRICE. Referenced in PRICE.

INCOML Lagged value of INCOM, won/person-year.
Located in (LOCAL). Defined in PRICE. Referenced in PRICE.

INIV Index array used in matrix inversion algorith, no units.
Located in (LOCAL). Referenced in PRICE.

IPLT Switch indicating whether plotting is to be done, no units.
Located in /CONTRC/. Defined in CONTR. Referenced in DEMAND.

ISFLD Number of DEMAND data variables whose values are to be changed, no units.
Located in /DCHDC/. Defined in CHDATA. Referenced in CHDATA, DEMAND, DEMI.

ITABPR Switch indicating whether annual tables are to be printed, no units.
Located in /CONTRC/. Defined in CONTR. Referenced in DEMAND.

ITER Solution iterations counter in price adjustment loop, no units.
Located in /VDEMC/. Defined in PRICE. Referenced in PRICE, DEMPRT.

ITIME Time cycle counter, no units.
Located in /CONTRC/. Defined in KASM3. Referenced in DEMCRT, DEMPRT, PRICE, RURDEM, TRADE.

JIN Array of indices of commodities with market-determined prices, no units.
Located in (LOCAL). Defined in PRICE. Referenced in PRICE.

JRELD Relative location of DEMAND data variable whose value is to be changed, no units.
Dimensioned by (variable). Located in /DCHDC/. Defined in CHDATA. Referenced in CHDATA, DEMAND, DEMI.

KCHECK Switch set in price adjustment loop indicating whether some prices are out of bounds, no units.
Located in (LOCAL). Defined in PRICE. Referenced in PRICE.

KDPRT Print switches for DEMAND annual output tables, no units.
Dimensioned by (table). Located in /CONTRC/. Defined in PRTSWT. Referenced in CONTR, DEMCRT, DEMPRT, PRICE.

KFT Consumer price policy indicator, no units.
Dimensioned by (commodity). Located in /PPDEMC/. Defined in DEMDAT. Referenced in PRICE, TRADE.

KNOR Number of data points in DEMAND table functions, no units.
Located in /PPDEMC/. Defined in DEMDAT. Referenced in PRICE, TRADE.

KNP Number of data points in population table functions, no units.
Located in /PKXDC/. Defined in PKXD. Referenced in DEMI, POPKX, POPMIG.

KPAVG Producer price policy indicator, no units.
Dimensioned by (commodity). Located in /PPDEMC/. Defined in DEMDAT. Referenced in PRICE.

KPOP Population model indicator switch, no units.
Located in /CONTRC/. Defined in CONTR. Referenced in DEMCRT.

KPP Number of data points in projected producer price and imports/exports table functions, no units.
Located in /DKXDC/. Defined in DKXD. Referenced in DEMKX.

KRAP Resource allocation and production model indicator switch, no units.
Located in /CONTRC/. Defined in CONTR. Referenced in DEMAND, DEMCRT, DEMPRT, PRICE, TRADE.

KTABI Number of data points in tracking period table functions, no units.
Located in /CONTRC/. Defined in CONTRD. Referenced in DEMI, PRICE, RURDEM, TRADE.

LAB Label indicating type of price solution, no units.
Located in (LOCAL). Defined in DEMPRT, PRPCON. Referenced in DEMPRT, PRPCON.

LPR Label indicating type of price solution, no units.
Dimensioned by (type). Located in (LOCAL). Defined in DEMPRT. Referenced in DEMPRT.

LUPLTI Logical unit scratch file which stores values of variables to be plotted, no units.
Located in /CONTRC/. Defined in CONTRD. Referenced in DEMAND.

LVSTPR Switch indicating whether special livestock time-series tables are to be printed, no units.
Located in /CONTRC/. Defined in CONTRD. Referenced in DEMAND.

MM Marketing margin, proportion of producer price.
Dimensioned by (commodity). Located in /CPDEMC/. Defined in DEMDAT. Referenced in DEMCRT, PRICE.

NCOM Number of commodities, no units.
Located in /CONTRC/. Defined in CONTRD. Referenced in DEMCRT, DEMI, DEMPRT, PRICE, TRADE.

NCOMAG Number of agricultural commodities, no units.
Located in /CONTRC/. Defined in CONTRD. Referenced in DEMCRT, DEMKX, DEMI, DEMPRT, PRICE, RURDEM, TRADE.

NCROP Number of crop commodities, no units.
Located in /CONTRC/. Defined in CONTRD. Referenced in DEMCRT, DEMI.

NIN Number of commodities whose consumer prices are market determined, no units.
Located in (LOCAL). Defined in PRICE. Referenced in PRICE.

NINT1 Number of equations in demand-price simultaneous system (=NIN+1), no units.
Located in (LOCAL). Defined in PRICE. Referenced in PRICE.

NLVST Number of livestock commodities, no units.
Located in /CONTRC/. Defined in CONTRD. Referenced in DEMI.

NREGN Number of regions, no units.
Located in /CONTRC/. Defined in CONTRD. Referenced in DEMCRT, DEMKX, DEMI, PRICE, RURDEM.

NVARD Number of DEMAND plot variables, no units.
Located in /PLOTK/. Defined in PLTPKG. Referenced in DEMAND, PLTPKG.

OVALD Standard value of DEMAND data variable changed, various units.
Dimensioned by (variable). Located in /DCHDC/. Defined in CHDATA. Referenced in CHDATA, DEMAND, DEMI.

PAVG Average yearly producer price, won/MT.
Dimensioned by (region, commodity). Located in /DEMKC/. Defined in DEMKX, PRICE. Referenced in RURDEM.

PAVGA Adjusted producer price, won/MT.
Located in (LOCAL). Defined in RURDEM. Referenced in RURDEM.

PCAL Percentage of calorie requirements consumed, percent.
Dimensioned by (population group). Located in /CVDEMC/. Defined in DEMCRT.

PCCON Per capita farm consumption, MT/person-year.
Dimensioned by (region, commodity). Located in /DEMSVC/. Defined in DEMI, RURDEM. Referenced in DEMI, RURDEM.

PCCONF Farm per capita consumption, MT/person-year.
Dimensioned by (commodity). Located in /VDEMC/. Defined in RURDEM. Referenced in DEMCRT, DEMPRT.

PCCONT Nonfarm consumption target, MT/person-year.
Dimensioned by (commodity). Located in /PPDEMC/. Defined in DEMDAT. Referenced in DEMI, DEMPRT, PRICE.

PCCONU Nonfarm per capita consumption, MT/person-year.
Dimensioned by (commodity). Located in /DEMSVC/. Defined in DEMI, PRICE. Referenced in DEMCRT, DEMI, DEMPRT, PRICE.

PCMEAN Mean of historical per capita consumption time series, MT/person-year.
Dimensioned by (commodity). Located in /TSTDC/. Defined in DEMCRT. Referenced in DEMCRT.

PCMIN Minimum farm nonfood consumption as a proportion of previous period's nonfood consumption, proportion.
Located in /CPDEMC/. Defined in DEMDAT. Referenced in DEMCRT.

PCNCON National average per capita consumption, MT/person-year. Dimensioned by (commodity). Located in /CVDEMC/. Defined in DEMCRT. Referenced in DEMCRT, DEMPRT.

PCONVG Grains conversion factor, proportion. Dimensioned by (commodity). Located in /CPDEMC/. Defined in DEMDAT. Referenced in PRICE, RURDEM.

PCPMAX Maximum allowable proportional increase in consumer price, proportion/year. Dimensioned by (commodity). Located in /PPDEMC/. Defined in DEMDAT. Referenced in PRICE.

PCPMIN Maximum allowable proportional decrease in consumer price, proportion/year. Dimensioned by (commodity). Located in /PPDEMC/. Defined in DEMDAT. Referenced in PRICE.

PCRT Farm consumption target, MT/person-year. Dimensioned by (commodity). Located in /PPDEMC/. Defined in DEMDAT. Referenced in DEMI, DEMPRT, RURDEM.

PCRT1 Farm consumption target for rice, MT/person-year. Located in /PPDEMC/. Defined in DEMDAT. Referenced in DEMPRT, RURDEM.

PCT1 Rice consumption target, MT/person-year. Located in (LOCAL). Defined in PRICE, RURDEM. Referenced in PRICE, RURDEM.

PCUT1 Nonfarm consumption target for rice, MT/person-year. Located in /PPDEMC/. Defined in DEMDAT. Referenced in DEMPRT, PRICE.

PEOLV Edible offal proportion of livestock product, proportion. Dimensioned by (livestock commodity). Located in /ADNXDC/. Defined in ADNXD. Referenced in DEMI, CLOUT.

PFDIMX Feed grains import price, won/MT. Located in /DEMKC/. Defined in DEMCRT. Referenced in OREDY.

PFLOSS Total loss rate on domestic production, proportion. Dimensioned by (commodity). Located in /RDDATC/. Defined in RDDAT. Referenced in ASET, CLOUT, DEMI, FRMAC, OREDY, PRODLV.

PFU — Proportion of unprocessed food consumption, proportion.
Dimensioned by (population group, commodity). Located in /CPDEMC/. Defined in DEMDAT. Referenced in DEMCRT.

PILOSS — Loss rate on imported commodities, proportion.
Dimensioned by (commodity). Located in /CPDEMC/. Defined in DEMDAT. Referenced in DEMI, TRADE.

PLANT — Index of plant sources of nutrients, no units.
Located in /CONTRC/. Defined in CONTRD. Referenced in DEMCRT, DEMPRT.

PLOSTD — Processing loss on tobacco consumed domestically, proportion.
Located in /CPDEMC/. Defined in DEMDAT. Referenced in DEMCRT, DEMI, TRADE.

POP — Population, persons.
Dimensioned by (location). Located in /POPKC/. Defined in POPKX, POPMIG. Referenced in DEMCRT, DEMPRT, PRICE.

POPL — Lagged value of POP.
Located in /POPKC/. Defined in POPKX, POPMIG. Referenced in DEMCRT, PRICE, RURDEM.

POPR — Farm population, persons.
Dimensioned by (region). Located in /POPKC/. Defined in POPKX, POPMIG. Referenced in DEMCRT, RURDEM.

PPRO — Percentage of protein requirements consumed, percent.
Dimensioned by (population group). Located in /CVDEMC/. Defined in DEMCRT.

PRFEXP — Export profits, won/year.
Dimensioned by (commodity). Located in /VDEMC/. Defined in TRADE. Referenced in DEMPRT, TRADE.

PRFIMP — Import profits, won/year.
Dimensioned by (commodity). Located in /VDEMC/. Defined in TRADE. Referenced in DEMPRT, TRADE.

PRFRMY — Regional proportion of farm household disposable income, proportion.
Dimensioned by (region). Located in /NECDYC/. Defined in FRMAC. Referenced in RURDEM.

PRO — Protein consumption, grams/person-day.
Dimensioned by (source, population). Located in /CVDEMC/. Defined in DEMCRT. Referenced in DEMCRT, DEMPRT.

PROPC Percentage contribution to protein consumption, percent. Dimensioned by (source, population). Located in /CVDEMC/. Defined in DEMCRT. Referenced in DEMPRT.

PRORPC Per capita protein requirements, grams/person-year. Dimensioned by (population group). Located in /POPKC/. Defined in POPMIG. Referenced in DEMCRT, POPPR2.

PROTPU Protein content of foods, grams/MT. Dimensioned by (commodity). Located in /CPDEMC/. Defined in DEMDAT. Referenced in DEMCRT.

PSUBCG Commodity-specific grain substitution proportions, proportion. Dimensioned by (commodity, commodity). Located in /CPDEMC/. Defined in DEMDAT. Referenced in PRICE, RURDEM.

PSUBCM Commodity-specific meat substitution proportions, proportion. Dimensioned by (commodity, commodity). Located in /CPDEMC/. Defined in DEMDAT. Referenced in PRICE, RURDEM.

PSUBTG Total grain substitution proportions, proportion. Dimensioned by (commodity). Located in /CPDEMC/. Defined in DEMDAT. Referenced in PRICE, RURDEM.

PSUBTM Total meat substitution proportions, proportion. Dimensioned by (commodity). Located in /CPDEMC/. Defined in DEMDAT. Referenced in PRICE, RURDEM.

PWHTIC Industrial consumption of wheat, proportion. Located in /CPDEMC/. Defined in DEMDAT. Referenced in DEMCRT, DEMI, TRADE.

PWLDD World food prices, \$/MT. Dimensioned by (commodity). Located in /VDEMC/. Defined in TRADE. Referenced in TRADE.

PWLDEX Export price of food, won/MT. Dimensioned by (commodity). Located in /VDEMC/. Defined in TRADE. Referenced in DEMPRT, PRICE, TRADE.

PWLDIM Import price of food, won/MT. Dimensioned by (commodity). Located in /VDEMC/. Defined in TRADE. Referenced in DEMCRT, DEMPRT, PRICE, TRADE.

PWPMAX Maximum allowable ratio of domestic consumer price to import price, proportion. Dimensioned by (commodity). Located in /PPDEMC/. Defined in DEMDAT. Referenced in PRICE.

PWPMIN Minimum allowable ratio of domestic consumer price to export price, proportion.
Dimensioned by (commodity). Located in /PPDEMC/. Defined in DEMDAT. Referenced in PRICE.

Q Nonfarm consumption, MT/year.
Dimensioned by (commodity). Located in /DEMSVC/. Defined in PRICE. Referenced in DEMCRT, DEMPRT, PRICE, TRADE.

QREQH Domestic crop production required for self-sufficiency in human consumption, MT/year.
Dimensioned by (commodity). Located in /CVDEMC/. Defined in DEMCRT.

QREQHF Domestic crop production required for self-sufficiency in human and animal consumption, MT/year.
Dimensioned by (commodity). Located in /CVDEMC/. Defined in DEMCRT.

QRQHFS Domestic crop production required for self-sufficiency in human and animal consumption and stock changes, MT/year.
Dimensioned by (commodity). Located in /CVDEMC/. Defined in DEMCRT.

QRQHLV Domestic livestock production required for self-sufficiency in human consumption, MT/year.
Dimensioned by (commodity). Located in /CVDEMC/. Defined in DEMCRT.

RATIOI Proportional change in farm income, proportion.
Located in (LOCAL). Defined in RURDEM. Referenced in RURDEM.

RDEM Farm consumption, MT/year.
Dimensioned by (commodity). Located in /VDEMC/. Defined in RURDEM. Referenced in DEMCRT, DEMPRT, PRICE, RURDEM, TRADE.

RFRMY Exponential average of farm disposable income, won/person-year.
Dimensioned by (region). Located in /DEMSVC/. Defined in RURDEM. Referenced in RURDEM.

RFRMYL Lagged value of RFRMY.
Located in (LOCAL). Defined in RURDEM. Referenced in RURDEM.

RFTEX Food proportion of nonfarm consumption expenditures, proportion.
Located in /CVDEMC/. Defined in DEMCRT. Referenced in DEMPRT.

RTHS Vector of independent variables (right-hand side) in demand-simultaneous equations system, various units.
Dimensioned by (equation). Located in (LOCAL). Defined in PRICE. Referenced in PRICE.

S1 Income constraint coefficient for nonfarm consumption, proportion.
Located in /DEMSVC/. Defined in DEMI, PRICE. Referenced in DEMPRT, PRICE.

SAVUPC Nonfarm savings per capita, won/person-year.
Located in /CVDEMC/. Defined in DEMCRT. Referenced in DEMPRT.

SDPHA Seed input, MT/ha-year.
Dimensioned by (region, crop). Located in /RDDATC/. Defined in RDDAT. Referenced in CROPAC, CLOUT, DEMI, OREDY.

SELFSU Self-sufficiency of domestic production for human food and animal feed, proportion.
Dimensioned by (commodity). Located in /CVDEMC/. Defined in DEMCRT. Referenced in DEMPRT.

STK Commodity stock levels, MT.
Dimensioned by (commodity). Located in /DEMSVC/. Defined in TRADE. Referenced in TRADE.

STKO Initial commodity stock levels, MT.
Dimensioned by (commodity). Located in /ICDEMC/. Defined in DEMDAT. Referenced in TRADE.

STKRT Months of consumption desired in stock, months.
Dimensioned by (commodity). Located in /PPDEMC/. Defined in DEMDAT. Referenced in TRADE.

SUPPLY Total supply for human consumption, MT/year.
Dimensioned by (commodity). Located in (LOCAL). Defined in DEMI. Referenced in DEMI.

T Simulated time, year.
Located in /CONTRC/. Defined in KASM3. Referenced in DEMI, DEMCRT, DEMPRT, PRICE, RURDEM, TRADE, DEMKX.

TARH Total crop area required to meet human consumption from domestic production, ha.
Located in /CVDEMC/. Defined in DEMCRT. Referenced in DEMCRT.

TARHF Total crop area required to meet human and feed demand from domestic production, ha.
Located in /CVDEMC/. Defined in DEMCRT. Referenced in DEMCRT.

TARHFS Total crop area required to meet human and feed demand and stock changes from domestic production, ha.
Located in /CVDEMC/. Defined in DEMCRT. Referenced in DEMCRT.

TARIFF Tariffs on food imports, proportion.
Dimensioned by (commodity). Located in /PPDEMC/. Defined in DEMDAT. Referenced in TRADE.

TARREV Tariff revenues on food imports, won/year.
Dimensioned by (commodity). Located in /VDEMC/. Defined in TRADE. Referenced in DEMPRT, TRADE.

TDEM National food demand, MT/year.
Dimensioned by (commodity). Located in /DEMSVC/. Defined in DEMCRT. Referenced in DEMCRT, TRADE.

TDEML Exponential average of national food demand, MT/year.
Dimensioned by (commodity). Located in /DEMSVC/. Defined in TRADE. Referenced in TRADE.

TDISAP Total utilization, MT/year.
Dimensioned by (commodity). Located in /CVDEMC/. Defined in DEMCRT. Referenced in DEMPRT.

TDSUP Total domestic supply for human consumption, MT/year.
Dimensioned by (commodity). Located in /PRDDC/. Defined in CLOUT, TRADE. Referenced in CLOUT, PRICE, TRADE.

TEXNF Total nonfarm expenditure on nonfood consumption, won/year.
Located in /CVDEMC/. Defined in DEMCRT. Referenced in DEMCRT, DEMPRT.

TEXNFC Total per capita nonfarm expenditure on nonfood consumption, won/person-year.
Located in /DEMNC/. Defined in DEMCRT. Referenced in ASNEIF, DEMPRT, NECPRT.

TEXPF Total nonfarm expenditure on food consumption, won/year.
Located in /CVDEMC/. Defined in DEMCRT. Referenced in DEMCRT, DEMPRT.

TEXPFC Nonfarm food consumption expenditures, won/person-year.
Located in /CVDEMC/. Defined in DEMCRT. Referenced in DEMPRT.

TEXPT Total nonfarm consumption expenditures, won/year.
Located in /CVDEMC/. Defined in DEMCRT. Referenced in DEMCRT, DEMPRT.

TEXPTC Total per capita nonfarm consumption expenditures, won/person-year.
Located in /CVDEMC/. Defined in DEMCRT. Referenced in DEMPRT.

TIMEI Initial year of simulation, year.
Located in /CONTRC/. Defined in CONTR. Referenced in DEMI, TRADE.

TOTHER Total utilization other than for human consumption, MT/year.
Dimensioned by (commodity). Located in /CVDEMC/. Defined in DEMCRT. Referenced in DEMCRT, DEMPRT.

TOTPOP Index of total population, no units.
Located in /CONTRC/. Defined in CONTRD. Referenced in DEMCRT, DEMPRT.

TOTSOR Index of total nutrient sources, no units.
Located in /CONTRC/. Defined in CONTRD. Referenced in DEMCRT, DEMPRT.

TOUTPT Total domestic production, MT/year.
Dimensioned by (commodity). Located in /PRDDNC/. Defined in CLOUT. Referenced in ASNEIF, CLOUT, DEMCRT, PRDPRT.

TPAVG National average agricultural producer prices, won/MT.
Dimensioned by (commodity). Located in /DEMKC/. Defined in DEMKX, PRICE. Referenced in ASNEIF, DEMCRT, DEMKX, DEMPRT, PRICE.

TPOP Total population, persons.
Located in /POPKC/. Defined in POPKX, POPMIG. Referenced in DEMCRT, PRICE, TRADE.

TRDPRF Profits on agricultural foreign trade, won/year.
Located in /VDEMC/. Defined in TRADE. Referenced in PRTRAD, TRADE.

TRPOP Total farm population, persons.
Located in /POPKC/. Defined in POPKX, POPMIG. Referenced in DEMCRT, FRMAC, POPKX, POPMIG, RURDEM.

TSTD Time-series tracking measure of goodness-of-fit, normalized sum of squared errors, no units.
Dimensioned by (commodity). Located in /TSTDC/. Defined in DEMCRT. Referenced in DEMCRT.

TSTDG Sum of TSTD for grains, no units.
Located in /TSTDC/. Defined in DEMCRT. Referenced in DEMCRT.

TSTDM Sum of TSTD for livestock and fisheries products, no units.
Located in /TSTDC/. Defined in DEMCRT. Referenced in DEMCRT.

TSTDO Sum of TSTD for fruits and vegetables, no units.
Located in /TSTDC/. Defined in DEMCRT. Referenced in DEMCRT.

TSTDT Sum of TSTD for all commodities, no units.
Located in /TSTDC/. Defined in DEMCRT. Referenced in DEMCRT.

TTABI Initial year of tracking period table functions, year.
Located in /CONTRC/. Defined in CONTRD. Referenced in DEMI, PRICE, RURDEM, TRADE.

TTAREV Tariff revenues from agricultural imports, won/year.
Located in /DEMNC/. Defined in TRADE. Referenced in TRADE, ASNEIF, NECPRT.

TVALDF Agricultural trade balance, won/year.
Located in /VDEMC/. Defined in TRADE. Referenced in PRCSD, PRTRAD, SENS, TRADE, VARAC.

UCONP Nonfarm consumption of processed foods, won/person-year.
Located in /DEMNC/. Defined in DEMCRT. Referenced in ASNEIF, NECPRT, DEMCRT.

UCONU Nonfarm consumption of unprocessed foods, won/person-year.
Located in /DEMNC/. Defined in DEMCRT. Referenced in ASNEIF, NECPRT, DEMCRT.

UPCI Nonfarm household per capita disposable income, won/person-year.
Located in /NECDYC/. Defined in ACCTG, NECDYX. Referenced in ASNEIF, DEMCRT, FRMAC, FYAPRT, NECPRT, PRICE.

URBAN Index of nonfarm population group, no units.
Located in /CONTRC/. Defined in CONTRD. Referenced in DEMCRT, DEMPRT.

VALDEF Value of agricultural imports/exports, won/year.
Dimensioned by (commodity). Located in /VDEMC/. Defined in DEMPRT, TRADE. Referenced in DEMCRT, DEMPRT.

VALDF1 Total value of agricultural exports, won/year.
Located in /DEMNC/. Defined in TRADE. Referenced in ASNEIF, DEMPRT, NECPRT, TRADE.

VALDF2 Total value of agricultural imports, won/year.
Located in /DEMNC/. Defined in TRADE. Referenced in ASNEIF, DEMPRT, NECPRT, TRADE.

VAREA Historical time series of crop area, ha.
Dimensioned by (year, region, crop commodity). Located in /FCAXDC/. Defined in FCAXD. Referenced in CLOUT, DEMI, FRACAX, YINTAL, YSET.

VCP Historical time series of nonfood price index, no units.
Dimensioned by (year). Located in /NDXDYC/. Defined in NDXD. Referenced in DEMI, NECDYX, PRICN.

VCPU Table function values of historical consumer food prices, won/MT.
Dimensioned by (year, commodity). Located in /ICDEMC/. Defined in DEMDAT. Referenced in DEMI, PRICE.

VCPUT Table function values of projected consumer prices, won/MT.
Dimensioned by (year, commodity). Located in /PPDEMC/. Defined in DEMDAT. Referenced in PRICE.

VCSTK Table function values of historical agricultural stock changes, MT/year.
Dimensioned by (year, commodity). Located in /ICDEMC/. Defined in DEMDAT. Referenced in DEMI, TRADE.

VDFC Historical time series of agricultural imports/exports, MT/year.
Dimensioned by (year, commodity). Located in /DKXDC/. Defined in DKXD. Referenced in DEMI, DEMKX.

VDFCT Table function values of projected agricultural imports/exports, MT/year.
Dimensioned by (year, commodity). Located in /DKXDC/. Defined in DKXD. Referenced in DEMKX, TRADE.

VECTOR Variables to be plotted, various units.
Dimensioned by (variable). Located in (LOCAL). Defined in DEMAND. Referenced in DEMAND.

VEFRMY Historical time series of farm consumption expenditures, won/person-year.
Dimensioned by (year). Located in /ICDEMC/. Defined in DEMDAT. Referenced in RURDEM.

VEINCM Historical time series of nonfarm consumption expenditures, won/person-year.
Dimensioned by (year). Located in /ICDEMC/. Defined in DEMDAT. Referenced in DEMI.

VFEED Historical time series of feed use of crop production, MT/year. Dimensioned by (year, crop commodity). Located in /ADNXDC/. Defined in ADNXD. Referenced in CLOUT, DEMI.

VFPIX Table function values of historical ratio of consumer to farm farm price indices, proportion. Dimensioned by (year). Located in /ICDEMC/. Defined in DEMDAT. Referenced in RURDEM.

VFSH Historical time series of fisheries production, MT/year. Dimensioned by (year). Located in /ADNXDC/. Defined in ADNXD. Referenced in CLOUT, DEMI.

VLIVST Historical time series of livestock production, MT/year. Dimensioned by (year, livestock commodity). Located in /FAYXDC/. Defined in FAYXD. Referenced in DEMI, FRAAYX, SET.

VMAX Temporary storage for maximum deviation of a consumer price from a bound, won/MT. Located in (LOCAL). Defined in PRICE. Referenced in PRICE.

VPCSUP Table function values of projected per capita consumption of food commodities, MT/person-year. Dimensioned by (year, commodity). Located in /PPDEMC/. Defined in DEMDAT. Referenced in TRADE.

VPCUFD Historical time series of food proportion of nonfarm consumption expenditures, proportion. Dimensioned by (year). Located in /ICDEMC/. Defined in DEMDAT. Referenced in DEMI, PRICE.

VPICF Historical time series of edible proportion of industrial crops production, proportion. Dimensioned by (year). Located in /ICDEMC/. Defined in DEMDAT. Referenced in DEMCRT, DEMI, TRADE.

VPOP Table function values of population, persons. Dimensioned by (year, population group). Located in /PKXDC/. Defined in PKXD. Referenced in DEMI, POPKX, POPMIG.

VPP Table function values of projected producer price, won/MT. Dimensioned by (year, commodity). Located in /DKXDC/. Defined in DKXD. Referenced in DEMKX.

VPRICE Table function values of historical producer prices, won/MT. Dimensioned by (year, commodity). Located in /DKXDC/. Defined in DKXD. Referenced in DEMI, DEMKX.

VPSUP Historical time series of per capita consumption, MT/person-year. Dimensioned by (year, commodity). Located in /ICDEMC/. Defined in DEMDAT. Referenced in DEMCRT, PRICE.

VPWI Historical time series of import price, dollars/MT. Dimensioned by (year, commodity). Located in /ICDEMC/. Defined in DEMDAT. Referenced in TRADE.

VPWLDI Table function values of projected agricultural import prices, $/MT. Dimensioned by (year, commodity). Located in /PPDEMC/. Defined in DEMDAT. Referenced in TRADE.

VPWLDX Table function values of projected agricultural export prices, $/MT. Dimensioned by (year, commodity). Located in /PPDEMC/. Defined in DEMDAT. Referenced in TRADE.

VPWX Historical time series of export price, dollars/MT. Dimensioned by (year, commodity). Located in /ICDEMC/. Defined in DEMDAT. Referenced in TRADE.

VYIELD Historical time series of crop yields, MT/ha-year. Dimensioned by (year, region, crop commodity). Located in /CRXDC/. Defined in CRXD. Referenced in CHGI, CHGRX, CLOUT, DEMI.

W Consumer price index weights, no units. Dimensioned by (commodity). Located in /CPDEMC/. Defined in DEMDAT. Referenced in DEMCRT.

WOND Foreign exchange rate, won/$. Located in /NECDYC/. Defined in FTPD, NECDYX. Referenced in DEMPRT, NECPRT, TRADE.

WPFI World food price index, no units. Located in /DEMNC/. Defined in DEMCRT. Referenced in ASNEIF, DEMCRT, DEMPRT, NECPRT.

WWP World food price index weights, no units. Dimensioned by (commodity). Located in /DEMSVC/. Defined in DEMCRT. Referenced in DEMCRT.

W1 Sum of price index weights of all commodities, no units. Located in (LOCAL). Defined in DEMCRT. Referenced in DEMCRT.

W2 Sum of price index weights of all food commodities, no units. Located in (LOCAL). Defined in DEMCRT. Referenced in DEMCRT.

W3 Sum of price index weights of grains other than rice, no units.
Located in (LOCAL). Defined in DEMCRT. Referenced in DEMCRT.

W4 Sum of price index weights of meat commodities, no units.
Located in (LOCAL). Defined in DEMCRT. Referenced in DEMCRT.

W5 Agricultural trade volume in initial year of simulation, won/year.
Located in (LOCAL). Defined in DEMCRT. Referenced in DEMCRT.

XDIFF Nonfarm savings, won/year.
Located in /CVDEMC/. Defined in DEMCRT. Referenced in DEMCRT, DEMPRT.

YDEL Lag time for exponential averaging income, years.
Located in /CPDEMC/. Defined in DEMDAT. Referenced in PRICE, RURDEM.

YEAR Simulated time, year.
Located in /CONTRC/. Defined in KASM3. Referenced in DEMPRT, PRICE, RURDEM, TRADE.

ZZ Total domestic supply, MT/year.
Dimensioned by (commodity). Located in /CVDEMC/. Defined in DEMCRT. Referenced in DEMCRT, DEMPRT.

APPENDIX B

DEMAND COMPUTER PROGRAM

This appendix contains FORTRAN listings of the computer program program of the demand-price-foreign trade model (DEMAND). Within each of the first two sections of this appendix, programs are listed alphabetically following the executive program. The first section begins with the executive routine DEMAND and includes programs particular to the DEMAND model. The second section begins with the overall executive program KASM3 and includes "overhead" routines of the KASM3 superstructure into which all the KASM models fit. Finally, the third section lists alphabetically the utility routines used by DEMAND.

DEMAND Routines

```
THIS UPDATE REQUIRED  34100B WORDS OF CORE.
      SUBROUTINE DEMAND
C
C DEMAND - PRICE - TRADE MODEL
C
C DECLARATIONS
C
      COMMON /CONTRC/ ANIMAL, BASEYR, DT, FARM, ITIME, KTRACK, KTABI,
     1                IRUN, ITTY, ICHDAT, ISENPR, IPLT, ITABPR,
     1                LVSTPR, MODE, MODNAM(9),
     1                KPOP, KCHG, KRAP, KDEM, KNEC, LINK, JPER,
     1                LUIN1, LUIN2, LUOUT, LUPLOT(3), LURLP1,
     2                LUPGEN, LUPOP,
     2                KPPRT(15), KCPRT(15), KFPRT(15), KAPRT(15),
     2                KDPRT(15), KYPRT(15), KNPRT(15),
     2                NCC, NCOM, NCOMAG, NCROP, NLVST, NNC, NREGN,
     3                NRUN, NS, NAME(20), PLANT,
     4                T, TIMEI, TIMEF, TTABI, YEAR,
     5                TOTSOR, TOTPOP, URBAN
      INTEGER ANIMAL, FARM, PLANT, URBAN, TOTPOP, TOTSOR
      EQUIVALENCE (LUPLT1,LUPLOT(1)),(LUPLT2,LUPLOT(2)),
     1            (LUPLT3,LUPLOT(3))
      COMMON /CPDEMC/ APCN, DELP, IGRAIN(5), IMEAT(5),
     2                MM(20), PILOSS(20), PLOSTD, PSUBCG(5,5,2), PCMIN,
     3                PSUFCM(5,5,2), PSUBTG(5,2), PSUBTM(5,2),PFU( 2,20),
     4                 PCONVG(5), PWHTIC, YDEL, CALPU(20), PROTPU( 20),
     5                W(20)
      REAL MM
      COMMON /VDEMC/ CHGSTK(20), DSTK(20), DVALDF, EFRMY,
     1               ELASP(20,20), ELASPR(20,20),
     1               ELASI(20), ELASIR(20), FRMY(1), INAPA(20),
     1               FGIVAL, ITER, PCCONF(20), TARREV(20),
     2               PRFEXP(20), PRFIMP(20), PWLDD(20), PWLDEX(20) ,
     3               PWLDIM(20), RDEM(20), TRDPRF, TVALDF, VALDEF( 20)
      COMMON /PLOTK/ NVARP, NVARC, NVARF, NVARA, NVARD, NVARY, NVARN,
     1          NVARK,NVAR,KMSW(7)
      COMMON /DCHDC/ JRELD(100), OVALD(100), CVALD(100), ISFLD
C
      DIMENSION VECTOR(1)
      EQUIVALENCE (VECTOR(1),CHGSTK(1))
C
C EXECUTION
C
      LINK = 3
      IF(ICHDAT.NE.0) CALL CHDATA(APCN,JRELD,OVALD,CVALD,ISFLD,7)
C
C    FARM DEMAND
      CALL RURDEM
C
C    STOCK CHANGE, WORLD PRICES AND CONSUMPTION POLICIES
      CALL TRADEI
C
C    NONFARM DEMAND AND PRICES
      CALL PRICE
C
C    FOREIGN TRADE
      IF(KRAP .NE. 0) CALL TRADEC
C
C    OUTPUTS AND PERFORMANCE CRITERIA
      CALL DEMCRT
C
C    PRINT TABLES
      IF(ITABPR .NE. 0) CALL DEMPRT
C
C    RECORD DEMAND PLOT VALUES
      IF(IPLT.NE.0 .OR. LVSTPR.NE.0) WRITE(LUPLT1) (VECTOR(I),I=1, NVARD)
C
C
      RETURN
      END
```

```
      BLOCKDATA ADNXD
C
C DECLARATIONS
C
      COMMON /ADNXDC/ ARGFSH(5), KFSH, VFSHT(5), VFSH(10), PEOLV(5 ),
     1                TORLI, ARGFD(5), KFD, VFEED(10,12),
     2                VFEEDT(5,12), FDFSHD
C
C DATA
C
      DATA TORLI / 4090444. /
C
      DATA PEOLV / .323, 0. , .166, .075, 0. /
C
      DATA FDFSHD / 5000. /
C
C              1970        1971        1972        1973        1974
C              1975        1976        1977        1978        1979
C
      DATA VFSH /
     1       935462. , 1073733. , 1343569. , 1686485. , 2026221. ,
     2      5*2134979. /
C
C
      DATA KFSH / 5 /
C
      DATA ARGFSH /
     1        1975. ,     1981. ,     1986. ,     1991. ,     2001. /
C
      DATA VFSHT /
     1     2500000. , 3562000. , 5000000. , 6000000. , 7000000. /
C
      DATA  KFD /5/
C
      DATA ARGFD /
     1        1976. ,     1981. ,     1986. ,     1991. ,     2001. /
C
C
      DATA VFEEDT /
     1          5*0. ,
     2       75000. ,  100000. ,  100000. ,  100000. ,  100000. ,
     3       50000. ,   75000. ,   75000. ,   75000. ,   75000. ,
     4      450000. ,  500000. ,  500000. ,  500000. ,  500000. ,
     5          5*0. ,
     6       30000. ,   50000. ,   50000. ,   50000. ,   50000. ,
     7          5*0. ,
     8      500000. ,  500000. ,  500000. ,  500000. ,  500000. ,
     *         20*0. /
C
C
C              1970        1971        1972        1973        1974
C              1975        1976        1977        1978        1979
C
      DATA VFEED /
     1         10*0. ,
     2      115000. ,  155000. ,  120000. ,   68000. ,   74000. ,
     2     5*60000. ,
     3       45000. ,   46000. ,   43000. ,   37000. ,   22000. ,
     3     5*27000. ,
     4      262000. ,  354000. ,  413000. ,  426000. ,  413000. ,
     4    5*418000. ,
     5         10*0. ,
     6       18000. ,   27000. ,   24000. ,   34000. ,   32000. ,
     6     5*46000. ,
     7         10*0. ,
     8      382000. ,  346000. ,  251000. ,  433000. ,  495000. ,
     8    5*495000. ,
     *         40*0. /
C
C
      END
```

```
      BLOCKDATA CBDEM
      COMMON /CPDEMC/ APCN, DELP, IGRAIN(5), IMEAT(5),
     2                MM(20), PILOSS(20), PLOSTD, PSUBCG(5,5,2), P CMIN,
     3                PSUBCM(5,5,2), PSUBTG(5,2), PSUBTM(5,2),PFU( 2,20),
     4                 PCONVG(5), PWHTIC, YDEL, CALPU(20), PROTPU( 20),
     5                W(20)
      REAL MM
      COMMON /PPDEMC/ ARG(5), KFT(20), KNOR, KPAVG(20),
     1               PCCUNT(20), PCPMAX(20), PCPMIN(20), PCRT(20),
     2               PCRT1, PCUT1, PWPMAX(20), PWPMIN(20), STKRT(2 0),
     3               TARIFF(20), VCPUT(5,20), VPCSUP(5,20),
     4               VPWLDI(5,20), VPWLDX(5,20)
      COMMON /ICDEMC/ ELISAV(20), ELISVR(20), EPAVGO(20),
     1                STKO(20), VCPU(10,20), VCSTK(10,20), VEFRMY( 10),
```

```
                   VFPIX(10), VPSUP(10,20), VPWI(10,20),
     2             VPWX(10,20), FNR(20), VPICF(10), VEINCM(10),
     3             VPCUFD(10)
     4
      COMMON /VDEMC/ CHGSTK(20), DSTK(20), DVALDF, EFRMY,
     1             ELASP(20,20), ELASPR(20,20),
     1             ELASI(20), ELASIR(20), FRMY(1), INAPA(20),
     1             FGIVAL, ITER, PCCONF(20), TARREV(20),
     2             PRFEXP(20), PRFIMP(20), PWLDD(20), PWLDEX(20) ,
     3             PWLDIN(20), RDEM(20), TRDPRF, TVALDF, VALDEF( 20)
      COMMON /CVDEMC/ AREQH(12), AREQHF(12), ARQHFS(12), CAL(3,3),
     1             PCAL(3), PPRO(3),
     1             CALPC(2,3), CMARKT(20), CMPI, COGPI, CPI, CR PI,
     2             GOVSUB(20), PCNCON(20), PRO(3,3),
     3             PROPC(2,3), QREQH(12), QREQHF(12), QRQHFS(12 ),
     4             QRQHLV(7), RFTEX, SAVUPC, SELFSU(20), TARH,
     5             TARHF, TARHFS, TDISAP(20), TEXNF, TEXPF, TEX PT,
     6             TEXPTC, TOTHER(20), TEXPFC, XDIFF, ZZ(20)
      COMMON /DEMSVC/ ALPHA, CPU(20), CPUO(20), DEMC(20),
     1             DEMT(1,20), EPAVG(1,20), INCOM, PCCON(1,20),
     2             PCCONU(20), Q(20), RFRMY(1), S1, STK(20),
     3             EINCOM, TDEM(20), TDEML(20), WWP(20)
      REAL INCOM
      END


      BLOCKDATA CRXD
C
C DATA FOR SUBROUTINE CHGRX
C
C DECLARATIONS
C
      COMMON /CRXDC/ AARGCR(5), FLBX(1,13,3), FXX2(1,13), FXX3(1,1 3),
     1               ARGCRD(5), KCRD, ARGCR(5), KCR, KKCR,
     2               VFERAP(5,1,13), VYIELD(10,1,12), VYLDT(5,1,12 ),
     3               VYPAS(5,1)
C
C DATA
C
      DATA KKCR /5/
C
      DATA FLBX /
     1          0.381,0.281,0.260,0.099,0.609,0.255,0.171,0.417,1.8 10,
     2          0.218,1.700,0.224,0.020,
     3          0.190,0.330,0.250,0.200,0.274,0.126,0.357,0.103,0.2 00,
     4          0.293,0.000,0.130,0.020,
     5          0.793,0.594,0.424,0.869,3.756,0.550,2.877,1.570,3.1 31,
     6          0.426,2.425,0.731,0.08/
C
      DATA FXX2 /
     1     3310., 0., 0., 0., 80064., 0., 4062., 0., 2760., 0.,
     2     4700., 1930., 0. /
C
      DATA FXX3 /
     1     8650., 7940., 6180., 940., 62890., 1590., 7202., 1505.,
     2     46620., 1590., 40520., 3360., 150. /
C
C
      DATA AARGCR /
     1          1970.,     1976.,     1981.,     1986.,     2001. /
C
      DATA VFERAP /
     1          .172,      .299,      .340,      .380,      .400,
     2          .182,      .286,      .330,      .360,      .380,
     3          .162,      .256,      .300,      .350,      .370,
     4          .158,      .223,      .240,      .260,      .270,
     5          .314,      .319,      .260,      .210,      .200,
     6          .066,      .093,      .110,      .120,      .130,
     7          .265,      .418,      .510,      .610,      .650,
     8          .147,      .220,      .270,      .300,      .320,
     9          .358,      .615,      .550,      .510,      .480,
     *          .271,      .407,      .500,      .520,      .530,
     A          .321,      .464,      .490,      .510,      .520,
     B          .108,      .170,      .200,      .210,      .220,
     C          .014,      .014,      .014,      .014,      .014 /
C
      DATA KCRD, KCR / 5,5 /
C
      DATA ARGCRD /
     1          1970.,     1976.,     1981.,     1986.,     2001./
C
      DATA VYPAS /
     1          6.00,      6.03,      9.78,      12.71,     12.71/
C
C
      DATA ARGCR /
     1          1976.,     1981.,     1986.,     1991.,     2001./
C
```

```
      DATA VYLDT /                                                      CRXD
     1          4.29,     4.34,     4.88,     4.96,     5.17,           CRXD
     2          2.30,     2.80,     3.40,     3.59,     3.95,           CRXD
     3          2.50,     3.05,     3.58,     3.70,     4.37,           CRXD
     4          1.35,     1.39,     1.87,     2.41,     3.38,           CRXD
     5          8.60,     8.90,     9.20,     9.40,     9.50,           CRXD
     6          1.15,     1.25,     1.69,     1.82,     1.90,           CRXD
     7         11.50,    13.50,    16.00,    18.00,    20.00,           CRXD
     8         16.00,    16.50,    17.50,    18.00,    19.00,           CRXD
     9          2.60,     2.80,     2.90,     3.00,     3.00,           CRXD
     *         25.00,    25.00,    25.00,    25.00,    25.00,           CRXD
     A           .50,      .55,      .60,      .65,      .65,           CRXD
     B           .85,     1.00,     1.30,     1.50,     1.50 /          CRXD
C                                                                       CRXD
C                                                                       CRXD
C               1970      1971      1972      1973      1974            CRXD
C               1975      1976      1977      1978      1979            CRXD
C                                                                       CRXD
C                                                                       CRXD
      DATA ((VYIELD(I,1,J),I=1,10),J=1,6) /                             CRXD
     1      3.301,    3.386,    3.350,    3.594,    3.690,              CRXD
     1      3.833,  4*4.290,                                            CRXD
     2      2.178,    2.230,    2.253,    2.167,    1.972,              CRXD
     2    5*2.391,                                                      CRXD
     3      2.261,    2.263,    2.371,    2.310,    2.031,              CRXD
     3    5*2.220,                                                      CRXD
     4      1.036,    1.128,    1.134,    1.160,    1.200,              CRXD
     4    5*1.302,                                                      CRXD
     5      7.091,    7.368,    8.329,    8.446,    7.748,              CRXD
     5    5*7.283,                                                      CRXD
     6       .764,     .785,     .774,     .772,    1.053,              CRXD
     6    5*1.062 /                                                     CRXD
      DATA ((VYIELD(I,1,J),I=1,10),J=7,12) /                            CRXD
     7      9.786,   11.403,   11.050,   10.361,   10.857,              CRXD
     7   5*11.531,                                                      CRXD
     8     15.322,   15.379,   15.975,   15.602,   15.558,              CRXD
     8   5*17.845,                                                      CRXD
     9      1.320,    1.559,    2.010,    2.026,    1.751,              CRXD
     9    5*1.934,                                                      CRXD
     *     10*25.,                                                      CRXD
     A       .254,     .306,     .345,     .389,     .422,              CRXD
     A     5*.397,                                                      CRXD
     B       .820,     .919,     .810,     .816,     .825,              CRXD
     B     5*.807 /                                                     CRXD
C                                                                       CRXD
      END                                                               CRXD
```

```
      SUBROUTINE DEMCRT
C                                                                       DEMCRT
C DECLARATIONS                                                          DEMCRT
C                                                                       DEMCRT
      COMMON /CONTRC/ ANIMAL, BASEYR, DT, FARM, ITIME, KTRACK, KTABI,   DEMCRT
     1                IRUN, ITTY, ICHDAT, ISENPR, IPLT, ITABPR,         CONTRC
     1                LVSTPR, MODE, MODNAM(9),                          CONTRC
     1                KPOP, KCHG, KRAP, KDEM, KNEC, LINK, JPER,         CONTRC
     1                LUIN1, LUIN2, LUOUT, LUPLOT(3), LURLP1,           CONTRC
     2                LUPGEN, LUPOP,                                    CONTRC
     2                KPPRT(15), KCPRT(15), KFPRT(15), KAPRT(15),       CONTRC
     2                KDPRT(15), KYPRT(15), KNPRT(15),                  CONTRC
     3                NCC, NCOM, NCOMAG, NCROP, NLVST, NNC, NREGN,      CONTRC
     3                NRUN, NS, NAME(20), PLANT,                        CONTRC
     4                T, TIMEI, TIMEF, TTABI, YEAR,                     CONTRC
     5                TOTSOR, TOTPOP, URBAN                             CONTRC
      INTEGER ANIMAL, FARM, PLANT, URBAN, TOTPOP, TOTSOR                CONTRC
      EQUIVALENCE (LUPLT1,LUPLOT(1)),(LUPLT2,LUPLOT(2)),                CONTRC
     1            (LUPLT3,LUPLOT(3))                                    CONTRC
      COMMON /DEMKC/ APCF, DEFCIT(20), PAVG(1,20), PFDIMX, TPAVG(20)    CONTRC
      COMMON /DEMNC/ CPFI, FCONP, FCONU, TTAREV, TEXNFC, FEXNFC,        DEMKC
     1               UCONP, UCONU, VALDF1, VALDF2, WPFI                 DEMNC
      COMMON /DEMYC/ CON(1,20), DSAVD(1)                                DEMNC
      COMMON /FYAKC/ LPER(1,4), PFRT(13), POI(1), PPEST(1),             DEMYC
     1               WR(1), YMIN(1), FSAV(1)                            FYAKC
      REAL LPER                                                         FYAKC
      COMMON /NECDYC/ AGINTX, CPUNF, FPCI, PRFRMY(1), UPCI, WOND,       FYAKC
     1                TAGDIP, TPTAX, TYG, TYNAG                         NECDYC
      COMMON /PRDDNC/ TOUTPT(20)                                        NECDYC
      COMMON /PRDDC/ AYLD(12), EFFPLV(7), EFFYLD(12), FDFSH, FEED( 12), PRDDNC
     1               FCIM, FLOSS(20), TAREA(12), TDSUP(20), TOUTRL ,    PRDDC
     2               TTAREA                                             PRDDC
      COMMON /RDDATC/ PFLOSS(20), SDPHA(1,12)                           PRDDC
      COMMON /CPDEMC/ APCN, DELP, IGRAIN(5), IMEAT(5),                  RDDATC
     2                MM(20), PILOSS(20), PLOSTD, PSUBCG(5,5,2), PCMIN, CPDEMC
     3                PSUBCM(5,5,2), PSUBTG(5,2), PSUBTM(5,2),PFU( 2,20), CPDEMC
     4                PCONVG(5), PWHTIC, YDEL, CALPU(20), PROTPU( 20),  CPDEMC
     5                W(20)                                             CPDEMC
      REAL MM                                                           CPDEMC
```

```
      COMMON /ICDEMC/ ELISAV(20), ELISVR(20), EPAVGO(20),
     1                STKO(20), VCPU(10,20), VCSTK(10,20), VEFRMY( 10),
     2                VFPIX(10), VPSUP(10,20), VPWI(10,20),
     3                VPWX(10,20), FNR(20), VPICF(10), VEINCM(10),
     4                VPCUFD(10)
      COMMON /DEMSVC/ ALPHA, CPU(20), CPUO(20), DEMC(20),
     1                DEMT(1,20), EPAVG(1,20), INCOM, PCCON(1,20),
     2                PCCONU(20), Q(20), RFRMY(1), S1, STK(20),
     3                EINCOM, TDEM(20), TDEML(20), WWP(20)
      REAL INCOM
      COMMON /TSTDC/ TSTD(20), PCMEAN(20), TSTDG, TSTDM, TSTDO, TS TDT
      COMMON /VDEMC/ CHGSTK(20), DSTK(20), DVALDF, EFRMY,
     1               ELASP(20,20), ELASPR(20,20),
     1               ELASI(20), ELASIR(20), FRMY(1), INAPA(20),
     1               FGIVAL, ITER, PCCONF(20), TARREV(20),
     2               PRFEXP(20), PRFIMP(20), PWLDD(20), PWLDEX(20) ,
     3               PWLDIM(20), RDEM(20), TRDPRF, TVALDF, VALDEF( 20)
      COMMON /CVDEMC/ AREQH(12), AREQHF(12), ARQHFS(12), CAL(3,3),
     1                PCAL(3), PPRO(3),
     1                CALPC(2,3), CMARKT(20), CMPI, COGPI, CPI, CR PI,
     2                GOVSUB(20), PCNCON(20), PRO(3,3),
     3                PROPC(2,3), QREQH(12), QREQHF(12), QRQHFS(12 ),
     4                QRQHLV(7), RFTEX, SAVUPC, SELFSU(20), TARH,
     5                TARHF, TARHFS, TDISAP(20), TEXNF, TEXPF, TEX PT,
     6                TEXPTC, TOTHER(20), TEXPFC, XDIFF, ZZ(20)
      COMMON /POPKC/ CALRPC(2), POP(2), POPL(2), POPR(1), PRORPC(2 ),
     1               TPOP, TRPOP, TPOPPC(2)
C
C
C              CALCULATION OF NATIONAL CRITERION VARIABLES
C
      PFDIMX = PWLDIM(4)
      DO 50 J=1,NCOMAG
      TDEM(J) = RDEM(J) + Q(J)
      PCNCON(J) = TDEM(J) / TPOP
   50 CONTINUE
      CPI    = 0.0
      WPFI = 0.
      CPFI   = 0.0
      COGPI  = 0.0
      CMPI   = 0.0
      W1     = 0.0
      W2     = 0.0
      W3     = 0.0
      W4     = 0.0
      W5 = 0
C              PRICE INDICES
C
      DO 60 J=1,NCOMAG
      CMARKT(J) = TPAVG(J) * MM(J)
      GOVSUB(J) = CMARKT(J) - (CPU(J) - TPAVG(J))
      IF(ITIME.NE.0 .OR.  KRAP.EQ.0) GO TO 60
      W5 = W5 + ABS(VALDEF(J))
   60 CONTINUE
C
      CRPI = CPU(1)/CPUO(1)
      DO 80 J=1,NCOM
      IF(KRAP.EQ.0 .OR. DEFCIT(J).EQ.0. .OR. J.GT.NCOMAG) GO TO 65
      IF(ITIME .EQ. 0) WWP(J) = ABS(DEFCIT(J)) / W5
      WPFI = WPFI + WWP(J) * ABS(VALDEF(J)) / ABS(DEFCIT(J))
   65 CONTINUE
      CPI = CPI + W(J)*(CPU(J)/CPUO(J))
      W1 = W1 + W(J)
      IF(J.GT.NCOMAG) GO TO 80
      CPFI = CPFI + W(J)*(CPU(J)/CPUO(J))
      W2 = W2 + W(J)
      IF(J.EQ.1 .OR. J.EQ.5 .OR. J.EQ.7 .OR. J.GE.9) GO TO 70
      COGPI = COGPI + W(J)*(CPU(J)/CPUO(J))
      W3 = W3 + W(J)
   70 CONTINUE
      IF((J.LT.13).OR.(J.GT.18)) GO TO 80
      CMPI = CMPI + W(J)*(CPU(J)/CPUO(J))
      W4 = W4 + W(J)
   80 CONTINUE
C
      CPI    = CPI/W1
      CPFI   = CPFI/W2
      COGPI  = COGPI/W3
      CMPI   = CMPI/W4
C
C         NONFARM CONSUMPTION EXPENDITURES AND PROCESSED/UNPROCESS ED
C
      FCONU = 0.
      FCONP = 0.
      UCONU = 0.
      UCONP = 0.
      TEXPF = 0.
      DO 85 J=1,NCOMAG
      FCONU = FCONU + PFU(1,J)*TPAVG(J)*RDEM(J)/POP(1)
      FCONP = FCONP + (1.-PFU(1,J))*TPAVG(J)*RDEM(J)/POP(1)
      UCONU = UCONU + PFU(2,J)*CPU(J)*Q(J)/(POP(2)*ALPHA)
```

```
      UCONP = UCONP + (1.-PFU(2,J))*CPU(J)*Q(J)/(POP(2)*ALPHA)
      TEXPF = TEXPF + CPU(J) * Q(J) / ALPHA
   85 CONTINUE
      TEXNF = CPU(NCOM) * Q(NCOM) / ALPHA
      TEXPT = TEXPF + TEXNF
      RFTEX = TEXPF/TEXPT
      XDIFF = POPL(2) * UPCI - TEXPT
      IF(ITIME .EQ. 0) XDIFF = POP(2) * (EINCOM/APC..) - TEXPT
C
      SAVUPC = XDIFF / POP(2)
      TEXPTC = TEXPT / POP(2)
      TEXPFC = TEXPF / POP(2)
      TEXNFC = TEXNF / POP(2)
C
      IF(ITIME.NE.0) CMIN = PCMIN * FEXNFC
      FEXNFC = EFRMY - FCONU - FCONP
      IF(ITIME.EQ.0) GO TO 88
      DSAV = AMAX1(CMIN - FEXNFC, 0. ) * TRPOP
      FEXNFC = AMAX1(FEXNFC,CMIN)
      DO 87 IR=1,NREGN
      DSAVD(IR) = -DSAV * POPR(IR) / TRPOP
   87 CONTINUE
   88 CONTINUE
C
C           SELF-SUFFICIENCY AND SELF-SUFFICIENCY REQUIREMENTS
C
      IF(KRAP .EQ. 0) GO TO 125
      TARH   = 0
      TARHF  = 0
      TARHFS = 0.
      DO 110 J=1,NCROP
      AREQH(J) = TDEM(J) / EFFYLD(J)
      IF(J .EQ.  3) AREQH(3) = AREQH(3) / (1.-PWHTIC)
      IF(J .EQ. 12) AREQH(12) = AREQH(12)/TABEL(VPICF,KTABI,TTABI, 1.,T)
      IF(J .EQ.  9) AREQH(9) = AREQH(9) / (1.-PLOSTD)
      QREQH(J) = AYLD(J) * AREQH(J)
      TARH = TARH + AREQH(J)
C
      AREQHF(J) = AREQH(J) + FEED(J) / EFFYLD(J)
      QREQHF(J) = AYLD(J) * AREQHF(J)
      TARHF = TARHF + AREQHF(J)
C
      ARQHFS(J) = AREQHF(J) + CHGSTK(J) / EFFYLD(J)
      QRQHFS(J) = AYLD(J) * ARQHFS(J)
      TARHFS = TARHFS + ARQHFS(J)
  110 CONTINUE
      DO 115 J=13,19
      QRQHLV(J-12) = TDEM(J) / EFFPLV(J-12)
  115 CONTINUE
C
      DO 120 J=1,NCOMAG
      ZZ(J) = TOUTPT(J) + DEFCIT(J)
      IF(J .EQ.  1) ZZ(1) = TOUTRL + DEFCIT(1)
      IF(J .EQ. 12) ZZ(12)= TOUTPT(12) * TABEL(VPICF,KTABI,TTABI,1. 0,T)
     1                      + DEFCIT(12)
      TOTHER(J) = ZZ(J) - TDEM(J)
      TDISAP(J) = TDEM(J) + TOTHER(J)
      SELFSU(J) = 0
      IF(ZZ(J)-CHGSTK(J) .NE. 0. )
     1   SELFSU(J) = (ZZ(J)-DEFCIT(J)) / (ZZ(J)-CHGSTK(J))
  120 CONTINUE
  125 CONTINUE
C
C        CALORIE AND PROTEIN CONSUMPTION PER CAPITA
C
      DO 90 IP=1,TOTPOP
      DO 90 IS=1,TOTSOR
      CAL(IS,IP) = 0.
      PRO(IS,IP) = 0.
   90 CONTINUE
C
      DO 94 JC=1,NCOMAG
C
      IF(JC .LE. NCROP)                                           GO TO  91
C
          THEN FOR CALORIES AND PROTEIN FROM PLANT SOURCES        GO TO  92
   91 CAL(PLANT,URBAN) = CAL(PLANT,URBAN)+ CALPU(JC) * PCCONU(JC)/ 365.
                                                                  CONTIN UE
      CAL(PLANT,FARM)  = CAL(PLANT,FARM) + CALPU(JC) * PCCONF(JC)/ 365.
      PRO(PLANT,URBAN) = PRO(PLANT,URBAN)+ PROTPU(JC)* PCCONU(JC)/ 365.
      PRO(PLANT,FARM)  = PRO(PLANT,FARM) + PROTPU(JC)* PCCONF(JC)/ 365.
C
         ELSE FOR CALORIES AND PROTEIN FROM ANIMAL SOURCES        GO TO  93
   92 CAL(ANIMAL,URBAN) = CAL(ANIMAL,URBAN)+ CALPU(JC) * PCCONU(JC )/365.
                                                                  CONTIN UE
      CAL(ANIMAL,FARM)  = CAL(ANIMAL,FARM) + CALPU(JC) * PCCONF(JC )/365.
      PRO(ANIMAL,URBAN) = PRO(ANIMAL,URBAN)+ PROTPU(JC)* PCCONU(JC )/365.
   93 PRO(ANIMAL,FARM)  = PRO(ANIMAL,FARM) + PROTPU(JC)* PCCONF(JC )/365.
C
         ENDIF                                                    CONTIN UE
   94 CONTINUE
```

```
      DO 98 IS=PLANT, ANIMAL
      DO 98 IP=FARM, URBAN
   98 PRO(IS, IP) = PRO(IS, IP) * 1.E6
C                NATIONAL AVERAGES
      DO 95 IS=PLANT, ANIMAL
      CAL(IS, TOTPOP) = (CAL(IS, URBAN)*POP(URBAN) +
     1                   CAL(IS, FARM)*POP(FARM)) / TPOP
      PRO(IS, TOTPOP) = (PRO(IS, URBAN)*POP(URBAN) +
     1                   PRO(IS, FARM)*POP(FARM)) / TPOP
   95 CONTINUE
C
      DO 96 IP=1, TOTPOP
      CAL(TOTSOR, IP) = CAL(PLANT, IP) + CAL(ANIMAL, IP)
      PRO(TOTSOR, IP) = PRO(PLANT, IP) + PRO(ANIMAL, IP)
   96 CONTINUE
C
C          COMPUTE PERCENTAGES FOR PLANT VS ANIMAL SOURCES
      DO 97 IP=1, TOTPOP
      DO 97 IS=PLANT, ANIMAL
      CALPC(IS, IP) = CAL(IS, IP) / CAL(TOTSOR, IP) * 100.
      PROPC(IS, IP) = PRO(IS, IP) / PRO(TOTSOR, IP) * 100.
   97 CONTINUE
C
C          PERCENTAGE OF CALORIE AND PROTEIN REQUIREMENTS CONSUMED
      DO 99 IP=1, TOTPOP
      PCAL(IP) = 0
      PPRO(IP) = 0
   99 CONTINUE
      IF(KPOP .EQ. 0) GO TO 105
      DO 100 IP=FARM, URBAN
      PCAL(IP) = 100. * CAL(TOTSOR, IP) /(CALRPC(IP)/365.)
      PPRO(IP) = 100. * PRO(TOTSOR, IP) /(PRORPC(IP)/365.)
  100 CONTINUE
      PCAL(TOTPOP) = 100. * CAL(TOTSOR, TOTPOP)/((CALRPC(FARM)*POP( FARM)
     1              +CALRPC(URBAN)*POP(URBAN))/(TPOP*365.))
      PPRO(TOTPOP) = 100. * PRO(TOTSOR, TOTPOP)/((PRORPC(FARM)*POP( FARM)
     1              +PRORPC(URBAN)*POP(URBAN))/(TPOP*365.))
  105 CONTINUE
C
C          TIME-SERIES TRACKING MEASURES OF FIT
      IF(ITIME .GT. 0) GO TO 150
      DO 140 I=1, NCOM
      TSTD(I) = 0
      PCMEAN(I) = 0.
  140 CONTINUE
      GO TO 190
  150 CONTINUE
      IF(T .GT. BASEYR+.0001) GO TO 190
      DO 160 I=1, NCOMAG
      IF(I.EQ.10 .OR. I.EQ.11) GO TO 160
      TSTD(I) = TSTD(I) + (PCNCON(I) - TABEL(VPSUP(1, I), KTABI, TTAB I,
     1          1.0, T))**2
      PCMEAN(I) = PCMEAN(I) + TABEL(VPSUP(1, I), KTABI, TTABI, 1.0, T)
  160 CONTINUE
      IF(ABS(BASEYR-T).GT. .0001) GO TO 190
      TSTDT = 0.
      DO 170 I=1, NCOMAG
      IF(I.EQ.10 .OR. I.EQ.11) GO TO 170
      PCMEAN(I) = PCMEAN(I) / FLOAT(ITIME)
      TSTD(I) = TSTD(I) / (PCMEAN(I)**2)
      TSTDT = TSTDT + TSTD(I)
  170 CONTINUE
      TSTDG = TSTD(1) + TSTD(2) + TSTD(3) + TSTD(4) + TSTD(6) + TS TD(8)
      TSTDO = TSTD(5) + TSTD(7)
      TSTDM = TSTD(13) + TSTD(14) + TSTD(15) + TSTD(16) + TSTD(17)
     1              + TSTD(18)
      IF(KLPRT(13) .EQ. 0) GO TO 190
      PRINT 180, TSTD, TSTDG, TSTDO, TSTDM, TSTDT
  180 FORMAT(1H1, T5, 36HDEMAND TIME-SERIES TRACKING MEASURES /
     1          T8, 13HBY COMMODITY , 10F10.4/21X, 10F10.4/
     2          T8, 13HFOR GRAINS   , F10.4 /
     3          T8, 13HFOR FRT + VEG, F10.4 /
     4          T8, 13HFOR LVSTK/FSH, F10.4 /
     5          T8, 13HFOR ALL COMM , F10.4 )
  190 CONTINUE
C
      RETURN
      END
```

```
      BLOCKDATA DEMDAT
C
C DEMAND DATA
C
C DECLARATIONS
C
      COMMON /CPDEMC/ APCN, DELP, IGRAIN(5), IMEAT(5),
```

```
     2                      MM(20), PILOSS(20), PLOSTD, PSUBCG(5,5,2), P CMIN,
     3                      PSUBCM(5,5,2), PSUBTG(5,2), PSUBTM(5,2),PFU( 2,20),
     4                       PCONVG(5), PWHTIC, YDEL, CALPU(20), PROTPU( 20),
     5                      W(20)
      REAL MM
      COMMON /ICDEMC/ ELISAV(20), ELISVR(20), EPAVGD(20),
     1                STKO(20), VCPU(10,20), VCSTK(10,20), VEFRMY( 10),
     2                VFPIX(10), VPSUP(10,20), VPWI(10,20),
     3                VPWX(10,20), FNR(20), VPICF(10), VEINCM(10),
     4                VPCUFD(1C)
      COMMON /PPDEMC/ ARG(5), KFT(20), KNOR, KPAVG(20),
     1                PCCONT(20), PCPMAX(20), PCPMIN(20), PCRT(20),
     2                PCRT1, PCUT1, PWPMAX(20), PWPMIN(20), STKRT(2 0),
     3                TARIFF(20), VCPUT(5,20), VPCSUP(5,20),
     4                VPWLDI(5,20), VPWLDX(5,20)
      COMMON /VDEMC/  CHGSTK(20), DSTK(20), DVALDF, EFRMY,
     1                ELASP(20,20), ELASPR(20,20),
     1                ELASI(20), ELASIR(20), FRMY(: .  'NAPA(2C),
     1                FGIVAL, ITER, PCCONF(20), T'RF.EV(20),
     2                PRFEXP(20), PRFIMP(20), P'.DD(20), PWLDEX(20) ,
     3                PWLDIM(20), RDEM(20), T.DPRF, TVALDF, VALDEF( 20)
C
C DATA
C
C   CONSTANT PARAMETERS
C
      DATA APCN / .85 /
      DATA CALPU  /
     1        .346E7,     .338E7,     .273E7,
     2        .382E7,     .330E6,     .950E6,     .362E7,     .740E6,
     3        0.,         .526E7,     .266E7,     0.,         0.,
     4        .770E6,     .142E7,     .480E6,     .590E6,     .482E7,
C                                                 .219E7,     0. /
      DATA DELP / 2.0 /
C
      DATA ELASP/
     1   -.3,   17*0 ,
     2   2*0.,   -.7,   17*0.,          0.,   -.2,   18*0.,
     3   4*0.,   -.85,   15*0.,         3*0 ,   -.6,   16*0.,
     4   6*0.,   -.10,   13*0.,         5*0.,   -.75,   14*0.,
     5   8*0 ,   -.50,   31*0.,         7*0.,   -.4,   12*0.,
     6   12*0.,   -1.4,   7*0.,         20*0.,   11*0.,   -1.10,   8*0.,
     7   14*0.,   -1.,   5*0.,          13*0.,   -1.5,   6*0.,
     8   16*0.,   -.3,   3*0.,          15*0.,   -1.2,   4*0.,
     9   20*0.,                         17*0.,   -.2,   2*0.,
C                                       19*0.,   -.4   /
      DATA ELASPR  /
     1   -.40,   19*0.,
     2   2*0.,   -.4,   17*0.,          0.,   -.20,   18*0.,
     3   4*0.,   -.35,   15*0.,         3*0.,   -.60,   16*0 ,
     4   6*0.,   -.10,   13*0.,         5*0.,   -.4,   14*0.,
     5   8*0 ,   -.20,   31*0.,         7*0.,   -.7,   12*0.,
     6   12*0.,   -1.8,   7*0.,         20*0.,   11*0.,   -.5,   8*0.,
     7   14*0.,   -.5,   5*0.,          13*0 ,   -1.5,   6*0.,
     8   16*0.,   -.4,   3*0 ,          15*0.,   -.8,   4*0.,
     9   20*0.,                         17*0 ,   -.3,   2*0.,
C                                       20*0.   /
      DATA IGRAIN   /
     1            1,   2,   3,   4,   8/
C
      DATA IMEAT   /
     1            13,  15,  16,  18,  19/
C
      DATA MM  /
     1     .02,  .06,  .10,  .96,  1.28,  .24,  .29,  1.03,  2.36,  0.,
     1     0.,  1.2,  .07,  1.09,  .57,  -.19,  .24,  2.72,  .35,  0. /
C
      DATA PCMIN /.80/
C
      DATA ((PFU(I,J),J=1,20),I=1,2) /
     1          .986,       .921,
     1          .290,       .962,       .028,       1.000,
     1          0.0,        0.0,        .924,       0.0,        .927,
     1          0.0,        1.000,      0.0,        .227,       0.0,
     2          .986,       .921,       .702,       .424,       0.0,
     2          .290,       .962,       .028,       1.000,      0.0,
     2          0.0,        0.0,        .924,       0.0,        .927,
     2          0.0,        1.000,      0.0,        .227,       0.0,
C                                       .702,       .424,       0.0,
      DATA W  /                                                 0.0/
     1          169.7,      15.2,
     2          13.5,       82.7,       23.5,       0.3,
     3          0.,         8.5,        6.8,        54.6,       15.5,
     4          6.5,        10.9,       25.5,       3.9,        0.,
C                                       49.6,       7.2,        12.0,
      DATA PCONVG   /                                           494.1  /
     :          1.00,       1.00,
C                                       .80,        1.00,       .27/
      DATA PILOSS /
     1   4*.006, 10,.006,.15,4*0.,.006,.02,.006,.02,.02,.02,.05,.03 ,0. /
C
```

```
      DATA PLOSTD / .107 /
C
      DATA PROTPU /
     1  .067,     .094,     .066,     .092,     .004,
     2  .362,     .015,     .016,     0.,       0.,
     3  0.,       098,      .180,     .029,     .112,
     4  .126,     .113,     .073,     .088,     0./
C
      DATA (((PSUBCG(I,J,K),J=1,5),I=1,5),K=1,2)   /
     1  .00,      55,       55,       .55,      .15,
     2  .65,      00,       .10,      .10,      .35,
     3  .25,      35,       .00,      .30,      .50,
     4  05,       .05,      .35,      .00,      .00,
     5  .05,      .05,      .00,      .05,      .00,
     1  .00,      .55,      55,       .50,      .15,
     2  .65,      .00,      .10,      .15,      .45,
     3  .30,      .45,      .00,      .30,      .40,
     4  .05,      .00,      .35,      .00,      .00,
     5  .00,      .00,      .00,      05,       .00/
C
      DATA (((PSUBCM(I,J,K),J=1,5),I=1,5),K=1,2)   /
     1  00,       .30,      .25,      .20,      .00,
     2  .50,      .00,      .50,      .60,      .00,
     3  .25,      .30,      .00,      .20,      .00,
     4  .25,      .40,      .25,      .00,      .00,
     5  .00,      .00,      .00,      .00,      .00,
     1  .00,      .30,      .30,      .30,      .00,
     2  .45,      .00,      .40,      .50,      .00,
     3  .20,      .20,      .00,      .20,      .00,
     4  .35,      .50,      .30,      .00,      .00,
     5  .00,      00,       .00,      .00,      .00/
C
      DATA PSUBTG   /
     1  .95,      .95,      .70,      .90,      .50,
     2  .95,      .95,      .70,      .90,      .33/
C
      DATA PSUBTM   /
     1  95,       95,       .50,      .33,      .00,
     2  95,       95,       .50,      .33,      .00/
C
      DATA PWHTIC  / .05 /
C
      DATA YDEL / 2.0 /
C
C
C     INITIAL CONDITION DATA
C
      DATA ELISAV /
     1  .20,      -.25,     .50,      -.30,     1.30,
     2  .45,      .40,      -.60,     1.20,     .00,
     3  .00,      .80,      1.40,     3.20,     .55,
     4  1.00,     .40,      4.00,     .00,      1.03/
C
      DATA ELISVR  /
     1  .10,      -.19,     .90,      -.10,     .75,
     2  .30,      .30,      -.50,     1.20,     .00,
     3  .00,      .30,      .94,      3.00,     .59,
     4  .80,      .40,      2.00,     .00,      .69/
C
      DATA EPAVGD /
     1  78500.,   41900.,   28100.,   42500.,   54300.,
     2  60000.,   31000.,   15255.,   207700.,  1.,
     3  500900.,  154600.,  653024.,  58000.,   165000.,
     4  351763.,  200300.,  68000.,   65000.,   1.  /
C
      DATA FNR /
     1  .83,      2.33,     .70,      1.70,     .28,
     2  .61,      .76,      2.83,     1.00,     1.00,
     3  1.00,     .32,      .11,      .01,      .38,
     4  .20,      .06,      .33,      .50,      1.00/
C
      DATA STKO / 88000., 1033000., 286000., 17*0.  /
C
C               1970      1971      1972      1973      1974
C               1975      1976      1977      1978      1979
C
      DATA ((VCPU(I,J),I=1,10),J=1,6) /
     1  79125.,   87775.,   102318.,  99388.,   102168.,
     1  5*114629.,
     2  48758.,   55622.,   59174.,   59963.,   51773.,
     2  5*55894.,
     3  30415.,   29202.,   29728.,   33341.,   42014.,
     3  5*41924.,
     4  60330.,   66554.,   86461.,   90309.,   95720.,
     4  5*88403.,
     5  133934.,  147189.,  128161.,  147734.,  134646.,
     5  5*139391.,
     6  98676.,   96621.,   101984.,  107310.,  109726.,
     6  5*112145.  /
C
      DATA ((VCPU(I,J),I=1,10),J=7,12) /
```

```
     7          59777.,     61616.,     45662.,     45218.,     44050.,
     7        5*60421.,
     8          39667.,     36652.,     44111.,     40571.,     44568.,
     8        5*45296.,
     9        1201345.,   1058454.,   947[illegible].,   821103.,    650913.,
     9       5*643430.,
     *              1.,          1.,          1.,          1.,          1.,
     *              1.,          1.,          1.,          1.,          1.,
     A         449100.,    504500.,    568300.,    943200.,       1.1E6,
     A          1.2E6,      1.2E6,      1.2E6,      1.2E6,      1.2E6,
     B         356667.,    455213.,    416666.,    415392.,    397704.,
     B       5*555846.  /
C
      DATA ((VCPU(I,J),I=1,10),J=13,20) /
     C         788333.,    829662.,    862250.,    875382.,    825133.,
     C       5*783833.,
     D         109301.,    104590.,    112912.,    110714.,    122605.,
     D       5*147275.,
     E         396667.,    455213.,    412723.,    447208.,    438704.,
     E       5*535101.,
     F         263000.,    266960.,    259464.,    292049.,    325953.,
     F       5*309769.,
     G         286000.,    236123.,    217666.,    246177.,    265683.,
     G       5*265096.,
     H         257100.,    271366.,    244326.,    270466.,    259156.,
     H       5*257857.,
     I          92357.,    147920.,     76966.,     68075.,     92437.,
     I        5*75000.,
     J           1.000,       .967,       .963,       .983,       .972,
     J          5*.918 /
      DATA VCSTK /
     1         237000.,     69000.,    219000.,     98000.,   -223000.,
     1       5*504000.,
     2          94000.,   -134000.,    137000.,     53000.,    -84000.,
     2       5*367000.,
     3          52000.,   -100000.,    -22000.,   -151000.,    174000.,
     3        5*31000.,
     *          170*0.  /
C
C
      DATA VFPIX /
     1           1.000,      1.011,      1.012,       .981,       .857,
     2          5*.856 /
C
      DATA VEINCM /
     1          65584.,     68468.,     67956.,     72365.,     70832.,
     2        5*74275.  /
C
      DATA VEFRMY /
     1          35096.,     36944.,     42770.,     45090.,     47320.,
     2          54317.,     60000.,     60000.,     60000.,     60000. /
C
      DATA VPCUFD /
     1            .436,       .443,       .440,       .442,       .461,
     2            .465,       .460,       .450,       .450,       .450/
C
      DATA VPICF /
     1            .576,       .688,       .634,       .631,       .728,
     2          5*.539  /
C
      DATA ((VPSUP(I,J),I=1,10),J=1,6) /
     1           .1258,      .1354,      .1206,      .1166,      .1241,
     1         5*.1155,
     2           .0383,      .0404,      .0433,      .0451,      .0451,
     2         5*.0418,
     3           .0398,      .0509,      .0565,      .0567,      .0401,
     3         5*.0466,
     4           .0040,      .0035,      .0036,      .0047,      .0067,
     4         5*.0038,
     5           .0118,      .0112,      .0131,      .0115,      .0152,
     5         5*.0164,
     6           .0086,      .0079,      .0072,      .0075,      .0079,
     6         5*.0096 /
C
      DATA ((VPSUP(I,J),I=1,10),J=7,12) /
     7           .0663,      .0749,      .0686,      .0644,      .0720,
     7         5*.0693,
     8           .0547,      .0560,      .0519,      .0421,      .0368,
     8         5*.0578,
     9           .0009,      .0010,      .0014,      .0023,      .0015,
     9         5*.0012,
     *             0.0,        0.0,        0.0,        0.0,        0.0,
     *             0.0,        0.0,        0.0,        0.0,        0.0,
     A             0.0,        0.0,        0.0,        0.0,        0.0,
     A             0.0,        0.0,        0.0,        0.0,        0.0,
     B           .0013,      .0013,      .0016,      .0016,      .0014,
     B         5*.0022 /
C
      DATA ((VPSUP(I,J),I=1,10),J=13,19) /
     C           .0012,      .0012,      .0012,      .0013,      .0015,
     C         5*.0020,
```

```
D     .0031,    .0024,    .0040,    .0032,    .0039,
D     5*.0047,
E     .0025,    .0024,    .0025,    .0025,    .0026,
E     5*.0027,
F     .0014,    .0015,    .0016,    .0015,    .0015,
F     5*.0015,
G     .0041,    .0041,    .0044,    .0039,    .0043,
G     5*.0044,
H     .0220,    .0249,    .0300,    .0360,    .0423,
H     5*.0420,
I     .0121,    .0070,    .0127,    .0166,    .0155,
I     5*.0135 /
C
      DATA ((VPWI(I,J),I=1,10),J=1,6) /
1     190.,     131.,     131.,     191.,     263.,
1     5*198.,
2     58.,      56.,      53.,      83.,      105.,
2     5*97.,
3     54.,      63.,      55.,      108.,     138.,
3     5*95.,
4     72.,      65.,      48.,      57.,      93.,
4     5*79.,
5     294.,     268.,     519.,     359.,     202.,
5     5*238.,
6     117.,     128.,     109.,     174.,     176.,
6     5*129  /
C
      DATA ((VPWI(I,J),I=1,10),J=7,12) /
7     1601.,    1106.,    1367.,    1316.,    775.,
7     5*680.,
8     98.,      121.,     184.,     54.,      66.,
8     5*50.,
9     368.,     1049.,    946.,     913.,     839.,
9     5*884.,
*     10*1.,
A     20688.,   18137.,   15871.,   19408.,   19402.,
A     5*11556.,
B     883.,     473.,     939.,     312.,     384.,
B     5*935.  /
C
      DATA ((VPWI(I,J),I=1,10),J=13,19) /
C     1128.,    1126.,    1278.,    462.,     2503.,
C     5*1846.,
D     1035.,    581.,     [illegible]    510.,     74.,
D     5*601.,
E     943.,     1312.,    1280.,    1888.,    1636.,
E     5*1514.,
F     850.,     775.,     665.,     740.,     825.,
F     5*665.,
G     925.,     685.,     560.,     625.,     675.,
G     5*560.,
H     550.,     576.,     544.,     625.,     472.,
H     5*430.,
I     115.,     108.,     120.,     143.,     222.,
I     5*227.  /
C
      DATA ((VPWX(I,J),I=1,10),J=1,6) /
1     146.,     101.,     101.,     147.,     202.,
1     5*152.,
2     45.,      43.,      41.,      64.,      81.,
2     5*75.,
3     45.,      53.,      46.,      90.,      115.,
3     5*79.,
4     60.,      54.,      40.,      48.,      78.,
4     5*66.,
5     245.,     223.,     433.,     299.,     168.,
5     5*198.,
6     98.,      107.,     91.,      145.,     147.,
6     5*108.  /
C
      DATA ((VPWX(I,J),I=1,10),J=7,12) /
7     1334.,    922.,     1139.,    1097.,    646.,
7     5*567.,
8     82.,      101.,     153.,     45.,      55.,
8     5*42.,
9     725.,     874.,     788.,     761.,     699.,
9     5*737.,
*     10*1.,
A     17[illegible]40.,   15114.,   13226.,   16173.,   16168.,
A     5*9630.,
B     736.,     394.,     783.,     260.,     320.,
B     5*779.  /
C
      DATA ((VPWX(I,J),I=1,10),J=13,19) /
C     940.,     938.,     1065.,    385.,     2086.,
C     5*1538.,
D     863.,     484.,     347.,     425.,     62.,
D     5*501.,
E     786.,     1093.,    1067.,    1573.,    1363.,
E     5*1262.,
F     710.,     645.,     555.,     615.,     690.,
```

```
     F          5*555.,
     G            770.,       570.,      465.,      520.,      565.,
     G          5*465.,
     H            458.,       480.,      453.,      521.,      393.,
     H          5*358.,
     I             96.,        90.,      100.,      119.,      185.,
     I          5*189. /
C
C
C    POLICY PARAMETERS
C
      DATA KFT /
     1     5, 5, 5, 3, 4, 3, 4, 4, 5, 5,
     2     5, 4, 3, 3, 3, 3, 3, 4, 5, 5/
C
      DATA KPAVG /
     1     1, 1, 1, 0, 0, 0, 0, 0, 1, 0,
     2     1, 0, 0, 0, 0, 0, 0, 0, 0, 0/
C
      DATA PCCONT /
     1          .150,      .025,      .055,      .002,      .040,
     2          .018,      .110,      .020,      .0025,     1.E-8,
     3         1.E-8,      .004,      .005,      .035,      .010,
     4          .005,      .015,      .070,      .030,    200000./
C
      DATA PCPMAX  /
     1     0., 0., 0., 0.1, 4.0, 0.05, 0.1, 0.15, 0.0, 0.0,
     2     0.0, 0.3, 0.1, 0.1, 0.1, 0.15, 0.15, 4.0, 0.0, 0.0/
C
      DATA PCPMIN  /
     1          0.,  0.,  0., .05, .1, .05, .05, .1, 0., 0.,
     2          0., .1, .05, .05, .1, .1, .1, 1., 0., 0./
C
      DATA PCRT  /
     1          .130,      .040,      .045,      .004,      .030,
     2          .012,      .100,      .050,      .0025,     1.E-8,
     3         1.E-8,      .003,      .002,      .007,      .005,
     4          .0025,     .005,      .040,      .015,    200000./
C
      DATA PCRT1  / .110 /
C
      DATA PCUT1 / .100 /
C
      DATA PWPMAX  /
     1     0.0, 0.0, 0.0, 5.0, 3.0, 2.8, 5.0, 5.0, 0.0, 0.0,
     2     0.0, 5.0, 1.0, 1.0, 5.0, 5.0, 5.0, 1.0, 0.0, 0.0/
C
      DATA PWPMIN  /
     1     0.0, 0.0, 0.0, 0.0, 1.65, 1.0, 0.0, 0.0, 0.0, 0.0,
     2     0.0, 0.0, 0.7, 0.3, 0.7, 0.0, 0.0, 1.0, 0.0, 0.0/
C
      DATA STKRT /
     1       1.5, 7.0, 2.0, 17*0.
C
      DATA TARIFF /
     1     .16, .20, .20,
     2     17* 0.0 /
C
      DATA KNOR / 5 /
C
C
      DATA ARG /
     1          1976.,     1981.,     1986.,     1991.,     2001. /
C
      DATA ((VCPUT(I,J),I=1,5),J=1,12) /
     1        123022.,   123022.,   125000.,   125000.,   125000.,
     2         65000.,    65000.,    65000.,    65000.,    65000.,
     3         43881.,    43881.,    45000.,    45000.,    45000.,
     4        103600.,    96400.,    96400.,    96400.,    96400.,
     5        140000.,   148000.,   148000.,   148000.,   148000.,
     6        104900.,   109800.,   109800.,   109800.,   109800.,
     7         39900.,    31700.,    31700.,    31700.,    31700.,
     8         51600.,    62100.,    62100.,    62100.,    62100.,
     9        700000.,   700000.,   700000.,   700000.,   700000.,
     *             1.,        1.,        1.,        1.,        1.,
     A          1.2E6,     1.2E6,     1.2E6,     1.2E6,     1.2E6,
     B        450900.,   501600.,   501600.,   501600.,   501600./
C
      DATA ((VCPUT(I,J),I=1,5),J=13,19) /
     C        913300.,  1324000.,  1324000.,  1324000.,  1324000.,
     D        117200.,   123900.,   123900.,   123900.,   123900.,
     E        467400.,   579800.,   579800.,   579800.,   579800.,
     F        305500.,   411100.,   411100.,   411100.,   411100.,
     G        283700.,   287400.,   287400.,   287400.,   287400.,
     H        250000.,   310000.,   310000.,   310000.,   310000.,
     I        100000.,   100000.,   100000.,   100000.,   100000. /
C
      DATA ((VPCSUP(I,J),I=1,5),J=1,12) /
     1          .1250,     .1250,     .1250,     .1250,     .1250,
     2          .0500,     .0500,     .0500,     .0500,     .0500,
     3          .0500,     .0500,     .0500,     .0500,     .0500,
```

```
     4      .0050,      .0050,      .0050,      .0050,      .0050,
     5      .0150,      .0150,      .0150,      .0150,      .0150,
     6      .0080,      .0080,      .0080,      .0080,      .0080,
     7      .0750,      .0750,      .0750,      .0750,      .0750,
     8      .0350,      .0350,      .0350,      .0350,      .0350,
     9      .0020,      .0020,      .0020,      .0020,      .0020,
     *      0.0,        0.0,        0.0,        0.0,        0.0,
     A      0.0,        0.0,        0.0,        0.0,        0.0,
     B      .0015,      .0015,      .0015,      .0015,      .0015/
C
      DATA ((VPCSUP(I,J), I=1,5), J=13,19) /
     C      .0015,      .0015,      .0015,      .0015,      .0015,
     D      .0050,      .0050,      .0050,      .0050,      .0050,
     E      .0030,      .0030,      .0030,      .0030,      .0030,
     F      .0015,      .0015,      .0015,      .0015,      .0015,
     G      .0045,      .0045,      .0045,      .0045,      .0045,
     H      .0450,      .0450,      .0450,      .0450,      .0450,
     I      .0150,      .0150,      .0150,      .0150,      .0150/
C
      DATA ((VPWLDI(I,J), I=1,5), J=1,12) /
     1      237.13,     136.57,     136.57,     136.57,     136.57,
     2       90.73,      97.95,      97.95,      97.95,      97.95,
     3       83.61,      82.57,      82.57,      82.57,      82.57,
     4       90.09,      67.03,      67.03,      67.03,      67.03,
     5      190.00,     185.00,     185.00,     185.00,     185.00,
     6      140.28,     125.43,     136.56,     145.00,     150.00,
     7       93.66,      87.47,      87.47,      87.47,      87.47,
     8       98.49,     132.68,     164.49,     200.88,     200.88,
     9      716.79,     763.72,     767.11,     770.00,     770.00,
     *        1.00,       1.00,       1.00,       1.00,       1.00,
     A    15510.00,   15510.00,   15510.00,   15510.00,   15510.00,
     B      830.00,     830.00,     830.00,     830.00,     830.00 /
C
      DATA ((VPWLDI(I,J), I=1,5), J=13,19) /
     C     2521.53,    2680.65,    3201.57,    3201.57,    3201.57,
     D      800.00,     800.00,     800.00,     800.00,     800.00,
     E     1364.28,    1557.16,    1729.53,    1810.00,    1810.00,
     F      664.79,     685.41,     695.72,     706.03,     706.03,
     G      561.70,     592.63,     613.25,     623.56,     623.56,
     H      578.70,     739.26,     802.20,     805.00,     805.00,
     I     5*268.92 /
C
      DATA ((VPWLDX(I,J), I=1,5), J=1,12) /
     1      212.13,     111.57,     111.57,     111.57,     111.57,
     2       65.73,      72.95,      72.95,      72.95,      72.95,
     3      658.61,      57.57,      57.57,      57.57,      57.57,
     4       65.09,      42.03,      42.03,      42.03,      42.03,
     5      175.00,     170.00,     170.00,     170.00,     170.00,
     6      115.28,     100.43,     111.56,     120.00,     125.00,
     7       83.66,      77.47,      77.47,      77.47,      77.47,
     8       88.49,     122.68,     154.49,     190.88,     190.88,
     9      706.79,     753.72,     757.11,     760.00,     760.00,
     *        1.00,       1.00,       1.00,       1.00,       1.00,
     A    15500.00,   15500.00,   15500.00,   15500.00,   15500.00,
     B      820.00,     820.00,     820.00,     820.00,     820.00/
C
      DATA ((VPWLDX(I,J), I=1,5), J=13,19) /
     C     2511.53,    2870.65,    3191.57,    3191.57,    3191.57,
     D      785.00,     785.00,     785.00,     785.00,     785.00,
     E     1354.28,    1547.16,    1717.53,    1800.00,    1800.00,
     F      654.79,     675.41,     685.72,     696.03,     696.03,
     G      551.70,     582.63,     603.25,     613.56,     613.56,
     H      593.70,     734.26,     797.20,     800.00,     800.00,
     I      258.92,     258.92,     253.92,     258.92,     258.92/
C
C
      END


      SUBROUTINE DEMI
C
C INITIALIZATION OF DEMAND
C
C DECLARATIONS
C
      COMMON /CONTRC/ ANIMAL, BASEYR, DT, FARM, ITIME, KTRACK, KTABI,
     1                IRUN, ITTY, ICHDAT, ISENPR, IPLT, ITABPR,
     1                LVSTPR, MODE, MODNAM(9),
     1                KPOP, KCHG, KRAP, KDEM, KNEC, LINK, JPER,
     1                LUIN1, LUIN2, LUOUT, LUPLOT(3), LURLP1,
     2                LUPGEN, LUPOP,
     2                KPPRT(15), KCPRT(15), KFPRT(15), KAPRT(15),
     2                KDPRT(15), KYPRT(15), KNPRT(15),
     2                NCC, NCOM, NCOMAG, NCROP, NLVST, NNC, NREGN,
     3                NRUN, NS, NAME(20), PLANT,
     4                T, TIMEI, TIMEF, TTABI, YEAR,
     5                TOTSOR, TOTPOP, URBAN
```

```
      INTEGER ANIMAL, FARM, PLANT, URBAN, TOTPOP, TOTSOR                CONTRC
      EQUIVALENCE (LUPLT1,LUPLOT(1)),(LUPLT2,LUPLOT(2)),                CONTRC
     1            (LUPLT3,LUPLOT(3))                                    CONTRC
      COMMON /CPDEMC/ APCN, DELP, IGRAIN(5), IMEAT(5),                  CPDEMC
     2                MM(20), PILOSS(20), PLOSTD, PSUBCG(5,5,2), P CMIN, CPDEMC
     3                PSUBCM(5,5,2), PSUBTG(5,2), PSUBTM(5,2),PFU( 2,20), CPDEMC
     4                PCONVG(5), PWHTIC, YDEL, CALPU(20), PROTPU( 20),  CPDEMC
     5                W(20)                                             CPDEMC
      REAL MM                                                           CPDEMC
      COMMON /ICDEMC/ ELISAV(20), ELISVR(20), EPAVGO(20),               ICDEMC
     1                STKO(20), VCPU(10,20), VCSTK(10,20), VEFRMY( 10), ICDEMC
     2                VFPIX(10), VPSUP(10,20), VPWI(10,20),             ICDEMC
     3                VPWX(10,20), FNR(20), VPICF(10), VEINCM(10),      ICDEMC
     4                VPCUFD(10)                                        ICDEMC
      COMMON /PPDEMC/ ARG(5), KFT(20), KNOR, KPAVG(20),                 PPDEMC
     1                PCCONT(20), PCPMAX(20), PCPMIN(20), PCRT(20),     PPDEMC
     2                PCRT1, PCUT1, PWPMAX(20), PWPMIN(20), STKRT(2 0), PPDEMC
     3                TARIFF(20), VCPUT(5,20), VPCSUP(5,20),            PPDEMC
     4                VPWLDI(5,20), VPWLDX(5,20)                        PPDEMC
      COMMON /DEMSVC/ ALPHA, CPU(20), CPUO(20), DEMC(20),               DEMSVC
     1                DEMT(1,20), EPAVG(1,20), INCOM, PCCON(1,20),      DEMSVC
     2                PCCONU(20), Q(20), RFRMY(1), S1, STK(20),         DEMSVC
     3                EINCOM, TDEM(20), TDEML(20), WWP(20)              DEMSVC
      REAL INCOM                                                        DEMSVC
      COMMON /PKXDC/ DP, DPOPRD(1), KNP, VPOP(31,2)                     PKXDC
      COMMON /DKXDC/ ARGPP(5),KPP,VDFC(10,20),VDFCT(5,20),VPRICE(1 0,20), DKXDC
     1                VPP(5,20),APCFD                                   DKXDC
      COMMON /NDXDYC/ ARGFX(4), ARGND(5), KFX, KND, VCP(10),            NDXDYC
     1                VCPT(5), VFI(10), VFIT(5), VUI(10), VUIT(5),      NDXDYC
     2                VWOND(10), VFXR(4                                 NDXDYC
      COMMON /CRXDC/  AARGCR(5), FLBX(1 13,3), FXX2(1,13), FXX3(1,1 3), CRXDC
     1                ARGCRD(5), KCRD, ARGCR(5), KCR, KKCR,             CRXDC
     2                VFERAP(5,1,13), VYIELD(10,1,12), VYLDT(5,1,12 ),  CRXDC
     3                VYPAS(5,1)                                        CRXDC
      COMMON /FAYXDC/ ARGRA(5), KRA, VLIVST(10,1,5), VLVSTT(5,1,5) ,    FAYXDC
     1                VRLIN(10,1),VRLINT(5,1)                           FAYXDC
      COMMON /ADNXDC/ ARGFSH(5), KFSH, VFSHT(5), VFSH(10), PEOLV(5 ),   ADNXDC
     1                TORLI, ARGFD(5), KFD, VFEED(10,12),               ADNXDC
     2                VFEEDT(5,12), FDFSHD                              ADNXDC
      COMMON /RDDATC/ PFLOSS(20), SDPHA(1,12)                           RDDATC
      COMMON /FCAXDC/ ARGRCR(5), KRCR, VAREA(10,1,13), VAREAT(5,1, 13)  FCAXDC
      COMMON /DCHDC/ JRELD(100), OVALD 100), CVALD(100), ISFLD          DCHDC
C                                                                       DEMI
C RUN INITIALIZATION                                                    DEMI
C                                                                       DEMI
      IF(ICHDAT.NE.0) CALL CHDATA(APCN,JRELD,OVALD,CVALD,ISFLD,7)       DEMI
C                                                                       DEMI
      X = 0.                                                            DEMI
      XX = 0.                                                           DEMI
      CPUO(NCUM) = 1.0                                                  DEMI
      DO 50 JC=1,NCOMAC                                                 DEMI
      IF(JC .GT. NCROP) GO TO 20                                        DEMI
      SUPPLY = 0.                                                       DEMI
      DO 10 IR=1,NREGN                                                  DEMI
      IF(JC .NE. 1)                                                     DEMI
     1    SUPPLY = SUPPLY + TABEL(VAREA(1,IR,JC),KTABI,TTABI,1.0,T IMEI) DEMI
     2         * TABEL(VYIELD(1 IR,JC),KTABI,TTABI,1.0,TIMEI) * (1      DEMI
     3           - PFLOSS(JC )) - TABEL(VAREA(1,IR,JC),KTABI,TTAB I,1.0, DEMI
     4           TIMEI)* SDPHA(IR,JC)                                   DEMI
      IF(JC .EQ. 1)                                                     DEMI
     1    SUPPLY = SUPPLY - TABEL(VAREA(1,IR,JC),KTABI,TTABI,1.0,T IMEI)* DEMI
     2           SDPHA(IR,JC)                                           DEMI
      IF(JC EQ 1 .AND. TIMEI.GT. TTABI+.9999)                           DEMI
     1         SUPPLY = SUPFLY + TABEL(VAREA(1,IR,1),KTABI,TTABI,1 .0,  DEMI
     2              TIMEI - 1.0) * TABEL(VYIELD(1,IR,1),KTABI,          DEMI
     3              TTABI, 1.0, TIMEI - 1.0) * (1.-PFLOSS(1))           DEMI
   10 CONTINUE                                                          DEMI
      IF(JC.EQ.1 .AND. TIMEI .LT. TTABI + .9999) SUPPLY = SUPPLY +      DEMI
     1         TORLI * (1 - PFLOSS(JC))                                 DEMI
      SUPPLY = SUPPLY + TABEL(VDFC(1,JC),KTABI,TTABI,1.0,TIMEI) - PILOSS DEMI
     1         (JC) * AMAX1(0., TABEL(VDFC(1,JC),KTABI,TTABI,1.0,T IMEI)) DEMI
     2         - TABEL(VFEED(1,JC),KTABI,TTABI,1.0,TIMEI) - TABEL( VCSTK( DEMI
     3         1,JC),KTABI,TTABI,1.0,TIMEI)                             DEMI
      IF(JC .EQ. 3) SUPPLY = (1. - PWHTIC) * SUPPLY                     DEMI
      IF(JC .EQ. 9) SUPPLY = (1. - PLOSTD) * SUPPLY                     DEMI
      IF(JC.EQ.10 .OR. JC.EQ.11) SUPPLY = 0.                            DEMI
      IF(JC .EQ. 12) SUPPLY = TABEL(VPICF,KTABI,TTABI,1.0,TIMEI) *      DEMI
     1         SUPPLY                                                   DEMI
      GO TO 40                                                          DEMI
   20 CONTINUE                                                          DEMI
      IF(JC .GT. NCROP+NLVST) GO TO 30                                  DEMI
      LC = JC - NCROP                                                   DEMI
      SUPPLY = 0.                                                       DEMI
      DO 25 IR=1,NREGN                                                  DEMI
      SUPPLY = SUPPLY + TABEL(VLIVST(1,IR,LC),KTABI,TTABI,1.0,TIME I) * ( DEMI
     1         1. - PFLOSS(JC))                                         DEMI
      XX = XX + PEOLV(LC) * TABEL(VLIVST(1,IR,LC),KTABI,TTABI,1.0, TIMEI) DEMI
   25 CONTINUE                                                          DEMI
      GO TO 35                                                          DEMI
   30 CONTINUE                                                          DEMI
      IF(JC .EQ. 18) SUPPLY = TABEL(VFSH,KTABI,TTABI,1.0 TIMEI) - FDFSHD DEMI
```

```
      IF(JC .EQ. 19) SUPPLY = XX
   35 CONTINUE
      SUPPLY = SUPPLY + TABEL(VDFC(1,JC),KTABI,TTABI,1.0,TIMEI) -  PILOSS
     1         (JC) * AMAX1(0. , TABEL(VDFC(1,JC),KTABI,TTABI,1.0,T IMEI))
     2         - TABEL(VCSTK(1,JC),KTABI,TTABI,1.0,TIMEI)
   40 CONTINUE
      PCCONU(JC) = SUPPLY / ((FNR(JC)*TABEL(VPOP(1,1),KNP,TTABI,DP ,
     1             TIMEI) + TABEL(VPOP(1,2),KNP,TTABI,DP,TIMEI))*1 000 )
      DO 45 IR=1,NREGN
      PCCON(IR,JC) = FNR(JC) * PCCONU(JC)
   45 CONTINUE
      CPUU(JC) = VCPU(1,JC)
      X = X + PCCONU(JC) * TABEL(VCPU(1,JC),KTABI,TTABI,1.0,TIMEI)
   50 CONTINUE
      EINCOM = TABEL(VEINCM,KTABI,TTABI,1.0,TIMEI)
      ALPHA = X / (EINCOM * TABEL(VPCUFD,KTABI,TTABI,1.0,TIMEI))
      PCCONU(NCOM) = (ALPHA * EINCOM - X) / TABEL(VCP,KTABI,
     1               TTABI,1.0,TIMEI)
      S1 = 1.0
      DO 60 JC=1,NCOM
      DEMC(JC) = 0
      IF(PCCONT(JC)-PCCONU(JC) .NE. 0. )
     1     DEMC(JC) = ELISAV(JC) / (PCCONT(JC) - PCCONU(JC))
      IF(JC .EQ. 2)
     1     DEMC(2) = DEMC(2) * (PCCONT(2)-PCCONU(2)) / (PCCONT(2)-. 0465)
      IF(JC .GT. NCOMAG) GO TO 60
      DO 55 IR=1,NREGN
      EPAVG(IR,JC) = EPAVGO(JC)
      IF(TIMEI .GT. TTABI+.9999)
     1     EPAVG(IR,JC) = TABEL(VPRICE(1,JC),KTABI,TTABI,1.0,TIMEI- 1.)
      DEMT(IR,JC) = 0
      IF(PCRT(JC)-PCCON(IR,JC) .NE. 0. )
     1     DEMT(IR,JC) = ELISVR(JC) / (PCRT(JC)-PCCON(IR,JC))
   55 CONTINUE
   60 CONTINUE
C
      RETURN
      END
```

```
      SUBROUTINE DEMKX
C
C EXOGENOUS PROJECTIONS OF DEMAND INPUTS TO CHANGE AND RAP
C
C DECLARATIONS
C
      COMMON /CONTRC/ ANIMAL, BASEYR, DT, FARM, ITIME, KTRACK, KTA BI,
     1                IRUN, ITTY, ICHDAT, ISENPR, IPLT, ITABPR,
     1                LVSTPR, MODE, MODNAM(9),
     1                KPOP, KCHG, KRAP, KDEM, KNEC, LINK, JPER,
     1                LUIN1, LUIN2, LUOUT, LUPLOT(3), LURLP1,
     2                LUPGEN, LUPOP,
     2                KPPRT(15), KCPRT(15), KFPRT(15), KAPRT(15),
     2                KDPRT(15), KYPRT(15), KNPRT(15),
     2                NCC, NCOM, NCOMAG, NCROP, NLVST, NNC, NREGN,
     3                NRUN, NS, NAME(20), PLANT,
     4                T, TIMEI, TIMEF, TTABI, YEAR,
     5                TOTSOR, TOTPOP, URBAN
      INTEGER ANIMAL, FARM, PLANT, URBAN, TOTPOP, TOTSOR
      EQUIVALENCE (LUPLT1,LUPLOT(1)),(LUPLT2,LUPLOT(2)),
     1            (LUPLT3,LUPLOT(3))
      COMMON /DEMKC/ APCF, DEFCIT(20), PAVG(1,20), PFDIMX, TPAVG(2 0)
      COMMON /DKXDC/ ARGPP(5),KPP,VDFC(10,20),VDFCT(5,20),VPRICE(1 0,20),
     1               VPP(5,20),APCFD
C
C EXECTTION
C
      DO 20 JC=1,NCOMAG
      IF(T .GT. BASEYR+.9999) GO TO 4
      TPAVG(JC) = TABEL(VPRICE(1,JC), KTABI, TTABI, 1.0, T)
      DEFCIT(JC)= TABEL(VDFC(1,JC)   , KTABI, TTABI, 1.0, T)
      GO TO 6
    4 CONTINUE
      TPAVG(JC) = TABUL(VPP(1,JC)   , KPP, ARGPP, T)
      DEFCIT(JC)= TABUL(VDFCT(1,JC), KPP, ARGPP, T)
    6 CONTINUE
      DO 10 IR=1,NREGN
      PAVG(IR,JC) = TPAVG(JC)
   10 CONTINUE
   20 CONTINUE
      APCF = APCFD
C
      RETURN
      END
```

```
C      SUBROUTINE DEMPRT
C
C PRINT DEMAND OUTPUT TABLES
C
C DECLARATIONS
C
       COMMON /CPDEMC/ APCN, DELP, IGRAIN(5), IMEAT(5),
      2                MM(20), PILOSS(20), PLOSTD, PSUBCG(5,5,2), P CMIN,
      3                PSUBCM(5,5,2), PSUBTG(5,2), PSUBTM(5,2),PFU( 2,20),
      4                PCONVG(5), PWHTIC, YDEL, CALPU(20), PROTPU( 20),
      5                W(20)
       REAL MM
       COMMON /PPDEMC/ ARG(5), KFT(20), KNOR, KPAVG(20),
      1                PCCONT(20), PCPMAX(20), PCPMIN(20), PCRT(20),
      2                PCRT1, PCUT1, PWPMAX(20), PWPMIN(20), STKRT(2 0),
      3                TARIFF(20), VCPUT(5,20), VPCSUP(5,20),
      4                VPWLDI(5,20), VPWLDX(5,20)
       COMMON /DEMYC/ CON(1,20), DSAVD(1)
       COMMON /DEMSVC/ ALPHA, CPU(20), CPUO(20), DEMC(20),
      1                DEMT(1,20), EPAVG(1,20), INCOM, PCCON(1,20),
      2                PCCONU(20), Q(20), RFRMY(1), S1, STK(20),
      3                EINCOM, TDEM(20), TDEML(20), WWP(20)
       REAL INCOM
       COMMON /POPKC/ CALRPC(2), POP(2), POPL(2), POPR(1), PRORPC(2 ),
      1                TPCP, TRPOP, TPOPPC(2)
       COMMON /VDEMC/ CHGSTK(20), DSTK(20), DVALDF, EFRMY,
      1                ELASP(20,20), ELASPR(20,20),
      1                ELASI(20), ELASIR(20), FRMY(1), INAPA(20),
      1                FGIVAL, ITER, PCCONF(20), TARREV(20),
      2                PRFEXP(20), PRFIMP(20), PWLDD(20), PWLDEX(20) ,
      3                PWLDIM(20), RDEM(20), TRDPRF, TVALDF, VALDEF( 20)
       COMMON /CVDEMC/ AREQH(12), AREQHF(12), ARQHFS(12), CAL(3,3),
      1                PCAL(3), PPRO(2),
      1                CALPC(2,3), CMARKT(20), CMPI, COGPI, CPI, CR PI,
      2                GOVSUB(20), PCNCON(20), PRO(3,3),
      3                PROPC(2,3), QREQH(12), QREQHF(12), QRQHFS(12 ),
      4                QRQHLV(7), RFTEX, SAVUPC, SELFSU(20), TARH,
      5                TARHF, TARHFS, TDISAP(20), TEXNF, TEXPF, TEX PT,
      6                TEXPTC, TOTHER(20), TEXPFC, XDIFF, ZZ(20)
       COMMON /CONTRC/ ANIMAL, BASEYR, DT, FARM, ITIME, KTRACK, KTA BI,
      1                IRUN, ITTY, ICHDAT, ISENFR, IPLT, ITABPR,
      1                LVSTPR, MODE, MODNAM(9),
      1                KPOP, KCHG, KRAP, KDEM, KNEC, LINK, JPER,
      1                LUIN1, LUIN2, LUOUT, LUPLOT(3), LURLP1,
      2                LUPGEN, LUPOP,
      2                KPPRT(15), KCPRT(15), KFPRT(15), KAPRT(15),
      2                KDPRT(15), KYPRT(15), KNPRT(15),
      2                NCC, NCOM, NCOMAG, NCROP, NLVST, NNC, NREGN,
      3                NRUN, NS, NAME(20), PLANT,
      3                T, TIMEI, TIMEF, TTABI, YEAR,
      4                TOTSOR, TOTPOP, URBAN
      5 INTEGER ANIMAL, FARM, PLANT, URBAN, TOTPOP, TOTSOR
       EQUIVALENCE (LUPLT1,LUPLOT(1)),(LUPLT2,LUPLOT(2)),
      1            (LUPLT3,LUPLOT(3))
       COMMON /DEMKC/ APCF, DEFCIT(20), PAVG(1,20), PFDIMX, TPAVG(2 0)
       COMMON /DEMNC/ CPFI, FCONP, FCONU, TTAREV, TEXNFC, FEXNFC,
      1                UCONP, UCONU, VALDF1, VALDF2, WPFI
       COMMON /NECDYC/ AGINTX, CPUNF, FPCI, PRFRMY(1), UPCI, WOND,
      1                TAGDIP, TPTAX, TYG, TYNAG
C
       DIMENSION VALDFM(20),DEFCTM(20)
       DIMENSION Z(20)
       DIMENSION LPR(4)
       DATA LPR / 1HU, 1HL, 1HM, 1HF /
C
C EXECUTION
C
C      UTILITY FORMAT
C
    1  FORMAT(1H )
    2  FORMAT(1H1)
    3  FORMAT(1H0)
    4  FORMAT(////)
C
       ISECT = 6HDEMAND
       ITAB = 0
C
C      PRICE INDICES
C      =============
C
C
       ITAB = ITAB + 1
       IF(KDPRT(1) .EQ. 0) GO TO 191
       WRITE(LUOUT,2)
       WRITE(LUOUT,1006) ISECT, ITAB, YEAR
       WRITE(LUOUT,1007) CPI,CPU(20), CPFI,CRPI,COGPI,CMPI,WPFI
C
 1006  FORMAT(T2,6HTABLE ,A6,1H ,I2,1H.,2X,14HPRICE INDICES ,F5.0)
 1007  FORMAT( T2,49(1H-),/,T2,5HINDEX,T46,5HVALUE,/,T2,49(1H-),/,
      1        T43, 8H1970=100,//,T2,20HCONSUMER PRICE INDEX,T41,+2 PF10.1
      1        //T4,7HNONFOOD,T41,+2PF10.1
```

```
                                  T6,4HRICE,T41,F10.1,    /
     2    / / T4,4HFOOD,T41,F10.1,//,T6,4HMEAT,T41,F10.1,  //,
     3        T6,12HOTHER GRAINS,T41,F10.1,//,T6,4HMEAT,T41,F10.1,//,T2,49(1H-)   )
     4        T2,22HWORLD FOOD PRICE INDEX,T41,F10.1,//,T2,49(1H-)   )
  191 CONTINUE
C
C        SUPPLY AND CONSUMPTION OF AGRICULTURAL COMMODITIES
C        ==================================================
C
C
      ITAB = ITAB + 1
      IF(KDPRT(2) EQ 0   OR  KRAF EQ 0) GO TO 193
      WRITE(LUOUT,2)
      WRITE(LUOUT,1008) ISECT, ITAB, YEAR
      WRITE(LUOUT,1009)
      DO 5 I=1,NCOMAG
      IF(I.EQ.10  .OR   I.EQ.11) GO TO 5
      Z(I) = ZZ(I)  - DEFCIT(I)
      WRITE(LUOUT,1010) NAME(I),Z(I),DEFCIT(I),ZZ(I),RDEM(I),Q(I),
     1          TOTHER(I),TDISAP(I),SELFSU(I)
      IF(MOD(I,5).EQ.0) WRITE(LUOUT,1)
    5 CONTINUE
      WRITE(LUOUT,1011)
 1008 FORMAT(T2,6HTABLE  ,A6,1H.,I2,1H.,2X,24HSUPPLY AND DISAPPEARA NCE,
     1      29H OF AGRICULTURAL COMMODITIES ,F5.0)
 1009 FORMAT( T2,104(1H-),/,T24,15HDOMESTIC SUPPLY,T65,13HDISAPPEA RANCE
     1         T102,4HSELF,/,T2,9HCOMMODITY,T16,30(1H-),5X,40(1H-),
     2         T99,7HSUFFIC-,/,T16,30HPRODUCTION    IMPORTS      TOTA L,
     3         T57,34HFARM      URBAN       OTHER*    TOTAL,T101,5HIEN CY,/
     4         T2,104(1H-),/,T16,34(1H ),1X,5HTH MT,1X,34(1H.),T99,
     5         7HPERCENT,// )
 1010 FORMAT(      T2,A6,T16,3(-3PF10.0),5X,4(-3PF10.0),5X,+2PF10.0 )
 1011 FORMAT(T2,104(1H-),/,T2,34H*NOTE:  OTHER INCLUDES FEED,SEED,   ,
     1         48HINDUSTRIAL CONSUMPTION, STOCK CHANGE, AND LOSSES   )
  193 CONTINUE
C
C        PER CAPITA CONSUMPTION AND CONSUMPTION TARGETS
C        ==============================================
C        (NATIONAL, URBAN, FARM) BY COMMODITY
C        ====================================
C
      ITAB = ITAB + 1
      IF(KDPRT(3)  .EQ.  0) GO TO 991
      WRITE(LUOUT,2)
      WRITE(LUOUT,1012) ISECT, ITAB, YEAR
      WRITE(LUOUT,1013)
      DO 2016 I=1,NCOMAG
      X = PCCONT(I)
      XX= PCRT(I)
      IF(YEAR.LT.1980.  .OR.  I.NE.1) GO TO 2010
      X = PCUT1
      XX= PCRT1
 2010 CONTINUE
      WRITE(LUOUT,1014) NAME(I),PCNCON(I),PCCONU(I),X,PCCONF(I),XX
      IF(MOD(I,5).EQ.0) WRITE(LUOUT,1)
 2016 CONTINUE
      WRITE(LUOUT,1015)
C
 1012 FORMAT(T2,6HTABLE  ,A6,1H.,I2,1H.,2X,23HPER CAPITA CONSUMPTIO N ,
     1         34HAND CONSUMPTION TARGETS (NATIONAL,,/,T16,12HURBAN,  FARM)
     2         13HBY COMMODITY ,F5.0)
 1013 FORMAT( T2,80(1H-),/,T18,8HNATIONAL,T37,  9HU R B A N,T67,7HF   A R M
     1         /,T2,9HCOMMODITY,T15,  11HCONSUMPTION,T30,24(1H-),4X,
     2         24(1H-),/,T30,11HCONSUMPTION,T48,6HTARGET,4X,
     3         11HCONSUMPTION,T76,6HTARGET,/,T2,80(1H-),/,T17,26(1H .),
     4         3X,9HKG/CAP-YR,1X,26(1H.),// )
 1014 FORMAT(   T2,A6,T16,+3PF10.1,2X,2(3X,3PF10.1),2X,2(3X,3PF10.1 ))
 1015 FORMAT(T2,80(1H-))
  991 CONTINUE
C
C        URBAN INCOME AND EXPENDITURES
C        =============================
C
      ITAB = ITAB + 1
      IF(KDPRT(4)  .EQ.  0) GO TO 993
      X = POP(2) * EINCOM / APCN
      XX = EINCOM / APCN
      WRITE(LUOUT,2)
      WRITE(LUOUT,2015) ISECT, ITAB, YEAR
      WRITE(LUOUT,1016) X,XX, XDIFF, SAVUPC, TEXPT, TEXPTC, TEXPF,
     1                  TEXPFC, TEXNF, TEXNFC, RFTEX
C
 2015 FORMAT(T2,6HTABLE  ,A6,1H.,I2,1H.,2X,23HURBAN INCOME AND EXPE ND,
     1         7HITURES  ,F5.0)
 1016 FORMAT( T2,41(1H-),/,T2,4HITEM,T25,18HTOTAL    PER CAPITA,/,
     1         T2,41(1H-),/,T24,19HBI WON    TH WON/CAP      //
     2         T2,6HINCOME,  T20,-9PF10.0,3X,-3PF10.1    ,  4(/),
     3         T2,6HSAVING,  T20,-9PF10.0,3X,-3PF10.1    ,  4(/),
     4         T2,11HEXPENDITURE,T20,-9PF10.0,3X,-3PF10.1,  //,
     5         T5,4HFOOD,     T20,-9PF10.0,3X,-3PF10.1   ,  //,
     6         T5,8HNON-FOOD,T20,-9PF10.0,3X,-3PF10.1    ,  3(/),
     7         T36,7HPERCENT,//,T2,10HFOOD/TOTAL,T33,2PF10.1,
```

```
     8
  993 CONTINUE                   /,T2,41(1H-)
C
C         CALORIE AND PROTEIN CONSUMPTION PER CAPITA
C         ==========================================
C
      ITAB = ITAB + 1
      IF(KDPRT(5) .EQ. 0) GO TO 995
      TPC = 100.0
C
      WRITE(LUOUT,2)
      WRITE(LUOUT,80) ISECT, ITAB, YEAR
      WRITE(LUOUT,81)
C
      WRITE(LUOUT,82) CAL(TOTSOR,TOTPOP), CAL(TOTSOR,URBAN),
     1           CAL(TOTSOR,FARM),TPC,TPC,TPC,CAL(PLANT,TOTPOP),
     2           CAL(PLANT,URBAN),CAL(PLANT,FARM),
     3           CALPC(PLANT,TOTPOP),CALPC(PLANT,URBAN),CALPC(PLANT, FARM),
     4           CAL(ANIMAL,TOTPOP),CAL(ANIMAL,URBAN),CAL(ANIMAL,FAR M),
     5           CALPC(ANIMAL,TOTPOP),CALPC(ANIMAL,URBAN),
     6           CALPC(ANIMAL,FARM),PRO(TOTSOR,TOTPOP),PRO(TOTSOR,UR BAN),
     7           PRO(TOTSOR,FARM),TPC,TPC,TPC, PRO(PLANT,TOTPOP),
     8           PRO(PLANT,URBAN),PRO(PLANT,FARM),PROPC(PLANT,TOTPOP ),
     9           PROPC(PLANT,URBAN),PROPC(PLANT,FARM),PRO(ANIMAL,TOT POP),
     *           PRO(ANIMAL,URBAN),PRO(ANIMAL,FARM),PROPC(ANIMAL,TOT POP),
     1           PROPC(ANIMAL,URBAN),PROPC(ANIMAL,FARM)
C
      WRITE(LUOUT,83)
C
   80 FORMAT(T2,6HTABLE ,A6,1H.,I2,1H.,2X,20HCALORIE AND PROTEIN ,
     1         23HCONSUMPTION PER CAPITA ,F5.0)
   81 FORMAT(T2,88(1H-),/,T30,8HQUANTITY,T69,7HPERCENT,/,T2,8HNUTR IENT,
     1         T19,32(1H-),T56,34(1H-),/,T19,8HNATIONAL,T34,5HURBAN,
     2         T47,4HFARM,T58,8HNATIONAL,T73,5HURBAN,T86,4HFARM, /,
     3         T2,88(1H-),/,T17,11(1H.),1X11HCAL/CAP-DAY,1X,10(1H.),
     4         T56,12(1H.),1X,7HPERCENT,1X,13(1H.),//)
   82 FORMAT(T2,8HCALORIES,T17,3(F10.1,2X),3X,3(F10.1,2X),     /
     5         T5,5HPLANT    ,T17,3(F10.1,2X),3X,3(F10.1,2X),    /
     2         T5,6HANIMAL   ,T17,3(F10.1,2X),3X,3(F10.1,2X),    /
     4         T2,7HPROTEIN  ,T17,3(F10.1,2X),3X,3(F10.1,2X),    //
     5         T5,5HPLANT    ,T17,3(F10.1,2X),3X,3(F10.1,2X),    /
     6         T5,6HANIMAL   ,T17,3(F10.1,2X),3X,3(F10.1,2X)      )
   83 FORMAT(T2,88(1H-)   )
  995 CONTINUE
C
C         MARKETING PRICES AND COSTS IN CONSTANT 1970 WON BY COMMOD ITY
C         ========================================================= ===
C
      ITAB = ITAB + 1
      IF(KDPRT(6) .EQ. 0) GO TO 997
      WRITE(LUOUT,2)
      WRITE(LUOUT,91) ISECT, ITAB, YEAR
      WRITE(LUOUT,92)
C
      DO 19 JC=1,NCOMAG
      LAB = LPR(INAPA(JC)+3)
      WRITE(LUOUT,93) NAME(JC),TPAVG(JC),CMARKT(JC),GOVSUB(JC),CPU (JC),
     1                LAB,PWLDEX(JC),PWLDIM(JC)
      IF(MOD(JC,5).EQ.0) WRITE(LUOUT,1)
   19 CONTINUE
      WRITE(LUOUT,94) ITER,S1
      WRITE(LUOUT,95)
C
   91 FORMAT(T2,6HTABLE ,A6,1H.,I2,1H.,2X,21HMARKETING PRICES AND ,
     1         40HCOSTS IN CONSTANT 1970 WON BY COMMODITY ,F5.0)
   92 FORMAT(T2,100(1H-),/,T19,22HPRODUCER       MARKETING,T50,5HGOV T.,
     1         T61,8HCONSUMER,T82,6HEXPORT,T96,6HIMPORT,/
     2         T2,9HCOMMODITY,T22,5HPRICE,T36,5HCOSTS,T48,7HSUBSIDY,
     3         T64,6HPRICE*,T83,5HPRICE,T97,5HPRICE,/
     4         T2,100(1H-),/,T19,36(1H.),11H TH WON/MT ,36(1H.) //)
   93 FORMAT(T2,A6,T17,3(-3PF10.1,4X),-3PF10.1,A1,8X,
     1         2(-3PF10.1,4X))
   94 FORMAT(T2,100(1H-),/,T2,22H*COMPUTATIONAL NOTE 0 ,I2,1X,
     1        32HITERATIONS TO OBTAIN CONVERGENCE,/,T24,5HS1 = ,F6.2 )
   95 FORMAT(1H ,30HL - PRICE AT LOWER BOUND            /
     1         1H ,30HU - PRICE AT UPPER BOUND          /
     2         1H ,30HM - PRICE DETERMINED BY MARKET    /
     3         1H ,30HF - FIXED PRICE                   )
C
  997 CONTINUE
C
C         ELASTICITIES URBAN AND RURAL BY COMMODITY
C         =========================================
C
      ITAB = ITAB + 1
      IF(KDPRT(7).EQ.0 .OR. ITIME.EQ.0) GO TO 97
      WRITE(LUOUT,2)
      WRITE(LUOUT,270) ISECT, ITAB, YEAR
      DO 173 JC=1,NCOM
      WRITE(LUOUT,271) NAME(JC),ELASP(JC,JC),ELASI(JC),ELASPR(JC,J C),
```

```
      1                   ELASIR(JC)
      F(MOD(JC,5).EQ.0) WRITE(LUOUT,1)
  173 (ONTINUE
      URITE(LUOUT,272)
C
  270 FORMAT(T2,6HTABLE ,A6,1H.,I2,1H.,2X,23HELASTICITIES URBAN AN D ,
             19HRURAL BY COMMODITY ,                5HURBAN,T65,5HRURAL, /,
     1       F5.0,/,T2,77(1H-),/,T36,
     2       T2,9HCOMMODITY,T28,2(23(1H-),5X),/,T26,12HDIRECT PRIC E,
     3       T45,6HINCOME,T54,12HDIRECT PRICE,T73,6HINCOME,/
     4       T28,11HELASTICITY*,2X,10HELASTICITY,5X,11HELASTICITY* ,
     5       2X,10HELASTICITY,/,T2,77(1H-),//)
     6
  271 FORMAT(T2,A6,T25,2(3X,F10.2),2X,2(3X,F10.2)   )
  272 FORMAT(/,T2,77(1H-),/T2,"*CONSTANT VALUE DURING RUN",)
   97 CONTINUE
C
C          PRINT URBAN CROSSELASTICITES
C          ============================
C
       TAB = ITAB + 1
       IF(KDPRT(8).EQ.0 .OR. ITIME.EQ.0) GO TO 999
       WRITE(LUOUT,2)
       WRITE(LUOUT,180) ISECT, ITAB, YEAR
C
       DO 1334 I=1,NCOM
       IJC = 0
       DO 1332 J=1,NCOM                                       GO TO  1331
       IF (I.NE.J .AND. ELASP(I,J).NE.0.)                     GO TO  1332
C
C          THEN                                               CONTIN UE
 1331
       WRITE(LUOUT,181) NAME(I), NAME(J), ELASP(I,J)
       IJC = IJC + 1                                          GO TO  1332
C
C          (ELSE)
 1332 CONTINUE
C          ENDIF
C          -----
       IF (IJC.GE.1) WRITE(LUOUT,1)
 1334 CONTINUE
C
       WRITE(LUOUT,182)
  180 FORMAT(T2,6HTABLE ,A6,1H.,I2,1H.,2X,20HCROSS ELASTICITIES, ,
     1           15HURBAN AND FARM ,F5.0 /T2,48(1H-) /
     2               T45,5HCROSS,/,T2,8HQUANTITY,2X,11HWITH  PRICE,T40 ,
     3               10HELASTICITY,/,T2,48(1H-),               /,T2,
     4               17HURBAN CONSUMPTION,/,T2,17(1H-),/                  )
  181 FORMAT( T2,A6,10X,A6,T40,F10.2 )
  182 FORMAT(T2,16HFARM CONSUMPTION,/,T2,16(1H-))
C
       DO 1338 I=1,NCOM
       IJC = 0
       DO 1336 J=1,NCOM
       IF(I.EQ.J .OR. ELASPR(I,J).EQ.0.) GO TO 1336
       WRITE(LUOUT,181) NAME(I),NAME(J),ELASPR(I,J)
       IJC = IJC + 1
 1336 CONTINUE
       IF(IJC .GT. 1) WRITE(LUOUT,1)
 1338 CONTINUE
C
  999 CONTINUE
C
C          AGRICULTURAL EXPORT-IMPORT BALANCE
C          ==================================
C
C
       ITAB = ITAB + 1
       IF(KDPRT(9).EQ.0 .OR. KRAP.EQ.0) GO TO 310
       WRITE(LUOUT,2)
       WRITE(LUOUT,1004) ISECT, ITAB, YEAR
       WRITE(LUOUT,1005) VALDF1,VALDF2,WOND,TVALDF,DVALDF,TRDPRF
C
 1004 FORMAT(T2,6HTABLE ,A6,1H.,I2,1H.,2X,20HAGRICULTURAL EXPORT-,
     1           15HIMPORT BALANCE ,F5.0)
 1005 FORMAT(T2,70(1H-),/,T2,4HITEM,T39,5HUNITS,T67,5HVALUE,/,T2,7 0(1H-)
     1          //,T2,20HAGRICULTURAL EXPORTS ,T39,6HBI WON,T62,-9PF1 0.0,//
     2               T2,20HAGRICULTURAL IMPORTS ,T39,6HBI WON,T62,-9PF1 0.0,//
     3               T2,13HEXCHANGE RATE,T39,10HWON/DOLLAR   ,T62,0PF10 .0 ,//
     4               T2,32HAGRICULTURAL BALANCE OF PAYMENTS  ,
     5                                   T39,6HBI WON,T62,-9PF10.0,       /
     6                                   T39, 9HMI DOLLAR,T62,-6PF10 .0,  //
     7               T2,30HPROFIT FROM AGRICULTURAL TRADE,
     8               T39,6HMI WON,T62,-6PF10.0,//,T2,70(1H-)     )
     9
C
  310 CONTINUE
C
C          AGRICULTURAL IMPORTS AND EXPORTS BY COMMODITY
C          =============================================
C
C
       ITAB = ITAB + 1
```

```
      IF(KDPRT(10).EQ.0 .OR. KRAP.EQ.0) GO TO 320
      WRITE(LUOUT,2)
      WRITE(LUOUT,160) ISECT, ITAB, YEAR
      WRITE(LUOUT,161)
C
      DO 162 JC=1,NCOMAG
      DEFCTM(JC) = 0.
      IF(DEFCIT(JC) .GE. 0.) GO TO 165
      DEFCTM(JC) = -DEFCIT(JC)
      DEFCIT(JC) = 0
  165 VALDFM(JC) = 0.
      IF(VALDEF(JC) .GE. 0.) GO TO 166
      VALDFM(JC) = -VALDEF(JC)
      VALDEF(JC) = 0
  166 CONTINUE
C
      WRITE(LUOUT,163) NAME(JC),DEFCIT(JC),PWLDIM(JC),TARREV(JC),
     1                 VALDEF(JC),PRFIMP(JC),DEFCTM(JC),PWLDEX(JC) ,
     2                 VALDFM(JC),PRFEXP(JC)
C
      IF(MOD(JC,5).EQ.0) WRITE(LUOUT,1)
C
  162 CONTINUE
      WRITE(LUOUT,164)
C
  160 FORMAT(T2,6HTABLE ,A6,1H.,I2,1H.,2X,24HAGRICULTURAL IMPORTS  AND,
     1         21HEXPORTS BY COMMODITY ,F5.0)
  161 FORMAT(T2,126(1H-),/,T36,13HI M P O R T S,T96,13HE X P O R T  S,/
     1         T2,9HCOMMODITY,1X,62(1H-),5X,49(1H-),/,T29,6HIMPORT,T 42,
     2         6HTARIFF,T96,6HEXPORT, /,T14,8HQUANTITY,T30,5HPRICE,T 40,
     3         8HREVENUES,T55,6HVALUE*,T68,6HPROFIT,T81,8HQUANTITY,T 97,
     4         5HPRICE,T110,5HVALUE,T122,6HPROFIT,/,T2,126(1H-),/,T1 7,
     5         5HTH MT,T29,6HTH WON,3(7X,6HMI WON),T84,5HTH MT,T96,
     6         6HTH WON,2(7X,6HMI WON) / )
  163 FORMAT(T2,A6,T12,-3PF10.3,3X,-3PF10.1,3(3X,-6PF10.1),5X,-3PF 10.3,
     1         3X,-3PF10.1,2(3X,-6PF10.1) )
  164 FORMAT(//,T2,126(1H-),/,T2,26H* INCLUDES TARIFF REVENUES  )
  320 CONTINUE
C
      RETURN
      END


      BLOCKDATA DKXD
C
C DATA FOR SUBROUTINE DEMKX
C
C DECLARATIONS
C
      COMMON /DKXDC/ ARGPP(5),KPP,VDFC(10,20),VDFCT(5,20),VPRICE(1 0,20),
     1               VPP(5,20),APCFD
C
C DATA
C
      DATA KPP / 5 /
      DATA APCFD / .80 /
C
C
C             1970        1971        1972        1973        1974
C             1975        1976        1977        1978        1979
C
      DATA ((VPRICE(I,J),I=1,10),J=1,6) /
     1        74689.,     82289.,     95639.,     97111.,    112914.,
     1      5*113989.,
     2        40610.,     49636.,     56800.,     56758.,     57622.,
     2       5*64950.,
     3        23939.,     25899.,     29912.,     31669.,     38992.,
     3       5*40027.,
     4        33375.,     40263.,     41883.,     45730.,     49295.,
     4       5*48884.,
     5        54052.,     60739.,     57118.,     63875.,     68605.,
     5       5*75582.,
     6        82634.,     74159.,     79079.,     89060.,     90041.,
     6       5*88964. /
      DATA ((VPRICE(I,J),I=1,10),J=7,12) /
     7        44115.,     44722.,     35204 ,     36287.,     37474.,
     7       5*44635.,
     8        15079.,     18134.,     19333.,     23860.,     29277.,
     8       5*25041.,
     9       205001.,    242291.,    344639.,    360856.,    353010.,
     9      5*285715.,
     *       10*1000.,
     A       449099.,    504491.,    568244.,    943119.,    849692.,
     A      5*678252.,
     B       146898.,    148522.,    176244.,    349335.,    204025.,
     B      5*207560. /
      DATA ((VPRICE(I,J),I=1,10),J=13,20) /
```

```
     C       659575. ,   715122. ,   862390. ,   911580. ,   793192. ,
     C     5*678218 ,
     D       54987. ,    52869 ,     50473. ,    57330. ,    52375. ,
     D     5*54001. ,
     E      261115 ,    281650. ,   228525. ,   313720 ,    291323. ,
     E     5*342642. ,
     F      366070 ,    354943. ,   298557. ,   338853. ,   391851. ,
     F     5*431132 ,
     G      216364 ,    189027. ,   170634. ,   207117. ,   230347. ,
     G     5*240933. ,
     H       70190 ,     72256. ,    63547 ,     84898. ,    62898. ,
     H     5*72654. ,
     I       68413. ,   109570. ,    57012. ,    50425. ,    68472. ,
     I     5*70000. ,
     J      10*1.  /
C
      DATA ((VDFC(I,J),I=1,10),J=1,6) /
     1      541000 ,    907000. ,   584000. ,   437000. ,   206000. ,
     1    5*481000 ,
     2           0 ,         0. ,   254000 ,    350000. ,   299000. ,
     2    5*354000. ,
     3     1254000. ,  1532000. ,  1881000. ,  1835000 ,   1592000. ,
     3    5*1703000 ,
     4      279000. ,   378000. ,   460000. ,   576000. ,    56900. ,
     4    5*554300 ,
     5        1000. ,     5000. ,     1000. ,    -1000. ,     4000. ,
     5      5*-900 ,
     6       36000. ,    61000. ,    31000. ,    73000. ,    66000. ,
     6    5*56500. /
C
      DATA ((VDFC(I,J),I=1,10),J=7,12) /
     7        5000. ,    -4000. ,    -8000. ,   -22000. ,   -27000. ,
     7    5*-27900 ,
     8        4000 ,      4000. ,     4000. ,     4000. ,     4000. ,
     8     5*68700 ,
     9      -27400 ,    -20900. ,    -9300. ,   -27600. ,   -54000. ,
     9    5*-61700 ,
     *        10*0. ,
     A       -2126 ,     -2319. ,    -3244. ,    -3743. ,    -2528. ,
     A     5*-703 ,
     B        -200. ,     2200. ,    -1900. ,    14500. ,     8700. ,
     B     5*-235. /
C
      DATA ((VDFC(I,J),I=1,10),J=13,19) /
     C         600. ,      600. ,      200. ,      600. ,        0. ,
     C           0. ,        0. ,        0. ,        0. ,        0. ,
     D       54400. ,    18900. ,    60400. ,     8700. ,    14600. ,
     D     5*10400. ,
     E        -100. ,     -100. ,    -4200. ,    -1700. ,    -3700. ,
     E     5*-8100. ,
     F           0. ,        0. ,        0. ,        0. ,        0. ,
     F           0. ,        0. ,        0. ,        0. ,        0. ,
     G       -1500. ,     -600. ,    -1300. ,    -1200. ,        0. ,
     G           0. ,        0. ,        0. ,        0. ,        0. ,
     H     -181000. ,  -194000. ,  -248000. ,  -355000. ,  -407000. ,
     H    5*-546400. ,
     I      374800. ,   207200. ,   406900. ,   550900. ,   517700. ,
     I    5*445500. /
C
C
      DATA ARGPP /
     1        1976. ,     1981. ,     1986. ,     1991. ,     2001.  /
C
      DATA ((VPP(I,J),I=1,5),J=1,10) /
     1      120000. ,   120000. ,   120000. ,   120000. ,   120000. ,
     2       60000. ,    60000. ,    60000. ,    60000. ,    60000. ,
     3       45000. ,    45000. ,    45000. ,    45000. ,    45000. ,
     4       48000. ,    48000. ,    48000. ,    48000. ,    48000. ,
     5       70000. ,    70000. ,    70000. ,    70000. ,    70000. ,
     6       90000. ,    90000. ,    90000. ,    90000. ,    90000. ,
     7       40000. ,    40000. ,    40000. ,    40000. ,    40000. ,
     8       30000. ,    30000. ,    30000. ,    30000. ,    30000. ,
     9      400000. ,   400000. ,   400000. ,   400000. ,   400000. ,
     *        1000. ,     1000. ,     1000. ,     1000. ,     1000. /
      DATA ((VPP(I,J),I=1,5),J=11,20) /
     A      855000. ,   855000. ,   855000. ,   855000. ,   855000. ,
     B      200000. ,   200000. ,   200000. ,   200000. ,   200000. ,
     C      800000. ,   800000. ,   800000. ,   800000. ,   800000. ,
     D       55000. ,    55000. ,    55000. ,    55000. ,    55000. ,
     E      300000. ,   300000. ,   300000. ,   300000. ,   300000. ,
     F      400000. ,   400000. ,   400000. ,   400000. ,   400000. ,
     G      250000. ,   250000. ,   250000. ,   250000. ,   250000. ,
     H       70000. ,    70000. ,    70000. ,    70000. ,    70000. ,
     I       74074. ,    74074. ,    74074. ,    74074. ,    74074. ,
     J           1. ,        1. ,        1. ,        1. ,        1.  /
C
      DATA ((VDFCT(I,J),I=1,10),J=1,12) /
     1      200000. ,        0. ,        0. ,        0. ,        0. ,
     2      100000. ,        0. ,        0. ,        0. ,        0. ,
     3     1900000. ,  1900000. ,  1900000. ,  1900000. ,  1900000. ,
     4      500000. ,   400000. ,   400000. ,   400000. ,   400000. ,
```

```
     5          -15000. ,   -25000. ,   -25000. ,   -25000. ,   -25000. ,
     6           50000. ,       0. ,       0. ,       0. ,       0. ,
     7               0. ,       0. ,       0. ,       0. ,       0. ,
     8               0. ,       0. ,       0. ,       0. ,       0. ,
     9          -60000. ,   -60000. ,   -60000. ,   -60000. ,   -60000. ,
     *               0. ,       0. ,       0. ,       0. ,       0. ,
     A           -2500. ,    -3500. ,    -4000. ,    -4500. ,    -5000. ,
     B               0. ,       0. ,       0. ,       0. ,       0. /
C
      DATA ((VDFCT(I,J),I=1,10),J=13,19) /
     C               0. ,       0. ,       0. ,       0. ,       0. ,
     D               0. ,       0. ,       0. ,       0. ,       0. ,
     E           -4000. ,    -4500. ,    -5000. ,    -5000. ,    -5000. ,
     F               0. ,       0. ,       0. ,       0. ,       0. ,
     G               0. ,       0. ,       0. ,       0. ,       0. ,
     H         -500000. ,  -600000. ,  -800000. ,  -800000. ,  -800000. ,
     I          520000. ,   520000. ,   520000. ,   520000. ,   520000.  /
C
      END


      BLOCKDATA FAYXD
C
C DATA FOR SUBROUTINE FRAAYX
C
C DECLARATIONS
C
      COMMON /FAYXDC/ ARGRA(5), KRA, VLIVST(10,1,5), VLVSTT(5,1,5) ,
     1                VRLIN(10,1),VRLINT(5,1)
C
C DATA
C
      DATA KRA / 5 /
C
C
      DATA ARGRA /
     1         1976. ,     1981. ,     1986. ,     1991. ,     2001.  /
C
      DATA VLVSTT /
     1        56900. ,   116200. ,   150000. ,   150000. ,   150000. ,
     2       192300. ,   499400. ,   850000. ,   850000. ,   850000. ,
     3       108800. ,   159600. ,   200000. ,   200000. ,   200000. ,
     4        59900. ,    84100. ,   120000. ,   120000. ,   120000. ,
     5       155000. ,   240700. ,   285000. ,   285000. ,   285000.  /
C
      DATA VRLINT /
     1        15000. ,    10000. ,     1000. ,     1000. ,     1000.  /
C
C              1970        1971        1972        1973        1974
C              1975        1976        1977        1978        1979
C
      DATA VLIVST /
     1        37300. ,    39500. ,    40200. ,    44900. ,    51506. ,
     1      5*70292. ,
     2        51900. ,    65300. ,    81100. ,   104100. ,   124745. ,
     2     5*160338. ,
     3        82500. ,    80900. ,    90200. ,    90100. ,    95353. ,
     3     5*106723. ,
     4        45200. ,    50000. ,    54300. ,    51800. ,    53209. ,
     4      5*55594. ,
     5       135100. ,   139400. ,   153500. ,   137500. ,   151525. ,
     5     5*159280.  /
C
      DATA VRLIN /
     1            0. ,        0. ,     6500. ,     9500. ,     5000. ,
     2         5000. ,     5000. ,     5000. ,     5000. ,     5000.  /
C
      END


      BLOCKDATA FCAXD
C
C DATA FOR SUBROUTINE FRACAX
C
C DECLARATIONS
C
      COMMON /FCAXDC/ ARGRCR(5), KRCR, VAREA(10,1,13), VAREAT(5,1, 13)
C
C DATA
C
      DATA KRCR / 5 /
C
C
```

```
      DATA ARGRCR /
     1     1976.,      1981.,      1986.,      1991.,      2001. /
C
      DATA VAREAT /
     1  1212400.,   1267000.,   1275000.,   1275000.,   1275000.,
     2   699800.,    706000.,    725000.,    725000.,    725000.,
     3    33100.,    108000.,    150000.,    150000.,    150000.,
     4    92500.,     89000.,     90000.,     90000.,     90000.,
     5    73400.,    133400.,    155000.,    155000.,    155000.,
     6   370000.,    395000.,    400000.,    400000.,    400000.,
     7   270000.,    273800.,    280000.,    280000.,    280000.,
     8   156100.,    181000.,    185000.,    185000.,    185000.,
     9    59900.,     64000.,     75000.,     75000.,     75000.,
     *        0.,         0.,         0.,         0.,         0.,
     A    90000.,     90000.,    100000.,    100000.,    100000.,
     B   100000.,    112900.,    125000.,    125000.,    125000.,
     P   690000.,    750000.,    750000.,    750000.,    750000. /
C
C
C          1970        1971        1972        1973        1974
C          1975        1976        1977        1978        1979
C
      DATA ((VAREA(I,1,J),I=1,10),J=1,6) /
     1  1203330.,  1190449.,  1191101.,  1181718.,  1204416.,
     1  5*1218012.,
     2   730269.,   677027.,   710174.,   665928.,   703690.,
     2  5*711010.,
     3    96740.,    86650.,    62735.,    43218.,    36477.,
     3  5*43710.,
     4   129721.,   104384.,    89950.,    95092.,    86021.,
     4  5*79270.,
     5    60185.,    55326.,    59189.,    65679.,    75129.,
     5  5*88448.,
     6   365182.,   337838.,   340115.,   369710.,   349392.,
     6  5*341492. /
      DATA ((VAREA(I,1,J),I=1,10),J=7,13) /
     7   254279.,   257126.,   247915.,   254213.,   274055.,
     7  5*276460.,
     8   180403.,   163291.,   147432.,   138241.,   123372.,
     8  5*147271.,
     9    43002.,    40974.,    58147.,    55542.,    54306.,
     9  5*53897.,
     *        0.,        0.,        0.,        0.,        0.,
     *        0.,        0.,        0.,        0.,        0.,
     A    84977.,    81356.,    78441.,    80267.,    88006.,
     A  5*90955.,
     B    89212.,    90581.,    81671.,    83288.,   107036.,
     B  5*100080.,
     P   250000.,   315000.,   415000.,   560000.,   735000.,
     P   570000.,   690000.,   530000.,   625000.,   715000. /
C
      END


      BLOCKDATA NDXD
C
C DATA FOR SUBROUTINE NECDX
C
C DECLARATIONS
C
      COMMON /NDXDYC/ ARGFX(4), ARGND(5), KFX, KND, VCP(10),
     1                VCPT(5), VFI(10), VFIT(5), VUI(10), VUIT(5),
     2                VWOND(10), VFXR(4)
C
C DATA
C
      DATA KND, KFX / 5, 4 /
C
      DATA ARGFX / 1976., 1977., 1981., 1986. /
C
C
C          1970        1971        1972        1973        1974
C          1975        1976        1977        1978        1979
C
      DATA VCP /
     1     1.000,      .959,       .946,       .952,       .922,
     2      .95,       .95,        .95,        .95,        .95 /
C
      DATA VFI /
     1    44028.,    52950.,    56534.,    61458.,    65000.,
     2    70000.,    70000.,    70000.,    70000.,    70000. /
C
      DATA VUI /
     1    84418.,    83735.,    88877.,    86737.,    90698.,
     2  5*92575. /
C
      DATA VWOND /
```

```
     1          3?4. 5,      316. 7,      373. 3,      398. 9,      399. 0,
     2          5*4?4. 0 /
C
C
      DATA ARGN? /
     1          1976. ,      1981. ,      1986. ,      1991. ,      2001.  /
C
      DATA VCPT  /
     1            . 95,        . 95,        . 95,        . 95,        . 95 /
C
      DATA VFI?  /
     1          70000. ,    103000. ,    151000. ,    193000. ,    314000.  /
C
      DATA VUIT /
     1          94118. ,    118244. ,    150912. ,    192606. ,    313735.  /
C
C
C               1976         1977         1981         1986
      DATA VFXR /
     1          485. 0,      500. 0,      500. 0,      500. 0 /
C
      END


      BLOCKDATA PKXD
C
C DATA FOR SUBROUTINE POPKX
C
C DECLARATIONS
C
      COMMON /PKXDC/ DP, DPOPRD(1), KNP, VPOP(31,2)
C
C DATA
C
      DATA DP, DPOPRD(1), KNP / 1.0, 1.0, 31 /
C
      DATA VPOP /
     1          14568. ,     14335. ,     14063. ,     13800. ,     13547. ,
     2          13309. ,     13089. ,     12893. ,     12685. ,     12543. ,
     3          12395. ,     12280. ,     12072. ,     11959. ,     11848. ,
     4          11739. ,     11628. ,     11514. ,     11396. ,     11273. ,
     5          11142. ,     11006. ,     10863. ,     10712. ,     10553. ,
     6          10386. ,     10216. ,     10042. ,      9866. ,      9687. ,
     7           9505. ,
     1          17766. ,     18619. ,     19501. ,     20358. ,     21190. ,
     2          21994. ,     22772. ,     23527. ,     24297. ,     25010. ,
     3          25743. ,     26457. ,     26906. ,     27620. ,     28344. ,
     4          29075. ,     29821. ,     30579. ,     31344. ,     32114. ,
     5          32885. ,     33696. ,     34424. ,     35188. ,     35947. ,
     6          36699. ,     37443. ,     38179. ,     38907. ,     39628. ,
     7          40342. /
C
      END


      SUBROUTINE PRICE
      COMMON /CONTRC/ ANIMAL, BASEYR, DT, FARM, ITIME, KTRACK, KTA BI,
     1                IRUN, ITTY, ICHDAT, ISENPR, IPLT, ITABPR,
     1                LVSTFR, MODE, MODNAM(9),
     1                KPOP, KCHG, KRAP, KDEM, KNEC, LINK, JPER,
     2                LUIN1, LUIN2, LUOUT, LUPLOT(3), LURLP1,
     2                LUPGEN, LUPOP,
     2                KPPRT(15), KCPRT(15), KFPRT(15), KAPRT(15),
     2                KDPRT(15), KYPRT(15), KNPRT(15),
     3                NCC, NCOM, NCOMAG, NCROP, NLVST, NNC, NREGN,
     3                NRUN, NS, NAME(20), PLANT,
     4                T, TIMEI, TIMEF, TTABI, YEAR,
     5                TOTSOR, TOTPOP, URBAN
      INTEGER ANIMAL, FARM, PLANT, URBAN, TOTPOP, TOTSOR
      EQUIVALENCE (LUPLT1,LUPLOT(1)),(LUPLT2,LUPLOT(2)),
     1            (LUPLT3,LUPLOT(3))
      COMMON /DEMKC/ APCF, DEFCIT(20), PAVG(1,20), PFDIMX, TPAVG(2 0)
      COMMON /NECDYC/ AGINTX, CPUNF, FPCI, PRFRMY(1), UPCI, WOND,
     1                TAGDIP, TPTAX, TYG, TYNAG
      COMMON /POPKC/ CALRPC(2), POP(2), POPL(2), POPR(1), PRORPC(2 ),
     1                TPOP, TRPOP, TPOPPC(2)
      COMMON /PRDDC/ AYLD(12), EFFPLV(7), EFFYLD(12), FDFSH, FEED( 12),
     1                FGIM, FLOSS(20), TAREA(12), TDSUP(20), TOUTRL ,
     2                TTAREA
      COMMON /CPDEMC/ APCN, DELP, IGRAIN(5), IMEAT(5),
     2                MM(20), PILOSS(20), PLOSTD, PSUBCQ(5,5,2), P CMIN,
     3                PSUBCM(5,5,2), PSUBTQ(5,2), PSUBTM(5,2),PFU( 2,20),
     4                PCONVQ(5), PWHTIC, YDEL, CALPU(20), PROTPU( 20),
```

```
     5                   W(20)
      REAL MM
      COMMON /ICDEMC/ ELISAV(20), ELISVR(20), EPAVGO(20),
     1                STKO(20), VCPU(10,20), VCSTK(10,20), VEFRMY( 10),
     2                VFPIX(10), VPSUP(10,20), VPWI(10,20),
     3                VPWX(10,20), FNR(20), VPICF(10), VEINCM(10),
     4                VPCUFD(10)
      COMMON /PPDEMC/ ARG(5), KFT(20), KNOR, KPAVG(20),
     1                PCCONT(20), PCPMAX(20), PCPMIN(20), PCRT(20),
     2                PCRT1, PCUT1, PWPMAX(20), PWPMIN(20), STKRT(2 0),
     3                TARIFF(20), VCPUT(5,20), VPCSUP(5,20),
     4                VPWLDI(5,20), VPWLDX(5,20)
      COMMON /DEMSVC/ ALPHA, CPU(20), CPUO(20), DEMC(20),
     1                DEMT(1,20), EPAVG(1,20), INCOM, PCCON(1,20),
     2                PCCONU(20), Q(20), RFRMY(1), S1, STK(20),
     3                EINCOM, TDEM(20), TDEML(20), WWP(20)
      REAL INCOM
      COMMON /VDEMC/  CHGSTK(20), DSTK(20), DVALDF, EFRMY,
     1                ELASP(20,20), ELASPR(20,20),
     1                ELASI(20), ELASIR(20), FRMY(1), INAPA(20),
     1                FGIVAL, ITER, PCCONF(20), TARREV(20),
     2                PRFEXP(20), PRFIMP(20), PWLDD(20), PWLDEX(20) ,
     3                PWLDIM(20), RDEM(20), TRDPRF, TVALDF, VALDEF( 20)
C
      REAL INCOML
      DIMENSION AE(20,20), AINTER(20), AINV(21,21), BDEM(20), RTHS (21),
     1          CPUMAX(20), CPUMIN(20), CPUL(20), INIV(21), JIN(20 )
C
C         INAPA CODES
C            -2 PRICE IS AT UPPER BOUND
C            -1 PRICE IS AT LOWER BOUND
C             0 PRICE IS DETERMINED BY MARKET MECHANISM
C             1 PRICE IS FIXED BY GOVERNMENT
C
      ITER = 0
      DO 2 I=1,NCOM
      IF(ITIME .GT. 0) CPUL(I) = CPU(I)
      INAPA(I) = 0
      IF(KFT(I).EQ.5 .OR. T.LT.BASEYR+.9999) INAPA(I) = 1
      IF(KRAP .EQ. 0) INAPA(I) = 1
    2 CONTINUE
C
C         CALCULATE CONSUMER PRICES  (CPU(J), J=1,19)
C
      DO 30 J=1,NCOMAG
      IF(T .LT. BASEYR+.9999)
     1   CPU(J) = TABEL(VCPU(1,J), KTABI,TTABI, 1.0, T)
      IF(T.GT.BASEYR+.9999 .AND. INAPA(J).EQ.1)
     1   CPU(J) = TABUL(VCPUT(1,J), KNOR, ARG, T)
   30 CONTINUE
      CPU(NCOM) = CPUNF
C
C         POLICY-SPECIFIED BOUNDS ON PRICES
C
      IF(T .LT. BASEYR+.9999) GO TO 36
      DO 35 J=1,NCOMAG
      IF(KFT(J) .NE. 3 .AND.KFT(J) .NE. 4)GO TO 35
      CPUMIN(J) = AMAX1(PWPMIN(J)*PWLDEX(J), (1.-DT*PCPMIN(J))*CPU L(J))
      CPUMAX(J) = AMIN1(PWPMAX(J)*PWLDIM(J), (1.+DT*PCPMAX(J))*CPU L(J))
   35 CONTINUE
   36 CONTINUE
      IF(ITIME .EQ. 0) GO TO 194
C
C         CALCULATE AE(I,J) AND B(I)
C
      DO 49 J=1,5
      JG = IGRAIN(J)
      JM = IMEAT(J)
      DO 48 I=1,5
      IF(I .EQ. J)GO TO 48
      IG = IGRAIN(I)
      IM = IMEAT(I)
      ELASP(IG,JG) = -PSUBTG(J,2) * PSUBCG(I,J,2) * ELASP(JG,JG)
     1               * PCCONU(JG) / PCCONU(IG) * (PCONVG(J) / PCONVG (I))
      ELASP(IM,JM) = -PSUBTM(J,2) * PSUBCM(I,J,2) * ELASP(JM,JM)
     1               * PCCONU(JM) / PCCONU(IM)
   48 CONTINUE
   49 CONTINUE
C
C         INCOME IS LAGGED
      IF(ITIME .EQ. 1) INCOM = (EINCOM/APCN) * POP(2) / POPL(2)
      EINCML = EINCOM
      EINCOM = APCN * UPCI * POPL(2) /POP(2)
      INCOML = INCOM
      INCOM = INCOM + (DT/YDEL) * (UPCI - INCOM)
C
C         INCOME ELASTICITIES
      DO 55 I=1,NCOM
      ELASI(I) = DEMC(I) * (PCCONT(I) - PCCONU(I))
C
```

```
      IF(I.NE.1 .OR. YEAR.LT.1980.) GO TO 55
      PCT1 = AMAX1(PCUT1,PCCONT(1)-(YEAR-1980.)*(PCCONT(1)-PCUT1)/ 20.)
      ELASI(1) = DEMC(1) * (PCT1 - PCCONU(1))
   55 CONTINUE
C
C        THE HOMOGENEITY POSTULATE FROM DEMAND THEORY +
C       "THE SUM OF INCOME PLUS OWN-PRICE PLUS CROSS-PRICE
C        ELASTICITIES FOR EACH COMMODITY MUST EQUAL ZERO "
C
      DO 80 I=1,NCOM
      IF(I.GT.NCOMAG) GO TO 60
C         NON-AGRICULTURAL CONSUMPTION IS TREATED UNIQUELY
C
      ELASP(NCOM,I) = -(ELASP(NCOM,NCOM) + ELASI(NCOM))*Q(I)*CPUL( I)/
     *                 (ALPHA*EINCML*POPL(2) - Q(NCOM)*CPUL(NCOM))
C
      ELASP(I,NCOM) = -ELASI(I)
C
      DO 50 J=1,NCOMAG
      ELASP(I,NCOM) = ELASP(I,NCOM) - ELASP(I,J)
   50 CONTINUE
C
   60 CONTINUE
      DO 70 J=1,NCOM
      AE(I,J) = PCCONU(I) * S1 * ELASP(I,J) / CPUL(J)
   70 CONTINUE
   80 CONTINUE
C
      DO 100 I=1,NCOM
      X = ELASI(I) * ALOG(INCOML)
      DO 90 J=1,NCOM
      X = X + ELASP(I,J) * ALOG(CPUL(J))
   90 CONTINUE
      AINTER(I) = PCCONU(I) * S1 * (ELASI(I) * (INCOM/INCOML) - X)
     1            + PCCONU(I)
      BDEM(I) = PCCONU(I) * X
  100 CONTINUE
C
C
C                    BEGIN PRICE ADJUSTMENT LOOP
C
   40 CONTINUE
      ITER = ITER + 1
C
      NIN = 0
      DO 46 I=1,NCOM
      IF(INAPA(I) .NE. 0)GO TO 46
      NIN = NIN + 1
      JIN(NIN) = I
   46 CONTINUE
      NINT1 = NIN + 1
C
      IF(NIN .EQ. 0) GO TO 191
C
C         MECHANISM TO CHANGE PRICES
C
C         DEFINE PRICE RESPONSE MATRIX
C
      DO 130 I=1,NIN
      K = JIN(I)
      RTHS(I) = (TDSUP(K) + DEFCIT(K)) / POP(2) - AINTER(K)
      DO 120 J=1,NCOM
      IF(INAPA(J) .EQ. 0)GO TO 120
      RTHS(I)=RTHS(I)-CPU(J)*AE(K,J)
  120 CONTINUE
      DO 130 J=1,NIN
      L = JIN(J)
      AINV(I,J) = AE(K,L)
  130 CONTINUE
      RTHS(NINT1) = ALPHA * EINCOM
      NN = 0
C
C          COMPLETE THE MATRIX FOR JOINT DETERMINATION OF CPU AN D S
C
      DO 160 N=1,NCOM
      IF(INAPA(N) .NE. 0)GO TO 150
      NN = NN + 1
      AINV(NN,NINT1) = BDEM(N)
      AINV(NINT1,NN) = (TDSUP(N) + DEFCIT(N)) / POP(2)
  150 CONTINUE
  160 CONTINUE
      AINV(NINT1,NINT1) = 0.
C
      DO 166 I=1,NCOM
      IF(INAPA(I) .EQ. 0)GO TO 166
      AINV(NINT1,NINT1) = AINV(NINT1,NINT1) + CPU(I) * BDEM(I)
      DO 162 L=1,NIN
      K=JIN(L)
      AINV(NINT1,L) = AINV(NINT1,L) + CPU(I) * AE(I,K)
  162 CONTINUE
      X = 0.0
      DO 164 J=1,NCOM
```

```
      IF(INAPA(J) .EQ. 0)GO TO 164
      X = X + AE(I,J) * CPU(J)
  164 CONTINUE
      RTHS(NINT1) = RTHS(NINT1) - CPU(I) * (AINTER(I) + X)
  166 CONTINUE
      DO 170 I=1,NINT1
      DO 170 J=1,NINT1
  170 AINV(I,J) = AINV(I,J) * (2**15)
C
C
      CALL MATINV(AINV, INIV, NCOM+1, NINT1)
C
C
      DO 180 I=1,NINT1
      DO 180 J=1,NINT1
  180 AINV(I,J) = AINV(I,J) * (2**15)
      IF(KDPRT(15) .EQ. 0) GO TO 184
      PRINT 1090,ITER
  184 CONTINUE
      DO 200 I=1,NIN
      K = JIN(I)
      CPU(K) = 0.0
      DO 200 J=1,NINT1
      CPU(K)=CPU(K)+AINV(I,J)*RTHS(J)
  200 CONTINUE
C
C          CONVERGENCE MECHANISM WITH BOUNDED PRICES
C
      KCHECK = 0
      IMAX = 0
      VMAX = 0
      DO 320 I=1,NIN
      K = JIN(I)
      IF(KDPRT(15) .EQ. 0) GO TO 204
      PRINT 1000,K,CPU(K),PCCONU(K),Q(K)
  204 CONTINUE
      IF(KFT(K) .LE. 2)GO TO 315
      IF(CPU(K) .GE. CPUMIN(K) .AND. CPU(K) .LE. CPUMAX(K))GO TO 3 15
      KCHECK = 1
      IF(CPU(K) .LT. CPUMIN(K))XX = (CPUMIN(K) - CPU(K)) / CPUMIN( K)
      IF(CPU(K) .GT. CPUMAX(K))XX = (CPU(K) - CPUMAX(K)) / CPUMAX( K)
      IF(XX .LE. VMAX)GO TO 315
      VMAX = XX
      IMAX = K
  315 CONTINUE
  320 CONTINUE
      IF(KCHECK .EQ. 0)GO TO 322
      IF(CPU(IMAX) .GE. CPUMIN(IMAX))GO TO 321
      INAPA(IMAX) = -1
      CPU(IMAX) = CPUMIN(IMAX)
      GO TO 40
  321 INAPA(IMAX) = -2
      CPU(IMAX) = CPUMAX(IMAX)
      GO TO 40
  322 CONTINUE
C
      S1 = 0.
      DO 190 J=1,NINT1
      S1 = S1 + AINV(NINT1,J) * RTHS(J)
  190 CONTINUE
      GO TO 193
  191 X = 0.
      Y = 0.
      Z = 0.
      DO 192 I=1,NCOM
      X = X + CPU(I) * AINTER(I)
      Y = Y + CPU(I) * BDEM(I)
      DO 192 J=1,NCOM
      Z = Z + CPU(I) * AE(I,J) * CPU(J)
  192 CONTINUE
      S1 = (ALPHA * EINCOM - X - Z) / Y
  193 CONTINUE
      IF(KDPRT(15) .EQ. 0) GO TO 194
      PRINT 1100, S1
      PRINT 1090,ITER
  194 CONTINUE
      DO 310 I=1,NCOM
      IF(ITIME .EQ. 0) GO TO 302
      Z = 0.
      DO 300 J=1,NCOM
      Z = Z + AE(I,J) * CPU(J)
  300 CONTINUE
      PCCONU(I) = AINTER(I) + Z + BDEM(I)*S1
  302 CONTINUE
      Q(I) = PCCONU(I) * POP(2)
      IF(KDPRT(15) .EQ. 0) GO TO 304
      PRINT 1000,I,CPU(I),PCCONU(I),Q(I)
  304 CONTINUE
  310 CONTINUE
      IF(KDPRT(14) .NE. 0) PRINT 1200, YEAR, S1, ITER
C          END OF PRICE ADJUSTMENT LOOP.
```

```
C
C
C        REINITIALIZE CONSUMPTION IN BASEYR
C
      IF(ABS(BASEYR-T).GT..0001 .OR. KTRACK.NE.0) GO TO 450
      X = 0.
      DO 410 I=1,NCOMAG
      PCCONU(I) = (TABEL(VPSUP(1,I), KTABI, TTABI, 1.0, T) * TPOP
     1            - RDEM(I)) / POP(2)
      X = X + CPU(I) * PCCONU(I)
      Q(I) = PCCONU(I) * POP(2)
  410 CONTINUE
      ALPHA = X/(EINCOM * TABEL(VPCUFD,KTABI,TTABI,1.0,BASEYR))
      PCCONU(NCOM) = ALPHA * EINCOM - X
      S1 = 1.0
  450 CONTINUE
C
C        FARM PRICES
C
      CALL DEMKX
      DO 340 J=1,NCOM
      IF(J.GT.NCOMAG .OR. KPAVG(J).NE.0 .OR. T.LT.BASEYR+.9999)GO  TO 340
      TPAVG(J) = CPU(J) / (1.+MM(J))
      DO 330 IR=1,NREGN
      PAVG(IR,J) = TPAVG(J)
  330 CONTINUE
  340 CONTINUE
      RETURN
 1000 FORMAT(/6X,I4,3(4X,E10.4))
 1090 FORMAT(9X,1HK,11X,3HCPU,8X,6HPCCONU,13X,1HQ,4X,6HITER =,I3)
 1100 FORMAT(1H1,10X,5HS1 = ,F6.2)
 1200 FORMAT(5X,F5.0,9H    S1  = ,F6.2,10H    ITER = ,I3)
C
      END



      BLOCKDATA RDDAT
C
C  DATA NEEDED IN BOTH RAP AND DEMAND
C
C  DECLARATIONS
C
      COMMON /RDDATC/ PFLOSS(20), SDPHA(1,12)
C
C  DATA
C
      DATA PFLOSS /
     1            .07, .07, .07, .03, .10, .03, .15, .14, 0.0, 0.0,
     2            .01, .03, .02, .036, .02, .02, .02, .05, .03, 0.0  /
C
      DATA SDPHA /
     1     .0288,     .0439,     .0550,     .0370,       0.0,     .0540,
     2       0.0,     .9800,       0.0,     .0200,       0.0,       0.0  /
C
      END



      SUBROUTINE RURDEM
      COMMON /CONTRC/ ANIMAL, BASEYR, DT, FARM, ITIME, KTRACK, KTA BI,
     1                IRUN, ITTY, ICHDAT, ISENPR, IPLT, ITABPR,
     1                LVSTPR, MODE, MODNAM(9),
     1                KPOP, KCHG, KRAP, KDEM, KNEC, LINK, JPER,
     1                LUIN1, LUIN2, LUOUT, LUPLOT(3), LURLP1,
     2                LUPGEN, LUPOP,
     2                KPPRT(15), KCPRT(15), KFPRT(15), KAPRT(15),
     2                KDPRT(15), KYPRT(15), KNPRT(15),
     3                NCC, NCOM, NCOMAG, NCROP, NLVST, NNC, NREGN,
     4                NRUN, NS, NAME(20), PLANT,
     5                T, TIMEI, TIMEF, TTABI, YEAR,
     5                TOTSOR, TOTPOP, URBAN
      INTEGER ANIMAL, FARM, PLANT, URBAN, TOTPOP, TOTSOR
      EQUIVALENCE (LUPLT1,LUPLOT(1)),(LUPLT2,LUPLOT(2)),
     1            (LUPLT3,LUPLOT(3))
      COMMON /DEMKC/ APCF, DEFCIT(20), PAVG(1,20), PFDIMX, TPAVG(2 0)
      COMMON /NECDYC/ AGINTX, CPUNF, FPCI, PRFRMY(1), UPCI, WOND,
     1                TAGDIP, TPTAX, TYG, TYNAG
      COMMON /POPKC/ CALRPC(2), POP(2), POPL(2), POPR(1), PRORPC(2 ),
     1               TPOP, TRPOP, TPOPPC(2)
      COMMON /PRDDC/ AYLD(12), EFFPLV(7), EFFYLD(12), FDFSH, FEED( 12),
     1               FGIM, FLOSS(20), TAREA(12), TLSUP(20), TOUTRL ,
     2               TTAREA
      COMMON /CPDEMC/ APCN, DELP, IGRAIN(5), IMEAT(5),
     2                MM(20), PILOSS(20), PLOSTD, PSUBCQ(5,5,2), P CMIN,
```

```
                       PSUBCM(5,5,2),  PSUBTG(5,2),  PSUBTM(5,2),PFU( 2,20),
     3                  PCONVG(5),  PWHTIC,  YDEL,  CALPU(20),  PROTPU( 20),
     4                 W(20)
     5
      REAL  MM
      COMMON /ICDEMC/  ELISAV(20),  ELISVR(20),  EPAVGO(20),
     1                 STKO(20),  VCPU(10,20),  VCSTK(10,20),  VEFRMY( 10),
     2                 VFPIX(10),  VPSUP(10,20),  VPWI(10,20),
     3                 VPWX(10,20),  FNR(20),  VPICF(10),  VEINCM(10),
     4                 VPCUFD(10)
      COMMON /PPDEMC/  ARG(5),  KFT(20),  KNOR,  KPAVG(20),
     1                PCCONT(20),  PCPMAX(20),  PCPMIN(20),  PCRT(20),
     2                PCRT1,  PCUT1,  PWPMAX(20),  PWPMIN(20),  STKRT(2 0),
     3                TARIFF(20),  VCPUT(5,20),  VPCSUP(5,20),
     4                VPWLDI(5,20),  VPWLDX(5,20)
      COMMON /DEMYC/  CON(1,20),  DSAVD(1)
      COMMON /DEMSVC/  ALPHA,  CPU(20),  CPUO(20),  DEMC(20),
     1                 DEMT(1,20),  EPAVG(1,20),  INCOM,  PCCON(1,20),
     2                 PCCONU(20),  Q(20),  RFRMY(1),  S1,  STK(20),
     3                 EINCOM,  TDEM(20),  TDEML(20),  WWP(20)
      REAL  INCOM
      COMMON /VDEMC/  CHGSTK(20),  DSTK(20),  DVALDF,  EFRMY,
     1                ELASP(20,20),  ELASPR(20,20),
     1                ELASI(20),  ELASIR(20),  FRMY(1),  INAPA(20),
     1                FGIVAL,  ITER,  PCCONF(20),  TARREV(20),
     2                PRFEXP(20),  PRFIMP(20),  PWLDD(20),  PWLDEX(20) ,
     3                PWLDIM(20),  RDEM(20),  TRDPRF,  TVALDF,  VALDEF( 20)
C
      DIMENSION EPAVGL(20)
C
      DO 40 JC=1,NCOMAG
      RDEM(JC) = 0.
   40 CONTINUE
      IF(ITIME .GT. 0) DCPUNF = (C[illegible] CPU(20)) / CPU(20)
C
      IF(ITIME .EQ. 0) EFRMY = TABEL([illegible]RMY,  KTABI,  TTABI,  1.0,  TIMEI)
      IF(ITIME .NE. 0) EFRMY = APCF * [illegible] * POPL(1) / TRPOP
C
      DO 100 IR=1,NREGN
      IF(ITIME .EQ. 0) GO TO 82
C
C              RURAL DEMAND COMPONENT
C              TIME-VARYING PER CAPITA CONSUMPTION FOR
C              RURAL SECTOR (COMPUTED YEARLY)
C
      IF(KRAP .EQ. 0) FRMY(IR) = FPCI
      IF(KRAP .NE. 0) FRMY(IR) = FPCI * TRPOP * PRFRMY(IR) / POPR( IR)
      IF(ITIME .EQ. 1) RFRMY(IR) = FRMY(IR) / 1.05
      FRMYA = FRMY(IR) * TABEL(VFPIX,  KTABI,  TTABI,  1.0,  T-DT)
      RFRMYL = RFRMY(IR)
      RFRMY(IR) = RFRMY(IR) + (DT/YDEL) * (FRMYA - RFRMY(IR))
      DO 75 J=1,NCOMAG
      PAVGA = PAVG(IR,J) * TABEL(VFPIX,  KTABI,  TTABI,  1.0,  T-DT)
      EPAVGL(J) = EPAVG(IR,J)
      EPAVG(IR,J) = EPAVG(IR,J) + (DT / DELP) *
     1             (PAVGA - EPAVG(IR,J))
   75 CONTINUE
      DO 78 J=1,5
      JG = IGRAIN(J)
      JM = IMEAT(J)
      DO 77 I=1,5
      IF(I .EQ. J)GO TO 77
      IG = IGRAIN(I)
      IM = IMEAT(I)
      ELASPR(IG,JG) = -PSUBTG(J,1) * PSUBCG(I,J,1) * ELASPR(JG,JG)
     1               * PCCON(IR,JG) / PCCON(IR,IG)
     2               * (PCONVG(J) / PCONVG(I))
      ELASPR(IM,JM) = -PSUBTM(J,1) * PSUBCM(I,J,1) * ELASPR(JM,JM)
     1               * PCCON(IR,JM) / PCCON(IR,IM)
   77 CONTINUE
   78 CONTINUE
C
      RATIOI = (RFRMY(IR) - RFRMYL) / RFRMYL  -  DCPUNF
      DO 79 J=1,NCOMAG
   79 ELASIR(J) = DEMT(IR,J) * (PCRT(J) - PCCON(IR,J))
      IF(YEAR .LT. 1980.) GO TO 82
      PCT1 = AMAX1(PCRT1,PCRT(1)-(YEAR-1980.)*(PCRT(1)-PCRT1)/20.)
      ELASIR(1) = DEMT(IR,1) * (PCT1 - PCCON(IR,1))
   82 CONTINUE
      DO 85 I=1,NCOMAG
      IF(ITIME .EQ. 0) GO TO 84
      X = 0.
      DO 80 J=1,NCOMAG
      X = X + ELASPR(I,J) * (EPAVG(IR,J) - EPAVGL(J)) / EPAVGL(J)
     1      - ELASPR(I,J) * DCPUNF
   80 CONTINUE
      PCCON(IR,I) = PCCON(IR,I) * (1. + X + ELASIR(I) * RATIOI)
   84 CONTINUE
      CON(IR,I) = PCCON(IR,I) * POPR(IR)
      RDEM(I) = RDEM(I) + CON(IR,I)
   85 CONTINUE
  100 CONTINUE
```

```
      DO 120 I=1,NCOMAG
      PCCONF(I) = RDEM(I) / TRPOP
  120 CONTINUE
C
      RETURN
      END



      SUBROUTINE TRADE
C
C STOCK CHANGES, WORLD PRICES, AND AGRICULTURAL TRADE
C
C DECLARATIONS
C
      COMMON /CONTRC/ ANIMAL, BASEYR, DT, FARM, ITIME, KTRACK, KTA BI,
     1                IRUN, ITTY, ICHDAT, ISENPR, IPLT, ITABPR,
     1                LVSTPR, MODE, MODNAM(9),
     1                KPOP, KCHG, KRAP, KDEM, KNEC, LINK, JPER,
     1                LUIN1, LUIN2, LUOUT, LUPLOT(3), LURLP1,
     2                LUPGEN, LUPOP,
     2                KPPRT(15), KCPRT(15), KFPRT(15), KAPRT(15),
     2                KDPRT(15), KYPRT(15), KNPRT(15),
     3                NCC, NCOM, NCOMAG, NCROP, NLVST, NNC, NREGN,
     4                NRUN, NS, NAME(20), PLANT,
     5                T, TIMEI, TIMEF, TTABI, YEAR,
                      TOTSOR, TOTPOP, URBAN
      INTEGER ANIMAL, FARM, PLANT, URBAN, TOTPOP, TOTSOR
      EQUIVALENCE (LUPLT1,LUPLOT(1)),(LUPLT2,LUPLOT(2)),
     1            (LUPLT3,LUPLOT(3))
      COMMON /DKXDC/ ARGPP(5),KPP,VDFC(10,20),VDFCT(5,20),VPRICE(1 0,20),
     1               VPP(5,20),APCFD
      COMMON /DEMKC/ APCF, DEFCIT(20), PAVG(1,20), PFDIMX, TPAVG(2 0)
      COMMON /DEMNC/ CPFI, FCONP, FCONU, TTAREV, TEXNFC, FEXNFC,
     1               UCONP, UCONU, VALDF1, VALDF2, WPFI
      COMMON /NECDYC/ AGINTX, CPUNF, FPCI, PRFRMY(1), UPCI, WOND,
     1                TAGDIP, TPTAX, TYG, TYNAG
      COMMON /POPKC/ CALRPC(2), POP(2), POPL(2), POPR(1), PRORPC(2 ),
     1               TPOP, TRPOP, TPOPPC(2)
      COMMON /PRDDC/ AYLD(12), EFFPLV(7), EFFYLD(12), FDFSH, FEED( 12),
     1               FGIM, FLOSS(20), TAREA(12), TDSUP(20), TOUTRL ,
     2               TTAREA
      COMMON /CPDEMC/ APCN, DELP, IGRAIN(5), IMEAT(5),
     2               MM(20), PILOSS(20), PLOSTD, PSUBCG(5,5,2), P CMIN,
     3               PSUBCM(5,5,2), PSUBTG(5,2), PSUBTM(5,2),PFU( 2,20),
     4                PCONVG(5), PWHTIC, YDEL, CALPU(20), PROTPU( 20),
     5               W(20)
      REAL MM
      COMMON /ICDEMC/ ELISAV(20), ELISVR(20), EPAVGO(20),
     1               STKO(20), VCPU(10,20), VCSTK(10,20), VEFRMY( 10),
     2               VFPIX(10), VPSUP(10,20), VPWI(10,20),
     3               VPWX(10,20), FNR(20), VPICF(10), VEINCM(10),
     4               VPCUFD(10)
      COMMON /PPDEMC/ ARG(5), KFT(20), KNOR, KPAVG(20),
     1               PCCONT(20), PCPMAX(20), PCPMIN(20), PCRT(20),
     2               PCRT1, PCUT1, PWPMAX(20), PWPMIN(20), STKRT(2 0),
     3               TARIFF(20), VCPUT(5,20), VPCSUP(5,20),
     4               VPWLDI(5,20), VPWLDX(5,20)
      COMMON /DEMSVC/ ALPHA, CPU(20), CPUO(20), DEMC(20),
     1               DEMT(1,20), EPAVG(1,20), INCOM, PCCON(1,20),
     2               PCCONU(20), Q(20), RFRMY(1), S1, STK(20),
     3               EINCOM, TDEM(20), TDEML(20), WWP(20)
      REAL INCOM
      COMMON /VDEMC/ CHGSTK(20), DSTK(20), DVALDF, EFRMY,
     1               ELASP(20,20), ELASPR(20,20),
     1               ELASI(20), ELASIR(20), FRMY(1), INAPA(20),
     1               FGIVAL, ITER, PCCONF(20), TARREV(20),
     2               PRFEXP(20), PRFIMP(20), PWLDD(20), PWLDEX(20) ,
     3               PWLDIM(20), RDEM(20), TRDPRF, TVALDF, VALDEF( 20)
C
      ENTRY TRADEI
C
C COMPUTE STOCK CHANGES
      DO 30 J=1,NCOMAG
      IF(ITIME .EQ. 0) STK(J) = STKO(J)
      IF(TIMEI.LT.TTABI+.0001 .OR. ITIME.NE.0) GO TO 15
C        INITIALIZE STOCK LEVES TO TIMEI IF TIMEI .GT. TTABI
      TX = TTABI
    5 CONTINUE
      CHGSTK(J) = TABEL(VCSTK(1,J), KTABI, TTABI, 1.0, TX)
      TX = TX + DT
      STK(J) = STK(J) + DT * CHGSTK(J)
      IF(TX .LT. TIMEI-.0001) GO TO 5
   15 CONTINUE
      IF(ITIME .EQ. 0) GO TO 25
      STK(J) = STK(J) + DT * CHGSTK(J)
      IF(ITIME .EQ. 1) TDEML(J) = TDEM(J)
   20 TDEML(J) = TDEML(J) + (DT/3.)*(TDEM(J)-TDEML(J))
```

```
      DSTK(J) = STKRT(J)*TDEML(J)/12
      CHGSTK(J) = (DSTK(J)-STK(J))/2.
   25 CONTINUE
      IF(T .LE. BASEYR+.9999)
     1    CHGSTK(J) = TABEL(VCSTK(1,J), KTABI, TTABI, 1.0,T)
      IF(KFAP .EQ. 0) GO TO 30
      IF(J .EQ. 3) TDSUP(3) = TDSUP(3) * (1.-PWHTIC)
      IF(J .EQ. 9) TDSUP(9) = TDSUP(9) * (1.-PLOSTD)
      IF(J .EQ 12) TDSUP(12)= TDSUP(12)* TABEL(VPICF,KTABI,TTABI,1 .0,T)
      TDSUP(J) = TDSUP(J) - CHGSTK(J) - RDEM(J)
   30 CONTINUE
C
      IF(KFAP .EQ. 0) GO TO 46
      DO 40 J=1,NCOM
      IF(T .GT. BASEYR+.9999) GO TO 35
      PWLDIM(J) = TABEL(VPWI(1,J), KTABI, TTABI, 1.0, T) * WOND
      PWLDD(J)  = TABEL(VPWX(1,J), KTABI, TTABI, 1.0, T)
      GO TO 38
   35 CONTINUE
      PWLDIM(J) = TABUL(VPWLDI(1,J),KNOR,ARG,YEAR) * WOND
      PWLDD(J) = TABUL(VPWLDX(1,J),KNOR,ARG,YEAR)
   38 CONTINUE
      PWLDEX(J) = PWLDD(J) * WOND
 40   CONTINUE
      IF(T .LE. BASEYR+.9999) GO TO 45
      DO 45 J=1,NCOMAG
      IF(KFT(J) .GT. 3) GO TO 45
      IF(KFT(J) .GE. 2)DEFCIT(J) = TABUL(VDFCT(1,J),KNOR,ARG,YEAR)
      IF(KFT(J) .EQ. 1)DEFCIT(J) = TPOP *
     1     TABUL(VPCSUP(1,J),KNOR ARG,YEAR) - TDSUP(J)
   45 CONTINUE
   46 CONTINUE
      RETURN
C
      ENTRY TRADEC
C
C                BALANCE OF PAYMENTS CALCULATIONS
C
      TVALDF = 0.0
      VALDF1 = 0.0
      VALDF2 = 0.0
      TRDPRF=0.0
      TTAREV = 0
C
C        OFFSET FEED GRAIN IMPORTS WITH SURPLUS BARLEY AND POTATO ES
      IF(KFAP.LE.0 .OR. ITIME.EQ.0) FGIM = Q(4) - TDSUP(4)
      IF(T .LE. BASEYR+.9999) GO TO 60
      DEFCIT(2) = AMAX1(0., TDSUP(2)-Q(2))
      DEFCIT(8) = AMAX1(0., TDSUP(8)-Q(8))
      X = FGIM
      IF(DEFCIT(2) .LE. 0.) GO TO 50
      FGIM = AMAX1(0., FGIM-DEFCIT(2))
      DEFCIT(2) = DEFCIT(2) - (X-FGIM)
      CHGSTK(2) = CHGSTK(2) + DEFCIT(2)
      TDSUP(2) = Q(2)
      X = FGIM
   50 CONTINUE
      IF(DEFCIT(8) .LE. 0.) GO TO 60
      FGIM = AMAX1(0., FGIM-DEFCIT(8))
      DEFCIT(8) = DEFCIT(8) - (X-FGIM)
      TDSUP(8) = Q(8) + DEFCIT(8)
   60 CONTINUE
      TDSUP(4) = Q(4) - FGIM
      FGIM = FGIM / (1.-PILOSS(4))
      FGIVAL = FGIM * PWLDIM(4)
C
      DO 110 J= 1,NCOMAG
      DEFCIT(J) = Q(J) - TDSUP(J)
      IF(J .EQ. 9) DEFCIT(9) = DEFCIT(9) / (1.-PLOSTD)
      IF (DEFCIT(J)) 90,90,80
   80 CONTINUE
      IF(YEAR.LE.BASEYR+.9999 .OR. J.GT.2) GO TO 85
      X = AMAX1(0.,CHGSTK(J)) + STK(J)/4.
      CHGSTK(J) = CHGSTK(J) - AMIN1(DEFCIT(J),X)
      DEFCIT(J) = AMAX1(0., DEFCIT(J)-X)
   85 CONTINUE
      DEFCIT(J) = DEFCIT(J) / (1.-PILOSS(J))
      IF(J .EQ. 3) DEFCIT(3) = DEFCIT(3)*(1.-PILOSS(3))
     1     /(1.-PILOSS(3)-PWHTIC)
      VALDEF(J) = DEFCIT(J)*PWLDIM(J)/(1.+TARIFF(J))
C              THESE COMMODITIES ARE IMPORTED
      TARREV(J) = VALDEF(J) * TARIFF(J)
      PRFIMP(J) = DEFCIT(J) * (CPU(J)-PWLDIM(J))
      PRFEXP(J) = 0.0
      VALDF2 = VALDF2 + VALDEF(J)
      GO TO 100
C              THESE COMMODITIES ARE EXPORTED
   90 CONTINUE
      IF(YEAR.LE.BASEYR+.9999 .OR. J.NE.1) GO TO 95
      CHGSTK(1) = CHGSTK(1) - DEFCIT(1)
      DEFCIT(1) = 0.
   95 CONTINUE
```

```
      VALDEF(J) = DEFCIT(J) * PWLDEX(J)
      TARREV(J) = 0.0
      PRFIMP(J) = 0.0
      PRFEXP(J) = DEFCIT(J) * (PWLDEX(J)-CPU(J))
      PRFEXP(J) = -PRFEXP(J)
      VALDF1 = VALDF1-VALDEF(J)
  100 TVALDF = TVALDF + VALDEF(J)
      TRDPRF=TRDPRF+PRFIMP(J)+PRFEXP(J)
      TTAREV = TTAREV + TARREV(J)
  110 CONTINUE
C
      TVALDF = -TVALDF
      DVALDF = TVALDF/WOND
C
      RETURN
      END
```

Executive Routines

```
      PROGRAM KASM3(INPUT=65,TAPE5=INPUT,OUTPUT=257,TAPE6=OUTPUT,
     1     DEBUG=OUTPUT,TAPE10=65,TAPE11,TAPE21=512,TAPE26,TAPE28,
     2     TAPE27=512,TAPE8=65,TAPE2=65,TAPE3=65)
C
C
C                            KASM 3  --  MARCH 1977
C
C     KASM3 IS A SYSTEM OF MODELS FOR AGRICULTURAL SECTOR DECISION
C     ANALYSIS IN KOREA. IT CONSISTS OF FIVE MODELS, EACH REPRESENTING
C     A COMPONENT OF AGRICULTURAL SECTOR ANALYSIS. EACH MODEL MAY BE
C     EXECUTED ALONE OR IN COMBINATION WITH ONE OR MORE OTHER MODELS
C     FOR PARTIAL ANALYSIS OR, WHEN ALL MODELS ARE RUN TOGETHER, FOR
C     COMPREHENSIVE SECTOR ANALYSIS. THE FIVE MODELS ARE --
C
C        1. POPULATION AND MIGRATION                    (POPMIG)
C        2. TECHNOLOGY CHANGE                           (CHANGE)
C        3. RESOURCE ALLOCATION AND PRODUCTION          (RAP)
C        4. DEMAND - PRICE - TRADE                      (DEMAND)
C        5. NATIONAL ECONOMY                            (NECON)
C
C     IN ADDITION, KASM3 HAS A FARM INCOME ACCOUNTING COMPONENT (FRMAC).
C     SWITCHES KPOP, KCHG, KRAP, KDEM, AND KNEC ARE SET BY THE USER
C     TO DESIGNATE THE MODE OF OPERATION OF EACH MODEL --
C
C        K... .EQ. 0   -- THE MODEL IS TO BE IGNORED, AND
C                      OUTPUTS OF THE MODEL WHICH ARE INPUTS TO OTHER
C                      MODELS ARE TO BE PROJECTED EXOGENOUSLY
C        K... .GT. 0   -- THE MODEL IS TO BE EXECUTED WITH ONE OR MORE
C                      OTHER MODELS, OR INDEPENDENTLY
C        KRAP .LT. 0   -- RESOURCE ALLOCATION IS TO BE PROJECTED
C                      EXOGENOUSLY, BUT THE PRODUCTION ACCOUNTING
C                      PART OF RAP IS TO BE EXECUTED
C
C     MOST OF KASM3 IS PROGRAMMED TO BE RUN IN A REGIONAL MODE, I. E.
C     WHERE KOREA IS DIVIDED INTO ONE OR MORE REGIONS. NOT ALL OUTPUT
C     ROUTINES ARE SO GENERALIZED, HOWEVER. ALSO, REGIONALIZATION WOULD
C     REQUIRE REGIONAL DATA INPUTS AND DIMENSION CHANGES.
C
C DECLARATIONS
C
      COMMON /CONTRC/ ANIMAL, BASEYR, DT, FARM, ITIME, KTRACK, KTABL,
     1                IRUN, ITTY, ICHDAT, ISENPR, IPLT, ITABPR,
     1                LVSTPR, MODE, MODNAM(9),
     1                KPOP, KCHG, KRAP, KDEM, KNEC, LINK, JPER,
     2                LUIN1, LUIN2, LUOUT, LUPLOT(3), LURLP1,
     2                LUPGEN, LUPOP,
     2                KPPRT(15), KCPRT(15), KFPRT(15), KAPRT(15),
     2                KDPRT(15), KYPRT(15), KNPRT(15),
     3                NCC, NCOM, NCOMAG, NCROP, NLVST, NNC, NREGN,
     4                NRUN, NS, NAME(20), PLANT,
     5                T, TIMEI, TIMEF, TTABI, YEAR,
     5                TOTSOR, TOTPOP, URBAN
      INTEGER ANIMAL, FARM, PLANT, URBAN, TOTPOP, TOTSOR
      EQUIVALENCE (LUPLT1,LUPLOT(1)),(LUPLT2,LUPLOT(2)),
     1            (LUPLT3,LUPLOT(3))
      COMMON /POPKC/ CALRPC(2), POP(2), POPL(2), POPR(1), PRORPC(2 ),
     1               TPOP, TRPOP, TPOPPC(2)
      COMMON /PLOTK/ NVARP, NVARC, NVARF, NVARA, NVARD, NVARY, NVARN,
     1        NVARK, NVAR, KMSW(7)
      COMMON /DCHKXC/ JRELKX(100), OVALKX(100), CVALKX(100), ISFLKX
```

```
      DIMENSION VECTOR(1)
      EQUIVALENCE (VECTOR(1),CALRPC(1))
C
      CALL SECOND(T1)
C
C JOB INITIALIZATION (JI)
C
C     BLOCKDATA CONTRD
      CALL CONTRJ
C
      ITIME = -2
      LINK = 0
      CALL SECOND(T3)
      IF(KPOP .GT. 0) CALL POPSTR
      CALL SECOND(T2)
      TIMEP = T2 - T3
      IF(KRAP .GT. 0) CALL FRAJI
      CALL SECOND(T3)
      TIMER = T3 - T2
      IF(KNEC .GT. 0) CALL NECJI
      CALL SECOND(T2)
      TIMEN = T2 - T3
      IF(IPLT.NE.0 .OR. LVSTPR.NE.0) CALL PLTPKG
      CALL SECOND(T3)
      TIMPL = T3 - T2
C
      TIMEC = 0.
      TIMEA = 0.
      TIMED = 0.
      TIMEY = 0.
C
C RUN LOOP
C
      DO 600 IRUN=1,NRUN
C
C RUN INITIALIZATION (RI)
C
      YEAR = TIMEI
      ITIME = -1
      LINK = 1
      IF(ICHDAT .NE. 0)CALL CHDATA(ANIMAL,JRELKX,OVALKX,CVALKX,ISF LKX,1)
      LINK = 2
      CALL SECOND(T3)
      IF(KPOP .GT. 0) CALL POPSET
      CALL SECOND(T2)
      TIMEP = TIMEP + T2 - T3
      IF(KCHG .GT. 0) CALL CHGI
      CALL SECOND(T3)
      TIMEC = TIMEC + T3 - T2
      IF(KRAP .GT. 0) CALL FRARI
      CALL SECOND(T2)
      TIMER = TIMER + T2 - T3
      IF(KRAP .NE. 0) CALL PRDAC
      CALL SECOND(T3)
      TIMEA = TIMEA + T3 - T2
      IF(KDEM .GT. 0) CALL DEMI
      CALL SECOND(T2)
      TIMED = TIMED + T2 - T3
      IF(KRAP .NE. 0) CALL FRMAC
      CALL SECOND(T3)
      TIMEY = TIMEY + T3 - T2
      IF(KNEC .GT. 0) CALL NECRI
      CALL SECOND(T2)
      TIMEN = TIMEN + T2 - T3
      IF(IPLT.NE.0.OR. LVSTPR.NE.0) CALL POSIT(LUPLOT(2),1,5)
C
      NTIME = 1.0001 + (TIMEF - TIMEI) / DT
      T = TIMEI - DT
C
C TIME LOOP
C
      DO 500 ITER=1,NTIME
      ITIME = ITER - 1
      T = T + DT
      YEAR = T
C
      CALL CONTRT
C
      IF(IPLT.NE.0 .OR. LVSTPR.NE.0) WRITE(LUPLT1) YEAR
C
C NATIONAL ECONOMY INPUTS TO AGRICULTURE
C
      IF(KNEC .GT. 0) GO TO 10
C
C     KNEC .EQ. 0 --NATIONAL ECONOMY EXOGENOUS
C
      IF(KCHG.GT.0 .OR. KRAP.NE.0) CALL NECCRX
      IF(KDEM.GT.0 .OR. KRAP.NE.0) CALL NECDYX
      GO TO 20
C
C     KNEC .GT. 0 -- NATIONAL ECONOMY ENDOGENOUS
C
```

```
C
   10 CONTINUE
      CALL SECOND(T3)
      LINK = 0
      CALL NECON
      CALL SECOND(T2)
      TIMEN = TIMEN + T2 - T3
C
C     POPULATION AND MIGRATION MODEL
C
   20 CONTINUE
      IF(KPOP .GT. 0) GO TO 30
C
C     KPOP .EQ. 0 --POPULATION EXOGENOUS
C
      IF(KRAP.NE.0 .OR. KDEM.GT.0 .OR. KNEC.GT.0) CALL POPKX
      IF(KRAP .GT. 0) CALL POPRX
      GO TO 50
C
C     KPOP .GT. 0 --POPULATION ENDOGENOUS
C
   30 CONTINUE
      CALL SECOND(T3)
      CALL POPMIG
      CALL SECOND(T2)
      TIMEP = TIMEP + T2 - T3
C
C     TECHNOLOGY CHANGE MODEL
C
   50 CONTINUE
      IF(KCHG.GT.0 .AND. ITIME.GT.0) GO TO 80
C
C     KCHG .EQ. 0 --TECHNOLOGY CHANGE EXOGENOUS
C
      IF(KRAP .NE. 0) CALL CHGRX
      IF(KRAP .GT. 0) CALL CHGFX
      IF(ITIME.EQ.0 .AND. (IPLT.NE.0 .OR. LVSTPR.NE.0) .AND. KCHG. GT.0)
     1   GO TO 80
      GO TO 100
C
C     KCHG .GT. 0 --TECHNOLOGY CHANGE ENDOGENOUS
C
   80 CONTINUE
      CALL SECOND(T3)
      CALL CHANGE
      CALL SECOND(T2)
      TIMEC = TIMEC + T2 - T3
C
C     FARM RESOURCE ALLOCATION AND PRODUCTION MODEL
C
  100 CONTINUE
      IF(KRAP.GT.0 .AND. ITIME.GT.0) GO TO 120
C
C     KRAP .LE. 0 --FARM RESOURCE ALLOCATION AND PRODUCTION EXOGE NOUS
C
      CALL FRACAX
      IF(KRAP .EQ. 0) GO TO 150
      CALL FRAAYX
      IF(ITIME.EQ.0 .AND. (IPLT.NE.0 .OR. LVSTPR.NE.0) .AND. KRAP. GT.0)
     1   GO TO 120
      GO TO 140
C
C     KRAP .GT. 0 --FARM RESOURCE ALLOCATION AND PRODUCTION ENDOG ENOUS
C
  120 CONTINUE
      CALL SECOND(T3)
      CALL FRESAL(NONOPT)
      CALL SECOND(T2)
      TIMER = TIMER + T2 - T3
C
C     KRAP .NE. 0 --PRODUCTION ACCOUNTING
C
  140 CONTINUE
      IF(ITIME .EQ. 0) CALL DEMKX
      CALL SECOND(T3)
      CALL PRDAC
      CALL SECOND(T2)
      TIMEA = TIMEA + T2 - T3
C
C     DEMAND - PRICE - TRADE MODEL
C
  150 CONTINUE
      IF(KDEM .GT. 0) GO TO 180
C
C     KDEM .EQ. 0 --PRODUCER PRICES EXOGENOUS
C
      IF(KCHG.GT.0 .OR. KRAP.NE.0) CALL DEMKX
      GO TO 200
C
C     KDEM .GT. 0 --DEMAND - PRICE - TRADE ENDOGENOUS
C
```

```
  180 CONTINUE
      CALL SECOND(T3)
      CALL DEMAND
      CALL SECOND(T2)
      TIMED = TIMED + T2 - T3
C
C     FARM INCOME ACCOUNTING
C
  200 CONTINUE
      IF(KRAP .EQ. 0) GO TO 250
      CALL SECOND(T3)
      CALL FRMAC
      CALL SECOND(T2)
      TIMEY = TIMEY + T2 - T3
C
C     NATIONAL ECONOMY MODEL
C
  250 CONTINUE
      IF(KNEC .EQ. 0) GO TO 300
C
C
C     KNEC .GT. 0  --NATIONAL ECONOMY ENDOGENOUS
C
      CALL SECOND(T3)
      LINK = 1
      CALL NECON
      CALL SECOND(T2)
      TIMEN = TIMEN + T2 - T3
C
C     RECORD SYSTEM PLOT VALUES AND MERGE PLOT RECORDS
C
  300 CONTINUE
      IF(IPLT.EQ.0 .AND. LVSTPR.EQ.0) GO TO 350
      WRITE(LUPLT1) (VECTOR(I),I=1,NVARK)
      CALL SECOND(T3)
      LINK = 0
      CALL PLTPKG
      CALL SECOND(T2)
      TIMPL = TIMPL + T2 - T3
C
C     END RUN IF LP SOLUTION NON-OPTIMAL
C
  350 CONTINUE
      IF(KRAP.GT.0 .AND. ITIME.GT.0 .AND. NONOPT.NE.0) GO TO 510
C
  500 CONTINUE
C
C     END TIME LOOP
C
      GO TO 520
  510 CONTINUE
      WRITE(LUOUT,515) IRUN, T
  515 FORMAT(/35H *** NONOPTIMAL LP SOLUTION -- RUN ,I2,11HTERMINA TED ,
     1       15HFOLLOWING TIME ,F5.0,4H *** )
      ITIME = ITIME - 1
  520 CONTINUE
C
C     CALL PLOT PACKAGE FOR PLOTTING AND/OR LIVESTOCK TIME SERIES TA BLES
C
      IF(IPLT.EQ.0 .AND. LVSTPR.EQ.0) GO TO 600
      CALL SECOND(T3)
      LINK = 1
      CALL PLTPKG
      CALL SECOND(T2)
      TIMPL = TIMPL + T2 - T3
C
  600 CONTINUE
C
C     END RUN LOOP
C
      CALL SECOND(T2)
      TIMEK = T2 - T1
      NTIME1 = NTIME - 1
      PRINT 700, TIMEK, TIMPL, TIMEP, TIMEC, TIMER, TIMEA, TIMED,
     1           TIMEY, TIMEN, NRUN, NTIME, NTIME1
  700 FORMAT(1H1,T10,27HEXECUTION TIMES IN CP SECS /
     1           T20,8HKASM     ,F8.2/
     2           T20,8HPLOT     ,F8.2/
     3           T20,8HPOPMIG   ,F8.2/
     4           T20,8HCHANGE   ,F8.2/
     5           T20,8HFRESAL   ,F8.2/
     6           T20,8HPRDAC    ,F8.2/
     7           T20,8HDEMAND   ,F8.2/
     8           T20,8HFRMAC    ,F8.2/
     9           T20,8HNECON    ,F8.2/
     *           T10,4HFOR ,I2,9H RUNS OF ,I3,22H TIME CYCLES PER  RUN. /
     *           T10,1H(,I3,36H TIME CYCLES FOR CHANGE AND FRESAL. ) )
C
C
      STOP
      END
```

```
      SUBROUTINE CHDATA(VECTOR, JREL, OVAL, CVAL, ISFL, KMOD)
C
C MAKES USER-SPECIFIED DATA CHANGES FOR SENSITIVITY AND POLICY RUN S
C
C DECLARATIONS
C
      COMMON /CONTRC/ ANIMAL, BASEYR, DT, FARM, ITIME, KTRACK, KTA BI,
     1                IRUN, ITTY, ICHDAT, ISENPR, IPLT, ITABPR,
     1                LVSTPR, MODE, MODNAM(9),
     1                KPOP, KCHG, KRAP, KDEM, KNEC, LINK, JPER,
     1                LUIN1, LUIN2, LUOUT, LUPLOT(3), LURLP1,
     2                LUPGEN, LUPOP,
     2                KPPRT(15), KCPRT(15), KFPRT(15), KAPRT(15),
     2                KDPRT(15), KYPRT(15), KNPRT(15),
     3                NCC, NCOM, NCOMAG, NCROP, NLVST, NNC, NREGN,
     3                NRUN, NS, NAME(20), PLANT,
     4                T, TIMEI, TIMEF, TTABI, YEAR,
     5                TOTSOR, TOTPOP, URBAN
      INTEGER ANIMAL, FARM, PLANT, URBAN, TOTPOP, TOTSOR
      EQUIVALENCE (LUPLT1,LUPLOT(1)),(LUPLT2,LUPLOT(2)),
     1            (LUPLT3,LUPLOT(3))
C
      DIMENSION VECTOR(1), JREL(1), OVAL(1), CVAL(1)
      DIMENSION IDESC(3), TTL(8)
C
C EXECUTION
C
C
C    REST TO OLD VALUE AT START OF RUN
      IF(LINK .EQ. 3) GO TO 300
      IF(IRUN.EQ.1 .OR. ISFL.EQ.0) GO TO 60
      DO 50 I=1,ISFL
      VECTOR(JREL(I)) = OVAL(I)
   50 CONTINUE
   60 CONTINUE
C
      IF(LINK .EQ. 2) GO TO 200
C
C TABLE HEADING
      IF(MODE .GT. 1) PRINT 105
  105 FORMAT(/38H ENTER DATA CHANGE TITLE IN FORMAT 8A9 / 2H *)
      READ(LUIN1,110) TTL
  110 FORMAT(8A9)
      WRITE(LUOUT,120) TTL, IRUN
  120 FORMAT(1H1,T10,8A9 //T20,31HINTER-RUN DATA CHANGES FOR RUN , I2/
     1       T5,11HDESCRIPTION,T43,9HOLD VALUE,6X,9HNEW VALUE )
C
C    MAKE CHANGES AS SPECIFIED ON INPUT AT START OF RUN
  200 CONTINUE
      WRITE(LUOUT,210) MODNAM(KMOD)
  210 FORMAT(//T5,A6,6H MODEL)
      ISFL = 0
  220 ISFL = ISFL + 1
      IF(MODE .GT. 1) PRINT 225, MODNAM(KMOD)
      READ(LUIN1,230) JREL(ISFL), CVAL(ISFL), IDESC
  225 FORMAT(/24H ENTER DATA CHANGES FOR ,A6/ 2H *)
  230 FORMAT(I5,E15.5,3A10)
      IF(JREL(ISFL) .EQ. 0) GO TO 260
      OVAL(ISFL) = VECTOR(JREL(ISFL))
      VECTOR(JREL(ISFL)) = CVAL(ISFL)
      WRITE(LUOUT,250) IDESC, OVAL(ISFL), CVAL(ISFL)
  250 FORMAT(T5,3A10,2X 2G15.4)
      GO TO 220
  260 ISFL = ISFL - 1
      GO TO 400
C
C SET TO NEW VALUE EACH TIME PERIOD
  300 CONTINUE
      IF(ISFL .EQ. 0) GO TO 400
      DO 350 I=1,ISFL
      VECTOR(JREL(I)) = CVAL(I)
  350 CONTINUE
C
C
  400 CONTINUE
      RETURN
      END


      SUBROUTINE CONTR
C
C SPECIFICATION OF VALUES FOR JOB, RUN, TIME AND PRINT/PLOT CONTRO L
C PARAMETERS
C
C DECLARATIONS
C
      COMMON /CONTRC/ ANIMAL, BASEYR, DT, FARM, ITIME, KTRACK, KTA BI,
```

```
                       IRUN, ITTY, ICHDAT, ISENPR, IPLT, ITABPR,
     1                 LVSTPR, MODE, MODNAM(9),
     1                 KPOP, KCHG, KRAP, KDEM, KNEC, LINK, JPER,
     1                 LUIN1, LUIN2, LUOUT, LUPLOT(3), LURLP1,
     1                 LUPGEN, LUPOP,
     2                 KPPRT(15), KCPRT(15), KFPRT(15), KAPRT(15),
     2                 KDPRT(15), KYPRT(15), KNPRT(15),
     2                 NCC, NCOM, NCOMAG, NCROP, NLVST, NNC, NREGN,
     2                 NRUN, NS, NAME(20), PLANT,
     3                 T, TIMEI, TIMEF, TTABI, YEAR,
     4                 TOTSOR, TUTPOP, URBAN
     5
      INTEGER ANIMAL, FARM, PLANT, URBAN, TOTPOP, TOTSOR
      EQUIVALENCE (LUPLT1,LUPLOT(1)),(LUPLT2,LUPLOT(2)),
     1            (LUPLT3,LUPLOT(3))
      COMMON /LPCNTR/ NFIX(30),TOL(5),ERR(4),IOUT(4),TOBJ,TIMLP
      EQUIVALENCE
     1            (NFIX( 1),ML )
     2           ,(NFIX( 2),NL )
     3           ,(NFIX( 3),M)
     4           ,(NFIX( 4),N)
     5           ,(NFIX( 5),INBASE)
     5           ,(NFIX(15),MLE)
     6           ,(NFIX(16),NCOST)
     8           ,(NFIX(18),HDRTXT)
     4           ,(NFIX(24),MSUB)
     5           ,(NFIX(25),NSUB)
     6           ,(NFIX(26),NRGRLP)
     7           ,(NFIX(27),MADD)
     8           ,(NFIX(28),NADD)
C
C
      DIMENSION IPYR(5), IPPRT(15), ICPRT(15), IFPRT(15), IAPRT(15 ),
     1          IDPRT(15), IYPRT(15), INPRT(15)
      DIMENSION ICMMT(8), MLAB(3)
      DATA MLAB /5HBATCH,5HTTY-W,5HTTY-N/
C
C     JOB CONTROL ENTRY
C
      ENTRY CONTRJ
C
      CALL DATE(IDAT)
      CALL TIME(ITIM)
C
      READ(LUIN1,90) MODE
      ITTY = 0
      IF(MODE .GT. 2) ITTY = 1
C
      IF(MODE .GT. 1) PRINT 200
      READ(LUIN1,100) NRUN, KTRACK, ICHDAT, ISENPR, IPLT, ITABPR
     1               ,LVSTPR
      IF(MODE .GT. 1) PRINT 201
      READ(LUIN1,100) KPOP, KCHG, KRAP, KDEM, KNEC
      IF(MODE .GT. 1) PRINT 202
      READ(LUIN1,120) ITI, ITF
      TIMEI = FLOAT(ITI)
      TIMEF = FLOAT(ITF)
      IF(ISENPR .NE. 0) ICHDAT = 1
      IF(ISENPR .NE. 0) ITABPR = 0
      IF(ITABPR .EQ. 0) GO TO 20
      IF(KPOP .EQ. 0) GO TO 2
      IF(MODE .GT. 1) PRINT 205, MODNAM(2)
      READ(LUIN1,100) IPPRT
    2 IF(KCHG .EQ. 0) GO TO 4
      IF(MODE .GT. 1) PRINT 205, MODNAM(3)
      READ(LUIN1,100) ICPRT
    4 IF(KRAP .LE. 0) GO TO 6
      NFIX(26) = NREGN
      IF(MODE .GT. 1) PRINT 205, MODNAM(5)
      READ(LUIN1,100) IFPRT
    6 IF(KRAP .EQ. 0) GO TO 8
      IF(MODE .GT. 1) PRINT 205, MODNAM(6)
      READ(LUIN1,100) IAPRT
      IF(MODE .GT. 1) PRINT 205, MODNAM(8)
      READ(LUIN1,100) IYPRT
    8 IF(KDEM .EQ. 0) GO TO 10
      IF(MODE .GT. 1) PRINT 205, MODNAM(7)
      READ(LUIN1,100) IDPRT
   10 IF(KNEC .EQ. 0) GO TO 12
      IF(MODE .GT. 1) PRINT 205, MODNAM(9)
      READ(LUIN1,100) INPRT
   12 IF(MODE .GT. 1) PRINT 206
      READ(LUIN1,120) IPDT, IPYR
C
   20 CONTINUE
      IF(MODE .GT. 1) PRINT 208
      IF(MODE .GT. 1) READ(LUIN1,130) ICMMT
C
      IF(MODE .GT. 1) GO TO 30
      WRITE(LUOUT,140) MODNAM(1),IDAT,ITIM,MLAB(MODE),ITI,ITF
      IF(KPOP .GT. 0) WRITE(LUOUT,150) MODNAM(2)
      IF(KCHG .GT. 0) WRITE(LUOUT,150) MODNAM(3)
```

```
      IF(KRAP  .GT.  0) WRITE(LUOUT,150)  MODNAM(4),MODNAM(8)
      IF(KRAP  .LT.  0) WRITE(LUOUT,150)  MODNAM(4),MODNAM(8)
      IF(KDEM  .GT.  0) WRITE(LUOUT,150)  MODNAM(6),MODNAM(8)
      IF(KNEC  .GT.  0) WRITE(LUOUT,150)  MODNAM(7)
      WRITE(LUOUT,170)                    MODNAM(9)
   25 READ(LUIN1,130)  ICMMT
      IF(ICMMT(1)  .EQ.  3HEND) GO TO 40
      WRITE(LUOUT,180)  ICMMT
      GO TO 25
   30 PRINT 140, MODNAM(1),IDAT,ITIM, MLAB(MODE),ITI,ITF
      IF(KPOP  .GT.  0) PRINT 150, MODNAM(2)
      IF(KCHG  .GT.  0) PRINT 150, MODNAM(3)
      IF(KRAP  .GT.  0) PRINT 150, MODNAM(4),MODNAM(8)
      IF(KRAP  .LT.  0) PRINT 150, MODNAM(6),MODNAM(8)
      IF(KNEC  .GT.  0) PRINT 150, MODNAM(9)
      IF(KDEM  .GT.  0) PRINT 150, MODNAM(7)
      PRINT 170
      PRINT 180, ICMMT
   40 CONTINUE
C
      RETURN
C
C     PRINT CONTROL ENTRY
C
      ENTRY CONTRT
C
      IF(ITABPR  .EQ.  0) GO TO 60
      IT = T + .0001
      IF(KPOP  .GT  0) CALL PRTSWT(IPPRT,KPPRT, IPDT,IPYR,IT,ITI,ITF ,15)
      IF(KCHG  .GT.  0) CALL PRTSWT(ICPRT,KCPRT,IPDT,IPYR,IT,ITI,ITF ,15)
      IF(KRAP  .GT.  0) CALL PRTSWT(IFPRT,KFPRT,IPDT,IPYR,IT,ITI,ITF ,15)
      IF(KRAP  .NE.  0) CALL PRTSWT(IAPRT,KAPRT,IPDT,IPYR,IT,ITI,ITF ,15)
      IF(KDEM  .GT.  0) CALL PRTSWT(IDPRT,KDPRT,IPDT,IPYR,IT,ITI,ITF ,15)
      IF(KRAP  .NE.  0) CALL PRTSWT(IYPRT,KYPRT,IPDT,IPYR,IT,ITI,ITF ,15)
      IF(KNEC  .GT.  0) CALL PRTSWT(INPRT,KNPRT,IPDT,IPYR,IT,ITI,ITF ,15)
      IF(KRAP  .LE.  0) GO TO 70
      NFIX(11) = KFPRT(12)
      NFIX(12) = KFPRT(13)
      GO TO 70
   60 CONTINUE
      IF(KRAP  .LE.  0) GO TO 70
      NFIX(11) = 0
      NFIX(12) = 0
   70 CONTINUE
C
   90 FORMAT(I1)
  100 FORMAT(20I2)
  120 FORMAT(6I5)
  130 FORMAT(7A10,A2)
  140 FORMAT(1H1////1H ,32(1H=)/
     1         33H KOREAN AGRICULTURAL SECTOR MODEL /
     2          1H ,32(1H=)//
     3          10H VERSION   ,A6//
     4          9H DATE RUN,A10//
     5          9H TIME RUN,A10//
     6           7H MODE   ,A5//
     7         13H INITIAL YEAR,I6//
     8         11H FINAL YEAR,I6//
     9         21H COMPONENT MODELS RUN )
  150 FORMAT(10X,A6)
  170 FORMAT(/16H JOB DESCRIPTION//)
  180 FORMAT(1H ,7A10,A2)
  200 FORMAT(/49H ENTER NRUN, KTRACK, ICHDAT, ISENPR, IPLT, ITABPR  ,
     1         22H, LVSTPR IN FORMAT 7I2 / 2H * )
  201 FORMAT(/35H ENTER KPOP, KCHG, KRAP, KDEM, KNEC ,
     1         14H IN FORMAT 5I2 / 2H * )
  202 FORMAT(/42H ENTER INITIAL TIME ITI AND FINAL TIME ITF ,
     1         14H IN FORMAT 2I5 / 2H * )
  205 FORMAT(/38H ENTER TABLE PRINT SWITCHES FOR MODEL ,A6,
     1         15H IN FORMAT 15I2 / 2H * )
  206 FORMAT(/45H ENTER TABLE PRINT INTERVAL IPDT (YEARS) AND,
     1         46H DESIGNATE FIVE YEARS FOR TABLE PRINTING IPYR ,
     2         14H IN FORMAT 6I5 / 2H * )
  208 FORMAT(/48H ENTER JOB DESCRIPTION COMMENT IN FORMAT 7A10,A2  /2H *)
C
      RETURN
      END
```

```
      BLOCKDATA CONTRD
C     DATA FOR CONTR
C     DECLARATIONS
      COMMON /CONTRC/ ANIMAL, BASEYR, DT, FARM, ITIME, KTRACK, KTA BI,
     1                IRUN, ITTY, ICHDAT, ISENPR, IPLT, ITABPR,
```

```
                          L\STPR, MODE, MODNAM(9),
     1                    KPOP, KCHG, KRAP, KDEM, KNEC, LINK, JPER,
     1                    LUIN1, LUIN2, LUOUT, LUPLOT(3), LURLP1,
     1                    LUPGEN, LUPOP,
     2                    KPPRT(15), KCPRT(15), KFPRT(15), KAPRT(15),
     2                    KDPRT(15), KYPRT(15), KNPRT(15),
     2                    NCC, NCOM, NCOMAG, NCROP, NLVST, NNC, NREGN,
     2                    NRUN, NS, NAME(20), PLANT,
     3                    T, TIMEI, TIMEF, TTABI, YEAR,
     4                    TOTSOR, TOTPOP, URBAN
     5
      INTEGER ANIMAL, FARM, PLANT, URBAN, TOTPOP, TOTSOR
      EQUIVALENCE (LUPLT1,LUPLOT(1)),(LUPLT2,LUPLOT(2)),
     1            (LUPLT3,LUPLOT(3))
      COMMON /LPCNTR/ NFIX(30),TOL(5),ERR(4),IOUT(4),TOBJ,TIMLP
      EQUIVALENCE
     1               (NFIX( 1),ML)
     2              ,(NFIX( 2),NL)
     3              ,(NFIX( 3),M)
     4              ,(NFIX( 4),N)
     5              ,(NFIX( 5),INBASE)
     5              ,(NFIX(15),MLE)
     5              ,(NFIX(16),NCOST)
     6              ,(NFIX(18),HDRTXT)
     0              ,(NFIX(24),MSUB)
     4              ,(NFIX(25),NSUB)
     5              ,(NFIX(26),NRGRLP)
     6              ,(NFIX(27),MADD)
     7              ,(NFIX(28),NADD)
     8
C
      COMMON /PMCNTR/ DTP, IALT, KMIG, NTIME, YEARO
C
C DATA
C
      DATA FARM, URBAN, TOTPOP / 1, 2, 3 /
C
      DATA PLANT, ANIMAL, TOTSOR / 1, 2, 3 /
C
      DATA NREGN, NCOM, NCOMAG, NCROP, NLVST, NS, NNC, NCC /
     1          1,   20,     19,    12,     5, 16,  11,   2
C
      DATA BASEYR, DT, KTABI, TTABI /
     1      1975., 1.0,    10, 1970  /
C
      DATA NAME /
     1              6HRICE  ,       6HBARLEY,       6HWHEAT ,       6HOGRA IN,
     2              6HFRUITS,       6HPULSES,       6HVEGTAB,       6HPOTA TO,
     3              6HTOBACO,       6HFORAGE,       6HCOCOON,       6HINDU ST,
     4              6HBEEF  ,       6HMILK  ,       6HPORK  ,       6HCHIC KN,
     5              6HEGGS  ,       6HFISH  ,       6HRESAGR,       6HNONA GR /
C
      DATA MODNAM /
     1        6HKASM3 , 6HPOPMIG, 6HCHANGE, 6HRAP   , 6HFRESAL,
     2        6HPRDAC , 6HDEMAND, 6HFRMAC , 6HNECON /
C
      DATA JPER /5/
C
      DATA LUIN1, LUIN2, LUOUT, LURLP1 / 5, 10, 6, 21 /
      DATA LUPLOT, LUPGEN / 26, 27, 8, 28 /
      DATA LUPOP, DTP, KMIG / 11, 1.0, 2 /
C
      DATA NFIX /
     1      60,61,60,61,1,400,120,100,1,0,0,0,1,0,
     2      62,61,1,6HNATION,0,0,0,0,1,57,61,1,3,0,0,0 /
C
      DATA TOL /
     1            1.E-7,   -1.E-4,    1.E-4,    1.E-10,    1.E6/
C
      END


      SUBROUTINE PRTSWT(IPRT,KPRT,IPDT,IPYR,IT,ITI,ITF,K)
C
C SET PRINT SWITCHES EACH TIME PERIOD
C
C DECLARATIONS
C
      DIMENSION IPRT(K), KPRT(K), IPYR(5)
C
C EXECUTION
C
      DO 90 I=1,K
      KPRT(I) = 0
      J = IPRT(I) + 1
      GO TO (90, 40, 50, 60, 70, 80) J
   40 CONTINUE
      IF(IT.LE.1976 .OR. IT.EQ.ITF) KPRT(I) = 1
      IF(IT.GT.1976 .AND. MOD(IT-1976,5).EQ.0) KPRT(I) = 1
```

```
      GO TO 90
   50 CONTINUE
      IF(MOD(IT-ITI,1) .EQ. 0) KPRT(I) = 1
      GO TO 90
   60 CONTINUE
      IF(IT.EQ.ITI .OR. IT.EQ.ITF) KPRT(I) = 1
      IF(MOD(IT-ITI,IPDT) .EQ. 0) KPRT(I) = 1
      GO TO 90
   70 CONTINUE
      IF(IT.EQ.ITI .OR. IT.EQ.ITF) KPRT(I) = 1
   80 CONTINUE
      DO 85 IY=1,5
      IF(IT .NE. IPYR(IY)) GO TO 85
      KPRT(I) = 1
      GO TO 90
   85 CONTINUE
   90 CONTINUE

      RETURN
      END
```

Utility Routines

```
      SUBROUTINE MATINV(A,INIV,M,N)
      DIMENSION A(M,M),INIV(N)
C
C         THIS ROUTINE INVERTS A MATRIX USING THE ROW
C         REDUCTION METHOD AND RETURNS THE RESULT IN THE
C         ARRAY PASSED TO IT.  THE ORIGINAL MATRIX NEED
C         NO LONGER BE DIMENSIONED M X 2M.
C
      DATA EPSIL/.1E-15/
C         RECORD SEQUENCE OF ROWS IN THE ORIGINAL MATRIX
      DO 1 I=1,N
    1 INIV(I) = I
      N1 = N - 1
C         BEGIN MAIN LOOP
      DO 50 IP=1,N
      IM = IP
      IST = IP + 1
      IF(IST.GT.N) GO  TO  10
C         FIND LARGEST ELEMENT OF KEY COLUMN
      DO 6  I=IST,N
      IF(ABS(A(I,1)) - ABS(A(IM,1)))8,8,7
    7 IM = I
    8 CONTINUE
C         CHECK FOR SINGULARITY
   10 IF(ABS(A(IM,1))-EPSIL)80,11,11
C         IF OFF-DIAGONAL, MOVE ROW TO DIAGONAL
   11 IF(IM.EQ.IP)GO TO 20
      IT = INIV(IP)
      INIV(IP) = INIV(IM)
      INIV(IM) = IT
      DO 15 J=1,N
      AL = A(IP,J)
      A(IP,J) = A(IM,J)
   15 A(IM,J) = AL
   20 CONTINUE
C         DIVIDE ROW BY DIAGONAL ELEMENT
      AL = A(IP,1)
      DO 25 J=1,N1
   25 A(IP,J) = A(IP,J+1) / AL
      A(IP,N) = 1. / AL
C         ZERO OUT OTHER ROWS IN THE COLUMN
      DO 40 I=1,N
      IF(I.EQ.IP)GO TO 40
      AL = A(I,1)
```

```
      DO 35 J=1,N1
35    A(I,J) = A(I,J+1) - AL * A(IP,J)
      A(I,N) = -AL * A(IP,N)
40    CONTINUE
50    CONTINUE
C         REORDER COLUMNS TO ORIGINAL SEQUENCE
      DO 75 I=1,N
      II = I + 1
      IF(INIV(I).EQ.I)GO TO 75
      DO 70 K=II,N
      IF(INIV(K).NE.I)GO TO 70
      INIV(K) = INIV(I)
      DO 65 J=1,N
      AL = A(J,I)
      A(J,I) = A(J,K)
65    A(J,K) = AL
      GO TO 75
70    CONTINUE
75    CONTINUE
      RETURN
80    PRINT 1001,EPSIL
      STOP
1001  FORMAT(1H0,5X,45HMATRIX IS SINGULAR OR HAS PIVOT SMALLER THAN ,
     1  E12.4)
      END
```

```
      FUNCTION MOD(I,MODE)
      L=I/MODE
      M=L*MODE
      MOD=I-M
      RETURN
      END
```

```
      FUNCTION TABEL(VAL, K, SMALL, DIFF, XARG)
  DELLARATIONS
      DIMENSION VAL(K)
C
C EXECUTION
C
      DUM = XARG - SMALL
      IF(DUM.LT.0.) DUM = 0.
      IF(DUM.GT.FLOAT(K-1)*DIFF) DUM = FLOAT(K-1)*DIFF
      I = 1. + DUM/DIFF
      IF(I.EQ.K) I = K-1
      TABEL = VAL(I) + ((VAL(I+1)-VAL(I))/DIFF)*(DUM-FLOAT(I-1)*DIFF)
C
      RETURN
      END
```

```
      FUNCTION TABUL (VAL,K,ARG,XARG)
      DIMENSION VAL(K),ARG(K)
      DUM = XARG
      IF(DUM .GT. ARG(K)) DUM = ARG(K)
      IF(DUM .LT. ARG(1)) DUM = ARG(1)
      DO 10 I=2,K
      IF (DUM .GT. ARG(I))GO TO 10
      TABUL  = VAL(I-1) + ((VAL(I) - VAL(I-1))/(ARG(I) - ARG(I-1)))*
     +         (DUM - ARG(I-1))
      RETURN
10    CONTINUE
      RETURN
      END
```

APPENDIX C

DEMAND STANDARD RUN OUTPUT

This appendix contains a listing of the output generated by the standard run of DEMAND. The standard run simulates the years 1970-1985, with output printed only for 1970 and 1985 for illustrative purposes. In addition to these annual output tables, the user has the option of specifying output in the form of plots and/or time-series tables generated by the KASM3 plot package.

```
KOREAN AGRICULTURAL SECTOR MODEL
================================

VERSION  KASM3

DATE RUN 05/09/77

TIME RUN.23.44.07.

MODE  BATCH

INITIAL YEAR  1970

FINAL YEAR  1985

COMPONENT MODELS RUN
          PRDAC
          FRMAC
          DEMAND

JOB DESCRIPTION

           DEMAND MODEL STANDARD RUN
```

TABLE DEMAND. 1. PRICE INDICES 1970.

INDEX	VALUE
	1970=100
CONSUMER PRICE INDEX	100.0
NONFOOD	100.0
FOOD	100.0
RICE	100.0
OTHER GRAINS	100.0
MEAT	100.0
WORLD FOOD PRICE INDEX	100.0

TABLE DEMAND. 2. SUPPLY AND DISAPPEARANCE OF AGRICULTURAL COMMODITIES 1970.

COMMODITY	DOMESTIC SUPPLY			DISAPPEARANCE				SELF SUFFIC-IENCY
	PRODUCTION	IMPORTS	TOTAL	FARM	URBAN	OTHER*	TOTAL	
	TH MT							PERCENT
RICE	4090.	541.	4631.	1648.	2422.	561.	4631.	93.
BARLEY	1591.	0.	1591.	813.	425.	352.	1591.	106.
WHEAT	219.	1257.	1476.	467.	813.	196.	1476.	15.
CGRAIN	134.	273.	413.	82.	59.	273.	413.	33.
FRUITS	420.	8.	428.	72.	313.	43.	428.	98.
PULSES	279.	36.	315.	90.	179.	46.	315.	89.
VEGTAB	2539.	5.	2544.	830.	1332.	382.	2544.	100.
POTATO	2764.	4.	2768.	1274.	549.	946.	2768.	100.
TOBACO	57.	-27.	29.	12.	14.	3.	29.	193.
INDUST	42.	-0.	42.	8.	32.	1.	42.	100.
BEEF	37.	1.	38.	3.	34.	1.	38.	98.
MILK	52.	54.	106.	1.	103.	2.	106.	49.
PORK	83.	-0.	82.	19.	62.	2.	82.	100.
CHICKN	45.	0.	45.	6.	38.	1.	45.	100.
EGGS	135.	-1.	134.	6.	125.	3.	134.	101.
FISH	935.	-134.	801.	160.	590.	52.	801.	117.
RESAGR	29.	376.	405.	114.	279.	12.	405.	7.

*NOTE: OTHER INCLUDES FEED,SEED, ,INDUSTRIAL CONSUMPTION, STOCK CHANGE, AND LOSSES

TABLE DEMAND. [illegible] CONSUMPTION AND CONSUMPTION TARGETS (NATIONAL, URBAN,FARM) BY COMMODITY 1970.

COMMODITY	NATIONAL CONSUMPTION	URBAN		FARM	
		CONSUMPTION	TARGET	CONSUMPTION	TARGET
	....	....	KG/CAP-YR	....	....
RICE	125.9	136.3	150.0	113.1	130.0
BARLEY	38.3	23.9	25.0	55.8	40.0
WHEAT	39.6	45.9	55.0	32.0	45.0
OGRAIN	4.4	3.3	2.0	5.6	4.0
FRUITS	11.9	17.6	40.0	4.9	30.0
PULSES	9.3	10.1	13.0	6.1	12.0
VEGTAB	66.9	75.0	110.0	57.0	100.0
POTATO	56.4	30.9	20.0	37.4	50.0
TOBACO	.8	.8	2.5	.8	2.5
FORAGE	0.0	0.0	.0	0.0	.0
COCOON	0.0	0.0	.0	0.0	.0
INDUST	1.3	1.8	4.0	.6	3.0
BEEF	1.1	1.9	5.0	.2	2.0
MILK	3.2	5.8	35.0	.1	7.0
PORK	2.5	3.5	10.0	1.3	5.0
CHICKN	1.4	2.1	5.0	.4	2.5
EGGS	[illegible]	[illegible]	15.0	.4	5.0
FISH	23.2	33.2	70.0	11.0	40.0
RESAGE	12.1	15.7	30.0	7.8	15.0

TABLE DEMAND. 4. URBAN INCOME AND EXPENDITURES 1970.

ITEM	TOTAL	PER CAPITA
	BI WON	TH WON/CAP
INCOME	1371.	77.2
SAVING	206.	11.6
EXPENDITURE	1165.	65.6
FOOD	508.	28.6
NON-FOOD	657.	37.0
		PERCENT
FOOD/TOTAL		43.6

TABLE DEMAND. 5. CALORIE AND PROTEIN CONSUMPTION PER CAPITA 1970.

NUTRIENT	QUANTITY			PERCENT		
	NATIONAL	URBAN	FARM	NATIONAL	URBAN	FARM
	.... CAL/CAP-DAY	....		.... PERCENT	....	
CALORIES	2392.2	2443.6	2329.5	100.0	100.0	100.0
PLANT	2228.7	2204.9	2246.5	93.0	90.2	96.4
ANIMAL	163.5	238.7	83.0	7.0	9.8	3.6
PROTEIN	66.0	71.2	59.7	100.0	100.0	100.0
PLANT	55.2	55.4	54.8	83.5	77.8	91.8
ANIMAL	10.9	15.8	4.9	16.5	22.2	8.2

TABLE DEMAND. 6. MARKETING PRICES AND COSTS IN CONSTANT 1970 WON BY COMMODITY 1970.

COMMODITY	PRODUCER PRICE	MARKETING COSTS	GOVT. SUBSIDY	CONSUMER PRICE*	EXPORT PRICE	IMPORT PRICE
	....	....	.... TH WON/MT	....	....	....
RICE	74.7	1.5	-2.9	79.1F	44.5	57.9
BARLEY	40.6	2.4	-5.7	48.8F	13.7	17.7
WHEAT	23.9	2.4	-4.1	30.4F	13.7	16.4
CGRAIN	33.4	32.0	5.1	60.3F	18.3	21.9
FRUITS	54.1	69.2	-10.7	133.9F	74.6	89.5
PULSES	82.6	19.9	3.8	98.7F	29.8	35.6
VEGTAB	44.1	12.8	-2.9	59.8F	406.2	487.5
POTATO	15.1	15.5	-9.1	39.7F	25.0	29.9
TOBACO	205.0	483.8	-512.5	1201.3F	220.2	264.3
FORAGE	1.0	0.0	1.0	.0F	.3	.3
COCOON	449.1	0.0	-.0	449.1F	5249.6	6299.5
INDUST	146.9	176.3	-33.5	356.7F	224.1	268.9
BEEF	659.6	46.2	-32.6	738.3F	286.2	343.5
MILK	55.0	59.9	5.6	109.3F	262.8	315.2
PORK	261.1	148.3	13.3	396.7F	239.3	287.1
CHICKN	366.1	-69.6	33.5	263.0F	216.2	258.8
EGGS	216.4	51.9	-17.7	236.0F	234.5	281.7
FISH	70.2	190.9	4.0	257.1F	139.5	167.5
RESAGR	68.4	23.9	.0	92.4F	29.2	35.0

*COMPUTATIONAL NOTE 0 0 ITERATIONS TO OBTAIN CONVERGENCE
S1 = 1.00

L - PRICE AT LOWER BOUND
U - PRICE AT UPPER BOUND
M - PRICE DETERMINED BY MARKET
F - FIXED PRICE

TABLE DEMAND. 9. AGRICULTURAL EXPORT-IMPORT BALANCE 1970.

ITEM	UNITS	VALUE
AGRICULTURAL EXPORTS	BI WON	137.
AGRICULTURAL IMPORTS	BI WON	85.
EXCHANGE RATE	WON/DOLLAR	305.
AGRICULTURAL BALANCE OF PAYMENTS	BI WON	52.
	MI DOLLAR	171.
PROFIT FROM AGRICULTURAL TRADE	MI WON	110727.

TABLE DEMAND.10. AGRICULTURAL IMPORTS AND EXPORTS BY COMMODITY 1970.

COMMODITY	IMPORTS QUANTITY	IMPORT PRICE	TARIFF REVENUES	VALUE*	PROFIT	EXPORTS QUANTITY	EXPORT PRICE	VALUE	PROFIT
	TH MT	TH WON	MI WON	MI WON	MI WON	TH MT	TH WON	MI WON	MI WON
RICE	541.000	57.3	4317.2	26382.4	11507.1	0.000	44.5	0.0	0.0
BARLEY	.000	17.7	.0	.0	.0	0.000	13.7	0.0	0.0
WHEAT	1257.153	19.4	3445.2	17226.1	17564.9	0.000	13.7	0.0	0.0
CGRAIN	279.000	21.9	0.0	6116.8	10715.3	0.000	19.3	0.0	0.0
FRUITS	8.132	89.5	0.0	733.4	363.5	0.000	74.6	0.0	0.0
PULSES	36.000	35.6	0.0	1282.6	2269.2	0.000	29.8	0.0	0.0
VEGTAB	5.000	487.5	0.0	2437.5	-2138.6	0.000	406.2	0.0	0.0
POTATO	4.000	29.5	0.0	113.4	39.3	0.000	25.0	0.0	0.0
TOBACO	0.000	264.3	0.0	0.0	0.0	27.400	220.2	6032.2	-26884.6
FORAGE	0.000	.3	0.0	0.0	0.0	0.000	.3	0.0	0.0
COCOON	0.000	6299.5	0.0	0.0	0.0	21.368	5249.6	112174.7	102578.2
INDUST	0.000	268.9	0.0	0.0	0.0	.115	224.1	25.8	-15.3
BEEF	.600	343.5	0.0	206.1	266.9	0.000	286.2	0.0	0.0
MILK	54.400	315.2	0.0	17144.6	-11195.6	0.000	262.8	0.0	0.0
PORK	0.000	287.1	0.0	0.0	0.0	.100	239.3	23.9	-15.7
CHICKN	.000	258.8	0.0	.0	.0	0.000	216.2	0.0	0.0
EGGS	0.000	281.7	0.0	0.0	0.0	1.500	234.5	351.7	-77.3
FISH	0.000	167.5	0.0	0.0	0.0	134.227	139.5	18719.4	-15790.3
RESAGR	375.701	35.0	0.0	13156.1	21542.9	0.000	29.2	0.0	0.0

* INCLUDES TARIFF REVENUES

DEMAND TIME-SERIES TRACKING MEASURES

BY COMMODITY	.0115	.0171	.1080	.1036	.0218	.0189	.0245	.0209	.3261	0.0000
	0.0000	.2896	.0200	.3915	.0036	.0096	.0486	.0430	.3099	0.0000
FOR GRAINS	.2600									
FOR FRT + VEG	.0464									
FOR LVSTK/FSH	.5165									
FOR ALL COMM	2.2706									

TABLE DEMAND. 1. PRICE INDICES 1985.

INDEX	VALUE
	1970=100
CONSUMER PRICE INDEX	116.0
NONFOOD	95.0
FOOD	130.4
RICE	157.5
OTHER GRAINS	120.5
MEAT	143.5
WORLD FOOD PRICE INDEX	156.4

TABLE DEMAND. 2. SUPPLY AND DISAPPEARANCE OF AGRICULTURAL COMMODITIES 1985.

COMMODITY	DOMESTIC SUPPLY			DISAPPEARANCE				SELF SUFFIC-IENCY
	PRODUCTION	IMPORTS	TOTAL	FARM	URBAN	OTHER*	TOTAL	
	TH MT							PERCENT
RICE	5932.	0.	5932.	1378.	3802.	751.	5932.	105.
BARLEY	2365.	0.	2366.	580.	1151.	634.	2366.	98.
WHEAT	492.	1758.	2250.	510.	1490.	250.	2250.	22.
CGRAIN	159.	0.	159.	55.	125.	-20.	159.	100.
FRUITS	1563.	-423.	1140.	139.	845.	156.	1140.	137.
PULSES	633.	-87.	552.	96.	366.	91.	552.	116.
VEGTAB	4321.	-203.	4112.	865.	2599.	648.	4112.	105.
POTATO	3187.	-232.	2954.	931.	753.	1270.	2954.	108.
TOBACO	210.	-128.	82.	24.	49.	9.	82.	256.
INDUST	82.	33.	114.	9.	103.	3.	114.	72.
BEEF	143.	-17.	126.	10.	114.	3.	126.	113.
MILK	780.	-26.	754.	23.	703.	28.	754.	103.
PORK	192.	-5.	187.	30.	154.	4.	187.	103.
CHICKN	113.	0.	113.	16.	95.	2.	113.	100.
EGGS	276.	-0.	276.	10.	261.	6.	276.	100.
FISH	4712.	-2031.	2681.	430.	2010.	241.	2631.	176.
RESAGR	87.	516.	602.	92.	492.	18.	602.	14.

*NOTE: OTHER INCLUDES FEED, SEED, ,INDUSTRIAL CONSUMPTION, STOCK CHANGE, AND LOSSES

TABLE DEMAND. 3. PER CAPITA CONSUMPTION AND CONSUMPTION TARGETS (NATIONAL, URBAN,FARM) BY COMMODITY 1985.

COMMODITY	NATIONAL CONSUMPTION	URBAN		FARM	
		CONSUMPTION	TARGET	CONSUMPTION	TARGET
	KG/CAP-YR				
RICE	125.9	130.8	100.0	117.4	110.0
BARLEY	42.4	39.6	25.0	49.4	40.0
WHEAT	49.0	51.3	55.0	43.5	45.0
OGRAIN	4.4	4.3	2.0	4.6	4.0
FRUITS	24.1	29.1	40.0	11.8	30.0
PULSES	11.3	12.6	18.0	8.2	12.0
VEGTAB	84.9	89.4	110.0	73.7	100.0
POTATO	41.3	25.9	20.0	79.3	50.0
TOBACO	1.8	1.7	2.5	2.0	2.5
FORAGE	0.0	0.0	.0	0.0	.0
COCOON	0.0	0.0	.0	0.0	.0
INDUST	2.7	3.6	4.0	.7	3.0
BEEF	3.9	3.9	5.0	.8	2.0
MILK	17.8	24.2	35.0	2.0	7.0
PORK	4.5	5.3	10.0	2.5	5.0
CHICKN	2.7	3.3	5.0	1.3	2.5
EGGS	6.6	9.0	15.0	.9	5.0
FISH	59.8	69.1	70.0	36.7	40.0
RESAGR	14.3	16.9	30.0	7.8	15.0

TABLE DEMAND. 4. URBAN INCOME AND EXPENDITURES 1985.

ITEM	TOTAL	PER CAPITA
	BI WON	TH WON/CAP
INCOME	3907.	134.4
SAVING	586.	20.2
EXPENDITURE	3321.	114.2
FOOD	1489.	51.2
NON-FOOD	1832.	63.0
		PERCENT
FOOD/TOTAL		44.8

TABLE DEMAND. 5. CALORIE AND PROTEIN CONSUMPTION PER CAPITA 1985.

NUTRIENT	QUANTITY			PERCENT		
	NATIONAL	URBAN	FARM	NATIONAL	URBAN	FARM
	.... CAL/CAP-DAY			.... PERCENT		
CALORIES	2703.2	2793.8	2478.7	100.0	100.0	100.0
PLANT	2397.1	2422.2	[illegible]	88.7	86.7	94.2
ANIMAL	306.1	371.6	143.9	11.3	13.3	5.8
PROTEIN	84.4	90.4	69.5	100.0	100.0	[illegible]
PLANT	61.7	63.1	58.2	73.1	[illegible]	[illegible]
ANIMAL	22.7	27.3	[illegible]	26.9	30.2	[illegible]

TABLE DEMAND. 6. MARKETING PRICES AND COSTS IN CONSTANT 1970 WON BY COMMODITY 1985.

COMMODITY	PRODUCER PRICE	MARKETING COSTS	GOVT. SUBSIDY	CONSUMER PRICE*	EXPORT PRICE	IMPORT PRICE
	....	....	....	TH WON/MT	....	....
RICE	120.0	2.4	-2.2	124.6F	55.6	68.3
BARLEY	60.0	3.6	-1.4	65.0F	36.5	49.0
WHEAT	45.0	4.5	4.7	44.8F	28.8	41.3
OGRAIN	35.5	52.1	-.0	167.6U	21.0	33.5
FRUITS	61.5	78.7	.0	140.3L	85.0	92.5
PULSES	72.6	17.4	.0	90.1L	54.7	67.2
VEGTAB	73.8	21.4	-.0	95.2M	38.7	43.7
POTATO	9.9	10.2	0.0	20.2L	74.1	79.1
TOBACO	400.0	944.0	644.0	700.0F	378.2	383.2
FORAGE	.0	0.0	0.0	.0F	.5	.5
COCOCN	855.0	0.0	-345.0	1200.0F	7750.0	7755.0
INDUST	158.7	190.5	0.0	349.2L	410.0	415.0
BEEF	1023.0	71.6	.0	1094.6L	1563.7	1568.7
MILK	56.3	61.4	-.0	117.8L	392.5	400.0
PORK	485.2	276.6	-.0	761.8M	842.5	347.5
CHICKN	553.8	-105.2	0.0	448.5M	341.8	346.8
EGGS	130.3	31.3	.0	161.5M	299.6	304.6
FISH	105.5	286.8	-.0	392.3L	392.3	394.8
RESAGR	74.1	25.9	0.0	100.0F	129.5	134.5

*COMPUTATIONAL NOTE 0 9 ITERATIONS TO OBTAIN CONVERGENCE
S1 = 1.04

L - PRICE AT LOWER BOUND
U - PRICE AT UPPER BOUND
M - PRICE DETERMINED BY MARKET
F - FIXED PRICE

174

TABLE DEMAND. 7. ELASTICITIES URBAN AND RURAL BY COMMODITY 1985

COMMODITY	URBAN		RURAL	
	DIRECT PRICE ELASTICITY*	INCOME ELASTICITY	DIRECT PRICE ELASTICITY*	INCOME ELASTICITY
RICE	-.30	.08	-.40	.05
BARLEY	-.20	-.16	-.20	-.12
WHEAT	-.70	.20	-.40	.11
CGRAIN	-.60	-.43	-.60	-.05
FRUITS	-.85	.66	-.35	.56
PULSES	-.75	.32	-.40	.21
VEGTAB	-.10	.26	-.10	.19
POTATO	-.40	-.42	-.70	-.35
TOBACO	-.50	.52	-.20	.37
FORAGE	0.00	0.00	0.00	0.00
COCOON	0.00	0.00	0.00	0.00
INDUST	-1.10	.18	-.50	.29
BEEF	-1.40	.54	-1.80	.64
MILK	-1.50	1.31	-1.50	2.31
PORK	-1.00	.41	-.50	.41
CHICKN	-1.20	.66	-.80	.46
EGGS	-.30	.31	-.40	.37
FISH	-.20	.10	-.30	.27
RESAGR	0.00	0.00	0.00	0.00
NONAGR	-.40	.71	0.00	0.00

*CONSTANT VALUE DURING RUN

TABLE DEMAND. 8. CROSS ELASTICITIES, URBAN AND FARM 1985.

QUANTITY	WITH PRICE	CROSS ELASTICITY
URBAN CONSUMPTION		
RICE	BARLEY	.03
RICE	WHEAT	.03
RICE	OGRAIN	.01
RICE	POTATO	.00
RICE	NONAGR	.09
BARLEY	RICE	.64
BARLEY	WHEAT	.05
BARLEY	OGRAIN	.01
BARLEY	POTATO	.01
BARLEY	NONAGR	-.35
WHEAT	RICE	.27
WHEAT	BARLEY	.08
WHEAT	OGRAIN	.02
WHEAT	POTATO	.01
WHEAT	NONAGR	.12
OGRAIN	RICE	.48
OGRAIN	WHEAT	1.81
OGRAIN	NONAGR	-1.26
FRUITS	NONAGR	.19
PULSES	NONAGR	.43
VEGTAB	NONAGR	-.16
POTATO	OGRAIN	.01
POTATO	NONAGR	.80
TOBACO	NONAGR	-.12
INDUST	NONAGR	.92
BEEF	PORK	.39
BEEF	CHICKN	.15
BEEF	FISH	.36
BEEF	NONAGR	-.03
MILK	NONAGR	.19
PORK	BEEF	.44
PORK	CHICKN	.14
PORK	FISH	.44
PORK	NONAGR	-.43
CHICKN	BEEF	.33
CHICKN	PORK	.32
CHICKN	FISH	.29
CHICKN	NONAGR	-.40
EGGS	NONAGR	-.01
FISH	BEEF	.03
FISH	PORK	.04
FISH	CHICKN	.01
FISH	NONAGR	.03
NONAGR	RICE	-.06

NONAGR	BARLEY	-.01
NONAGR	WHEAT	-.01
NONAGR	OGRAIN	-.00
NONAGR	FRUITS	-.02
NONAGR	PULSES	-.00
NONAGR	VEGTAB	-.03
NONAGR	POTATO	-.00
NONAGR	TOBACO	-.00
NONAGR	INDUST	-.01
NONAGR	BEEF	-.02
NONAGR	MILK	-.01
NONAGR	PORK	-.01
NONAGR	CHICKN	-.01
NONAGR	EGGS	-.01
NONAGR	FISH	-.10
NONAGR	RESAGR	-.01

FARM CONSUMPTION

RICE	BARLEY	.04
RICE	WHEAT	.05
RICE	OGRAIN	.01
RICE	POTATO	.01
BARLEY	RICE	.58
BARLEY	WHEAT	.02
BARLEY	OGRAIN	.01
BARLEY	POTATO	.05
WHEAT	RICE	.32
WHEAT	BARLEY	.10
WHEAT	OGRAIN	.02
WHEAT	POTATO	.10
OGRAIN	RICE	.46
OGRAIN	BARLEY	.10
OGRAIN	WHEAT	.70
POTATO	RICE	.11
POTATO	BARLEY	.02
POTATO	OGRAIN	.01
BEEF	PORK	.44
BEEF	CHICKN	.17
BEEF	FISH	.51
PORK	BEEF	.27
PORK	CHICKN	.11
PORK	FISH	.88
CHICKN	BEEF	.26
CHICKN	PORK	.27
CHICKN	FISH	.55
FISH	BEEF	.01
FISH	PORK	.01
FISH	CHICKN	.00

TABLE DEMAND. 9. AGRICULTURAL EXPORT-IMPORT BALANCE 1985.

ITEM	UNITS	VALUE
AGRICULTURAL EXPORTS	BI WON	1395.
AGRICULTURAL IMPORTS	BI WON	143.
EXCHANGE RATE	WON/DOLLAR	500.
AGRICULTURAL BALANCE OF PAYMENTS	BI WON	1252.
	MI DOLLAR	2504.
PROFIT FROM AGRICULTURAL TRADE	MI WON	309746.

TABLE DEMAND.10. AGRICULTURAL IMPORTS AND EXPORTS BY COMMODITY 1985.

COMMODITY	IMPORTS					EXPORTS			
	QUANTITY	IMPORT PRICE	TARIFF REVENUES	VALUE*	PROFIT	QUANTITY	EXPORT PRICE	VALUE	PROFIT
	TH MT	TH WON	MI WON	MI WON	MI WON	TH MT	TH WON	MI WON	MI WON
RICE	0.000	68.3	0.0	0.0	0.0	0.000	55.8	0.0	0.0
BARLEY	0.000	49.0	0.0	0.0	0.0	0.000	36.5	0.0	0.0
WHEAT	1758.305	41.3	12098.6	60493.0	6138.6	0.000	28.8	0.0	0.0
CGRAIN	0.000	33.5	0.0	0.0	0.0	0.000	21.0	0.0	0.0
FRUITS	0.000	92.5	0.0	0.0	0.0	423.231	85.0	35974.6	-23333.5
PULSES	0.000	67.2	0.0	0.0	0.0	86.904	54.7	4750.8	-3075.6
VEGTAB	0.000	43.7	0.0	0.0	0.0	208.471	38.7	8075.1	-11778.6
POTATO	0.000	79.1	0.0	0.0	0.0	232.444	74.1	17215.7	12524.8
TOBACO	0.000	383.2	0.0	0.0	0.0	127.851	378.2	48355.3	-41140.4
FORAGE	0.000	.5	0.0	0.0	0.0	0.000	.5	0.0	0.0
COCOON	0.000	7755.0	0.0	0.0	0.0	57.242	7750.0	443624.0	374933.8
INDUST	32.533	415.0	0.0	13503.5	-2139.8	0.000	410.0	0.0	0.0
BEEF	0.000	1568.7	0.0	0.0	0.0	16.857	1563.7	26359.3	7907.8
MILK	0.000	400.0	0.0	0.0	0.0	25.956	392.5	10187.8	7131.5
PORK	0.000	847.5	0.0	0.0	0.0	4.900	842.5	4128.4	395.4
CHICKN	.000	346.8	0.0	.0	.0	0.000	341.8	0.0	0.0
EGGS	0.000	304.6	0.0	0.0	0.0	.000	299.6	.0	.0
FISH	0.000	394.3	0.0	0.0	0.0	2031.065	392.3	796738.9	0.0
RESAGR	515.618	134.5	0.0	69330.0	-17768.2	0.000	129.5	0.0	0.0

* INCLUDES TARIFF REVENUES

APPENDIX D

COMPUTATION OF CROSS-PRICE ELASTICITIES

DEMAND computes cross-price elasticities for grains and for meats and for farm and nonfarm consumers based on own-price elasticities, assumed substitution proportions and relative consumption levels. Since relative consumption changes over time in DEMAND, so do the cross-price elasticities. The development of these computations, discussed below, was motivated by two factors: 1) the need for a way to more easily utilize "soft" data, such as informed judgements of knowledgeable people familiar with Korean consumption patterns and behavior, to estimate demand elasticities; and 2) the desirability of maintaining consistency between proportional changes in consumption, represented by the elasticities, and realistic quantity changes.

First, enormous difficulties were encountered in trying to estimate price elasticities from time-series data only [7]. Hundreds of computer runs of statistical packages, eating up computer time and staff time, had little to show except bad signs, statistically insignificant results and low, even negative, R^2s. The decision was made to take as much as we reasonably could from these "hard" data results and to rely for the rest on "soft" data, i.e., informed guesstimates. The reasonableness of elasticity estimates made in this way could then be measured by time-series tracking tests of the model, as reported in the main body of this documentation.

It was felt that it would be easier to elicit elasticity estimates from "soft" data sources indirectly through such reasoning as illustrated below, i.e., in quantity terms rather than proportional terms, than it would be to ask directly for elasticity values. Indeed, the workability of this approach was verified in the case of substitution behavior for meats (beef, pork, chicken and fish). Reasoning in this way, officials of the Livestock Bureau were able to come up with substitution proportions (see below) which gave very good results when used in the model.

The second factor motivating development of this approach was the obvious observation that cross-price elasticities measure the strength of substitution behavior between pairs of commodities. For any pair of commodities, this strength depends on the relative importance of, or preference for, those two commodities, a relative importance which can be measured by the relative consumption levels. In long-run analyses, relative consumption levels can change substantially, especially for developing countries. Therefore, constant price elasticities would not be consistent with changing consumption patterns over time.

In DEMAND, the farm and nonfarm demand components each have a 20x 20 matrix of price elasticities. For each, two submatrices are estimated in the following manner. One, of dimension 5x5, is for the grains, rice, barley, wheat, other grains and potatoes; the other, of dimension 4x4, is for the meats, beef, pork, chicken and fish. Rows and columns for the other commodities are assumed to have only the direct-price elasticity on the diagonal. Income elasticities are estimated separately and are assumed to neither affect nor depend on the price elasticities. The homogeneity condition is insured by computing, as a residual, the cross-price elasticity of demand for each commodity with respect to the nonfood price index.

First, own-price elasticities are assumed given, and assumptions are made about quantity substitutions. Then, cross-price elasticities are derived consistent with those assumptions. Since grains and meats are handled in exactly the same way, only cross-price elasticities for grains (Figure D.1) will be derived in detail here.

Assume a change occurs in the price of rice, with all other prices (and income) remaining unchanged. Then, only the first column of the matrix in Figure D.1 is effective.

Prices

Quantities	Rice	Barley	Wheat	Other Grains	Potatoes
Rice	ε_{11}	ε_{12}	ε_{13}	ε_{14}	ε_{15}
Barley	ε_{21}	ε_{22}	ε_{23}	ε_{24}	ε_{25}
Wheat	ε_{31}	ε_{32}	ε_{33}	ε_{34}	ε_{35}
Other grains	ε_{41}	ε_{42}	ε_{43}	ε_{44}	ε_{45}
Potatoes	ε_{51}	ε_{52}	ε_{53}	ε_{54}	ε_{55}

Figure D.1. Submatrix of Grain Price Elasticities

Further, assume the price change and direct elasticity ε_{11} (taken as given) are such that rice consumption decreases 10 kg per capita. That is,

$$(1) \quad dC_1 = \frac{\partial C_1}{\partial P_1} dP_1 = \varepsilon_{11} \frac{C_1}{P_1} dP_1 = -10$$

or in difference equation form

$$(2) \quad \Delta C_1 = C_1(t) - C_1(t-1) = \varepsilon_{11} \frac{\Delta P_1}{P_1(t-1)} C_1(t-1) = -10$$

These equations follow from the definition of the elasticity as shown in equation (3) below.

Now, what will happen to total grain consumption as a result of consuming 10 kg less rice? We would expect a substitution of other grains for rice, but how much? Suppose our best information (including judgement) tells us 9 kg. That is, there will be a net decrease in total grain consumption of 1 kg per capita resulting from the assumed rice price change (increase). Thus, we set the value of the parameter α to .9. That is,

$$(3) \quad \Delta C_2 + \Delta C_3 + \Delta C_4 + \Delta C_5 = -\alpha_1 \Delta C_1 = .9 \times 10 = 9$$
$$= -\alpha_1 \varepsilon_{11} \frac{\Delta P_1}{P_1(t-1)} C_1(t-1)$$

where we have used equation (2) to replace ΔC_1. Next, we need to know how much of the 9 kg will be barley, wheat flour, and other grains and potatoes (in grain equivalents), respectively. Suppose our data tells us 5 kg, 2 kg, 1 kg, and 1 kg, respectively. Thus, we set the values of β_{21}, β_{31}, β_{41} and β_{51} to 5/9, 2/9, 1/9 and 1/9, respectively, and

$$(4) \quad \Delta C_2 = C_2(t) - C_2(t-1) = -\beta_{21} \alpha_1 \varepsilon_{11} \frac{\Delta P_1}{P_1(t-1)} C_1(t-1) = 5$$

$$\Delta C_3 = C_3(t) - C_3(t-1) = -\beta_{31}\,\alpha_1\,\varepsilon_{11}\,\frac{\Delta P_1}{P_1(t-1)}\,C_1(t-1) = 2$$

$$\Delta C_4 = C_4(t) - C_4(t-1) = -\beta_{41}\,\alpha_1\,\varepsilon_{11}\,\frac{\Delta P_1}{P_1(t-1)}\,C_1(t-1) = 1$$

$$\Delta C_5 = C_5(t) - C_5(t-1) = -\beta_{51}\,\alpha_1\,\varepsilon_{11}\,\frac{P_1}{P_1(t-1)}\,C_1(t-1) = 1$$

The cross elasticities in the first column of Figure D.1 can now be computed.

$$(5)\quad \varepsilon_{11}(t-1) = \frac{\Delta C_i}{C_i(t-1)} \Big/ \frac{\Delta P_1}{P_1(t-1)} = -\beta_{11}\,\alpha_1\,\varepsilon_{11}\,\frac{C_1(t-1)}{C_i(t-1)} \quad \text{for } i = 2, 3, 4, 5$$

Through similar reasoning, we can estimate α_2, β_{12}, β_{32}, β_{42} and β_{52} for the barley column (a change only in barley price); α_3, β_{13}, β_{23}, β_{43} and β_{53} for wheat; α_4, β_{14}, β_{24}, β_{34} and β_{54} for other grains; and α_5, β_{15}, β_{25}, β_{35} and β_{45} for potatoes. From these, the full set of cross elasticities is easily computed as in (5). Actually, (5) is modified to (6) in order to convert from wheat grain to wheat flour and from potatoes as potatoes to potatoes in grain equivalents. (The conversion ratios for rice, barley and other grains (α_1, α_2 and α_4) are taken as 1.0.)

$$(6)\quad \varepsilon_{ij}(t-1) = -\beta_{ij}\alpha_j\varepsilon_{jj}\,\frac{C_j(t-1)\cdot\gamma_j}{C_i(t-1)\cdot\gamma_i} \quad \text{for } i,j = 1, 2, 3, 4, 5 \text{ and } i \neq j$$

The same procedure is followed for meats (although no conversion ratios are used) and for farm and nonfarm consumers. In the linearized, difference equation demand model, then, consumption at time t depends on elasticities at time t-DT, where DT=1 in the above equations. Notice that the cross elasticities now change over time as relative consumptions change.

Also, notice that with this method we now have more parameters than we had before. For n commodities (here, n=5), instead of n^2 direct and cross-price elasticities, we have n direct elasticiites, n αs and n^2-n βs for a total of n^2+n. However, through the type of reasoning illustrated above, we can more confidently use soft data to arrive at reasonable and mutually consistent cross elasticities by first estimating the substitution proportions α and β.

In addition, basic changes in consumption patterns can be projected (or assumed) by projecting changes in the αs and βs over time. We must remember in doing so, however, that we are talking here only of changes in price substitution behavior. Change in consumer behavior associated with rising incomes is the subject of income elasticities, not price elasticities, although income could indirectly affect price responses if the αs, βs and/or direct-price elasticities were made functions of income in later model refinements.

A final point can be made relating the substitution proportions (the αs and βs) to partial derivatives. As noted above, the elasticity of demand for commodity i with respect to the price of commodity j can be defined as

$$(7) \quad \varepsilon_{ij} = \frac{\partial C_i}{\partial P_j} \cdot \frac{P_j}{C_i}$$

or the partial derivative of C_i with respect to P_j as

(8) $$\frac{\partial C_i}{\partial P_j} = \varepsilon_{ij} \frac{C_i}{P_j}$$

From equation (6)

(9) $$\frac{\partial C_i}{\partial P_j} = -\beta_{ij}\, \alpha_j\, \varepsilon_{jj} \frac{\gamma_j C_j}{\gamma_i C_i} \cdot \frac{C_i}{P_j}$$

$$= -\beta_{ij}\, \alpha_j \frac{\partial C_j}{\partial P_j} \frac{P_j}{C_j} \frac{\gamma_j C_j}{\gamma_i P_j}$$

$$= -\beta_{ij}\, \alpha_j \frac{\gamma_j}{\gamma_i} \frac{\partial C_j}{\partial P_j}$$

$$= \mu_{ij} \frac{\partial C_j}{\partial P_j}$$

where $\mu_{ij} = -\beta_{ij}\, \alpha_j\, \gamma_j/\gamma_i \leq 0$ is constant. Thus, the cross-price partial derivative of i with respect to j is proportional to j's own-price partial derivative. That is, the slope of the curve of C_i plotted against P_j is proportional to that of C_j plotted against P_j and of opposite sign.

APPENDIX E

DATA DOCUMENTATION

The data for the demand-price-foreign trade model (DEMAND), as for any model, can be classified into four categories: 1) initial conditions, 2) system parameters, 3) policy inputs and 4) other inputs. Some initial conditions and constant parameters are computed within the model at the start of a simulation to insure consistency among them and with the real world at time zero. These are described in the technical description of DEMAND--for example, the constant coefficient ALPHA of the nonfarm consumption expenditure constraint (equation (14) in Chapter 2) is computed in equation (27), and initial per capita consumption levels are computed in equations (28)-(30). We shall describe here only the numbers that must be supplied to the model.

The model is currently programmed so that all DEMAND data, with exceptions as noted in the tables, are entered as data statements in BLOCKDATA DEMDAT (see Appendix B, Program List). These data may be easily changed for policy runs, sensitivity testing or updating.

Initial Conditions

Three categories of data are needed to initialize DEMAND: 1) initial conditions of state variables, 2) data used to insure initial consistency in consumption, and 3) time series values of certain variables to be used in the tracking period of the model.

State Variable Initialization

Five state variables in DEMAND must be initialized at 1970: 1) stock levels $STKO_i$, 2) exponentially averaged producer prices $EPAVGO_i$, 3) farm income elasticities $ELISVAR_i$, and 4) nonfarm income elasticities $ELISAV_i$ for each commodity i, and 5) the income constraint parameter S1. Data for STKO, EPAVGO, ELISVAR and ELISAV are shown in Table E.1. Since the coefficient ALPHA is computed within the model to insure initial consistency among income, per capita consumption and prices, S1 is initialized at its nominal value 1.0 by a replacement statement in subroutine DEMI. Other state variables of DEMAND are initialized either within the model (e.g., per capita consumption) or as tracking data (e.g., prices). If a simulation is to begin at a time later than 1970 (i.e., TIMEI > 1970, where TIMEI is the desired starting time), tracking period values of stock changes and producer prices are used by the model instead of STKO and EPAVGO to initialize stock levels and lagged producer prices.

Initial Consistency Data

Per capita consumption is initialized by the model to be consistent with the food balance data for 1970 (or later initial year if desired). This means consistency with total supply for human consumption and with ratios of farm per capita consumption to nonfarm per capita consumption (FNR_i) for each commodity i. Total supply is production plus imports minus losses minus animal feed minus stock increases minus seed requirements minus industrial consumption (for wheat) minus nonfood portions (for industrial crops). All of this data is considered a part of the production models of KASM and is not shown here; Table E.1 shows the initial ratio of farm to nonfarm per capita consumption.

Tracking Period Time Series

In KASM as a whole, simulations begin in 1970. Empirical time series data is stored in the model, however, for selected variables from 1970 to the latest year available (called BASEYR in KASM). At the time of this writing, data has been compiled up to 1975 (i.e., BASEYR = 1975), although storage is reserved in the data arrays for new values to be added as they become available up through 1979. When simulating, the period 1970 through BASEYR is called the tracking period, and the period BASEYR + 1 to the end of the simulation is called the projection period.

The tracking period data included in DEMAND is shown in Table E.2. The variables shown in Table E.2 fall into four categories, depending on how the model uses the time series data.

1) First are variables computed by DEMAND in the projection period but whose empirical values are used during the tracking period. These include farm and nonfarm consumption expenditures VEFRMY and VEINCM, producer and consumer prices VPRICE and VCPU, and stock changes VCSTK.

2) Next are DEMAND inputs (discussed below) and time-varying parameters. These include a) variables computed in other models of KASM but projected exogenously if DEMAND is run independently of those models (farm and nonfarm income VFI and VUI, and the nonfood price index VCP); and b) variables and parameters exogenous to KASM (import and

export prices VPWI and VPWX, the foreign exchange rate VWOND, the food proportion of nonfarm consumption expenditures VPCUFD, the edible proportion of industrial crops VPICF, and the ratio of nonfarm to farm consumer price indices VFPIX).

3) Third are variables computed by DEMAND in both the tracking and projection periods but whose simulated and empirical tracking period values may be compared as a verification test. Included in this group are per capita consumption VPSUP and imports/exports VDFC.

4) Finally, time series data on the all-cities all-commodities consumer price index VCPI is included for reference purposes only. It is not used directly in KASM, but it is the implicit deflator for all prices and price indices used in the model.

System Parameters

The system parameters of DEMAND are constants which determine the strength of behavioral and technical relationships among variables. The user may want to experiment with the model to investigate the effects of making these parameters time-varying. This could be done without too much difficulty by modifying the computer program. One might want to do this for, for example, direct-price elasticities or substitution proportions.

The values of commodity-specific system parameters are displayed in Tables E.3 and E.4. Table E.3 shows, for all commodities, own-price elasticities, consumption targets, marketing margins, nutritional contents of foods, import losses, proportions consumed unprocessed, and price index weights. For grains and meats, Table E.4 shows the substitution proportions used by the model to determine cross-price elasticities. Finally, Table E.5 lists other, miscellaneous parameters.

Policy and Other Inputs

Policy inputs to DEMAND are discussed in detail in Chapters 2 and 4 of this documentation. No further discussion will be made here. The reader is referred to Chapter 5 for a table of base run values of policy parameters (Table 5.2); those values produced the sample output of Appendix C.

Other inputs to DEMAND include those projected exogenously in table functions and those which are computed elsewhere in the sector model (KASM). Here we will present only data for those inputs (import and export prices) projected exogenously (Table E.6). Table E.7 lists those variables which are input from other components of KASM. These include population (farm, nonfarm and total), per capita income (farm and nonfarm), the foreign exchange rate, and total supply of domestically produced agricultural commodities for human consumption.

TABLE E.1

Demand-Price-Trade Model
Initial Conditions

	STKO[1] (MT)	EPAVGO[2] (W/MT)	ELISVR[3]	ELISAV[3]	FNR[4]
1. Rice	88,000	78,500	.10	.20	.83
2. Barley	103,300	41,900	-.19	-.25	2.33
3. Wheat	286,000	28,100	.90	.50	.70
4. Other grain	0	42,500	-.10	-.30	1.70
5. Fruit	0	54,300	.75	1.30	.28
6. Pulses	0	60,600	.30	.45	.61
7. Vegetables	0	31,000	.30	.40	.76
8. Potatoes	0	15,255	-.50	-.60	2.83
9. Tobacco	0	207,700	1.20	1.20	1.00
12. Ind. crops	0	154,600	.30	.80	.32
13. Beef	0	653,024	.94	1.40	.11
14. Milk	0	58,000	3.00	3.20	.01
15. Pork	0	165,000	.59	.55	.38
16. Chicken	0	351,763	.80	1.00	.20
17. Eggs	0	200,300	.40	.40	.06
18. Fish	0	68,000	2.00	4.00	.33
19. Residual[5]	0	65,000	0	0	.49
20. Nonfood	0	-	.69	1.03	-

Notes and Sources:

(1) Stock level, defined as carry-in as of November 1, 1969 (beginning of Rice Year 1970), including stocks held in farm and urban households, in customs clearance and by the government; it does not include stocks held in private market channels. The model currently considers stocks only for rice, barley and wheat.

Source: MAF Food Bureau, Grain Statistics.

(2) Exponential average of recent producer prices (in 1970 constant won/MT), initialized as 1969 producer prices inflated to 1970 prices by the all-cities consumer price index for 1969 of .862 (1970 = 1.000).

Source: Crops, milk and fish--value of production divided by quantity produced, MAF, Yearbook of Agriculture and Forestry Statistics; other livestock products--surveyed prices received by farmers, NACF, Monthly Review.

Table E.1 (continued)

(3) Initial values of farm (ELISVR) and nonfarm (ELISAV) income elasticities.

Source: Based on cross-section analyses by Alan R. Thodey et al. ("Demand Relationships for Food in Korea, 1965-1974," KASS Special Report 12, 1976) and then adjusted in tuning the model to track the 1970-74 period.

(4) Farm per capita consumption as a proportion of nonfarm per capita consumption.

Source: Alan R. Thodey, "Food and Nutrition in Korea, 1965-1974," KASS Special Report 11, December 1975.

(5) The residual food commodity includes edible offal from doemstic meat production; and sugar, tea/coffee/cocoa, animal fat and vegetable oil, which are largely imported.

TABLE E.2

Demand-Price-Trade Model
Empirical Time Series[1]

Variable	1970	1971	1972	1973	1974	1975	1976	1977	1978	1979
VCPI[2]	1.000	1.135	1.268	1.308	1.626	2.037				
VFPIX[3]	1.000	1.011	1.012	.981	.857	.856				
VCP[4]	1.000	.967	.963	.983	.972	.918				
VWOND[5]	304.5	316.7	373.3	398.9	399.0	484.0				
VEINCM[6]	65,584	68,468	67,956	72,365	70,832	74,275				
VEFRMY[7]	35,096	36,944	42,770	45,090	47,320	54,317				
VPCUFD[8]	.436	.443	.440	.442	.461	.465				
VFI[9]	42,075	52,801	58,247	62,852	71,699	74,355				
VUI[10]	80,551	79,948	85,136	83,881	87,382					
Commodities										
1. Rice										
VPRICE[11]	74,689	82,289	95,639	97,111	112,914	113,989				
VCPU[12]	79,125	87,775	102,318	99,388	102,168	114,629				
VPSUP[13]	.1258	.1354	.1206	.1166	.1241	.1155				
VCSTK[14]	237,000	69,000	219,000	98,000	-223,000	504,000				
VDFC[15]	541,000	907,000	584,000	437,000	206,000	481,000				
VPWI[16]	190	131	131	191	263	198				
VPWX[17]	146	101	101	147	202	152				
2. Barley										
VPRICE	40,610	49,636	56,800	56,758	57,622	64,950				
VCPU	48,758	55,622	59,174	59,963	51,773	55,894				
VPSUP	.0383	.0404	.0433	.0451	.0451	.0418				
VCSTK	94,000	-134,000	137,000	53,000	-84,000	367,000				
VDFC	0	0	254,000	350,000	299,000	354,000				
VPWI	58	56	53	83	105	97				
VPWX	45	43	41	64	81	75				

Table E.2--continued

Variable	1970	1971	1972	1973	1974	1975	1976	1977	1978	1979
3. Wheat										
VPRICE	23,939	25,899	29,912	31,669	38,992	40,027				
VCPU	30,415	29,202	29,728	33,341	41,014	41,924				
VPSUP	.0398	.0509	.0565	.0567	.0401	.0466				
VCSTK	52,000	-100,000	-22,000	-151,000	174,000	31,000				
VDFC	1,254,000	1,532,000	1,881,000	1,835,000	1,592,000	1,703,000				
WPWI	54	63	55	108	138	95				
VPWX	45	53	46	90	115	79				
4. O. Grain										
VPRICE	33,375	40,263	41,883	45,730	49,295	48,884				
VCPU	60,330	66,554	86,461	90,309	95,720	88,403				
VPSUP	.0040	.0035	.0036	.0047	.0067	.0038				
VDFC	279,000	378,000	460,000	575,000	569,000	554,293				
VPWI	72	65	48	57	93	79				
VPWX	60	54	40	48	78	66				
5. Fruit										
VPRICE	54,052	60,739	57,118	63,875	68,605	75,582				
VCPU	133,934	147,189	128,161	147,734	134,646	139,391				
VPSUP	.0118	.0112	.0131	.0115	.0152	.0164				
VDFC	1,000	5,000	1,000	-1,000	4,000	-932				
VPWI	294	268	519	359	202	238				
VPWX	245	223	433	299	168	198				
6. Pulses										
VPRICE	82,634	74,159	79,079	89,060	90,041	88,964				
VCPU	98,676	96,621	101,984	107,310	109,726	112,145				
VPSUP	.0086	.0079	.0072	.0075	.0079	.0096				
VDFC	36,000	61,000	31,000	73,000	66,000	56,450				
VPWI	117	128	109	174	176	129				
VPWX	98	107	91	145	147	108				

Table E.2--continued

Variable	1970	1971	1972	1973	1974	1975	1976	1977	1978	1979
7. Vegetables										
VPRICE	44,115	44,722	35,204	36,287	37,474	44,635				
VCPU	59,777	61,616	45,662	45,218	44,050	60,421				
VPSUP	.0663	.0749	.0686	.0644	.0720	.0693				
VDFC	5,000	-4,000	-8,000	-22,000	-27,000	-27,888				
VPWI	1,601	1,106	1,367	1,316	775	680				
VPWX	1,334	992	1,139	1,097	646	567				
8. Potatoes										
VPRICE	15,079	18,134	19,333	23,860	29,277	25,041				
VCUP	39,667	36,652	44,111	40,571	44,568	45,296				
VPSUP	.0547	.0560	.0519	.0421	.0368	.0578				
VDFC	4,000	1,000	0	6,000	63,000	68,681				
VPWI	98	121	184	54	66	50				
VPWX	82	101	153	45	55	42				
9. Tobacco										
VPRICE	205,001	242,291	344,639	360,856	353,010	285,715				
VCPU	1,201,345	1,058,454	947,433	821,103	650,913	643,430				
VPSUP	.0009	.0010	.0014	.0023	.0015	.0012				
VDFC	-27,400	-20,900	-9,300	-27,600	-54,000	-61,717				
VPWI	868	1,049	946	913	839	884				
VPWX	723	874	788	761	699	737				
11. Silk										
VPRICE	449,099	504,491	568,244	943,110	849,692	678,252				
VCPU	-	-	-	-	-	-				
VPSUP	0	0	0	0	0	0				
VDFC	-2,126	-2,319	-3,244	-3,743	-2,528	-703				
VPWI	20,688	18,137	15,871	19,408	19,402	11,556				
VPWX	17,240	15,114	13,226	16,173	16,168	9,630				

Table E.2--continued

Variable	1970	1971	1972	1973	1974	1975	1976	1977	1978	1979
12. Ind. Crops										
VPRICE	146,898	148,522	176,244	349,335	204,025	207,580				
VCPU	356,667	455,213	416,666	415,392	397,704	555,846				
VPSUP	.0013	.0013	.0016	.0016	.0014	.0022				
VDFC	-200	2,200	-1,900	14,500	8,700	-235				
VPWI	883	473	939	312	384	935				
VPWX	736	394	783	260	320	779				
VPICF[18]	.576	.688	.634	.631	.728	.539				
13. Beef										
VPRICE	659,575	715,122	862,390	911,580	793,192	678,218				
VCPU	788,333	829,662	862,250	875,382	825,133	783,833				
VPSUP	.0012	.0012	.0012	.0013	.0015	.0020				
VDFC	600	600	200	600	0	-3				
VPWI	1,128	1,126	1,278	462	2,503	1,846				
VPWX	940	938	1,065	385	2,086	1,538				
14. Milk										
VPRICE	54,987	52,869	50,473	57,330	52,375	54,001				
VCPU	109,301	104,590	112,912	110,714	122,605	147,275				
VPSUP	.0031	.0024	.0040	.0032	.0039	.0047				
VDFC	54,400	18,900	60,400	8,700	14,600	10,441				
VPWI	1,035	581	416	510	74	601				
VPWX	863	484	347	425	62	501				
15. Pork										
VPRICE	261,115	281,650	228,525	313,720	291,323	342,642				
VCUP	396,567	455,213	412,723	447,208	438,704	535,101				
VPSUP	.0025	.0024	.0025	.0025	.0026	.0027				
VDFC	-100	-100	-4,200	-1,900	-3,700	-8,053				
VPWI	943	1,312	1,280	1,888	1,636	1,514				
VPWX	786	1,093	1,067	1,573	1,363	1,262				

Table E.2--continued

Variable	1970	1971	1972	1973	1974	1975	1976	1977	1978	1979
16. Chicken										
VPRICE	366,070	354,943	298,557	338,853	391,851	431,132				
VCPU	263,000	266,960	259,464	292,049	325,953	309,769				
VPSUP	.0014	.0015	.0016	.0015	.0015	.0015				
VDFC	0	0	0	0	0	0				
VPWI	-	-	-	-	-	-				
VPWX	-	-	-	-	-	-				
17. Eggs										
VPRICE	216,364	189,027	170,634	207,117	230,347	240,933				
VCUP	286,000	236,123	217,666	246,177	265,683	265,096				
VPSUP	.0041	.0041	.0044	.0039	.0043	.0044				
VDFC	-1,500	-600	-1,300	-1,200	0	0				
VPWI	-	-	-	-	-	-				
VPWX	-	-	-	-	-	-				
18. Fish										
VPRICE	70,190	72,256	63,547	84,898	62,898	72,654				
VCUP	257,100	271,366	244,326	270,466	259,156	257,857				
VPSUP	.0220	.0249	.0300	.0360	.0423	.0420				
VDFC	-181,000	-194,000	-248,000	-355,000	-407,000	-546,439				
VPWI	550	576	544	625	472	430				
VPWX	458	480	453	521	393	358				
19. Residual[19]										
VPRICE	68,413	109,570	57,012	50,425	68,472	70,000				
VCUP	92,357	147,920	76,966	68,075	92,437	95,000				
VPSUP	.0121	.0070	.0127	.0166	.0155	.0135				
VDFC	374,812	207,216	406,862	550,908	517,665	445,525				
VPWI	115	108	120	143	222	227				
VPWX	96	90	100	119	185	189				

Table E.2--continued

Notes and Sources:

1. These are recorded data used by DEMAND for the tracking period in the model. Each year, as new data become available, they will be added to the table and to the DATA statements in the computer program of KASM.

2. VCPI = all-cities, all-commodities consumer price index.
   Located in BLOCKDATA DKXD.
   Source: EPB, Results of Annual Price Survey.

3. VFPIX = ratio of VCPI to index of prices paid by farmers for household goods.
   Located in BLOCKDATA DEMDAT.
   Source: VCPI and NACF, Monthly Review.

4. VCP = all-cities consumer price index of all commodities except food and tobacco, deflated by VCPI.
   Located in BLOCKDATA NDXD.
   Source: VCPI and EPB, Results of Annual Price Survey.

5. VWOND = won-dollar exchange rate, won/dollar.
   Located in BLOCKDATA NDXD.
   Source: BOK, Economic Statistics Yearbook.

6. VEINCM = total nonfarm household consumption expenditure, won/person-year, deflated by VCPI.
   Located in BLOCKDATA DEMDAT.
   Source: VCPI and EPB, Annual Report on the Family Income and Expenditure Survey.

7. VEFRMY = total farm hosuehold consumption expenditure, won/person-year, deflated by VCPI.
   Located in BLOCKDATA DEMDAT.
   Source: VCPI and MAF, Report on the Results of Farm Household Economy Survey.

8. VPCUFD = food and tobacco proportion of nonfarm consumption expenditures.
   Located in BLOCKDATA DEMDAT.
   Source: EPB, Annual Report on the Family Income and Expenditure Survey.

9. VFI = farm household disposable income won/person-year, deflated by VCPI.
   Located in BLOCKDATA NDXD.
   Source: VCPI and MAF, Report on the Results of Farm Household Economy Survey.

10. VUI = nonfarm household disposable income, won/person-year, deflated by VCPI.
    Located in BLOCKDATA NDXD.
    Source: VEINCM at t + 1 divided by an assumed average propensity to consume of .85.

Table E.2 --continued

11. VPRICE = annual average producer price, won/MT, deflated by VPI.
Located in BLOCKDATA DKXD.
Source: VCPI and for

crops: value of production/quantity produced; MAF, Yearbook of Agriculture Statistics.

beef: prices received by farmers; females weighted .5 and males weighted .5; meat content of 120 kg/female and 160 kg/male; NACF, Monthly Review.

milk: same as crops.

pork: prices received by farmers; meat content of 52 kg/head; NACF, Monthly Review.

chicken: prices received by farmers; hens and roosters both weighted .5; meat content of 1.4 kg/head; NACF, Monthly Review.

eggs: prices received by farmers; 55 grams/egg; NACF, Monthly Review.

fish and seaweed: value of total catch/total catch; Office of Fisheries, Yearbook of Fisheries Statistics.

12. VCPU = annual average consumer price, won/MT, deflated by VCPI.
Located in BLOCKDATA DEMDAT.
Source: VCPI and Seoul retail prices from EPB, unpublished; commodities included are other

grains: Italian millet.

fruit: 1970-1972 weighted by 1970 quantities produced: apple (.52), pear (.14), peach (.19), grape (.08), persimmon (.07). Since 1973 weighted by 1973 quantities produced: apple (.53), pear (.10), peach (.16), grape (.10), persimmon (.06), orange (.05).

pulses: weighted by 1970 quantities consumed: soybeans (.85), red beans (.08), green beans (.07).

vegetables: weighted by 1973 quantities produced: radish (.31), Chinese cabbage (.31), cabbage (.02), Welsh onion (.03), onion (.04), spinach (.01), cucumber (.04), pumpkin (.05), tomato (.02), red pepper (.03), garlic (.04), sweet melon (.04), watermelon (.06).

potatoes: weighted by 1970 quantities consumed: white potatoes (.25), sweet potatoes (.75).

tobacco: using 1970 Seoul price and published all-cities price index for cigarettes.

silk: no price is given; it is not a food commodity.

industrial crops: sesame.

milk: fresh milk.

fish and seaweed: using 1970 Seoul price and published all-cities price indices for fresh fish, dried and pickled fish, and seaweed.

13. VPSUP = national average per capita consumption, MT/person-year.
Located in BLOCKDATA DEMDAT.
Source: derived as a residual from food balance sheet analyses considering domestic production, imports and exports, farm and marketing losses, stock changes, seed and feed use, and industrial consumption.

14. VCSTK = stock change for rice, barley and wheat, MT/year.
Located in BLOCKDATA DEMDAT.
Source: carry-out minus carry-in on a rice-year basis (November 1 to October 31) considering stocks held on farms, in urban households, by the government and at ports awaiting customs clearance, but not including stocks held in private market channels; MAF Food Bureau, Grain Statistics.

15. VDFC = net imports (imports minus exports), MT/year.
Located in BLOCKDATA DKXD.
Source: for rice, barley and wheat, imports based on port arrivals; MAF, Food Bureau, Grain Statistics.
for fish, Office of Fisheries, Yearbook of Fisheries Statistics.
for other commodities, imports based on customs clearance; Bureau of Customs Administration, Statistical Yearbook of Foreign Trade.

16. VPWI = annual average import price for commodities primarily imported, $/MT, deflated by VCPI. For other commodities, a 20% spread is assumed between import and export prices (30% for rice and barley).
Located in BLOCKDATA DEMDAT.
Source: VCPI and value of imports/quantity imported, Bureau of Customs Administration, Statistical Yearbook of Foreign Trade.

17. VPWX = annual average export price for commodities primarily exported, $/MT, deflated by VCPI. For other commodities, a 20% spread is assumed between import and export prices.
Located in BLOCKDATA DEMDAT.
Source: VCPI and value of exports/quantity exported; Bureau of Customs Administration. Statistical Yearbook of Foreign Trade; and Office of Fisheries, Yearbook of Fisheries Statistics.

18. VPICF = edible proportion of industrial crops.
Located in BLOCKDATA DEMDAT.
Source: MAF, Yearbook of Agriculture Statistics.

19. The residual food commodity includes edible offal from domestic meat production; and sugar, tea/coffee/cocoa, animal fat and begetable oil, which are largely imported.

TABLE E.3

Demand-Price-Trade Model
System Parameters

	ELASP[1]	ELASPR[2]	PCCONT[3] (PCUT1)	PCRT[4] (PCRT1)	MM[5]	CALPU[6]	PROTPU[7]	PILOSS[8]	PFU[9]	W[10]
1. Rice	-.3	-.4	.150 (.100)	.130 (.110)	.02	3460	.067	.006	.986	169.7
2. Barley	-.2	-.2	.030	.050	.06	3380	.094	.006	.921	15.2
3. Wheat	-.7	-.4	.055	.045	.10	2730	.066	.006	.028	23.5
4. Other grain	-.6	-.6	.002	.004	.96	3620	.092	.006	1.000	.3
5. Fruit	-.85	-.35	.040	.030	1.28	740	.004	.100	.927	15.5
6. Pulses	-.75	-.4	.018	.012	.24	3820	.362	.006	.290	13.5
7. Vegetables	-.1	-.1	.110	.100	.29	330	.015	.150	.962	82.7
8. Potatoes	-.4	-.7	.020	.050	1.03	950	.016	0	.924	6.8
9. Tobacco	-.5	-.2	.0025	.0025	2.36	0	0	0	0	54.6
12. Ind. crops	-1.1	-.5	.004	.003	1.20	5260	.098	.005	0	8.5
13. Beef	-1.4	-1.8	.005	.002	.07	2660	.180	.020	0	25.5
14. Milk	-1.5	-1.5	.035	.007	1.09	590	.029	.006	.227	3.9
15. Pork	-1.0	-.5	.010	.005	.57	4820	.112	.020	0	12.0
16. Chicken	-1.2	-.8	.005	.0025	-.19	700	.126	.020	0	6.5
17. Eggs	-.3	-.4	.015	.005	.24	1420	.113	.020	1.000	10.9
18. Fish	-.2	-.3	.070	.040	2.71	480	.073	.050	.702	49.6
19. Residual	0	0	.303	.015	.35	2190	.088	.030	.424	7.2
20. Nonfood	-.4	-	-	-	-	-	-	-	-	494.1

Table E.3--continued

Notes and Sources:

1. Nonfarm direct-price elasticities.
   Source: Time-series regressions produced few usable results (Alan R. Thodey et al., "Demand Relationships for Food in Korea, 1965-1974," KASS Special Report 12, 1976). Mostly based on discussions with knowledgeable people in MAF, NAERI and universities, and adjusted in tuning the model to track consumptions over the 1970-1974 period.

2. Farm direct-price elasticities.
   Source: See note 1 above.

3. Nonfarm consumption targets (MT/person-year). For rice, the target changes, beginning in 1980, from PCCONT to PCUT1 over a 20-year period.
   Source: Alan R. Thodey, "Food and Nutrition in Korea, 1965-1974," KASS Special Report 11, December 1975.

4. Farm consumption targets (MT/person-year). For rice, see note 3 above.
   Source: Alan R. Thodey, op. cit.

5. Marketing margins (proportion). Five-year averages of 1970-1974 values (the ratio of consumer to producer prices, Table E.2) are used as constants in the model. Marketing margins must be interpreted with care; they are a poor representation of the pricing behavior of the complex dynamic and speculative marketing system. Some of the factors causing exceptionally large, small or even negative margins in any year (see Table E.2) are:

   a) Time lags
   Rice is an example of a crop produced in one year but largely consumed in the next. A better margin to consider might be consumer price in time t relative to producer price in t-1.

   b) Government intervention
   In the case of barley, small or negative margins are the result of the government's dual price policy, where the government sells barley at prices lower than the purchase price plus storage and handling costs. This is also true to a lesser extent for rice.

c) Commodity definitions

Perhaps the biggest problem interpreting margins is due to commodity definitions and weights which differ for producers and consumers. For crops, all commodities produced in a given crop category (e.g., fruits, vegetables) are included in value of production to get producer prices. Only commodities consumed, however, are included in the average price of a category. Major differences arise, then, in fruits where imported fruits are included in consumption, and in industrial crops where only a fraction of industrial crop production is edible.

d) Location

Consumer prices are based on retail prices for Seoul, while producer prices are national averages.

e) Chicken

It is not clear why there appears to be an inconsistency between producer and consumer prices for chicken meat. It could be a problem of converting national average of farm prices for live hens and roosters as surveyed by NACF to Seoul retail prices of chicken meat surveyed by EPB.

6. Calorie content of food (Kcal/kg).
   Source: Alan R. Thodey, op. cit.

7. Protein content of food (proportion).
   Source: Alan R. Thodey, op. cit.

8. Losses on imports (proportion).
   Source: Alan R. Thodey, op. cit.

9. Proportion of food consumed unprocessed.
   Source: Figures shown are for urban consumers, derived from EPB, Annual Report on the Family Income and Expenditure Survey. The same values are assumed for farm consumers for lack of further information.

10. Price index weights (total = 1000.0).
    Source: EPB, Results of Annual Price Survey.

TABLE E.4

Demand-Price-Trade Model
Substitution Proportions

A. Grains

	Rice	Barley	Wheat	Other Grain	Potatoes
IGRAIN[1]	1	2	3	4	8
PSUBTG[2]					
Farm	.95	.95	.70	.90	.50
Nonfarm	.95	.95	.70	.90	.33
PSUBCG[3]					
Farm					
Rice	-	.55	.55	.55	.15
Barley	.65	-	.10	.10	.35
Wheat	.25	.35	-	.30	.50
Other grain	.05	.05	.35	-	0
Potatoes	.05	.05	0	.05	-
Nonfarm					
Rice	-	.55	.55	.50	.15
Barley	.65	-	.10	.15	.45
Wheat	.30	.45	-	.30	.40
Other grain	.05	0	.35	-	0
Potatoes	0	0	0	.05	-
PCONVG[4]	1.00	1.00	.80	1.00	.27

B. Meats

	Beef	Pork	Chicken	Fish
IMEAT[1]	13	15	16	18
PSUBTM[2]				
Farm	.95	.95	.50	.33
Nonfarm	.95	.95	.50	.33
PSUBCM[3]				
Farm				
Beef	-	.30	.25	.20
Pork	.50	-	.50	.60
Chicken	.25	.30	-	.20
Fish	.25	.40	.25	-
Nonfarm				
Beef	-	.30	.30	.30
Pork	.45	-	.40	.50
Chicken	.20	.20	-	.20
Fish	.35	.50	.30	-

Table E.4--continued

1. Commodity subscript indicator.

2. Total substitution proportions, i.e., the proportion of the change in consumption of a commodity, due to a price change in that commodity alone, which is made up with substitute commodities.

   Source: Discussions with knowledgeable people in MAF and NAERI; adjusted in tuning the model to track consumption over the 1970-1974 period.

3. Commodity substitution proportions, i.e., allocates total substitutions to specific commodities.

   Source: See note 2 above.

4. Conversion ratios for grains. Essentially converts wheat to flour and potatoes to grain equivalents.

TABLE E.5

Demand-Price-Trade Model
Miscellaneous System Parameters

Variable	Value	Variable	Value
APCFD[1]	.80	PWHTIC[5]	.05
APCN[2]	.85	PLOSTD[6]	.107
DELP[3]	2.0	PCMIN[7]	.80
YDEL[4]	2.0		

Notes and Sources:

1. Average propensity to consume, farm.
   Located in BLOCKDATA DKXD.
   Source: An average of 1970-1975 values from MAF, <u>Report on the Results of Farm Household Economy Survey</u>.

2. Average propensity to consume, nonfarm.
   Source: Guesstimate based on indications from EPB, <u>Annual Report on the Family Income and Expenditure Survey</u>.

3. Mean lag time used for exponentially averaging producer prices (years).
   Source: Guesstimate.

4. Mean lag time used for exponentially averaging income (years).
   Source: Guesstimate.

5. Industrial consumption of wheat as a proportion of domestically-produced wheat available for both human and industrial consumption.
   Source: Alan R. Thodey, "Food and Nutrition in Korea, 1965-1974," KASS Special Report 11, December 1975.

6. Loss in tobacco in converting from tobacco as produced to tabacco as consumed (proportion).
   Source: Alan R. Thodey, <u>op. cit</u>.

7. Minimum farm nonfood consumption expenditures as a proportion of initial such expenditures.
   Source: Guesstimate.

TABLE E.6

Projections of Export and Import
Prices for Korean Commodities[1]

Commodity	1975[2]	1976[3]	1981[3]	1986+[3]
1. Rice				
Export	152.00	174.10	100.02	100.02
Import	198.00	226.33	130.02	130.02
2. Barley				
Export	75.00	44.62	48.17	48.17
Import	97.00	58.01	62.62	62.62
3. Wheat				
Export	79.00	69.94	69.07	69.07
Import	95.00	83.92	82.88	82.88
4. Other grain				
Export	66.00	74.22	55.22	55.22
Import	79.00	89.07	66.27	66.27
5. Fruit				
Export	198.00	270.42	256.90	256.90
Import	238.00	324.50	308.28	308.28
6. Pulses				
Export	108.00	117.06	106.54	115.82
Import	129.00	140.47	127.85	138.98
7. Vegetables				
Export	576.00	570.00	570.00	570.00
Import	680.00	680.00	680.00	680.00
8. Potatoes				
Export	42.00	58.33	58.33	58.33
Import	50.00	70.00	70.00	70.00
9. Tobacco				
Export	737.00	706.37	754.59	757.99
Import	884.00	847.64	905.51	909.58
11. Silk				
Export	9,630.00	10,000.00	10,000.00	10,000.00
Import	11,556.00	10,000.00	10,000.00	10,000.00
12. Ind. crops				
Export	779.00	258.63	151.57	165.10
Import	935.00	310.36	181.88	198.12

Table E.6--continued

Commodity	1975[2]	1976[3]	1981[3]	1986+[3]
13. Beef				
Export	1,538.00	2,007.56	2,344.58	2,600.09
Import	1,846.00	2,409.07	2,813.50	3,120.11
14. Milk				
Export	501.00	416.67	416.67	416.67
Import	601.00	500.00	500.00	500.00
15. Pork				
Export	1,262.00	1,312.26	1,532.56	1,699.58
Import	1,514.00	1,574.71	1,839.08	2,039.49
16. Chicken				
Export	-	-	-	-
Import	-	-	-	-
17. Eggs				
Export	-	-	-	-
Import	-	-	-	-
18. Fish & seaweed				
Export	358.00	351.90	442.67	479.98
Import	430.00	422.28	531.21	575.97
19. Residual[4]				
Export	189.00	189.00	189.00	189.00
Import	227.00	227.00	227.00	227.00

Notes and Sources:

1. All prices are in 1970 constant \$/MT. Export prices are VPWLDX and import prices are VPWLDI, set in DATA statements in BLOCKDATA DEMDAT.

2. Values for 1975 are actual prices from Table 5.2 with an exchange rate of W484/\$1.00.

3. Where possible, projections are based on percent changes (from the 1974 base) in constant dollar commodity prices projected by the IBRD, Commodities and Export Projections Division, November 1974. In all cases, a margin of 20% is assumed between export and import prices (rice and barley, 30%).

Table E.6--continued

Rice--IBRD
Barley--IBRD, wheat
Wheat--IBRD
Other grain--IBRD, corn
Fruit--IBRD, citrus
Pulses--IBRD, soybeans
Vegetables--assumed to remain essentially unchanged
Potatoes--assumed increasing trend continues to 1976, then remain unchanged
Tobacco--IBRD
Silk--assumed to remain essentially unchanged
Industrial crops (edible oils)--IBRD, peanut oil
Beef--IBRD
Milk--assumed to remain essentially at a past average
Pork--IBRD, beef
Chicken--not traded
Eggs--not traded
Fish--IBRD, fish meal
Residual--assumed to remain essentially unchanged

4. The residual food commodity includes edible offal from domestic meat production; and sugar, tea/coffee/cocoa, animal fat and vegetable oil, which are largely imported.

TABLE E.7

Inputs to DEMAND From
Other KASM Models

Variable	Definition	Where Computed[1]
FPCI	Per capita farm disposable income (W/person-year)	FRMAC or NECDYX (if RAP not linked)
UPCI	Per capita nonfarm disposable income (W/person-year)	NECON or NECDYX (if (NECON not linked)
WOND	Foreign exchange rate (W/$)	NECON or NECDYX (if NECON not linked)
CPUNF	Nonfood price index (1970 = 1.000)	NECON or NECDYX (if NECON not linked)
POPR	Regional farm population (persons)	POPMIG or POPKX (if POPMIG not linked)
TRPOP= POP(1)	National farm population (persons)	POPMIG or POPKX (if POPMIG not linked)
POP(2)	Nonfarm population (persons)	POPMIG or POPKX (if POPMIG not linked)
TPOP	Total national population (persons)	POPMIG or POPKX (if POPMIG not linked)
TDSUP	Supply of domestically produced food commodities for human consumption (MT/year)	PRDAC or ignored (if RAP not linked)
PRFRMY	Regional farm disposable income as a proportion of national farm disposable income	FRMAC or ignored (if RAP not linked)

Notes:

1. RAP = resource allocation and production model
   FRMAC = farm income accounting component of RAP
   PRDAC = production accounting component of RAP
   NECON = national economy model
   POPMIG = population and migration model

BIBLIOGRAPHIC REFERENCES

[1] Carroll, Tom W., and Sloboda, John E. Population and Migration Model of KASM3: Technical Documentation. KASS Special Report 14. Seoul: National Agricultural Economics Research Institute; and East Lansing: Department of Agricultural Economics, Michigan State University, June, 1977.

[2] Abkin, Michael H.; Ingvaldson, Gary R.; Carroll, Tom W.; de Haen, Hartwig; and Lee, Jeung-Han. Korean Agricultural Sector Model, Version KASM3: Introduction to Technical Documentation. KASS Special Report 13. Seoul: National Agricultural Economics Research Institute; and East Lansing: Department of Agricultural Economics, Michigan State University, June, 1977.

[3] Lee, Jeung-Han. Crop Technology Change Model of KASM3: Technical Documentation. KASS Special Report 15. Seoul: National Agricultural Economics Research Institute; and East Lansing: Department of Agricultural Economics, Michigan State University, June, 1977.

[4] de Haen, Hartwig. Farm Resource Allocation and Production Accounting Model of KASM3: Technical Documentation. KASS Special Report 16. Seoul: National Agricultural Economics Research Institute; and East Lansing: Department of Agricultural Economics, Michigan State University, June, 1977.

[5] Abkin, Michael H. National Economy Model of KASM3: Technical Documentation. KASS Special Report 18. Seoul: National Agricultural Economics Research Institute; and East Lansing: Department of Agricultural Economics, Michigan State University, June, 1977.

[6] Thodey, Alan R. Food and Nutrition in Korea, 1965-74. KASS Special Report 11. Seoul: National Agricultural Economics Research Institute; and East Lansing: Department of Agricultural Economics, Michigan State University, n.d. (circa 1976).

[7] Thodey, Alan R.; Kim, Chul Ho; and Han, Won Hee. Demand Relationships for Food in Korea, 1965-1974. KASS Special Report 12. Seoul: National Agricultural Economics Research Institute; and East Lansing: Department of Agricultural Economics, Michigan State University, June, 1977.

[8] Rossmiller, George E., ed. A Systems Approach to Agricultural Sector Development Decision-Making: Building and Institutionalizing an Investigative Capacity. State-of-the-Art Series, International Institute for Applied Systems Analysis, Laxenburg, Austria (forthcoming).

[9] Teigen, Lloyd D. "The Annual Price Determination Mechanism." KASS Working Paper 74-2. Seoul: National Agricultural Economics Research Institute; and East Lansing: Department of Agricultural Economics, Michigan State University, July, 1974.