

Mini-USAGE: reducing barriers to entry in dynamic CGE modelling

by

Peter B. Dixon and Maureen T. Rimmer*

Centre of Policy Studies, Monash University

Paper prepared for the 8th Annual Conference on Global Economic Analysis

Lübeck, Germany

June 9 – 11, 2005

Revised May 23, 2005

1. Introduction

USAGE (U.S. Applied General Equilibrium) is a dynamic computable general equilibrium model of the United States. It is based on the MONASH model of Australia which has been applied widely in forecasting, policy analysis, estimation of technological trends and analysis of historical events. USAGE has many special features and is normally run with about 500 industries.

The detail included in USAGE and earlier MONASH-style models is important for practical, policy-oriented analyses. However, the detail means that comprehensive documentation is necessarily voluminous.¹ While this documentation is essential for keeping track of theoretical and empirical details, it is not ideal for disseminating the principal ideas in the models or the techniques involved in building and applying them.

This paper sets out a miniature version of the USAGE model, Mini-USAGE. Our primary aim in creating Mini-USAGE is to reduce barriers to entry into the world of MONASH-style dynamic, general equilibrium modelling. By providing Mini-USAGE, we hope to give modellers a tool for understanding the theory, construction and computing of

* We thank Laurent Cretegny who encouraged us to build Mini-USAGE. He provided a starting point for our work by implementing in GEMPACK an earlier miniature set out in Dixon and Parmenter (1996).

¹ An annotated version of the complete TABLO code for MONASH is available on the CoPS website at <http://www.monash.edu.au/policy/monbook2.htm>. See also Dixon and Rimmer (2002). A folder of papers documenting USAGE is available from the authors.

MONASH-style models and for interpreting results. For these purposes, Mini-USAGE is unencumbered by detail. We hope that for some modellers, Mini-USAGE will be a useful skeleton onto which they can graft their own detail. A secondary aim of our work on Mini-USAGE is to provide a vehicle for model development. We have found that it is efficient to test and refine new modelling features in Mini-USAGE before embedding them in a full-scale model.

While stripped of detail, Mini-USAGE retains the main theoretical features of USAGE. These include: physical capital accumulation and rate-of-return-sensitive investment; foreign debt accumulation and the balance of payments; public debt accumulation and the public sector deficit; and dynamic adjustment of wage rates in response to gaps between the demand for and supply of labour.

In forming Mini-USAGE, we aggregated the database underlying USAGE from 500 to 5 industries. This allows the miniature database to be set out on a couple of pages and Mini-USAGE simulations to be run in a few seconds. With the miniature data being an aggregated version of the full-dimension data, we find at the macro level that results from Mini-USAGE are a good guide to those obtained from USAGE.

As with full-scale MONASH-style models, Mini-USAGE can be run with 4 basic closures. With these closures, MONASH-style models produce: estimates of changes in technologies and consumer preferences (historical closure); explanations of historical developments such as rapid growth international trade (decomposition closure); forecasts for industries, regions, occupations and households (forecast closure); and projections of the deviations from forecast paths that would be caused by the implementation of proposed policies and by other shocks to the economic environment (policy closure).

The paper is organized as follows. In section 2 we set out the Mini-USAGE database and the computational method. Section 3 describes aspects of the Mini-USAGE theory. Because the *intra*-period equations in Mini-USAGE are quite familiar from earlier models, we concentrate only on *inter*-period or dynamic equations. In section 4 we provide an example of the development role of Mini-USAGE. We describe a problem with Divisia indexes. These indexes are used in MONASH-style models (and are favoured by statistical agencies) for a wide variety of macro variables. However, as we demonstrate, they can lead to distorted results, especially in dynamic policy analysis. We use Mini-USAGE to experiment with alternative indexes. Concluding remarks are in section 5. An annotated version of the TABLO code for Mini-USAGE is presented in the Appendix. This code can be

downloaded from <http://www.monash.edu.au/policy/models.htm> together with data, forecast and policy closures, shock files and instructions for running the model.²

The available version of Mini-USAGE is set up only for year-on-year forecasting and policy analysis. Forecasts can be made as a single sequence of year-on-year solutions (a run of the model). Policy analysis requires two runs: a forecast run and an alternative run that includes the policy shock. Differences between the policy and forecast runs show the effects of the policy. Future downloadable versions of Mini-USAGE will include facilities for running historical and decomposition simulations.

2. Database and computational method

We represent Mini-USAGE and other MONASH-style models as

$$F(V(t)) = 0 \quad , \quad (2.1)$$

where $V(t)$ is a vector of length n referring to prices, quantities and other variables for year t and F is a vector function of length m , $m < n$. The computational approach that we have adopted for MONASH-style models depends on being able to solve the model one period (usually one year) at a time. For year t we specify values for $n-m$ exogenous variables and solve (2.1) for the remaining m endogenous variables. By obtaining a sequence of linked solutions for years τ , $\tau+1$, $\tau+2$, ..., we generate time paths for variables. Links between the annual solutions are provided by lags. For example, we assume that capital stocks at the beginning of year $\tau+1$ (variables in the solution for year $\tau+1$) equal capital stocks at the end of year τ (variables in the solution for year τ).³

Within any sequence of solutions, we obtain the solution for year t [i.e., we solve (2.1)] by the Johansen/Euler⁴ method. This method requires an initial solution, $\bar{V}(t)$, satisfying (2.1). Starting from this initial solution, we obtain the required solution for year t by calculating the effects on endogenous variables of moving the exogenous variables away from their values in the initial solution to their values in the required solution. In most MONASH-style computations, the initial solution for year t is the final solution for year $t-1$

² Mini-USAGE can be run by licenced GEMPACK users of RunDynam or RunMONASH.

³ We may also wish to impose forward links. For example, we may wish to assume that profit expectations held in year τ (variables in the τ solution) depend on profit outcomes in year $\tau+1$ (variables in the $\tau+1$ solution). Forward links pose difficulties for our one-year-at-a-time computational method. These can be tackled by an iterative method, see Dixon and Rimmer (2002, section 21) and Dixon *et al.* (2003).

⁴ So named in recognition of the contributions of Johansen (1960) who applied a version of this method to solve his CGE model of Norway, and Euler, the eighteenth century mathematician who set out the theory of the method as an approach to numerical integration. Early examples of applications of the Johansen/Euler method include Taylor and Black (1974), Staelin (1976), Keller (1980) and Dixon *et al.* (1982). Variants of the Johansen/Euler method underlie the GEMPACK programs [Pearson (1988) and Harrison and Pearson (1996, 2002 and 2005)].

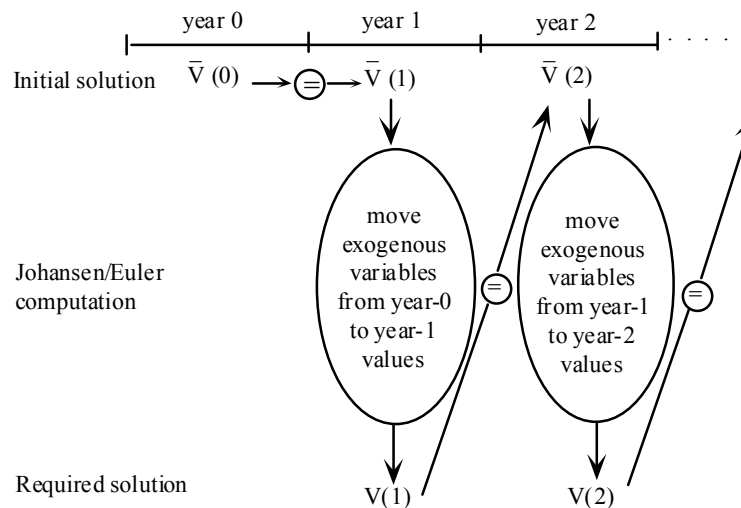
(see Figure 2.1). Thus the computation for year t consists of working out the effects of moving the exogenous variables from their values in year $t-1$ to their values in year t . The initial solution for year 1 comes from input-output data and data on the government accounts, the balance of payments and capital stocks. These data provide a numerical picture (solution) for year 0, which is the initial solution for year 1.

The bulk of the data for Mini-USAGE are set out in Tables 2.1 to 2.4.⁵ These data refer to the U.S. economy in 1998 (year 0). Table 2.1 shows the input-output data, consisting of an absorption matrix and a joint production or make matrix. Tables 2.2, 2.3 and 2.4 contain the government accounts, the current account and the values of capital stocks.

The five industries and commodities in Mini-USAGE are:

1. low-protection goods
2. high-protection goods
3. construction
4. services
5. government

Figure 2.1. A sequence of solutions using the final solution for year $t-1$ as the initial solution for year t



⁵ As can be seen from the Read statements in the TABLO code presented in the Appendix, Mini-USAGE requires some other data items such as starting values for various prices and wage rates. These can usually be set at 1. As well as introducing data items, Read statements are also used to set parameter values, e.g. Armington elasticities.

Table 2.1. Mini-USAGE Input-Output Database for 1998 (\$millions)

		Absorption matrix														
		Producers (industries)					Investors (industries)					H'hlds	Govt.	Exports	Duty	Sales
		1	2	3	4	5	1	2	3	4	5					
basic flows domestic	1	1384278	73479	231271	506579	123708	112300	5854	35437	293758	56540	655092	0	506982		3985278
	2	39893	67239	2393	33400	8102	36	0	0	27	41	138468	0	32674		322273
	3	34505	2577	885	172576	52352	64846	2115	576	514027	94119	0	0	80		938658
	4	477650	46491	94656	2108502	253636	11673	503	1592	72772	1901	3735309	0	326263		7130948
	5	0	0	0	0	121	0	0	0	0	0	0	1257059	0		1257180
basic flows imported	1	305628	7257	29901	89848	14305	40912	2678	12973	91254	8365	179943	0	0	-9591	773473
	2	7370	9319	1110	6578	978	3	0	0	5	5	89635	0	0	-10991	104012
	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	4	25858	648	1088	158010	11989	94	4	17	198	0	74591	0	0	-23	272474
	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
margin flows on domestic goods	1	213740	8147	62980	111367	12882	23485	1193	7126	63991	5836	397910	0	89892		998549
	2	6778	6626	1116	7164	1347	13	0	0	12	12	106738	0	4892		134698
	3	0	0	0	0	0	0	0	0	0	0	0	0	1		1
	4	9399	86	16	6466	254	0	0	0	0	0	37794	0	1836		55851
	5	0	0	0	0	0	0	0	0	0	0	0	0	0		0
margin flows on imported goods	1	67207	1239	11092	28873	2841	10019	529	3014	26853	2659	131239	0	0		285565
	2	1481	1284	746	2882	169	1	0	0	3	1	98301	0	0		104868
	3	0	0	0	0	0	0	0	0	0	0	0	0	0		0
	4	3261	50	7	5298	39	0	0	0	0	0	2558	0	0		11213
	5	0	0	0	0	0	0	0	0	0	0	0	0	0		0
taxes on domestic flows	1	8632	348	5788	21806	398	1560	89	864	5415	161	61363	0	2411		108835
	2	144	65	130	1282	34	0	0	0	0	1	19939	0	668		22263
	3	0	0	0	0	0	0	0	0	0	0	0	0	0		0
	4	6566	653	753	32164	3270	30	1	2	436	0	34204	0	592		78671
	5	0	0	0	0	0	0	0	0	0	0	0	0	0		0
taxes on imported flows	1	1969	63	935	4280	55	580	36	349	1936	58	17088	0	0		27349
	2	44	13	82	206	2	0	0	0	0	0	8798	0	0		9145
	3	0	0	0	0	0	0	0	0	0	0	0	0	0		0
	4	19	1	0	333	3	0	0	0	0	0	289	0	0		645
	5	0	0	0	0	0	0	0	0	0	0	0	0	0		0
labor		1041640	66025	438252	3581575	783888										5911380
capital		444643	26160	55457	1587960	150427										2264647
Total		4080705	317770	938658	8467149	1420800	265552	13002	61950	1070687	169699	5789259	1257059	966291	-20605	

... Table 2.1 continues

Table 2.1 continued

		Joint production matrix					
		industries					Total
		1	2	3	4	5	
commodities	1	3943067	8810	0	27831	5570	3985278
	2	5281	306932	0	9841	219	322273
	3	0	0	938658	0	0	938658
	4	132357	2028	0	8429477	157831	8721693
	5	0	0	0	0	1257180	1257180
Total		4080705	317770	938658	8467149	1420800	15225083

Table 2.2. Government outlays and revenue: 1998 (\$million)

Public consumption	1257059	
Public investment	225743	
Transfers	1214268	
made up of:		
benefits		1087928
Net interest on public-sector debt		126340
Total outlays	2697070	
Indirect taxes	267513	
made up of taxes on:		
Intermediate inputs		90038
Inputs to investment		11518
Consumption		141681
Exports		3671
Imports		20605
Direct taxes	2190275	
made up of taxes on:		
Labour		1850309
Capital		339966
Other revenue	150427	
Total revenue	2608215	
Public-sector deficit (outlays less revenue)	88855	

Table 2.3. Calculation of the current account deficit: 1998 (\$million)

Imports	1149959
– Exports	-966291
+ Servicing of U.S. foreign liabilities	35494
Current account deficit (CAD)	219162
Current account deficit as percent of GDP	2.6 %
Net foreign liabilities (start of year)	1314599

Table 2.4 . Value of capital stock, start of 1998 (\$million)

industries					
1	2	3	4	5	
2495648	147632	320379	14402637	3639893	

As can be seen from the joint production matrix, each industry specializes in the production of the commodity of the same name, but most of the industries can produce more than one commodity. Commodity 4 is used both directly and as a margin service. All commodity flows are valued at basic prices. For domestic commodities this is the price received by the producer (the price at the factory door) and for imports it is the landed-duty-paid price.

By looking at Tables 2.1 to 2.4, we can check various properties of the data including the following:

Sales of good 1 = 3985278 (absorption matrix)

Production of good 1 = 3985278 (joint production matrix)

Direct sales of good 4 plus margin sales = $7130948 + 998549 + 134698 + 1 + 55851 + 0 + 285565 + 104868 + 0 + 11213 + 0 = 8721693$ (absorption matrix)

Production of good 4 = 8721693 (joint production matrix)

Total inputs to industry 1 = 4080705 (absorption matrix)

Total output of industry 1 = 4080705 (joint production matrix)

Government consumption = 1257059 (absorption matrix)

Public consumption = 1257059 (government outlays and revenue table)

Indirect taxes (including import duty) = $20605 + 108835 + 22263 + 0 + 78671 + 0 + 27349 + 9145 + 0 + 645 + 0 = 267513$ (absorption matrix)

Indirect taxes (including import duty) = 267513 (government outlays and revenue table)

Imports (c.i.f. value) = $773473 + 104012 + 0 + 272474 + 0 = 1149959$ (absorption matrix)

Imports = 1149959 (calculation of the current account deficit table)

Exports = 966291 (absorption matrix)

Exports = 966291 (calculation of the current account deficit table)

GDP (income) = wages + returns to capital + indirect taxes = $5911380 + 2264647 + 267513 = 8443540$ (absorption matrix)

GDP (expenditure) = $C + I + G + X - M = 5789259 + (265552 + 13002 + 61950 + 1070687 + 169699) + 1257059 + 966291 - 1149959 = 8443540$ (absorption matrix)

Gross rate of return on capital in industry 1 = $100 \times 444643 / 2495648 = 17.8\%$

Gross rate of return on capital in industry 2 = $100 \times 26160 / 147632 = 17.7\%$

Gross rate of return on capital in industry 3 = $100 \times 55457 / 320379 = 17.3\%$

Gross rate of return on capital in industry 4 = $100 \times 1587960 / 14402637 = 11.0\%$

Gross rate of return on capital in industry 5 = $100 \times 150427 / 3639893 = 4.1\%$

Implied rate of interest on net foreign liabilities = $100 \times 35494 / 1314599 = 2.7\%$

3. Dynamic aspects of Mini-USAGE

Mini-USAGE contains 4 dynamic mechanisms or links between successive years:

(a) capital at the start of year t equals capital at the end of year $t-1$;

(b) net foreign liabilities at the start of year t equals net foreign liabilities at the end of year $t-1$;

(c) public sector debt at the start of year t equals public sector debt at the end of year $t-1$; and

(d) the deviation in the real wage rate away from its forecast path in year t caused by a policy shock equals the deviation in year $t-1$ plus a term reflecting the gap in year t between the employment deviation and the deviation in labour supply.

3.1. Capital

In Mini-USAGE, investment in year t for industry i is a function of the expected rate of return in industry i . This is determined as a function of the rental and asset prices of i 's capital in year t . Capital stock in industry i at the end of year t [$K1_i(t)$] equals capital stock at the start of year t depreciated [$K0_i(t)*(1-D_i)$] plus investment during year t [$I_i(t)$]:

$$K1_i(t) = K0_i(t) * (1 - D_i) + I_i(t) \quad (3.1)$$

Without loss of generality we can assume that capital units are chosen so that the price of a unit of capital in year zero is one. This means that quantities of investment and start-of-year capital stocks for year zero can be read as data items from Tables 2.1 and 2.4. As can be discovered from the parameter files for Mini-USAGE, the depreciation rate for all industries is set at 0.07. Quantities of end-of-year capital stocks are calculated in the GEMPACK code according to (3.1). For example, the quantity of end-of-year capital stock for industry 1 in year 0 (1998) is

$$\begin{aligned} K1_1(1998) &= K0_1(1998) * (1 - 0.07) + I_1(1998) \\ &= 2495648 * (1 - 0.07) + 265552 = 2586505 \end{aligned} \quad (3.2)$$

How do we ensure that the start-of-year capital stock in industry 1 in year 1 (1999) is 2586505?

Our initial solution for year 1 is the database for year 0. The start-of-year quantity of capital in industry 1 in our final solution should be 3.64 per cent higher than in our initial solution [$= 100 * (2586505 / 2495648 - 1)$]. One possibility is to treat start-of-year capital stocks as exogenous variables in the year 1 computation. With this approach we would shock the start-of-year capital stock for industry 1 by 3.64 per cent. An alternative method, which we have found convenient, is to include in the GEMPACK code an equation of the form

$$d_K0_i(t) = [K1_B_i(t) - K0_B_i(t)] * del_unity \quad (3.3)$$

or equivalently,

$$K0_i(t) * ko_i(t) = 100 * [K1_B_i(t) - K0_B_i(t)] * del_unity \quad (3.4)$$

where

$K0_B_i(t)$ and $K1_B_i(t)$ are the initial values for the start-of-year and end-of-year quantities of industry i 's capital stock in the initial solution (or database) for year t ;

$K0_i(t)$ is the start-of-year quantity of industry i 's capital stock in the current step in the computation of the final solution for year t ;
 $d_K0_i(t)$ is the change in the start-of-year quantity of industry i 's capital stock between the initial and final solutions for year t ;
 $ko_i(t)$ is the percentage change in the start-of-year quantity of industry i 's capital stock between the initial and final solutions for year t ; and
 del_unity is a variable that we move in the year t computation from an initial value of zero to a final value of 1.

3.2. Net foreign liabilities

We assume that net foreign liabilities at the end of year t [$NFL1(t)$] equals net foreign liabilities at the start of year t [$NFL0(t)$] plus the current account deficit for year t [$CADEF(t)$]:

$$NFL1(t) = NFL0(t) + CADEF(t) \quad . \quad (3.5)$$

To ensure that net foreign liabilities at the start of year t equals net foreign liabilities at the end of year $t-1$, we include in Mini-USAGE the equation:

$$d_NFL0(t) = [NFL1_B(t) - NFL0_B(t)] * del_unity \quad (3.6)$$

where

$NFL0_B(t)$ and $NFL1_B(t)$ are the initial values for the start-of-year and end-of-year net foreign liabilities in the initial solution (or database) for year t ;
 $d_NFL0(t)$ is the change in the start-of-year net foreign liabilities between the initial and final solutions for year t ; and
 del_unity is, as before, a variable that we move in the year t computation from an initial value of zero to a final value of 1.

As indicated in Table 2.3, we model the current account deficit for year t as imports minus exports plus servicing of foreign liabilities. In Mini-USAGE all foreign liabilities are assumed to be debt repayable in U.S. currency. In calculating servicing charges on the foreign debt we apply an interest rate to the start-of-year foreign debt.

3.3. Public sector debt

Public sector debt at the end of year t [$PSD1(t)$] equals public sector debt at the start of year t [$PSD0(t)$] plus the public sector deficit for year t [$GOVDEF(t)$]:

$$PSD1(t) = PSD0(t) + GOVDEF(t) \quad . \quad (3.7)$$

To ensure that public sector debt at the start of year t equals public sector debt at the end of year $t-1$, we include in Mini-USAGE the equation:

$$d_PSD0(t) = [PSD1_B(t) - PSD0_B(t)] * del_unity \quad (3.8)$$

where

$PSD0_B(t)$ and $PSD1_B(t)$ are the initial values for the start-of-year and end-of-year public sector debt in the initial solution (or database) for year t ;

$d_PSD0(t)$ is the change in the start-of-year public sector debt between the initial and final solutions for year t ; and
 del_unity is, as before, a variable that we move in the year t computation from an initial value of zero to a final value of 1.

The components in our modelling of the public sector deficit can be seen in Table 2.2. In calculating net interest on public sector debt, we apply an interest rate to the start-of-year public sector debt.

3.4. Wage determination in policy runs

In many general equilibrium analyses of the effects of changes in policy instruments and other changes in the economic environment, one of the following two assumptions is made:

- a. real wages adjust so there is no effect on employment; or
- b. real wages remain unaffected and employment adjusts.

In MONASH-style models, we can take an intermediate position between a and b. We can assume that real wages are sticky in the short run and flexible in the long run. In this case, favourable shocks generate short-run gains in aggregate employment and long-run gains in real wages.

More specifically, in MONASH-style policy simulations, we usually assume that the deviation in the real wage rate from its basecase forecast level increases at a rate which is proportional to the deviation in aggregate hours of employment from its basecase forecast level. The coefficient of proportionality is chosen so that the employment effects of a shock to the economy are largely eliminated after 5 years. This labour market assumption is consistent with conventional macro-economic modelling in which the NAIRU is either exogenous or only weakly dependent on real wage rates.

The version of this theory that we include in Mini-USAGE can be written algebraically as:

$$\left\{ \frac{W^P(t)}{W^f(t)} - 1 \right\} = \left\{ \frac{W^P(t-1)}{W^f(t-1)} - 1 \right\} + \alpha \left\{ \frac{E^P(t)}{E^f(t)} - 1 \right\} \quad (3.9)$$

where

$W^P(\tau)$ and $W^f(\tau)$ are the real before-tax wage rate in the policy and forecast runs in year τ ;

$E^P(\tau)$ and $E^f(\tau)$ are aggregate employment in the policy and forecast runs in year τ ;

and

α is a positive parameter.

In implementing the Johansen/Euler solution method, we need a change form of (3.9). As in our earlier representations of dynamic processes, we make use of base period coefficients and the del_unity variable. In our code for Mini-USAGE we represent (3.9) as:

$$\begin{aligned} \frac{W^P(t)}{W^f(t)}[w^P(t) - w^f(t)] = 100 * \left[\frac{WB^P(t)}{WB^f(t)} - \frac{WLB^P(t)}{WLB^f(t)} \right] * \text{del_unity} \\ + \alpha * \frac{E^P(t)}{E^f(t)}[e^P(t) - e^f(t)] \end{aligned} \quad (3.10)$$

In this equation

$WB^P(t)$ is the initial solution for the real before-tax wage rate in year t in the policy run, that is the final solution for this variable in the policy run for year $t-1$.⁶

$WLB^P(t)$ is the initial solution for the lagged real before-tax wage rate in year t in the policy run. This is the final solution for the lagged real before-tax wage in year $t-1$, which is the final solution for the real before-tax wage rate in the policy run for year $t-2$.

$WB^f(t)$ and $WLB^f(t)$ are the corresponding coefficients for the forecast run.

$w^P(t)$ and $w^f(t)$ are the percentage changes in the real before-tax wage rate between the initial and final solutions for year t in the policy and forecast runs.

$e^P(t)$ and $e^f(t)$ are the percentage changes in aggregate employment between the initial and final solutions for year t in the policy and forecast runs.

4. Divisia indexes: Mini-USAGE in its development role

4.1. Divisia problems

MONASH-style models contain many Divisia indexes for macro quantities and prices. Divisia indexes or approximations to them are also favoured by statistical agencies. However, they lead to problems. First they can indicate differences in the effects of alternative policies on macro aggregates for a given year where no differences exist. Second they can indicate spurious policy-induced changes in long-run growth rates.

To illustrate the first problem, we start by considering a Divisia price index for consumption. The percentage movement (p_t) in such an index between years $t-1$ and t is defined by

$$p_t = \sum_i S_{it} p_{it} \quad , \quad (4.1)$$

where

p_{it} is the percentage movement in the price of good i between years $t-1$ and t , and

⁶ For year zero, “the final solution” is the initial database.

S_{it} is the budget (or value) share of good i , averaged across years $t-1$ and t .⁷

The problem with Divisia price indexes (or other variable weight price indexes) is that their movement between years 0 and τ cannot be determined simply from information on prices and value shares in years 0 and τ . This is illustrated in Table 4.1 which shows basecase forecasts and two policy simulations. In the forecast simulation there are no movements in prices and value shares through the simulation period. In the policy simulations the prices of both commodities increase from 1 to 2. The time paths of these price increases are the same in the two policy simulation (although this is not an essential aspect of this example). In the first policy simulation there is a policy-induced taste change in year 2 favouring good 2 and leading to a change in budget shares. In the second policy simulation there is a similar policy-induced taste change but it takes place in year 1 rather than year 2. Although the two policy simulations have the same values for prices and budget shares in year 2, the Divisia price index in year 2 is 1.8750 in the first policy simulation and 2.1875 in the second. We are in danger of concluding that the first policy causes an 87.5 per cent increase in the price level after two years and that the second causes a 118.75 per cent increase. However, it is clear that both policies cause a 100 per cent increase.

To illustrate how a Divisia index can indicate spurious long-run growth effects, we consider a quantity index for capital. In the Divisia formulation, the percentage movement in aggregate capital from year $t-1$ to year t in simulation j is given by

$$k^j(t) = \sum_i S_i^j(t) * k_i^j(t) \quad (4.2)$$

where

$k_i^j(t)$ is the rate of growth from year $t-1$ to year t in industry i 's capital stock in simulation j either the forecast simulation ($j = f$) or the policy simulation ($j = p$); and

$S_i^j(t)$ is the share in simulation j of industry i in the aggregate capital stock for year t .

Equation (4.2) implies that the difference between the growth rates in year t of aggregate capital stock in policy and forecast simulations is given by:

$$k^p(t) - k^f(t) = \sum_i [S_i^p(t) - S_i^f(t)] * k_i^{ave}(t) + \sum_i [k_i^p(t) - k_i^f(t)] * S_i^{ave}(t) , \quad (4.3)$$

where the superscript *ave* denotes the average of values in the policy and forecast simulations.

⁷ In an N step Johansen/Euler computation, we can think of the average budget share for good i across years $t-1$ and t as being an N th of the sum of the i -budget shares at the N steps.

Table 4.1. Paths of Divisia Indexes in Dynamic Simulations

	Year 0	Year 1	Year 2
<i>Forecast</i>			
P1, level	1	1	1
P2, level	1	1	1
S1, share	0.5	0.5	0.5
S2, share	0.5	0.5	0.5
p1, % change	0	0	0
p2, % change	0	0	0
p, % change in index	0	0	0
P, level of index	1	1	1
<i>Policy case 1 (taste change in year 2)</i>			
P1, level	1	1	2
P2, level	1	2	2
S1, share	0.5	0.25	0.25
S2, share	0.5	0.75	0.75
p1, % change	0	100	100
p2, % change	100	0	0
p, % change in index	50	25	25
P, level of index	1	1.5	1.8750
<i>Policy case 2 (taste change in year 1)</i>			
P1, level	1	1	2
P2, level	1	2	2
S1, share	0.25	0.25	0.25
S2, share	0.75	0.75	0.75
p1, % change	0	100	100
p2, % change	100	0	0
p, % change in index	75	25	25
P, level of index	1	1.75	2.1875

We refer to the first term on the RHS of (4.3) as a share effect. Via the share effect, the growth rate of aggregate capital in year t in the policy simulation can fall short of that in the forecast simulation $[k^p(t) - k^f(t) < 0]$ if the policy reduces the capital shares $[S_i^p(t) - S_i^f(t) < 0]$ of industries with fast rates of capital growth [high $k_i^{ave}(t)$]. We refer to the second term on the RHS of (4.3) as a quantity effect. The quantity effect will be negative if the policy reduces capital growth rates $[k_i^p(t) - k_i^f(t) < 0]$.

In an application of a MONASH-style model in which the policy shock is unfavourable to industry i , we will find in early years of the simulation period that $k_i^p(t) - k_i^f(t) < 0$. Eventually the gaps between the policy and forecast growth rates close $[k_i^p(t) - k_i^f(t) \text{ becomes approximately zero for all } i]$. Thus, the quantity effect makes no further contribution to the gap between $k^p(t)$ and $k^f(t)$. However the share effect can remain

nonzero indefinitely. For example, if the policy reduces the long-run share of industry i in the value of capital [$S_i^p(t) - S_i^f(t) < 0$ for all $t > 5$, say] and industry i 's capital is relatively fast-growing in the forecast run [$k_i^{ave}(t)$ is large], then the policy can permanently reduce the growth rate of the Divisia measure of aggregate capital [$k^p(t) - k^f(t) < 0$ for all $t > 5$, say]. Thus we are in danger of concluding that the policy causes a permanent change in the long-run growth rate of the economy's capital stock, even though it has no effect on long-run capital growth in any industry.

4.2. Providing alternative indexes that avoid the Divisia problem

If we want to look at the effects of a policy in year t , then we should compare the state of the economy in year t with the policy in place with its state without the policy. The fundamental problem in using year- t , policy-induced deviations in Divisia indexes is that their values depend on the path by which the economy arrived at year t . They do not depend just on the situation *in* year t .

As a step towards providing full-scale MONASH-style models with legitimate indicators of policy-induced changes in macro variables, we have added code (see Appendix) to Mini-USAGE for calculating Laspeyres and Paasche indexes. Values of these indexes for year t depend only on prices and quantities in year t .

We define policy-induced deviations for year t in Laspeyres quantity and prices indexes [$Lq(t)$ and $Lp(t)$] as forecast-share-weighted averages of percentage deviations in component quantities and prices:

$$Lq(t) = 100 * \frac{\sum_i P_i^f(t) * X_i^p(t) - \sum_i P_i^f(t) * X_i^f(t)}{\sum_i P_i^f(t) * X_i^f(t)} \quad (4.4)$$

$$= 100 * \frac{\sum_i P_i^f(t) * X_i^f(t)}{\sum_j P_j^f(t) * X_j^f(t)} * \left[\frac{X_i^p(t) - X_i^f(t)}{X_i^f(t)} \right] \quad (4.5)$$

and

$$Lp(t) = 100 * \frac{\sum_i P_i^p(t) * X_i^f(t) - \sum_i P_i^f(t) * X_i^f(t)}{\sum_i P_i^f(t) * X_i^f(t)} \quad (4.6)$$

$$= 100 * \frac{\sum_i P_i^f(t) * X_i^f(t)}{\sum_j P_j^f(t) * X_j^f(t)} * \left[\frac{P_i^p(t) - P_i^f(t)}{P_i^f(t)} \right]. \quad (4.7)$$

where $P_i^f(t)$, $P_i^p(t)$, $X_i^f(t)$ and $X_i^p(t)$ are the prices and quantities of i in year t in the forecast and policy runs.

To define policy-induced deviations for year t in Paasche quantity and prices indexes we recall that a macro value can be expressed as the product of a Laspeyres quantity index and a Paasche price index and as the product of a Paasche quantity index and a Laspeyres price index. Thus we can compute the policy-induced deviations for year t in Paasche quantity and prices indexes [$Pq(t)$ and $Pp(t)$] via:

$$1 + \frac{V^p(t) - V^f(t)}{V^f(t)} = \left[1 + \frac{Lq(t)}{100} \right] * \left[1 + \frac{Pp(t)}{100} \right] = \left[1 + \frac{Pq(t)}{100} \right] * \left[1 + \frac{Lp(t)}{100} \right] \quad (4.8)$$

where $V^f(t)$ and $V^p(t)$ are macro values in year t in the forecast and policy runs.⁸

As can be seen from the Appendix, the code for the Laspeyres and Paasche indexes is quite cumbersome. For testing and refining it, Mini-USAGE provided a valuable service by allowing quick turnaround. In the rest of this section we briefly look at some Mini-USAGE results generated during the testing procedure.

For our test experiment, we conducted forecast and policy runs for 1999 to 2007. In the forecast run we used actual data for most exogenous variables for the years up to 2004. Beyond 2004, the forecasts for the exogenous variables were based mainly on trends. In the policy run, the shock was a 5 per cent reduction in the power of the tariff on the high-protection good (a reduction in the tariff rate from about 10 per cent to about 5 per cent) imposed in 1999.

Chart 4.1 shows percentage deviations generated in the test experiment for Divisia, Laspeyres and Paasche indexes of real GDP. The Divisia results emerge naturally in MONASH-style models solved by the Johansen/Euler method. They are obtained as percentage differences between the policy and forecast levels of Divisia real GDP indexes. These are calculated by accumulating year-to-year real GDP growth. In run j (j = policy or forecast), growth from year $t-1$ to year t in real GDP is obtained from:

$$gdpr^j(t) = S_c^j(t) * cr^j(t) + S_i^j(t) * ir^j(t) + S_g^j(t) * gr^j(t) + S_x^j(t) * xr^j(t) - S_m^j(t) * mr^j(t) \quad (4.9)$$

⁸ Paasche quantity and price deviations are defined by

$$Pq(t) = 100 * \frac{\sum_i P_i^p(t) * X_i^f(t)}{\sum_j P_j^f(t) * X_j^f(t)} * \left[\frac{X_i^p(t) - X_i^f(t)}{X_i^f(t)} \right] \quad \text{and} \quad Pp(t) = 100 * \frac{\sum_i P_i^f(t) * X_i^p(t)}{\sum_j P_j^f(t) * X_j^p(t)} * \left[\frac{P_i^p(t) - P_i^f(t)}{P_i^f(t)} \right].$$

where $cr^j(t)$, $ir^j(t)$, $gr^j(t)$, $xr^j(t)$, and $mr^j(t)$ are rates of growth in run j from year $t-1$ to year t in Divisia quantity indexes for private consumption, investment, government consumption, exports and imports; and

$S_c^j(t)$, $S_i^j(t)$, $S_g^j(t)$, $S_x^j(t)$ and $S_m^j(t)$ are GDP shares in year t of run j .

The Laspeyres and Paasche results for real GDP deviations are calculated in the new code being tested in Mini-USAGE. For the Laspeyres deviations, we use a Laspeyres combination of deviations in components of real GDP, with the deviations in the components themselves being Laspeyres deviations. Similarly, for the Paasche deviations we use a Paasche combination of deviations in components of real GDP, with the deviations in the components being Paasche deviations.

Charts 4.2 to 4.4 show deviation results for aggregate employment and aggregate capital calculated as percentage deviations in Divisia, Laspeyres and Paasche indexes of industry employment and capital. As can be seen in these charts, the tariff cut generates a short-run increase in employment and a sustained increase in capital. The explanation of these results is not relevant for our present purposes. Here we focus on the deviations in the alternative indexes of real GDP.

A good starting point for thinking about policy-induced deviations in real GDP is the aggregate production function:

$$Y = A * F(K, L) \quad (4.10)$$

where Y is real GDP, K and L are inputs of capital and labour, and A is a total-factor-productivity term. In our tariff-cut simulation, we assumed no policy-induced changes in technology (the technology variables follow the same paths in the forecast and policy runs). Nevertheless, the tariff cut can still affect A . This is because A reflects not only technology but also captures the effects on GDP of changes in the efficiency with which capital and labour are allocated between industries and expenditure is allocated between goods. On the basis of the usual consumer-surplus diagram, we would expect the total-factor productivity or efficiency effect in our tariff-cut experiment to be approximated by:

$$eff(t) = \left(\frac{T_2^p(t) + T_2^f(t)}{200} \right) * SM_2(t) * m_2(t) \quad (4.11)$$

where

$eff(t)$ is the percentage increase in real GDP in year t generated by the policy-induced reduction in the deadweight loss associated with the tariff on the high protection good (good 2);

Chart 4.1. Policy-induced percentage deviations in real GDP calculated with different indexes

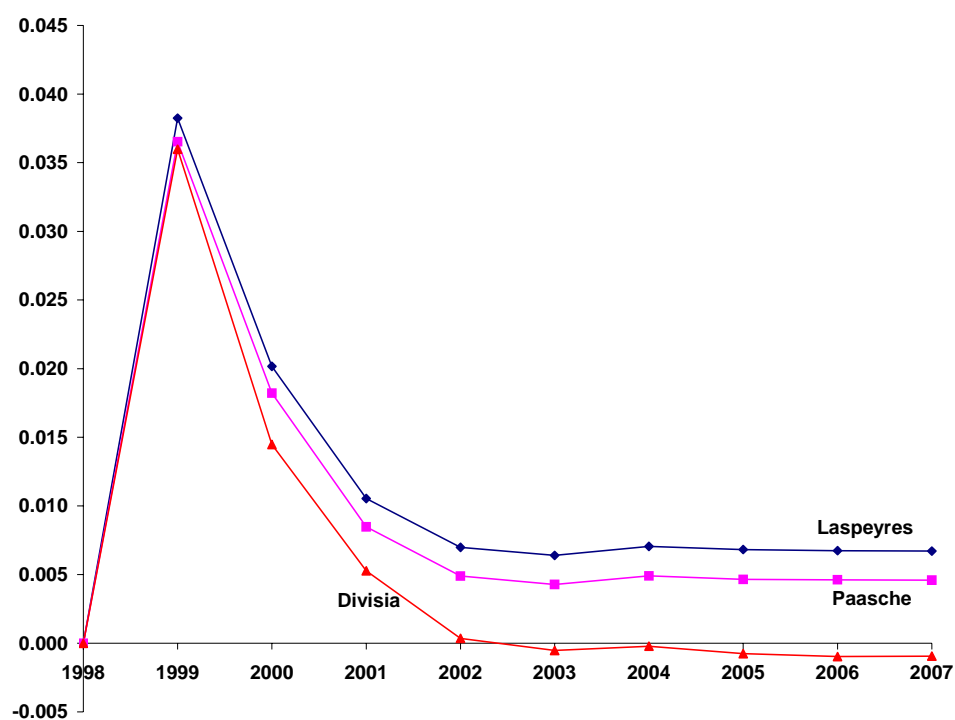
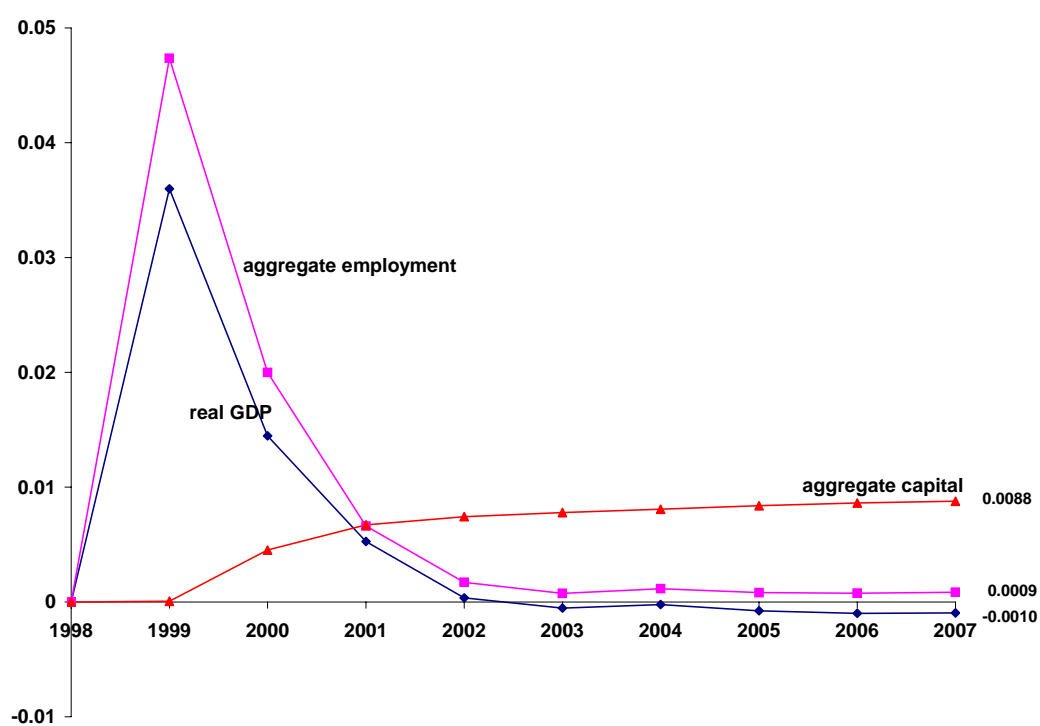
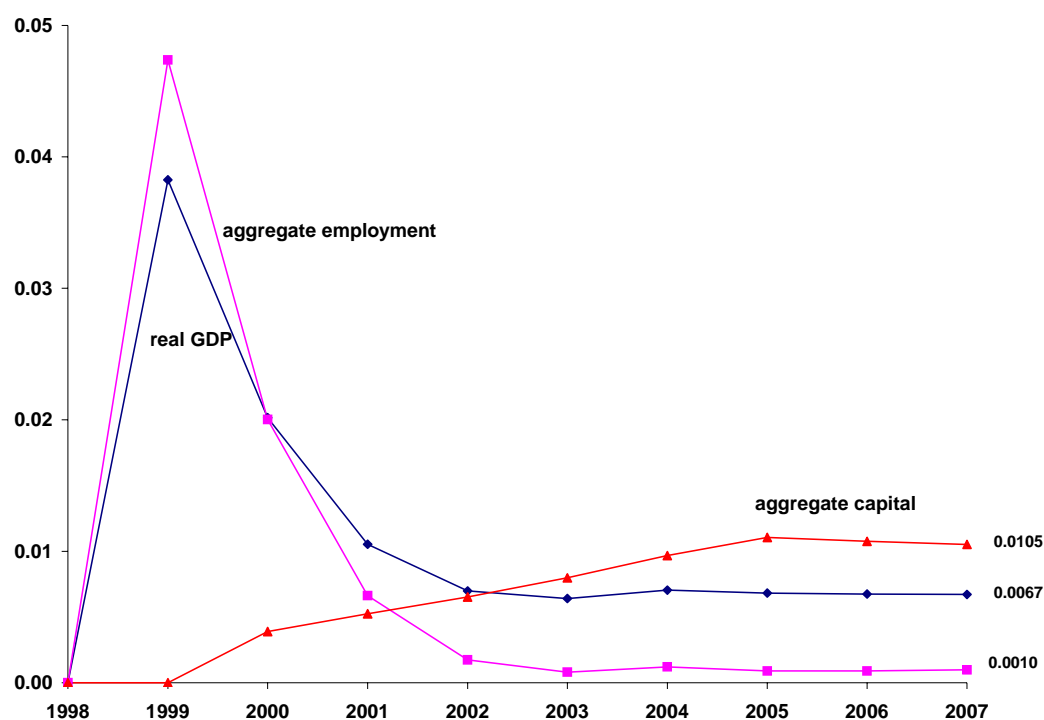


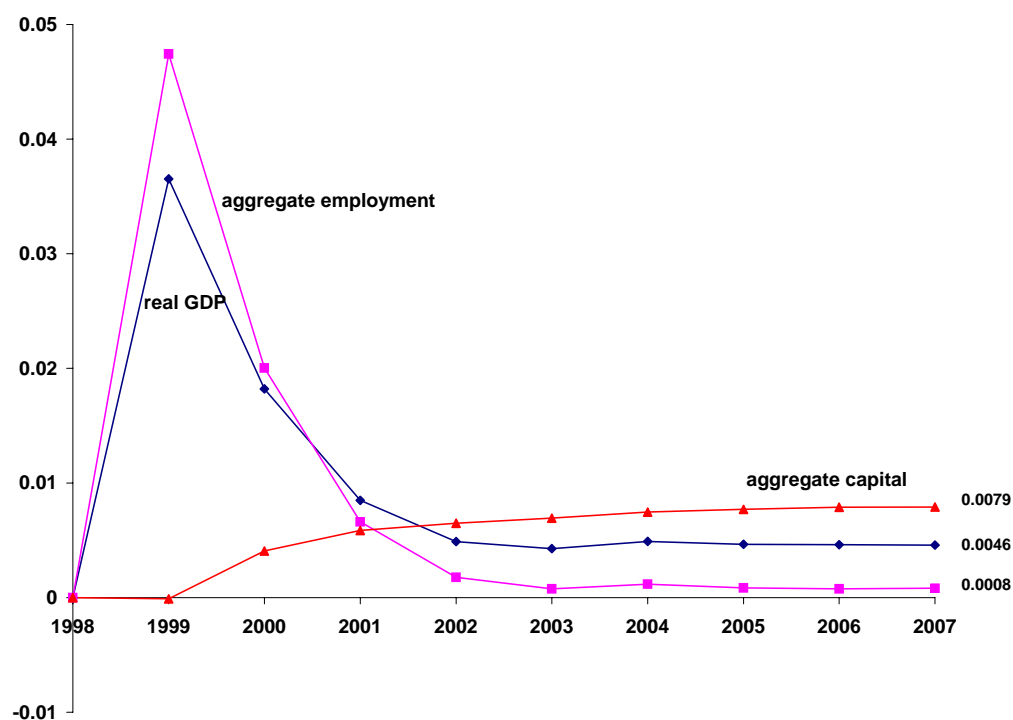
Chart 4.2. Policy-induced percentage deviations in real GDP, employment and capital: Divisia indexes



**Chart 4.3. Policy-induced percentage deviations in real GDP, employment and capital:
Laspeyres indexes**



**Chart 4.4. Policy-induced percentage deviations in real GDP, employment and capital:
Paasche indexes**



$T_2^p(t)$ and $T_2^f(t)$ are the tariff rates applying to good 2 in year t of the policy and forecast runs;

$SM_2(t)$ is the c.i.f. value of imports of good 2 in year t expressed as a share of GDP; and

$m_2(t)$ is the policy-induced percentage deviation in the quantity of imported good 2.

In all simulation years, $T_2^p(t)$ is 5 per cent and $T_2^f(t)$ is 10 per cent. From the Mini-USAGE database, we find that $SM_2(t)$ is about 0.0123 [= 104012/8443540]. From our simulation results we find that $m_2(t)$ is about 3.7 per cent for all t . Bringing all these numbers together in (4.11), gives

$$\text{eff}(t) = \left(\frac{5+10}{200} \right) * 0.0123 * 3.7 = 0.0034 \quad . \quad (4.12)$$

Thus we would expect our results to show a policy-induced total-factor-productivity increase (increase in A) for all years of about 0.0034 per cent.

With a long-run policy-induced increase in K , no long-run reduction in L and a policy-induced increase in A , real GDP as commonly understood should show a long-run policy-induced increase. This happens with both the Laspeyres and Paasche measures, but not with the Divisia measure.

Quantitatively, we would expect the long-run policy-induced percentage increase in real GDP to be approximated by

$$\text{gdpr_dev} = S_k * k_dev + S_\ell * \ell_dev + a_dev \quad (4.13)$$

where

gdpr_dev , k_dev and ℓ_dev are long-run policy-induced percentage deviations in real GDP and the quantities of capital and labour;

S_k and S_ℓ are capital and labour shares in GDP; and

a_dev is the long-run policy-induced percentage deviation in factor productivity.

In our database, S_k and S_ℓ are approximately 0.275 and 0.725. Substituting these numbers into (4.13) and using the long-run capital and labour deviations shown in Charts 4.2 to 4.4 together with the efficiency calculation from (4.12), we can obtain back-of-the-envelope estimates of deviations in real GDP. As can be seen from Table 4.2, the Laspeyres and Paasche calculations accord with our expectations but the Divisia measure of the real GDP deviation is very much at odds with what would be expected on the basis of (4.13).

Table 4.2. Long-run real GDP deviations

	From (4.13)	Simulation result
Divisia	0.0064	-0.0010
Laspeyres	0.0069	0.0067
Paasche	0.0061	0.0046

What is it that causes the Divisia measure to go wrong? We traced the problem to the share effect [see the discussion of (4.3)] associated with aggregate imports. The contribution of this share effect to the difference between the real GDP growth rates $[\text{gdpr}^p(t) - \text{gdpr}^f(t)]$ in year t in the policy and forecast runs is:

$$\text{Share effect, imports} = -[S_m^p(t) - S_m^f(t)] * \left[\frac{\text{mr}^p(t) + \text{mr}^f(t)}{2} \right] \quad (4.14)$$

The cut in protection permanently raises the share of imports in GDP, that is

$$S_m^p(t) > S_m^f(t) \quad \text{for all } t, \quad (4.15)$$

and in our forecasts, real imports are a fast growing component of real GDP. We found that this was sufficient to cause the tariff cut in 1999 to have a permanent long-run negative effect on the rate of growth of the Divisia index of real GDP. Thus when we compare Divisia indexes of real GDP in the policy and forecast runs, we find ever increasing negative policy-induced deviations.

5. Concluding remarks

An effective way to find out about an economic model is to run simulations and to look at results. Not many people have the time or inclination to do this with detailed MONASH-style models such as USAGE. The effort is just too large. However, running mini-USAGE and looking at results requires relatively little effort for people familiar with GEMPACK. Thus, by creating Mini-USAGE we hope to expose USAGE mechanisms and results to a wider audience. We also hope that Mini-USAGE will be a useful starting point for other modelling efforts.

In our own research, we are finding that Mini-USAGE is useful vehicle for testing new developments. As described in Section 4, we have found that the Divisia indexes used in MONASH-style models can sometimes produce misleading results. We are planning to provide alternative indexes in full-scale MONASH-style models. Mini-USAGE has been valuable in developing and testing these alternatives.

The documentation provided here for Mini-USAGE is brief. We are catering for people who like to try things out and to look at instructions for occasional guidance. We hope that users of Mini-USAGE who would like more depth on particular aspects of MONASH-style models will refer some of the detailed documentation cited in our first footnote.

References

- Dixon P.B. and B.R. Parmenter (1996), “Computable General Equilibrium Modelling for Policy Analysis and Forecasting”, Chapter 1, pp.3-85 in H.M. Amman, D.A. Kendrick and J. Rust (eds), *Handbook in Computational Economics*, Volume 1, Elsevier, Amsterdam.
- Dixon P.B. and M.T. Rimmer (2002), *Dynamic General Equilibrium Modelling for Forecasting and Policy: a Practical Guide and Documentation of MONASH*, Contributions to Economic Analysis 256, North-Holland Publishing Company, Amsterdam, pp. xiv+338.
- Dixon, P.B., B.R. Parmenter, J. Sutton and D.P. Vincent (1982), *ORANI: A Multisectoral Model of the Australian Economy*, Contributions to Economic Analysis 142, North-Holland Publishing Company, Amsterdam, pp. xviii+372.
- Dixon P.B., K.R. Pearson, M.R. Picton and M.T. Rimmer (2003), “Rational expectations for large models: a Practical Algorithm and a Policy Application”, presented at the 6th Annual Conference on Global Economic analysis, available from the authors, Centre of Policy Studies, MONASH University, Clayton 3800, Australia (forthcoming in *Economic Modelling*).
- Harrison, W.J. and K.R. Pearson (1996), “Computing Solutions for large General Equilibrium Models using GEMPACK”, *Computational Economics*, Vol. 9, pp. 83-127.
- Harrison, W.J. and K.R. Pearson (2002), Gempack documents GPD-1 (6th edition), GPD-2 (4th edition), GPD-3 (2nd edition), GPD-4 (2nd edition), and GDP-8 (3rd edition), downloadable from the web at <http://www.monash.edu.au/policy/gpdoc.htm>.
- Harrison, W.J. and K.R. Pearson (2005), Gempack documents GPD-5 (1st edition), GPD-6 (12th edition) and GDP-7 (9th edition), downloadable from the web at <http://www.monash.edu.au/policy/gpdoc.htm>.
- Johansen, L. (1960), *A Multisectoral Study of Economic Growth*, Contributions to Economic Analysis 21, North-Holland Publishing Company, Amsterdam, pp. x+177 (enlarged edition, 1974).
- Keller, W.J. (1980), *Tax Incidence: A General Equilibrium Approach*, North-Holland, Amsterdam.
- Pearson, K.R. (1988), “Automating the Computation of Solutions of Large Economic Models”, *Economic Modelling*, Vol. 7, pp. 385-395.
- Taylor, L. and S.L. Black (1974), “Practical General Equilibrium Estimation of Resources Pulls under Trade Liberalization”, *Journal of International Economics*, Vol. 4(1), April, pp. 37-58.
- Staelin, C.P. (1976), “A General Equilibrium Model of Tariffs in a Noncompetitive Economy”, *Journal of International Economics*, Vol. 6(1), pp. 39-63.

Appendix

The Mini-USAGE model: TABLO code
by
Peter B. Dixon and Maureen T. Rimmer
Centre of Policy Studies, Monash University
April 29, 2005

Detailed explanations of equations similar to those presented here can be found in the annotated MONASH code at <http://www.monash.edu.au/policy/monbook2.htm> and in the MONASH book [Dixon P.B. and M.T. Rimmer, Dynamic General Equilibrium Modelling for forecasting and Policy, North-Holland, 2002].

```
!-----!
! Section 1: Files                                     !
File
MMDATA # Input-output flows #;
PARAMS # Main file of non-updatable data, e.g. substitution elasticities #;
EXTRA # Supplementary updatable data, e.g. capital stocks, ROR #;
EXTRA3 # Alternative EXTRA file, generated in forecast & used in policy #;
EXTRA4 # Lagged version of EXTRA file #;
EXTRA5 # Lagged version of EXTRA3 file #;
SRF # Macro data e.g. balance of payments, government accounts #;

! Section 2: Sets and subsets                         !
Set
COM      # Commodities #      (LowPro, HighPro, Construct, Services, Govern);
IND      # Industries #      (LowPro, HighPro, Construct, Services, Govern);
MAR      # Margin Commodities #      (Services);
SRC      # Source of Commodities #      (dom, imp);
NONMAR   # Non-Margin Commodities #      (LowPro, HighPro, Construct, Govern);
GOVT     # Coverment commodities/industries #(Govern);

Subset GOVT is subset of IND;
MAR is subset of COM;
NONMAR is subset of COM;

Set NONGOVT = IND - GOVT;

! Section 3: Coefficient declaration listed alphabetically !
Coefficient
      AGGCAPRIV # Aggregate returns on private capital #;
      AGGINVG # Value of public sector investment #;
      AGGOTH # Total value of government consumption #;
(Parameter) ALPHA1 # Controls wage response to gaps between labour D & S #;
      AV_PROP_CON # Ave propensity to consume Hdy #;
      BENEFITS # Social security benefits #;
      CADEF # Current account deficit #;
(Parameter)(All,i,IND)
      COEFF_SL(i) # Coeff. in capital supply curve #;
(All,c,COM) DELTA(c) # Normalised marginal budget shares #;
(Parameter)(All,i,IND)
      DEP(i) # Depreciation rates #;
(Parameter) DIFF # Used in setting maximum feasible capital growth rates #;
      EMPLOY # Aggregate employment: 1 in init. sol. for yr 1 #;
      EMPLOY_OLD # Aggregate employmen in t, forecast #;
(All,c,COM) EPS(c) # Household expenditure elasticities #;
(All,c,COM)(All,k,COM)
      ETA(c,k) # Household price elasticities #;
```

```

(Parameter) (All, c, COM)
    EXP_ELAST(c) # Export demand elasticities #;
    FDATT # Net foreign liabilities at start of year #;
    FDATT_1 # Net foreign liabilities end of year #;
(Parameter)
    FDATT_1_B # Net foreign liabilities at end of year, base #;
(Parameter)
    FDATT_B # Net foreign liabilities at start of year, base #;
    FRISCH # Frisch 'parameter' #;
(All, i, IND)
    G_VINVEST(i) # Value of government investment in industry i #;
    GNP # Gross national product #;
    GOV_DEF # Government deficit or financing transaction #;
(All, i, IND)
    HOURS(i) # Labour input by industry, hours #;
    HOURSTOT # Total labour input, hours #;
    HOUS_DIS_INC # Household disposable income #;
    INCTAX # Income tax #;
    INF # Rate of inflation #;
    INT_PSD # Nominal rate of interest on public sector debt #;
    INTFD # $US value of interest on f'gn debt #;
(All, i, IND)
    K_GR(i) # Growth in capital through the year #;
(Parameter) (All, i, IND)
    K_GR_MAX(i) # Upper bound on growth for capital #;
(Parameter) (All, i, IND)
    K_GR_MIN(i) # Lower bound on growth for capital #;
(All, c, COM) (All, k, COM)
    KD(c, k) # Kroneker's delta: 1 if c=k, else 0 #;
    LEV_CPI # Level of the CPI #;
(Parameter)
    LEV_CPI_B # Level of the CPI, base, usually CPI for t-1 #;
    LEV_CPI_L # Lagged level of the CPI, that is level in t-1 #;
(Parameter)
    LEV_CPI_L_B # Lagged level of the CPI, base #;
(All, c, COM) (All, i, IND)
    MAKE(c, i) # Commodity outputs by industry #;
(All, i, IND)
    MAKE_C(i) # Output by industry #;
(All, c, COM)
    MAKE_I(c) # Output by commodity #;
    NET_TAXTOTG # Net income and indirect taxes collected #;
    NETINT_G # Net interest paid by government #;
    OTHGOVREV # Other govt. revenue, e.g. sale of 2nd hand assets #;
(All, i, IND)
    PCAP_J(i) # Asset price of capital in i, average in year #;
    PSDATT # Public sector debt, start of year #;
(Parameter)
    PSDATT_1_B # Public sector debt, end of year, base #;
(Parameter)
    PSDATT_B # Public sector debt, start of year, base #;
    PSDATTPLUS1 # Public sector debt, end of year #;
(All, i, IND)
    QCAP(i) # Capital stock at start of year #;
(Parameter) (All, i, IND)
    QCAP_B(i) # Capital stock at start of year, base #;
(All, i, IND)
    QCAP1(i) # Capital stock at end of year #;
(Parameter) (All, i, IND)
    QCAP1_B(i) # Capital stock at end of year, base #;
(All, i, IND)
    QINVEST(i) # Quantity of investment #;
(Parameter)
    R_DEFGDP_B # Ratio public sector deficit to GDP in base year #;
    R_PSDGDP # Ratio of st-of-yr public sector debt to GDP #;
(Parameter)
    R_PSDGDP_B # Ratio st-of-yr public sector debt to GDP, base #;
(Parameter)
    RINT # Real interest rate #;
    RINT_PSD # Real rate of interest, on public sector debt #;
(Parameter)
    ROIFOREIGN # Interest rate on foreign debt #;
    RWAGE # Real wage in year t, CPI deflated #;
(Parameter)
    RWAGE_B # Real wage in year t, CPI deflated, base #;
(Parameter)
    RWAGE_L_B # Real wage in year t-1, CPI deflated, base #;
    RWAGE_OLD # Real wage in year t, CPI deflated, forecast #;
(Parameter)
    RWAGE_OLD_B # Real wage in t, CPI deflated, forecast, base #;
(Parameter)
    RWAGE_O_L_B # Real wage in t-1, CPI deflated, forecast, base #;
(All, c, COM) (All, s, SRC) (All, i, IND)
    S1(c, s, i) # Source shares, intermediate #;

```

```

(All,c,COM)(All,s,SRC)(All,i,IND)
    S2(c,s,i) # Source shares, investment #;
(All,c,COM)(All,s,SRC)
    S3(c,s) # Source shares, households #;
(All,c,COM)
    S3_S(c) # Budget shares, composites #;
(All,c,COM)
    SALES(c) # Sales of domestic commodities #;
(All,s,SRC)
    SDOM(s) # Equals 1 if s = "dom", else 0 #;
(Parameter)(All,c,COM)
    SIGMA1(c) # Armington elasticities: intermediate #;
(Parameter)(All,i,IND)
    SIGMA0(i) # CET transformation elasticities #;
(Parameter)(All,i,IND)
    SIGMA1PRIM(i) # Substitutionn elasticities, prim. factors #;
(Parameter)(All,c,COM)
    SIGMA2(c) # Armington elasticities: investment #;
(Parameter)(All,c,COM)
    SIGMA3(c) # Armington elasticities: households #;
(Parameter)
    SMURF # Recip. of slope of cap. supply curve at K_GR=TREND_K #;
(All,c,COM)
    SS3_S(c) # Subsistence expenditure as share of budget #;
(Parameter)
    SUMDELTA # Initial sum of marg. budget shares #;
    TAX_CAP # Tax collected from capital income #;
    TAX_K_RATE # Rate of tax on capital income #;
    TAX_L_RATE # Rate of tax on labour income, year t #;
    TAX_LAB # Tax collected from labour income #;
    TINY # Tiny no. used to avoid zero divides #;
    TRANS # Transfers from the government #;
(Parameter)(All,i,IND)
    TREND_K(i) # Trend growth rate for capital #;
(All,c,COM)
    V0CIF(c) # Imports by commodity excl. duty #;
    V0CIF_C # Total c.i.f. imports, excludes tariffs #;
    V0GDPEXP # Nominal GDP, expenditure side #;
    V0GDPINC # Nominal GDP from income side #;
(All,c,COM)
    V0IMP(c) # Basic-value of imports #;
    V0IMP_C # Total basic value of imports, includes tariffs #;
(All,c,COM)
    V0MAR_CSI(c) # Total margins usage #;
(All,c,COM)
    V0TAR(c) # Tariff revenue #;
    V0TAR_C # Aggregate tariff revenue #;
    V0TAX_CSI # Aggregate tax revenue #;
(All,c,COM)(All,s,SRC)(All,i,IND)
    V1BAS(c,s,i) # Basic flows, intermediate #;
(All,i,IND)
    V1CAP(i) # Capital rentals #;
    V1CAP_I # Total rental payments to capital #;
(All,i,IND)
    V1LAB(i) # Wage bill by industry #;
    V1LAB_I # Total wage bill #;
(All,c,COM)(All,s,SRC)(All,i,IND)(All,m,MAR)
    V1MAR(c,s,i,m) # Margins, intermediate #;
(All,i,IND)
    V1PRIM(i) # Total primary factor cost by industry #;
    V1PRIM_I # Total primary factor cost #;
(All,c,COM)(All,s,SRC)(All,i,IND)
    V1PUR(c,s,i) # Purchasers' value, intermediate #;
(All,c,COM)(All,i,IND)
    V1PUR_S(c,i) # Purchasers' value, intermed. Composite #;
(All,c,COM)(All,s,SRC)(All,i,IND)
    V1TAX(c,s,i) # Tax revenue, intermediate #;
    V1TAX_CSI # Aggregate tax revenue, intermediate #;
(All,i,IND)
    V1TOT(i) # Total cost by industry #;
(All,c,COM)(All,s,SRC)(All,i,IND)
    V2BAS(c,s,i) # Basic flows, investment #;
(All,c,COM)(All,s,SRC)(All,i,IND)(All,m,MAR)
    V2MAR(c,s,i,m) # Margins, investment #;

```



```

(All,c,COM)(All,s,SRC)(All,i,IND)
    V2PUR(c,s,i) # Purchasers' value, investment #;
(All,c,COM)(All,i,IND)
    V2PUR_S(c,i) # Purchasers' value, invest. composite #;
(All,c,COM)(All,s,SRC)(All,i,IND)
    V2TAX(c,s,i) # Tax revenue, investment #;
    V2TAX_CSI # Aggregate tax revenue, investment #;
(All,i,IND)
    V2TOT(i) # Total investment by industry #;
    V2TOT_I # Total investment #;
(All,c,COM)(All,s,SRC)
    V3BAS(c,s) # Basic flows, households #;
(All,c,COM)(All,s,SRC)(All,m,MAR)
    V3MAR(c,s,m) # Margins, households #;
(All,c,COM)(All,s,SRC)
    V3PUR(c,s) # Purchasers' value, households #;
(All,c,COM)
    V3PUR_S(c) # Purchasers' value, households composite #;
(All,c,COM)(All,s,SRC)
    V3TAX(c,s) # Tax revenue, households #;
    V3TAX_CS # Aggregate tax revenue, households #;
    V3TOT # Total purchases by households #;
(All,c,COM)
    V4BAS(c) # Basic flows, export #;
(All,c,COM)(All,m,MAR)
    V4MAR(c,m) # Margins, export #;
(All,c,COM)
    V4PUR(c) # Purchasers' value, exports #;
(All,c,COM)
    V4TAX(c) # Tax revenue, exports #;
    V4TAX_C # Aggregate tax revenue, export #;
    V4TOT # Total export earnings #;
(All,i,COM)(All,s,SRC)
    V5BAS(i,s) # Basic flows, government #;
    V5TOT # total purchases by government #;
(All,c,COM)
    V5PUR_S(c);
(All,i,IND)
    VCAP(i) # Value of capital stock at start of year #;
(All,i,IND)
    WAGE(i) # Wage rate by industry #;

```

! Section 4: Read instructions for coefficients listed alphabetically

!

READ

```

ALPHA1      from file PARAMS Header "ALPH";
BENEFITS    from file SRF Header "BENS";
DELTA       from file MMDATA Header "P044";
DEP         from File MMDATA Header "DEP";
DIFF        from file PARAMS Header "DIFF";
EMPLOY      from file EXTRA Header "EMPTY";
EMPLOY_OLD  from file EXTRA3 Header "EMPTY";
EXP_ELAST   from File MMDATA Header "P018";
FDATT       from file SRF Header "FDAT";
FRISCH      from File MMDATA Header "P021";
G_VINVEST   from file SRF Header "0044";
LEV_CPI     from file EXTRA Header "LCPI";
LEV_CPI_L   from file EXTRA Header "LCPL";
MAKE        from File MMDATA Header "MAKE";
NETINT_G    from file SRF Header "NING";
PCAP_J      from file EXTRA Header "PCAP";
PSDAT       from file SRF Header "0037";
RINT        from file EXTRA Header "RINT";
ROIFOREIGN  from file PARAMS Header "FINT";
RWAGE       from file EXTRA Header "RWAG";
RWAGE_L_B   from file EXTRA4 Header "RWAG";
RWAGE_O_L_B from file EXTRA5 Header "RWAG";
RWAGE_OLD   from file EXTRA3 Header "RWAG";

```

```

SIGMA0      from File MMDATA Header "SCET";
SIGMA1      from File MMDATA Header "1ARM";
SIGMA1PRIM  from File MMDATA Header "P028";
SIGMA2      from File MMDATA Header "2ARM";
SIGMA3      from File MMDATA Header "3ARM";
SMURF       from file PARAMS Header "MURF";
TAX_K_RATE  from file EXTRA Header "TAXK";
TAX_L_RATE  from file EXTRA Header "TAXL";
TREND_K     from file EXTRA Header "GRSE";
V0TAR       from File MMDATA Header "0TAR";
V1BAS       from File MMDATA Header "1BAS";
V1CAP       from File MMDATA Header "1CAP";
V1LAB       from File MMDATA Header "1LAB";
V1MAR       from File MMDATA Header "1MAR";
V1TAX       from File MMDATA Header "1TAX";
V2BAS       from File MMDATA Header "2BAS";
V2MAR       from File MMDATA Header "2MAR";
V2TAX       from File MMDATA Header "2TAX";
V3BAS       from File MMDATA Header "3BAS";
V3MAR       from File MMDATA Header "3MAR";
V3TAX       from File MMDATA Header "3TAX";
V4BAS       from File MMDATA Header "4BAS";
V4MAR       from File MMDATA Header "4MAR";
V4TAX       from File MMDATA Header "4TAX";
V5BAS       from file MMDATA Header "5BAS";
VCAP        from File MMDATA Header "VCAP";
WAGE        from file EXTRA Header "WAGE";

```

! Section 5: Formulas in thematic groups

! Subsection 5a: Dummies

Formula

```

TINY = 0.0000000000001;
SDOM("dom") = 1.0;
SDOM("imp") = 0.0;

```

! Subsection 5b: Aggregates from input-output data including tax collections, purchasers' values and macro coefficients

Formula

```

(All,i,IND) MAKE_C(i) = Sum(c,COM,MAKE(c,i));
(All,c,COM) MAKE_I(c) = Sum(i,IND,MAKE(c,i));
(All,c,COM)V0IMP(c) = Sum(i,IND,V1BAS(c,"imp",i) + V2BAS(c,"imp",i))
                    + V3BAS(c,"imp");
V0IMP_C = Sum(c,COM,V0IMP(c));
(All,m,MAR) V0MAR_CSI(m) = Sum(c,COM, V4MAR(c,m) + Sum(s,SRC,V3MAR(c,s,m)
                    + Sum(i,IND,V1MAR(c,s,i,m) + V2MAR(c,s,i,m))));
(All,n,NONMAR) V0MAR_CSI(n) = 0.0;
V0TAR_C = Sum(c,COM,V0TAR(c));
AGGCAPRIV = Sum(i,NONGOVT, V1CAP(i));
V1CAP_I = Sum(i,IND,V1CAP(i));
V1LAB_I = Sum(i,IND,V1LAB(i));
(All,i,IND) V1PRIM(i) = V1LAB(i) + V1CAP(i);
V1PRIM_I = V1LAB_I + V1CAP_I;
(All,c,COM)(All,s,SRC)(All,i,IND) V1PUR(c,s,i) = V1BAS(c,s,i) + V1TAX(c,s,i)
                    + Sum(m,MAR,V1MAR(c,s,i,m));
(All,c,COM)(All,i,IND) V1PUR_S(c,i) = Sum(s,SRC,V1PUR(c,s,i));
V1TAX_CSI = Sum(c,COM,Sum(s,SRC,Sum(i,IND,V1TAX(c,s,i))));
(All,i,IND) V1TOT(i) = Sum(c,COM,V1PUR_S(c,i)) + V1PRIM(i);
(All,c,COM)(All,s,SRC)(All,i,IND) V2PUR(c,s,i) = V2BAS(c,s,i) + V2TAX(c,s,i)
                    + Sum(m,MAR,V2MAR(c,s,i,m));
(All,c,COM)(All,i,IND) V2PUR_S(c,i) = Sum(s,SRC,V2PUR(c,s,i));

```

```

V2TAX_CSI = Sum(c, COM, Sum(s, SRC, Sum(i, IND, V2TAX(c, s, i))));
(All, i, IND) V2TOT(i) = Sum(c, COM, V2PUR_S(c, i));
V2TOT_I = Sum(i, IND, V2TOT(i));
(All, c, COM) (All, s, SRC) V3PUR(c, s) = V3BAS(c, s) + V3TAX(c, s)
                                         + Sum(m, MAR, V3MAR(c, s, m));
(All, c, COM) V3PUR_S(c) = Sum(s, SRC, V3PUR(c, s));
V3TAX_CS = Sum(c, COM, Sum(s, SRC, V3TAX(c, s)));
V3TOT = Sum(c, COM, V3PUR_S(c));
(All, c, COM) V4PUR(c) = V4BAS(c) + V4TAX(c) + Sum(m, MAR, V4MAR(c, m));
V4TAX_C = Sum(c, COM, V4TAX(c));
V4TOT = Sum(c, COM, V4PUR(c));
(All, c, COM) V5PUR_S(c) = Sum(s, SRC, V5BAS(c, s));
V5TOT = Sum(c, COM, V5PUR_S(c));
(All, c, COM) SALES(c) = Sum(i, IND, V1BAS(c, "dom", i) + V2BAS(c, "dom", i)
                             + V3BAS(c, "dom") + V4BAS(c) + V5BAS(c, "dom") + V0MAR_CSI(c));
(All, c, COM) V0CIF(c) = V0IMP(c) - V0TAR(c);
V0CIF_C = Sum(c, COM, V0CIF(c));
V0TAX_CSI = V1TAX_CSI + V2TAX_CSI + V3TAX_CS + V4TAX_C + V0TAR_C;
TAX_LAB = TAX_L_RATE*(V1LAB_I);
TAX_CAP = TAX_K_RATE*AGGCAPRIV;
INCTAX = TAX_LAB + TAX_CAP;
(Initial) LEV_CPI_B = LEV_CPI;
(Initial) LEV_CPI_L_B = LEV_CPI_L;
INF = LEV_CPI/LEV_CPI_L - 1;
V0GDPEXP = V3TOT + V2TOT_I + V4TOT + V5TOT - V0CIF_C;
V0GDPINC = V1PRIM_I + V0TAX_CSI;

```

! Subsection 5c: Current account and foreign debt

!

Formula

```

INTFD=FDATT*ROIFOREIGN;
CADEF = V0CIF_C - V4TOT + INTFD;
FDATT_1= FDATT + CADEF;
(Initial) FDATT_1_B =FDATT_1;
(Initial)FDATT_B =FDATT;

```

! Subsection 5d: Import/domestic shares

!

Zerodivide Default 0.5;

Formula

```

(All, c, COM) (All, s, SRC) (All, i, IND) S1(c, s, i) = V1PUR(c, s, i)/V1PUR_S(c, i);
(All, c, COM) (All, s, SRC) (All, i, IND) S2(c, s, i) = V2PUR(c, s, i)/V2PUR_S(c, i);
(All, c, COM) (All, s, SRC) S3(c, s) = V3PUR(c, s)/V3PUR_S(c);
Zerodivide Off;

```

! Subsection 5e: Coefficients for consumer demand equations

!

Formula

```

(Initial) SUMDELTA = Sum(c, COM, DELTA(c));
(Initial) (All, c, COM) DELTA(c) = DELTA(c)/SUMDELTA;
(All, c, COM) S3_S(c) = V3PUR_S(c)/V3TOT;
Zerodivide Default 1.0;
Formula (All, c, COM) EPS(c) = DELTA(c)/S3_S(c);
Zerodivide Off;
Formula (All, c, COM) SS3_S(c) = S3_S(c) + DELTA(c)/FRISCH;
(All, c, COM) (All, k, COM) KD(c, k) = 0.0;
(All, c, COM) KD(c, c) = 1.0;
(All, c, COM) (All, k, COM) ETA(c, k) = KD(c, k)*EPS(c)/FRISCH
                                         - EPS(c)*[S3_S(k) + DELTA(k)/FRISCH];

```

! Subsection 5f: Capital, investment and rates of return

!

Formula

```

(All, i, IND) QINVEST(i) = V2TOT(i)/PCAP_J(i);
(All, i, IND) QCAP(i) = VCAP(i)/PCAP_J(i);

```

```

(All,i,IND) QCAP1(i) = QCAP(i)*(1 - DEP(i)) + QINVEST(i);
(Initial)(All,i,IND) QCAP_B(i) = QCAP(i);
(Initial)(All,i,IND) QCAP1_B(i) = QCAP1(i);
(Initial)(All,i,IND) K_GR_MIN(i) = - DEP(i);
(Initial)(All,i,IND) K_GR_MAX(i) = TREND_K(i) + DIFF;
(All,i,IND) K_GR(i) = [QCAP1(i)/QCAP(i)] - 1;
(Initial)(All,i,IND) COEFF_SL(i) = SMURF*
(K_GR_MAX(i)-K_GR_MIN(i))/[(K_GR_MAX(i)-TREND_K(i))*(TREND_K(i)-K_GR_MIN(i))];

! Subsection 5g: Calculations of coefficients used in government accounts !
OTHGOVREV = Sum(i,GOVT,V1CAP(i));
AGGINVG = Sum(i,IND,G_VINVEST(i));
AGGOTH = Sum(i,COM,Sum(s,SRC, V5BAS(i,s)));
NET_TAXTOTG = V0TAX_CSI + INCTAX ;
PSDATTPLUS1 = PSDATT + AGGOTH + AGGINVG - NET_TAXTOTG
- OTHGOVREV + BENEFITS + NETINT_G;
INT_PSD = NETINT_G/PSDATT;
RINT_PSD = (1+INT_PSD)/(1 + INF) - 1;
TRANS = BENEFITS + NETINT_G;
GOV_DEF = AGGOTH + AGGINVG - NET_TAXTOTG - OTHGOVREV + TRANS;
(Initial) R_DEFGDP_B = GOV_DEF/V0GDPEXP;
(Initial) PSDATT_1_B = PSDATTPLUS1;
(Initial) PSDATT_B = PSDATT;
R_PSDGDP = PSDATT/V0GDPEXP;
(Initial) R_PSDGDP_B = R_PSDGDP;

! Subsection 5h: GNP, saving, disposable income and propensities to !
save and consume !
Formula
GNP = V0GDPEXP - INTFD;
HOUS_DIS_INC = V0GDPEXP + TRANS - NET_TAXTOTG - OTHGOVREV - INTFD;
AV_PROP_CON = V3TOT/HOUS_DIS_INC;

! Subsection 5i: Wage rates and employment !
Formula
(Initial) RWAGE_B = RWAGE;
(Initial) RWAGE_OLD_B = RWAGE_OLD;
(All,i,IND) HOURS(i) = V1LAB(i)/WAGE(i);
HOURSTOT = Sum(i,IND, HOURS(i));

! Section 6: Variable declarations in alphabetical order !
Variable
(All,i,IND) a(i) # Technical change in industry i #;
a_cont # Contribution of tech change to GDP #;
(All,c,COM)(All,i,IND)
a0(c,i) # Com-c-output-augmenting techn. chge #;
(All,i,IND) a0_c(i) # Output-augmenting technical change by industry #;
(All,c,COM)(All,i,IND)
a1_s(c,i) # Input-c-saving tech change in industry i #;
(All,i,IND) allk(i) # Aggregate of labour and capital saving tech ch #;
(All,c,COM)(All,s,SRC)(All,i,IND)
alcsi(c,s,i) # Input-c,s-saving tech change in industry i #;
(All,c,COM)(All,i,IND)
alcsi_s(c,i) # Aggregate of c,s-saving tech change in i #;
(All,i,IND) alcap(i) # Capital-saving technical change #;
(All,i,IND) allab(i) # Labour-saving technical change #;
(All,c,COM)(All,s,SRC)(All,i,IND)(All,m,MAR)
almar(c,s,i,m) # Technical change in margins, intermediate #;
(All,i,IND) alprim(i) # All-primary-factor augmenting tech.change #;
(All,c,COM)(All,i,IND)
a2_s(c,i) # c-saving tech change in capital creation in i #;

```

```

(All,c,COM)(All,s,SRC)(All,i,IND)
    a2csi(c,s,i) # c,s-saving tech ch. in capital creation in i #;
(All,c,COM)(All,i,IND)
    a2csi_s(c,i) # Agg. of c,s-saving tech ch. in i's cap creat. #;
(All,c,COM)(All,s,SRC)(All,i,IND)(All,m,MAR)
    a2mar(c,s,i,m) # Technical change in margins, investment #;
(All,c,COM)
    a3com(c) # Combined change in household tastes #;
(All,c,COM)(All,s,SRC)
    a3cs(c,s) # Commodity c,s-augmenting change in hhld pref. #;
(All,c,COM)
    a3cs_s(c) # Agg. of c,s-augmenting change in hhld pref. #;
(All,c,COM)(All,s,SRC)(All,m,MAR)
    a3mar(c,s,m) # Technical change in margins, consumption #;
(All,c,COM)(All,m,MAR)
    a4mar(c,m) # Technical change in margins, export #;
(All,c,COM)
    ac(c) # Commodity-c-using technical change #;
    agginv_g # Value of public sector investment #;
    agginv_rg # Real public investment #;
    apc # Average prop. to consume out of Hhold disposable income #;
    apc_gnp # Ratio of total consumption to GNP #;
    aps_gnp # Average propensity to save out of GNP #;
    ave_a3com # Average value of a3com #;
    bens # Social security benefits #;
(All,i,IND)
    cap_t(i) # Start-of-year capital stock #;
(All,i,IND)
    cap_t1(i) # End-of-year capital stock #;
(Change)
    d_cad # Current account deficit #;
(Change)
    d_def_gdp_r # Change in ratio of govt. deficit to GDP #;
(Change)(All,i,IND)
    d_f_cap_t(i) # Shifter in year-to-year capital growth eqn #;
(Change)
    d_f_eeqror # Scalar shift variable in K-growth/ROR equation #;
(Change)(All,i,IND)
    d_f_eeqror_j(i) # Vector shifter in K-growth/ROR equation #;
(Change)
    d_f_p3tot_l
    # Shift for p3tot_l, make endog. if init solut. not for t-1 #;
(Change)
    d_f_fd_t # Shifter in start-of-year foreign debt equation #;
(Change)
    d_f_psd_t # Endo in pub-sect-def eqn if init sol. not for t-1 #;
(Change)
    d_fd_t # Foreign debt at start of year #;
(Change)(All,i,COM)
    d_gamma(i) # Change in subsistence cons. as % of consumption #;
(Change)
    d_gov_def # Public sector deficit #;
(Change)
    d_housav # Household savings #;
(Change)
    d_inf # Rate of inflation #;
(Change)
    d_int_fd # Interest on foreign debt #;
(Change)
    d_int_psd # Rate of interest on public sector debt #;
(Change)
    d_net_int_g # Net interest paid by government #;
(Change)
    d_net_tax_gdp # Change in the net tax to GDP ratio #;
(Change)
    d_psd_t # Public sector debt, start of year #;
(Change)
    d_r_psdgdp # Ratio of st-of-yr public sector debt to GDP #;
(Change)
    d_rint # Real interest rate #;
(Change)
    d_rint_psd # Real rate of interest on public sector debt #;
(Change)
    del_b # Balance of Trade #;
(Change)
    del_b_gdp # (Balance of Trade)/GDP #;
(Change)
    del_f_wage_c # Shift in labour supply, pre-tax #;
(Change)(All,i,IND)
    del_ff_rate(i) # Shifter in equal rate-of-return equation #;
(Change)(All,i,IND)
    del_k_gr(i) # Capital growth through year #;
(Change)
    del_r_tot # Allows for equal changes in rates of return #;
(Change)(All,i,IND)
    del_ror(i) # Expected rate of return # ;
(Change)
    del_unity # Normally shocked from zero to one#;

```

```

(All,i,COM)      deltapc(i) # Change in marginal budget shares#;
                  emp_hours # Aggregate employment, hours #;
                  emp_hours_o # Aggregate employment in hours, forecast #;
                  employ_i # Aggregate employment #;
(All,i,IND)      f_alcap(i) # Shift for capital-saving technical change #;
(All,c,COM)(All,s,SRC)(All,i,IND)
                  f_alcsi(c,s,i) # Allows exog. of csi tech change in cur prod #;
(All,i,IND)      f_allab(i) # Shift for labour-saving technical change #;
(All,c,COM)(All,s,SRC)(All,i,IND)
                  f_a2csi(c,s,i) # Allows exog. of csi tech change in cap creat #;
(All,c,COM)(All,s,SRC)
                  f_a3cs(c,s) # Allows exog, of cs pref change by Hholds #;
(All,i,IND)      f_alprim(i) # Vector shift prim-fac augmenting tech.change #;
                  f_emp_o # Shift in eqn that sets the value of emp_hours_o #;

                  f_rwage_o # Shift in eqn that sets the value of real_wage_c_o #;
(All,c,COM)      f_t0imp(c) # Vector shifter on power of tariff #;
                  f_tax_r # Ratio of capital to labour tax rates # ;
(All,c,COM)      f0tax_s(c) # General sales tax shifter #;
(All,i,IND)      f1lab(i) # Wage shift variable #;
                  f1lab_i # Overall wage shifter #;
                  f1tax_csi # Tax shift, intermediate usage #;
                  f2tax_csi # Tax shift, investment #;
                  f2tot_i # Shift in IR/CR ratio #;
                  f3tax_cs # Tax shift, household usage #;
                  f4gen # Uniform vertical shifter on export demand curves #;
(All,c,COM)      f4p(c) # Export demand shift, price or vertical #;
(All,c,COM)      f4q(c) # Export demand shift, quantity or horizontal #;
(All,c,COM)      f4tax(c) # Export tax shifter #;
                  f4tax_c # Tax shift, exports #;
(All,c,COM)      f5(c) # Shift terms for government demands domestic #;
                  f5gen # Overall shift term for government demands #;
(All,c,COM)      falc(c) # Shifter: input-c-saving tech change in production #;
(All,c,COM)(All,i,IND)
                  falci(c,i) # Shifter: input-c-sav tech change in i's product. #;
(All,c,COM)(All,s,SRC)(All,i,IND)(All,m,MAR)
                  falmar(c,s,i,m) # Shifter tech change margins, intermediate #;
(All,c,COM)      falmarc(c) # Shifter tech change in margins, intermediate #;
(All,c,COM)      fa2c(c) # Shifter: input-c-saving tech change in investment #;
(All,c,COM)(All,i,IND)
                  fa2ci(c,i) # Shifter: input-c-sav tech change in i's invest. #;
(All,c,COM)(All,s,SRC)(All,i,IND)(All,m,MAR)
                  fa2mar(c,s,i,m) # Shifter tech change margins, investment #;
(All,c,COM)      fa2marc(c) # Shifter tech change in margins, investment #;
(All,c,COM)(All,s,SRC)(All,m,MAR)
                  fa3mar(c,s,m) # Shifter tech change margins, consumption #;
(All,c,COM)      fa3marc(c) # Shifter tech change in margins, consumption #;
(All,c,COM)(All,m,MAR)
                  fa4mar(c,m) # Shifter tech change margins, exports #;
(All,c,COM)      fa4marc(c) # Shifter tech change in margins, exports #;
                  ff_alprim # Scalar shift prim-fac augmenting tech.change #;
                  ff_t0imp # Scalar shifter on power of tariff #;
(All,c,COM)      ftwist_src(c) # Import/domestic twist shift #;
(All,i,IND)      ftwistlk(i) # Labour/capital twist shift #;
                  gnpnom # Gross national product #;
                  hdy # Household disposable income #;
                  lev_def_gdp_r # Level of public sector deficit to GDP ratio #;
                  lev_r_psdgdp # Level of ratio of public sector debt to GDP #;
                  net_tax_tot # Net collection of income and indirect taxes #;
                  oth_gov_rev # Govt. revenue apart from taxes & interest #;

```

```

(All,c,COM)(All,s,Src)
    p0(c,s) # Basic price #;
(All,i,IND)    p0_e(i) # Effective price of ind output in com supply eqns #;
               p0cif_c # Imports price index, CIF, $US #;
(All,c,COM)    p0dom(c) # Basic price of domestic commodity #;
               p0gdpexp # GDP price index, expenditure #;
(All,c,COM)    p0imp(c) # Basic price of imported commodity #;
               p0imp_c # Duty-paid imports price index #;
(All,i,IND)    p0ind(i) # Price of industry outputs #;
               p0realdev # Real devaluation #;
               p0toft # Terms of trade #;
(All,c,COM)(All,s,Src)(All,i,IND)
    p1(c,s,i) # Intermediate purchasers' price #;
(All,c,COM)(All,i,IND)
    p1_s(c,i) # Intermediate composite prices #;
(All,i,IND)    p1cap(i) # Rental price of capital #;
               p1cap_i # Average capital rental #;
(All,i,IND)    p1lab(i) # Wage #;
(All,i,IND)    p1prim(i) # Price in primary-factor demand eqn #;
(All,c,COM)(All,s,Src)(All,i,IND)
    p2(c,s,i) # Purchasers' price, investment #;
(All,c,COM)(All,i,IND)
    p2_s(c,i) # Investment composite prices #;
(All,i,IND)    p2tot(i) # Costs of units of Capital #;
               p2tot_g # Price index for government investment expenditure #;
               p2tot_i # Investment price index #;
(All,c,COM)(All,s,Src)
    p3(c,s) # Purchasers' price, households #;
(All,c,COM)    p3_s(c) # Household composite prices #;
               p3tot # Consumer price index (CPI) #;
               p3tot_l # Lagged CPI, usually CPI in year t-1 #;
(All,c,COM)    p4(c) # Purchasers' price, exports $US #;
               p4tot # Exports price index #;
(All,c,COM)(All,s,Src)
    p5(c,s) # Purchasers' prices paid by govt by com and source #;
    p5tot # Government demands price index #;
(All,c,COM)    pf0cif(c) # CIF import prices, $F #;
    phi # Exchange rate, mid year, $Foreign/$US #;
    pop # U.S. population #;
    q # Number of Households #;
(All,i,IND)    r_inv_cap(i) # Invest/capital ratio shifters #;
               r_inv_cap_u # Uniform shifter in I/K ratio #;
               real_wage_c # Real wage for consumers #;
               real_wage_c_o # Real wage in forecast sim #;
(All,c,COM)    t0imp(c) # Power of tariffs, (1 + tariff rate/100) #;
(All,c,COM)(All,s,Src)(All,i,IND)
    t1(c,s,i) # Tax power, intermediate #;
(All,c,COM)(All,s,Src)(All,i,IND)
    t2(c,s,i) # Tax power, investment #;
(All,c,COM)(All,s,Src)
    t3(c,s) # Tax power household #;
(All,c,COM)    t4(c) # Tax power, export #;
               tax_k_r # Rate of tax on capital income #;
               tax_l_r # Rate of tax on labour income #;
               taxindg # Aggregate revenue from all indirect taxes #;
               taxrev_inc # Income tax revenue #;
               transfers # Transfers from the government #;
               twist_c # Scalar shift for import/domestic twist #;
               twist_i # Scalar shift for labour/capital twist #;
(All,c,COM)    twist_src(c) # Used to impose cost-neutral imp/dom pref twist #;
(All,i,IND)    twistlk(i) # Used to impose cost-neutral L/K tech twists #;

```



```

w0cif_c # CIF $US value of imports #;
w0gdpxp # Nominal GDP, expenditure side #;
w0gdpinc # Nominal GDP, Income side #;
w0imp_c # Value of imports plus Duty #;
w0tar_c # Aggregate tariff revenue #;
w0tax_csi # Aggregate indirect-tax revenue #;
w1cap_i # Aggregate payments to capital #;
w1lab_i # Aggregate payments to labour #;
w1tax_csi # Tax revenue, intermediate #;
w2tax_csi # Tax revenue, investment #;
w2tot_i # Aggregate nominal investment #;
w3tax_cs # Tax revenue, households #;
w3tot # Total household expenditure #;
(Change) d_w4tax_c # Tax revenue, export #;
w4tot # $US border value of exports #;
w5tot # Aggregate nominal value of government demands #;
(All,c,COM)(All,i,IND)
x0(c,i) # Commodity Output by Industry #;
x0cif_c # Import volume index, CIF weights #;
(All,c,COM) x0dom(c) # Supplies of domestic goods #;
x0gdpxp # Real GDP, expenditure Side #;
x0gdpinc # Real GDP, income side #;
(All,c,COM) x0imp(c) # Supplies of imported goods #;
x0imp_c # Import volume index, duty-paid weights #;
(All,c,COM)(All,s,SRC)(All,i,IND)
x1(c,s,i) # Intermediate demand #;
(All,c,COM)(All,i,IND)
x1_s(c,i) # Intermediate composites #;
(All,i,IND) x1cap(i) # Start-of-year capital stock #;
x1cap_i # Aggregate capital stock, start of year, rental wgts #;
(All,i,IND) x1lab(i) # Employment #;
(All,c,COM)(All,s,SRC)(All,i,IND)(All,m,MAR)
x1mar(c,s,i,m) # Margins, intermediate #;
(All,i,IND) x1prim(i) # Primary factor composite #;
(All,c,COM)(All,s,SRC)(All,i,IND)
x2(c,s,i) # Investment demand #;
(All,c,COM)(All,i,IND)
x2_s(c,i) # Investment composites #;
(All,c,COM)(All,s,SRC)(All,i,IND)(All,m,MAR)
x2mar(c,s,i,m) # Margins, investment #;
(All,i,IND) x2tot(i) # Investment by using industry #;
x2tot_i # Aggregate real investment #;
(All,c,COM)(All,s,SRC)
x3(c,s) # Household demand #;
(All,c,COM) x3_s(c) # Household composites #;
(All,c,COM)(All,s,SRC)(All,m,MAR)
x3mar(c,s,m) # Margins, household #;
x3tot # Real household consumption #;
(All,c,COM) x4(c) # Export demand #;
(All,c,COM)(All,m,MAR)
x4mar(c,m) # Margins, export #;
x4tot # Export volume index #;
(All,c,COM)(All,s,SRC)
x5(c,s) # Government demands by commodity and source #;
x5tot # Aggregate real government demands #;
(All,i,IND) z(i) # Activity level #;

! Section 7: Updates in alphabetical order !
Update
BENEFITS = bens;
(All,i,COM) DELTA(i) = deltapc(i);

```



```

EMPLOY = emp_hours;
EMPLOY_OLD = emp_hours_o;
(Change) FDATT = d_fd_t;
(Explicit) FRISCH =
-100/[100 - Sum(i,COM,SS3_S(i)*(100+p3_s(i)-w3tot+q)+ S3_S(i)*d_gamma(i))];
(All,j,IND) G_VINVEST(j) = x2tot(j)*p2tot(j);
(Change) NETINT_G = d_net_int_g;
(All,c,COM)(All,i,IND)
MAKE(c,i) = p0(c,"dom")*x0(c,i);
LEV_CPI = p3tot;
LEV_CPI_L = p3tot_l;
(All,i,IND) PCAP_J(i) = p2tot(i);
(Change) PSDATT = d_psd_t;
RWAGE = real_wage_c;
RWAGE_OLD = real_wage_c_o;
TAX_K_RATE = tax_k_r;
TAX_L_RATE = tax_l_r;
(Change)(All,c,COM)
V0TAR(c) = V0TAR(c)*{x0imp(c)+pf0cif(c)- phi}/100
+ V0IMP(c)*t0imp(c)/100;
(All,c,COM)(All,s,SRC)(All,i,IND)
V1BAS(c,s,i) = p0(c,s)*x1(c,s,i);
(All,i,IND) V1CAP(i) = p1cap(i)*x1cap(i);
(All,i,IND) V1LAB(i) = p1lab(i)*x1lab(i);
(All,c,COM)(All,s,SRC)(All,i,IND)(All,m,MAR)
V1MAR(c,s,i,m) = p0(m,"dom")*x1mar(c,s,i,m);
(Change)(All,c,COM)(All,s,SRC)(All,i,IND)
V1TAX(c,s,i) = V1TAX(c,s,i)*{x1(c,s,i) + p0(c,s)}/100
+ {V1BAS(c,s,i) + V1TAX(c,s,i)}*t1(c,s,i)/100;
(All,c,COM)(All,s,SRC)(All,i,IND)
V2BAS(c,s,i) = p0(c,s)*x2(c,s,i);
(All,c,COM)(All,s,SRC)(All,i,IND)(All,m,MAR)
V2MAR(c,s,i,m) = p0(m,"dom")*x2mar(c,s,i,m);
(Change)(All,c,COM)(All,s,SRC)(All,i,IND)
V2TAX(c,s,i) = V2TAX(c,s,i)*{x2(c,s,i)+ p0(c,s)}/100
+ {V2BAS(c,s,i)+V2TAX(c,s,i)}*t2(c,s,i)/100;
(All,c,COM)(All,s,SRC)
V3BAS(c,s) = p0(c,s)*x3(c,s);
(All,c,COM)(All,s,SRC)(All,m,MAR)
V3MAR(c,s,m) = p0(m,"dom")*x3mar(c,s,m);
(Change)(All,c,COM)(All,s,SRC)
V3TAX(c,s) = V3TAX(c,s)*{x3(c,s) + p0(c,s)}/100
+ {V3BAS(c,s)+V3TAX(c,s)}*t3(c,s)/100;
(All,c,COM) V4BAS(c) = p0(c,"dom")*x4(c);
(All,c,COM)(All,m,MAR)
V4MAR(c,m) = p0(m,"dom")*x4mar(c,m);
(Change)(All,c,COM)
V4TAX(c) = V4TAX(c)*{x4(c) + p0(c,"dom")}/100
+ {V4BAS(c)+V4TAX(c)}*t4(c)/100;
(All,i,IND) VCAP(i) = cap_t(i)*p2tot(i);
(All,c,COM)(All,s,SRC)
V5BAS(c,s) = p0(c,s)*x5(c,s);
(All,i,IND) WAGE(i) = p1lab(i);

```

```

! Section 8: Equations in thematic order !

! Subsection 8.1: Industry decisions !
! Demands for primary factors !
! Producers choose labour [X1LAB(i)] and capital [X1CAP(i)] !
! to minimise primary-factor costs !
! [(P1LAB(i)*A1LAB(i))*(X1LAB(i)/A1LAB(i))+ !
! P1CAP(i)*A1CAP(i))*(X1CAP(i)/A1CAP(i))] !
! subject to a CES aggregation function: !
! X1PRIM(i) = CES [ X1LAB(i)/A1LAB(i), X1CAP(i)/A1CAP(i) ] !

Equation E_xllab # Industry demands for labour #
(All,i,IND)
xllab(i) - allab(i) = x1prim(i) - SIGMA1PRIM(i)*[p1lab(i) - plprim(i)]
-SIGMA1PRIM(i)*[allab(i) - allk(i)];

Equation E_xlcap # Industry demands for capital #
(All,i,IND)
xlcap(i) - alcap(i) = x1prim(i) - SIGMA1PRIM(i)*[plcap(i) - plprim(i)]
-SIGMA1PRIM(i)*[alcap(i) - allk(i)]
! - [V1LAB(i)/V1PRIM(i)]*twistlk(i)!!;

Equation E_plprim # Price term in factor demand equations #
(All,i,IND)
V1PRIM(i)*plprim(i) = V1LAB(i)*p1lab(i) + V1CAP(i)*plcap(i) ;

Equation E_alprim # Price term in factor demand equations #
(All,i,IND)
V1PRIM(i)*allk(i) = V1LAB(i)*allab(i) + V1CAP(i)*alcap(i) ;

! Intermediate demands for domestic and imported inputs !
! Producers choose domestic and imported intermediate inputs of each commodity !
! to minimise costs !
! Sum(s,SRC, P1(c,s,i)*X1(c,s,i) !
! subject to a CES aggregation function: !
! X1_S(c,i) = CES [ All,s,SRC: X1(c,s,i) ] !

Equation E_x1 # Source-specific demands for intermediate inputs #
(All,c,COM)(All,s,SRC)(All,i,IND)
x1(c,s,i) = x1_s(c,i) - SIGMA1(c)*{pl(c,s,i) - pl_s(c,i)}
+alcsi(c,s,i) -SIGMA1(c)*{alcsi(c,s,i) - alcsi_s(c,i)};

Equation E_pl_s # Effective price of composite intermediate input #
(All,c,COM)(All,i,IND)
pl_s(c,i) = Sum(s,SRC,S1(c,s,i)*pl(c,s,i));

Equation E_alcsi_s # Effective price of composite intermediate input #
(All,c,COM)(All,i,IND)
alcsi_s(c,i) = Sum(s,SRC,S1(c,s,i)*alcsi(c,s,i));

! Demands for composite primary factors and composite intermediate inputs !
! The top nest of the production functions is Leontief: !
! Z(i) = MIN [ All,c,COM: X1_S(c,i)/A1_S(c,i), X1PRIM(i)/A1PRIM(i) ] !

Equation E_x1_s # Intermediate demands for commodity composites #
(All,c,COM)(All,i,IND)
x1_s(c,i) - al_s(c,i) = z(i);

Equation E_xlprim # Demands for primary factor composite #
(All,i,IND)
x1prim(i) - alprim(i) = z(i);

```

```

! Commodity supplies !
! Producers in multiproduct industries choose their commodity outputs !
! to maximize revenue !
! Sum(c,COM, (P0(c,dom)/A0(c,i))*(X0(c,i)*A0(c,i)) ) !
! subject to CET transformation frontiers: !
! Z(i) = CET [ All,c,COM: x0(c,i)/A0(c,i) ] !

Equation E_x0 # Supplies of commodities by industries #
(All,c,COM)(all,i,IND)
x0(c,i) + a0(c,i) = z(i) + SIGMA0(i)*(p0(c,"dom") - a0(c,i) - p0_e(i));

Equation E_p0_e # Effective price for the commodity supply equations #
(All,i,IND)
MAKE_C(i)*p0_e(i) = Sum(c,COM,MAKE(c,i)*(p0(c,"dom") - a0(c,i)));

! Subsection 8.2: Creation of capital !

! Investors choose source-specific inputs to capital formation to !
! minimize costs !
! Sum(s,SRC, X2(c,s,i)*P2(c,s,i) ) !
! subject to CES aggregation functions: !
! X2_S(c,i) = CES [ All,s,SRC: X2(c,s,i) ] !

Equation E_x2 # Source-specific demands for inputs to capital creation #
(All,c,COM)(All,s,SRC)(All,i,IND)
x2(c,s,i) = x2_s(c,i) - SIGMA2(c)*{p2(c,s,i) - p2_s(c,i)}
+ a2csi(c,s,i) - SIGMA2(c)*{a2csi(c,s,i) - a2csi_s(c,i)};

Equation E_p2_s # Effective price term for inputs-to-capital equation #
(All,c,COM)(All,i,IND)
p2_s(c,i) = Sum(s,SRC,S2(c,s,i)*p2(c,s,i));

Equation E_a2csi_s # Effective price of composite intermediate input #
(All,c,COM)(All,i,IND)
a2csi_s(c,i) = Sum(s,SRC,S2(c,s,i)*a2csi(c,s,i));

! The top nest of the capital production functions is Leontief: !
! X2TOT(i) = MIN [ All,c,COM: X2_S(c,i) ] !

Equation E_x2_s # Demands for composite commodities for use in capital creation#
(All,c,COM)(All,i,IND)
x2_s(c,i) - a2_s(c,i) = x2tot(i);

! Subsection 8.3: Household demands !

! Households choose source-specific supplies of commodities !
! to minimize composite-commodity costs !
! Sum(s,SRC, X3(c,s)*P3(c,s)) !
! subject to CES aggregation functions: !
! X3_S(c) = CES [ All,s,SRC: X3(c,s) ] !

Equation E_x3 # Source-specific commodity demands by household #
(All,c,COM)(All,s,SRC)
x3(c,s) = x3_s(c) - SIGMA3(c)*{p3(c,s) - p3_s(c)}
+ a3cs(c,s) - a3cs_s(c) - SIGMA3(c)*{a3cs(c,s) - a3cs_s(c)} ;

Equation E_p3_s # Effective price term of dom/imp-household demand equation #
(All,c,COM)
p3_s(c) = Sum(s,SRC,S3(c,s)*p3(c,s));

```

```

Equation E_a3cs_s # Effective price of composite intermediate input #
  (All,c,COM)
  a3cs_s(c) = Sum(s, SRC, S3(c,s)*a3cs(c,s));

! Commodity composition of household demand undifferentiated by source. !
! Households choose commodities to maximize a Stone-Geary utility function !
! Sum(c,COM,DELTA(c)*Ln[X3_S(c)/Q - GAMMA(c)] !
! subject an aggregate-expenditure constraint !
! X3TOT = Sum(c,COM, P3_S(c)*X3_S(c)) !
! This leads to the Linear Expenditure System !

Equation E_x3_s # Household demands for composite commodities #
  (All,c,COM)
  x3_s(c) - q = EPS(c)*(w3tot-q) + Sum(k,COM,ETA(c,k)*p3_s(k))
                                     + a3com(c) - ave_a3com;

Equation E_ave_a3com # Average value of a3com #
  ave_a3com = Sum(k,COM, S3_S(k)*a3com(k));

Equation E_deltapc
  # Movements in marginal budget shares in the linear expenditure system #
  (All,c,COM) deltapc(c) = a3com(c) - Sum(k,COM, DELTA(k)*a3com(k));

Equation E_d_gamma
  # Movements in subsistence variables in the linear expenditure system #
  (All,c,COM) d_gamma(c) = [1 + EPS(c)/FRISCH]*[a3com(c)-ave_a3com];

! Subsection 8.4: Export demands !

! Foreigners have constant-elast.-of-demand functions for the economy's exports!
Equation E_x4 # Export demand functions #
  (All,c,COM)
  x4(c) = EXP_ELAST(c)*[p4(c) + phi - f4p(c)- f4gen] + f4q(c);

! Subsection 8.5: Government demands !

Equation E_x5 # Government demand, undifferentiated by source #
  (All,c,COM)(All,s, SRC)
  x5(c,s) = SDOM(s)*f5(c) + f5gen;

! Subsection 8.6: Margin demands !

! Margins services are in fixed proportions to the transactions facilitated. !
Equation E_x1mar # Margins on inputs to production#
  (All,c,COM)(All,s, SRC)(All,i, IND)(All,m, MAR)
  x1mar(c,s,i,m) = x1(c,s,i) + almar(c,s,i,m);

Equation E_x2mar # Margins on inputs to capital creation #
  (All,c,COM)(All,s, SRC)(All,i, IND)(All,m, MAR)
  x2mar(c,s,i,m) = x2(c,s,i) + a2mar(c,s,i,m);

Equation E_x3mar # Margins on flows of commodities to households #
  (All,c,COM)(All,s, SRC)(All,m, MAR)
  x3mar(c,s,m) = x3(c,s) + a3mar(c,s,m);

Equation E_x4mar # Margins on flows of exports to ports of exit #
  (All,c,COM)(All,m, MAR)
  x4mar(c,m) = x4(c) + a4mar(c,m);

```

```

! Subsection 8.7: Zero pure profits !

! Producers, investors and importers earn zero pure profits. !
! With constant returns to scale, the zero-pure-profit conditions !
! can be written as relationships between price and technology variables !
! without quantity variables. !

Equation E_p0ind # Price of industry outputs #
(All,i,IND)
MAKE_C(i)*p0ind(i) = Sum(c,COM,MAKE(c,i)*p0(c,"dom"));

Equation E_z # Zero pure profits in production #
(All,i,IND)
V1TOT(i)*p0ind(i) = Sum(c,COM,V1PUR_S(c,i)*(p1_s(c,i)))
+ V1LAB(i)*p1lab(i) + V1CAP(i)*p1cap(i) + V1TOT(i)*a(i);

Equation E_a # Technical change by industry #
(All,i,IND)
V1TOT(i)*a(i) = Sum(c,COM,MAKE(c,i)*a0(c,i))
+ Sum(c,COM,V1PUR_S(c,i)*a1_s(c,i))
+ Sum(c,COM,Sum(s,SRC,V1PUR(c,s,i)*alcsi(c,s,i)))
+ V1LAB(i)*a1lab(i) + V1CAP(i)*a1cap(i) + V1PRIM(i)*a1prim(i);

Equation E_p2tot # Zero pure profits in capital creation #
(All,i,IND)
V2TOT(i)*p2tot(i) = Sum(c,COM,Sum(s,SRC,
V2PUR(c,s,i)*[p2_s(c,i) + a2_s(c,i)+a2csi(c,s,i)]));

Equation E_p0imp # Zero pure profits in importing #
(All,c,COM)
p0imp(c) = pf0cif(c) - phi + t0imp(c);

! Zero pure profits in movement of commodities. !
! Purchasers' prices are equal to sums of basic prices, margins costs and !
! indirect taxes. TINY takes care of potential zero divides. !

Equation E_p0 # Convenient definition of basic prices of commodities #
(All,c,COM)(All,s,SRC)
p0(c,s) = SDOM(s)*p0dom(c) + (1-SDOM(s))*p0imp(c);

Equation E_p1 # Purchasers prices of inputs to producers #
(All,c,COM)(All,s,SRC)(All,i,IND)
(V1PUR(c,s,i)+TINY)*p1(c,s,i)
= (V1BAS(c,s,i)+V1TAX(c,s,i)+TINY)*[p0(c,s)+t1(c,s,i)]
+ Sum(m,MAR,V1MAR(c,s,i,m)*[p0(m,"dom")+a1mar(c,s,i,m)]);

Equation E_p2 # Purchasers prices of inputs to capital creation #
(All,c,COM)(All,s,SRC)(All,i,IND)
(V2PUR(c,s,i)+TINY)*p2(c,s,i)
= (V2BAS(c,s,i)+V2TAX(c,s,i)+TINY)*[p0(c,s)+t2(c,s,i)]
+ Sum(m,MAR,V2MAR(c,s,i,m)*[p0(m,"dom")+a2mar(c,s,i,m)]);

Equation E_p3 # Purchasers prices for commodity flows to households #
(All,c,COM)(All,s,SRC)
(V3PUR(c,s)+TINY)*p3(c,s) = (V3BAS(c,s)+V3TAX(c,s)+TINY)*[p0(c,s)+t3(c,s)]
+ Sum(m,MAR,V3MAR(c,s,m)*[p0(m,"dom")+a3mar(c,s,m)]);

Equation E_p4 # Fob export prices #
(All,c,COM)
(V4PUR(c)+TINY)*p4(c) = (V4BAS(c)+V4TAX(c)+TINY)*[p0(c,"dom")+t4(c)]
+ Sum(m,MAR,V4MAR(c,m)*[p0(m,"dom")+a4mar(c,m)]);

```

```

Equation E_p5 # Effective price term of dom/imp-household demand equation #
  (All,c,COM) (All,s,SRC)
  p5(c,s) = p0(c,s);

! Subsection 8.8: Demands equal supplies !

! Supplies of domestic commodities are sums of supplies from industries !
Equation E_x0dom # Total output of domestic commodities #
  (All,c,COM)
  (MAKE_I(c)+TINY)*x0dom(c) = Sum(i,IND,MAKE(c,i)*x0(c,i));

! Supplies of domestic commodities equal demands !
! Margins and non-margins commodities are treated separately. !

Equation E_p0dom_NM # Demand equals supply for non margin commodities #
  (All,n,NONMAR)
  (SALES(n)+TINY)*x0dom(n) = Sum(i,IND,V1BAS(n,"dom",i)*x1(n,"dom",i)
    + V2BAS(n,"dom",i)*x2(n,"dom",i))
    + V3BAS(n,"dom")*x3(n,"dom") + V4BAS(n)*x4(n)+V5BAS(n,"dom")*x5(n,"dom");

Equation E_p0dom_M # Demand equals supply for margin commodities #
  (All,m,MAR)
  (SALES(m)+TINY)*x0dom(m) = Sum(i,IND,V1BAS(m,"dom",i)*x1(m,"dom",i)
    + V2BAS(m,"dom",i)*x2(m,"dom",i))
    + V3BAS(m,"dom")*x3(m,"dom") + V4BAS(m)*x4(m)+V5BAS(m,"dom")*x5(m,"dom")
    + Sum(c,COM,V4MAR(c,m)*x4mar(c,m) + Sum(s,SRC,V3MAR(c,s,m)*x3mar(c,s,m)
    + Sum(i,IND,V1MAR(c,s,i,m)*x1mar(c,s,i,m)+V2MAR(c,s,i,m)*x2mar(c,s,i,m)))));

Equation E_x0imp # Supplies of imports equal demands for imports #
  (All,c,COM)
  (V0IMP(c)+TINY)*x0imp(c) = Sum(i,IND,V1BAS(c,"imp",i)*x1(c,"imp",i)
    + V2BAS(c,"imp",i)*x2(c,"imp",i)) + V3BAS(c,"imp")*x3(c,"imp");

Equation E_cap_t # Demand equals supply for capital #
  (All,i,IND)
  x1cap(i) = cap_t(i);

Equation E_employ_i # Aggregate employment, wage bill weights #
  V1LAB_I*employ_i = Sum(i,IND,V1LAB(i)*x1lab(i));

! Subsection 8.9: Capital stocks, investment and rates of return !

! Capital investment accumulation relationship !
! The underlying levels equation is: !
! QCAP1(i) = QCAP(i)*(1-DEP(i)) + QINVEST(i) !

Equation E_cap_t1 # End-of-year capital stock #
  (All,i,IND)
  (QCAP1(i)+TINY)*cap_t1(i) = (1-DEP(i))*QCAP(i)*cap_t(i)+ QINVEST(i)*x2tot(i);

! In year-to-year simulations where the initial solution for year t is the !
! final solution for year t-1, the appropriate shock for the start-of-year !
! capital in year t is the difference between the end- and start-of-year !
! capital stocks in year t-1. The shock to the start-of-year capital in year t !
! is conveniently imposed by shocking del_unity by one in the following: !

Equation E_d_f_cap_t # Capital Stock for current production #
  (All,i,IND)
  QCAP(i)*cap_t(i) = 100*(QCAP1_B(i) - QCAP_B(i))*del_unity + 100*d_f_cap_t(i);

```

```

Equation E_x2tot # Investment/capital ratios by industry #
  (All,i,IND)
  x2tot(i) = x1cap(i) + r_inv_cap(i) + r_inv_cap_u;

! When the shift variables are exogenous, this equation imposes a positive !
! relationship between expected rates of return and growth in capital stock. !
! Details can be found in the MONASH book, section 21.1. !

Equation E_d_f_eeqr_ror_j # Change in expected rate of return in forecast year #
  (All,i,IND) del_ror(i) =
    (1/COEFF_SL(i))*[1/(K_GR(i)-K_GR_MIN(i))+1/(K_GR_MAX(i)-K_GR(i))]*del_k_gr(i)
    + d_f_eeqr_ror_j(i) + d_f_eeqr_ror;

Equation E_plcap # Changes in expected rors by industry: static exp. #
  (All,i,IND) 100*del_ror(i) =
    (1/(1 + RINT))*[V1CAP(i)/VCAP(i)]*[plcap(i) - p2tot(i)];

Equation E_del_k_gr # Capital growth through a year #
  (All,i,IND) del_k_gr(i) =
    [{QCAP1(i)/QCAP(i)}/100]*[cap_t1(i) - cap_t(i)];

Equation E_del_ff_rate # Allows for equalization of changes in rates of return #
  (All,i,IND) del_ror(i) = del_r_tot + del_ff_rate(i);

Equation E_f2tot_i # Can be used to lock agg. invest. & consumption together #
  x2tot_i = x3tot + f2tot_i ;

! Subsection 8.10: Government accounts !

! Government deficit !
Equation E_d_gov_def
  # Public sector deficit, or public sector financing transactions #
  100*d_gov_def = AGGOTH*w5tot + AGGINVG*agginv_g
    - NET_TAXTOTG*net_tax_tot - OTHGOVREV*oth_gov_rev + TRANS*transfers;

Equation E_d_def_gdp_r
  # Change in the ratio of the government deficit to GDP #
  100*d_def_gdp_r = (100/V0GDPEXP)*d_gov_def - (GOV_DEF/V0GDPEXP)*w0gdpepx;

Equation E_lev_def_gdp_r
  # Level of the ratio of the government deficit to GDP #
  lev_def_gdp_r = R_DEFGDP_B*del_unity + d_def_gdp_r;

! Government investment !
Equation E_agginv_g # In each ind, Gov invest is a fixed share of total invest #
  AGGINVG*agginv_g = Sum(i,IND,G_VINVEST(i)*(p2tot(i)+x2tot(i)));

Equation E_agginv_rg # Aggregate real public investment #
  AGGINVG*agginv_rg = Sum(j,IND,G_VINVEST(j)*x2tot(j));

Equation E_p2tot_g # Price index for government investment expenditure #
  AGGINVG*p2tot_g = Sum(i,IND,G_VINVEST(i)*p2tot(i));

! Government revenue net of subsidies !
Equation E_net_tax_tot
  # Net collection of indirect taxes and income taxes #
  NET_TAXTOTG*net_tax_tot
    = V0TAX_CSI*taxindg + INCTAX*taxrev_inc ;

```

```

Equation E_taxindg
V0TAX_CSI*taxindg = V1TAX_CSI*w1tax_csi + V2TAX_CSI*w2tax_csi
+ V3TAX_CS*w3tax_cs + 100*d_w4tax_c+ V0TAR_C*w0tar_c;

Equation E_d_net_tax_gdp # Net tax to GDP ratio #
100*d_net_tax_gdp = (NET_TAXTOTG/V0GDPEXP)*(net_tax_tot - w0gdpepx);

Equation E_taxrev_inc # Income tax revenue #
INCTAX*taxrev_inc = TAX_LAB*(w1lab_i + tax_l_r ) + TAX_CAP*(w1cap_i+tax_k_r) ;

Equation E_tax_k_r
# Rate of tax on capital income related to rate of tax on labour income #
tax_k_r = tax_l_r + f_tax_r;

Equation E_oth_gov_rev
# Other government revenue, e.g. income from public enterprises #
OTHGOVREV*oth_gov_rev = Sum(i,GOVT,V1CAP(i)*(x1cap(i)+p1cap(i)) );
;

! Transfers from Government !
Equation E_transfers # Transfers from the government #
TRANS*transfers = BENEFITS*bens + 100*d_net_int_g;

Equation E_bens # Social security paid by government #
bens = pop + p3tot + real_wage_c ;

Equation E_d_net_int_g # Net interest payments from government #
d_net_int_g = PSDATT*d_int_psd + INT_PSD*d_psd_t;

Equation E_d_int_psd # Nominal rate of interest on public sector debt #
d_int_psd = (1+INF)*d_rint_psd + (1+RINT_PSD)*d_inf;

Equation E_d_inf # Rate of inflation #
100*d_inf =(1+INF)*(p3tot - p3tot_1);

Equation E_d_f_p3tot_1
# Lagged value of the CPI if initial sol for year t is sol for year t-1 #
LEV_CPI_L*p3tot_1 = 100*(LEV_CPI_B - LEV_CPI_L_B)*del_unity + 100*d_f_p3tot_1;

Equation E_d_rint_psd
# Link between real rates of interest on PSD and business borrowing #
d_rint_psd = d_rint ;

Equation E_d_f_psd_t
# Gives shock to start-of-year public sector debt, yr-to-yr sims #
d_psd_t = (PSDATT_1_B - PSDATT_B)*del_unity + d_f_psd_t;

Equation E_d_r_psdgdp
# Ratio of st-of-yr public sector debt to GDP #
d_r_psdgdp = (1/V0GDPEXP)*d_psd_t - 0.01*R_PSDGDP*w0gdpepx;

Equation E_lev_r_psdgdp
# Level of ratio of st-of-yr public sector debt to GDP #
lev_r_psdgdp = R_PSDGDP_B*del_unity + d_r_psdgdp;

! Subsection 8.11: Current account deficit and foreign debt !
Equation E_d_cad # Current account of the balance of payments #
100*d_cad = V0CIF_C*w0cif_c - V4TOT*w4tot + 100*d_int_fd ;

```



```

Equation E_d_int_fd
# $US value of interest on f'gn debt repayable in f'gn curr. #
d_int_fd = ROIFOREIGN*d_fd_t; ! Interest rate on fgn debt is a parameter !

Equation E_d_f_fd_t
# Gives shock to start-of-year f'gn debt in year-to-year sims #
d_fd_t = (FDATT_1_B - FDATT_B)*del_unity + d_f_fd_t;

! Subsection 8.12: GNP, household disposable income and saving, and
propensities to save and consume !

Equation E_gnpnom # Gross national product #
GNP*gnpnom = V0GDPEXP*w0gdpexp - 100*d_int_fd;

Equation E_hdy # Household disposable income #
HOUS_DIS_INC*hd_y = V0GDPEXP*w0gdpexp + TRANS*transfers - NET_TAXTOTG*net_tax_tot
- OTHGOVREV*oth_gov_rev - 100*d_int_fd;

Equation E_d_housav # Household saving #
100*d_housav = HOUS_DIS_INC*hd_y - V3TOT*w3tot;

Equation E_apc # Consumption related to disposable income #
V3TOT*w3tot = AV_PROP_CON*HOUS_DIS_INC*(apc + hd_y);

Equation E_x3tot # Relates total private consumption to GNP #
w3tot = gnpnom + apc_gnp;

Equation E_aps_gnp # Consumption and saving propensities add to one #
0 = V3TOT*apc_gnp + (GNP - V3TOT)*aps_gnp ;

! Subsection 8.13: Employment and sticky wage adjustment !

Equation E_del_f_wage_c
# Relates deviation in CPI-deflated pre-tax wage to deviation in employment #
(RWAGE/RWAGE_OLD)*(real_wage_c - real_wage_c_o) =
100*((RWAGE_B/RWAGE_OLD_B) - (RWAGE_L_B/RWAGE_O_L_B))*del_unity
+ ALPHA1*(EMPLOY/EMPLOY_OLD)*(emp_hours - emp_hours_o) + del_f_wage_c;

Equation E_fllab_i # Aggregate employment, hours #
HOURSTOT*emp_hours = Sum(i,IND, HOURS(i)*xllab(i));

Equation E_real_wage_c # Economy-wide real wage rate for consumers #
V1LAB_I*[real_wage_c + p3tot] = Sum(i,IND, V1LAB(i)*p1lab(i));

Equation E_real_wage_c_o
# Introduces forecast CPI-deflated pre-tax wage into policy simulation #
real_wage_c_o = real_wage_c + f_rwage_o;

Equation E_emp_hours_o # Introduces forecast employment into policy simulation #
emp_hours_o = emp_hours + f_emp_o;

Equation E_p1lab # Can be used to vary wage movements across industries #
(All,i,IND)
p1lab(i) = p3tot + fllab_i + fllab(i);

! Subsection 8.14: Powers of taxes and tax collections !

! Equations to allow commod.-specific and user-specific shifts in tax rates !
Equation E_t1 # Power of tax on sales to intermediate #
(All,c,COM)(All,s,SRC)(All,i,IND)
t1(c,s,i) = f0tax_s(c) + f1tax_csi;

```

```

Equation E_t2 # Power of tax on sales to investment #
  (All,c,COM)(All,s,SRC)(All,i,IND)
  t2(c,s,i) = f0tax_s(c) + f2tax_csi;

Equation E_t3 # Power of tax on sales to households #
  (All,c,COM)(All,s,SRC)
  t3(c,s) = f0tax_s(c) + f3tax_cs;

Equation E_t4 # Power of tax on exports #
  (All,c,COM)
  t4(c) = f0tax_s(c) + f4tax(c) + f4tax_c;

Equation E_wltax_csi # Revenue from indirect taxes on flows to intermediate #
  V1TAX_CSI*wltax_csi = Sum(c,COM, Sum(s,SRC, Sum(i,IND,
    V1TAX(c,s,i)*{p0(c,s)+x1(c,s,i)} + {V1TAX(c,s,i)+V1BAS(c,s,i)}*t1(c,s,i)))));

Equation E_w2tax_csi # Revenue from indirect taxes on flows to investment #
  V2TAX_CSI*w2tax_csi = Sum(c,COM, Sum(s,SRC, Sum(i,IND,
    V2TAX(c,s,i)*{p0(c,s)+x2(c,s,i)} + {V2TAX(c,s,i)+V2BAS(c,s,i)}*t2(c,s,i)))));

Equation E_w3tax_cs # Revenue from indirect taxes on flows to households #
  [V3TAX_CS+TINY]*w3tax_cs = Sum(c,COM, Sum(s,SRC,
    V3TAX(c,s)*{p0(c,s)+x3(c,s)} + {V3TAX(c,s)+V3BAS(c,s)}*t3(c,s)))));

Equation E_w4tax_c # Revenue from indirect taxes on exports #
  100*d_w4tax_c = Sum(c,COM,
    V4TAX(c)*{p0(c,"dom")+x4(c)} + {V4TAX(c)+ V4BAS(c)}*t4(c));

Equation E_w0tar_c # Tariff revenue #
  V0TAR_C*w0tar_c = Sum(c,COM, V0TAR(c)*{pf0cif(c) - phi + x0imp(c)}
    + V0IMP(c)*t0imp(c));

! Subsection 8.15: Nominal and real GDP from the income side !

Equation E_wllab_i # Aggregate payments to labour #
  V1LAB_I*wllab_i = Sum(i,IND,V1LAB(i)*{xllab(i)+p1lab(i)});

Equation E_wlcap_i # Aggregate payments to capital #
  V1CAP_I*wlcap_i = Sum(i,IND,V1CAP(i)*{xlcap(i)+p1cap(i)});

Equation E_w0tax_csi # Aggregate value of indirect taxes #
  V0TAX_CSI*w0tax_csi = V1TAX_CSI*wltax_csi + V2TAX_CSI*w2tax_csi
    + V3TAX_CS*w3tax_cs + 100*d_w4tax_c + V0TAR_C*w0tar_c;

Equation E_w0gdpinc # Aggregate nominal GDP from income side #
  V0GDPINC*w0gdpinc = V1CAP_I*wlcap_i + V1LAB_I*wllab_i + V0TAX_CSI*w0tax_csi;

Equation E_a0_c # Output-augmenting technical change by industry #
  (All,i,IND)
  MAKE_C(i)*a0_c(i) = Sum(c,COM, MAKE(c,i)*a0(c,i));

Equation E_x0gdpinc # Real GDP from income side #
  V0GDPINC*x0gdpinc = V1CAP_I*xlcap_i + V1LAB_I*employ_i
    - Sum(i,IND, V1TOT(i)*a(i))
    - Sum(m,MAR, Sum(c,COM,Sum(s,SRC,
      Sum(i,IND, V1MAR(c,s,i,m)*almar(c,s,i,m)+V2MAR(c,s,i,m)*a2mar(c,s,i,m))
      + V3MAR(c,s,m)*a3mar(c,s,m) ) + V4MAR(c,m)*a4mar(c,m)))
    - Sum(c,COM,Sum(i,IND,V2PUR_S(c,i)*a2_s(c,i)))
    - Sum(c,COM,Sum(s,SRC, Sum(i,IND,V2PUR(c,s,i)*a2csi(c,s,i))))
    + Sum{c,COM,Sum[s,SRC,Sum(i,IND, V1TAX(c,s,i)*x1(c,s,i)+V2TAX(c,s,i)*x2(c,s,i))
      + V3TAX(c,s)*x3(c,s) ] + V4TAX(c)*x4(c)+V0TAR(c)*x0imp(c) } ;

```

! Subsection 8.16: Nominal and real GDP from the expenditure side *!*
! and quantity and price indexes, expenditure aggregates *!*

```
Equation E_x2tot_i # Total real investment #
  V2TOT_I*x2tot_i = Sum(i,IND,V2TOT(i)*x2tot(i));

Equation E_p2tot_i # Investment price index #
  V2TOT_I*p2tot_i = Sum(i,IND,V2TOT(i)*p2tot(i));

Equation E_w2tot_i # Total nominal investment #
  w2tot_i = x2tot_i + p2tot_i;

Equation E_p3tot # Consumer price index #
  V3TOT*p3tot = Sum(c,COM,Sum(s,SRC,V3PUR(c,s)*p3(c,s)));

Equation E_w3tot # Household expenditure (budget constraint) #
  w3tot = x3tot + p3tot;

Equation E_x4tot # Export volume index #
  V4TOT*x4tot = Sum(c,COM,V4PUR(c)*x4(c));

Equation E_p4tot # Exports price index, $US #
  V4TOT*p4tot = Sum(c,COM,V4PUR(c)*p4(c));

Equation E_w4tot # $US border value (f.o.b) of exports #
  w4tot = x4tot + p4tot;

Equation E_x5tot # Government volume index #
  V5TOT*x5tot = Sum(c,COM,Sum(s,SRC, V5BAS(c,s)*x5(c,s)));

Equation E_p5tot # Government price index #
  V5TOT*p5tot = Sum(c,COM,Sum(s,SRC, V5BAS(c,s)*p5(c,s)));

Equation E_w5tot # Government expenditure #
  w5tot = x5tot + p5tot;

Equation E_x0cif_c # C.i.f. import volume index #
  V0CIF_C*x0cif_c = Sum(c,COM,V0CIF(c)*x0imp(c));

Equation E_p0cif_c # $US c.i.f. imports, price index #
  V0CIF_C*p0cif_c = Sum(c,COM,V0CIF(c)*{-phi+pf0cif(c)});

Equation E_w0cif_c # $US c.i.f. value of imports #
  w0cif_c = x0cif_c + p0cif_c;

Equation E_x0gdpexp # Real GDP, expenditure side #
  V0GDPEXP*x0gdpexp = V3TOT*x3tot + V2TOT_I*x2tot_i
  + V4TOT*x4tot + V5TOT*x5tot - V0CIF_C*x0cif_c;

Equation E_p0gdpexp # Price index for GDP, expenditure side #
  V0GDPEXP*p0gdpexp = V3TOT*p3tot + V2TOT_I*p2tot_i
  + V4TOT*p4tot +V5TOT*p5tot - V0CIF_C*p0cif_c;

Equation E_w0gdpexp # Nominal GDP from expenditure side #
  w0gdpexp = x0gdpexp + p0gdpexp;

Equation E_del_b # Change in the balance of trade #
  100*del_b = V4TOT*w4tot - V0CIF_C*w0cif_c;
```

```

Equation E_del_b_gdp # (Balance of trade)/GDP #
100*V0GDPEXP*del_b_gdp = V4TOT*w4tot - V0CIF_C*w0cif_c
- (V4TOT-V0CIF_C)*w0gdpepx;

Equation E_x0imp_c # Import volume index, duty-paid weights #
V0IMP_C*x0imp_c = Sum(c,COM,V0IMP(c)*x0imp(c));

Equation E_p0imp_c # Duty-paid imports, price index #
V0IMP_C*p0imp_c = Sum(c,COM,V0IMP(c)*p0(c,"imp"));

Equation E_w0imp_c # Value of imports, duty-paid #
w0imp_c = x0imp_c + p0imp_c;

Equation E_xlcap_i # Aggregate usage of capital, rental weights #
V1CAP_I*xlcap_i = Sum(i,IND,V1CAP(i)*xlcap(i));

Equation E_plcap_i # Average capital rental #
V1CAP_I*plcap_i = Sum(i,IND,V1CAP(i)*plcap(i));

Equation E_p0toft # Terms of trade #
p0toft = p4tot - p0cif_c;

Equation E_p0realdev # Real devaluation #
p0realdev = p0cif_c - p0gdpepx;

! Subsection 8.17: Equation to facilitate historical and/or forecast sims !

Equation E_twist_src # Import/domestic twist and shift #
(All,c,COM)
twist_src(c) = twist_c+ ftwist_src(c);

Equation E_f_alprim # Used to impose uniform shifts in prim fac tech change #
(All,i,IND)
alprim(i) = f_alprim(i) + ff_alprim;

Equation E_a_cont # Contribution of tech change to GDP #
a_cont = -(1/V0GDPINC)*{ Sum(i,IND, V1TOT(i)*a(i))
+ Sum(m,MAR, Sum(c,COM,Sum(s,SRC,
Sum(i,IND, V1MAR(c,s,i,m)*almar(c,s,i,m)+V2MAR(c,s,i,m)*a2mar(c,s,i,m))
+ V3MAR(c,s,m)*a3mar(c,s,m) ) + V4MAR(c,m)*a4mar(c,m)))
+ Sum(c,COM,Sum(i,IND,V2PUR_S(c,i)*a2_s(c,i)))
+ Sum(c,COM,Sum(s,SRC, Sum(i,IND,V2PUR(c,s,i)*a2csi(c,s,i)))));

Equation E_twistlk # Labour/capital twist #
(All,i,IND)
twistlk(i) = twist_i+ ftwistlk(i);

Equation E_t0imp # Allows uniform tariff changes #
(All,c,COM)
t0imp(c) = f_t0imp(c) + ff_t0imp;

Equation E_al_s # Allows equilization of tech change in intermediate uses of c #
(All,c,COM)(All,i,IND)
al_s(c,i) = ac(c)+ falc(c) + falci(c,i);

Equation E_a2_s # Allows equilization of tech change in investment uses of c #
(All,c,COM)(All,i,IND)
a2_s(c,i) = ac(c)+ fa2c(c) + fa2ci(c,i);

```

Equation E_almar

Allows equilization of tech change in intermediate margin uses of m #
(All,c,COM)(All,s,SRC)(All,i,IND)(All,m,MAR)
almar(c,s,i,m) = ac(m)+ falmarc(m) + falmar(c,s,i,m);

Equation E_a2mar

Allows equilization of tech change in investment margin uses of m #
(All,c,COM)(All,s,SRC)(All,i,IND)(All,m,MAR)
a2mar(c,s,i,m) = ac(m)+ fa2marc(m) + fa2mar(c,s,i,m);

Equation E_a3mar

Allows equilization of tech change in consumption uses of c #
(All,c,COM)(All,s,SRC)(All,m,MAR)
a3mar(c,s,m) = ac(m)+ fa3marc(m) + fa3mar(c,s,m);

Equation E_a4mar

Allows equilization of tech change in export uses of c #
(All,c,COM)(All,m,MAR)
a4mar(c,m) = ac(m)+ fa4marc(m) + fa4mar(c,m);

! Subsection 8.18: Equation for converting twists into tech/taste changes !

! In forecasting it is convenient to use twist variables to accommodate information on imports and on primary factor inputs. The twists on imports, for example, allow exogenously specified growth in aggregate imports to be accommodated by a cost neutral change in the import/domestic quantity share in all uses of imports. The cost neutral changes are equivalent to domestic- and import-saving technical and taste changes. These are calculated in the equations below when the shift variables are set exogenously on zero. In policy we usually endogenize the shift variables and exogenize the technical and taste changes. This ensures that the technical and taste changes in policy are the same as those in forecast. If instead we exogenize the twists in policy, then we impart different tech changes in policy than in forecast. This is because import/domestic shares are different in policy than in forecast. With different tech changes in policy than in forecast, interpretation of the policy results becomes difficult. !

Equation E_alcsi_dom # Converts twist into tech change, intermed. domestic

(All,c,COM)(All,i,IND)
alcsi(c,"dom",i) = [S1(c,"imp",i)/(SIGMA1(c)-1)]*twist_src(c)
+ f_alcsi(c,"dom",i);

Equation E_alcsi_imp # Converts twist into tech change, intermed. imported

(All,c,COM)(All,i,IND)
alcsi(c,"imp",i) = -[S1(c,"dom",i)/(SIGMA1(c)-1)]*twist_src(c)
+ f_alcsi(c,"imp",i);

Equation E_a2csi_dom # Converts twist into tech change, cap creat. domestic

(All,c,COM)(All,i,IND)
a2csi(c,"dom",i) = [S2(c,"imp",i)/(SIGMA2(c)-1)]*twist_src(c)
+ f_a2csi(c,"dom",i);

Equation E_a2csi_imp # Converts twist into tech change, cap creat. imported

(All,c,COM)(All,i,IND)
a2csi(c,"imp",i) = -[S2(c,"dom",i)/(SIGMA2(c)-1)]*twist_src(c)
+ f_a2csi(c,"imp",i);

Equation E_a3cs_dom # Converts twist into taste change, H'hold domestic

(All,c,COM)
a3cs(c,"dom") = [S3(c,"imp")/(SIGMA3(c)-1)]*twist_src(c) + f_a3cs(c,"dom");

```

Equation E_a3cs_imp # Converts twist into taste change, H'hold imported #
  (All,c,COM)
  a3cs(c,"imp") = -[S3(c,"dom")/(SIGMA3(c)-1)]*twist_src(c) + f_a3cs(c,"imp");

Equation E_allab # Converts twist into tech change, production labour #
  (All,i,IND)
  allab(i) = [V1CAP(i)/(V1PRIM(i)*(SIGMA1PRIM(i)-1))]*twistlk(i) + f_allab(i);

Equation E_alcap # Converts twist into tech change, production capital #
  (All,i,IND)
  alcap(i) = -[V1LAB(i)/(V1PRIM(i)*(SIGMA1PRIM(i)-1))]*twistlk(i) + f_alcap(i);

!*****!
! Note that the next section is designed for investigating the Divisia !
! problem. It can be ignored by uses of Mini-USAGE who are not concerded !
! with this problem. !
!*****!

! Subsection 8.19: Equation for facilitating the computation of policy
! deviations in comparative static price and quantity indexes !
! Note. Deviation results for Laspeyres and Paasche quantity and price indexes
! appear as year-on-year growth rates in the policy simulation !

Coefficient
DUM_YEAR1 # Zero in first year, then one RU#;
(Parameter) ADJDUMYEAR1 # Adjusts DUM_YEAR1, one in first year, then zero FIu#;
Read
DUM_YEAR1 from file SRF Header"0045";
Formula
(Initial) ADJDUMYEAR1 = 0.0 + If( DUM_YEAR1 LT 0.1, 1.0);
Variable
(Change)d_dum_year1 # One in year one, zero in later years #;
Update
(Change) DUM_YEAR1 = d_dum_year1;

Equation E_d_dum_year1 # One in year one, zero in later years #
  d_dum_year1 = ADJDUMYEAR1*del_unity;

! Household consumption quantity and price indexes !
Coefficient
(All,c,COM)(All,s,SRC)
  LEVX3(c,s) # Quantities of consumption by commodity/source #;
(Parameter)(All,c,COM)(All,s,SRC)
  LEVX3B(c,s) # Quantities of consumption by comm/source, base #;
(Parameter)(All,c,COM)(All,s,SRC)
  LEVX3B_O(c,s) # Quantities of cons by com/src in f'cast, base #;
(All,c,COM)(All,s,SRC)
  LEVX3_O(c,s) # Quantities of cons by com/src in f'cast #;
(All,c,COM)(All,s,SRC)
  LEVP3(c,s) # Price of consumption by commodity/source #;
(Parameter)(All,c,COM)(All,s,SRC)
  LEVP3B(c,s) # Price of consumption by commodity/source, base #;
(Parameter)(All,c,COM)(All,s,SRC)
  LEVP3B_O(c,s) # Price of cons by com/src in f'cast, base #;

```

```

(All,c,COM)(All,s,SRC)
    LEVP3_O(c,s) # Price of cons by com/src in f'cast #;
(All,c,COM)(All,s,SRC)
    B3SH(c,s) # Shares in H'hold consumption #;
(All,c,COM)(All,s,SRC)
    B3SHO(c,s) # Shares in H'hold consumption, forecast #;
(Parameter)(All,c,COM)(All,s,SRC)
    B3SHBO(c,s) # Shares in H'hold consumption, forecast base #;
(Parameter)
    I_3QL # Household consumption, Laspeyres quantity #;
(Parameter)
    I_3PL # Household consumption, Laspeyres price #;
(Parameter)
    I_3QP # Household consumption, Paasche quantity #;
(Parameter)
    I_3PP # Household consumption, Paasche price #;
    V3_T # Household consumption, value #;
    V3_T_O # Household consumption, forecast value #;
(Parameter)
    V3_TB # Household consumption, value, base #;
(Parameter)
    V3_TB_O # Household consumption, forecast value, base #;
(Parameter)
    DEV_3V # Household consumption, deviation value #;

```

Read

```

LEVX3 from file EXTRA Header "LVX3";
LEVX3_O from file EXTRA3 Header "LVX3";
B3SH from file EXTRA Header "B3SH";
B3SHO from file EXTRA3 Header "B3SH";
LEVP3 from file EXTRA Header "LVP3";
LEVP3_O from file EXTRA3 Header "LVP3";
V3_T from file EXTRA Header "V3TT";
V3_T_O from file EXTRA3 Header "V3TT";

```

Formula

```

(Initial)(All,c,COM)(All,s,SRC) LEVX3B(c,s) =LEVX3(c,s);
(Initial)(All,c,COM)(All,s,SRC) LEVX3B_O(c,s) = LEVX3_O(c,s);
(Initial)(All,c,COM)(All,s,SRC) B3SH(c,s) =
    (1-DUM_YEAR1)*{V3PUR(c,s)/V3TOT} + DUM_YEAR1*B3SH(c,s);
(Initial)(All,c,COM)(All,s,SRC) B3SHO(c,s) =
    (1-DUM_YEAR1)*{V3PUR(c,s)/V3TOT} + DUM_YEAR1*B3SHO(c,s);
(Initial)(All,c,COM)(All,s,SRC) B3SHBO(c,s) = B3SHO(c,s);
(Initial)(All,c,COM)(All,s,SRC) LEVP3B(c,s) =LEVX3(c,s);
(Initial)(All,c,COM)(All,s,SRC) LEVP3B_O(c,s) = LEVP3_O(c,s);
(Initial) I_3QL=Sum(c,COM,Sum(s,SRC,
    B3SHBO(c,s)*(LEVX3B(c,s)-LEVX3B_O(c,s))/LEVX3B_O(c,s)));
(Initial) I_3PL=Sum(c,COM,Sum(s,SRC,
    B3SHBO(c,s)*(LEVP3B(c,s)-LEVP3B_O(c,s))/LEVP3B_O(c,s)));
(Initial) V3_T = (1-DUM_YEAR1)*V3TOT + DUM_YEAR1*V3_T;
(Initial) V3_T_O = (1-DUM_YEAR1)*V3TOT + DUM_YEAR1*V3_T_O;
(Initial) V3_TB = V3_T;
(Initial) V3_TB_O = V3_T_O;
(Initial)DEV_3V= (V3_TB - V3_TB_O)/V3_TB_O;
(Initial)I_3PP = (1+DEV_3V)/(1+I_3QL)-1;
(Initial)I_3QP = (1+DEV_3V)/(1+I_3PL)-1;

```

Variable

```

(All,c,COM)(All,s,SRC)
    x3_o(c,s) # Consumption by com/source in forecast simulation #;
(All,c,COM)(All,s,SRC)
    f_x3_o(c,s) # Allows exogenization of forecast cons #;
(All,c,COM)(All,s,SRC)
    b3shr_o(c,s) # Consumption shares, forecast # ;
(All,c,COM)(All,s,SRC)
    b3shr(c,s) # Consumption shares # ;
(All,c,COM)(All,s,SRC)
    f_b3shr_o(c,s) # Allows exog. of forecast budget shares #;

```

```

        x_3la # Laspeyres quantity index for consumption #;
        x_3pa # Paasche quantity index for consumption #;
(All,c,COM)(All,s,SRC)
        p3_o(c,s) # Cons. prices by commodity/source in forecast #;
(All,c,COM)(All,s,SRC)
        f_P3_o(c,s) # Allows exogenization of forecast cons prices #;
        p_3la # Laspeyres price index for consumption #;
        p_3pa # Paasche price index for consumption #;
        w3tot_o # Value of consumption, forecast # ;
        f_w3tot_o # Allows exog. of value of consumption, forecast #;

```

Update

```

(All,c,COM)(All,s,SRC) LEVX3(c,s) =x3(c,s);
(All,c,COM)(All,s,SRC) LEVX3_O(c,s) =x3_o(c,s);
(All,c,COM)(All,s,SRC) B3SH(c,s) =b3shr(c,s);
(All,c,COM)(All,s,SRC) B3SHO(c,s) =b3shr_o(c,s);
V3_T = w3tot;
V3_T_O = w3tot_o;
(All,c,COM)(All,s,SRC) LEVP3(c,s) =P3(c,s);
(All,c,COM)(All,s,SRC) LEVP3_O(c,s) =P3_o(c,s);

```

Equation E_x_3la

```

# Laspeyres quantity index for consumption #
x_3la = 100*I_3QL*del_unity;

```

Equation E_x_3pa

```

# Paasche quantity index for consumption #
x_3pa = 100*I_3QP*del_unity;

```

Equation E_p_3la

```

# Laspeyres price index for consumption#
p_3la = 100*I_3PL*del_unity;

```

Equation E_p_3pa

```

# Paasche price index for consumption #
p_3pa = 100*I_3PP*del_unity;

```

Equation E_b3shr # Budget shares in consumption

```

(All,c,COM)(All,s,SRC) b3shr(c,s) = x3(c,s)+p3(c,s) - w3tot;

```

Equation E_x3cs_o

```

# Introduces forecasts of consumption by com/source into policy sims. #
(All,c,COM)(All,s,SRC) x3_o(c,s) = x3(c,s) + f_x3_o(c,s);

```

Equation E_b3sh_o

```

# Introduces forecasts of budget shares by com/source into policy sims. #
(All,c,COM)(All,s,SRC) b3shr_o(c,s) = b3shr(c,s)+ f_b3shr_o(c,s);

```

Equation E_P3cs_o

```

# Introduces forecasts of consumer prices into policy sims. #
(All,c,COM)(All,s,SRC) p3_o(c,s) = p3(c,s) + f_p3_o(c,s);

```

Equation E_w3tot_o

```

# Introduces forecasts of nominal consumption into policy sims. #
w3tot_o = w3tot + f_w3tot_o;

```

! Government consumption quantity and price indexes!

Coefficient

```

(All,c,COM)(All,s,SRC)
        LEVX5(c,s) # Quantities of Govt cons by commodity/source #;

```



```

(Parameter)(All,c,COM)(All,s,SRC)
    LEVX5B(c,s) # Quantities of Govt cons by comm/source, base #;
(Parameter)(All,c,COM)(All,s,SRC)
    LEVX5B_O(c,s) # Quantities of Govt cons in f'cast, base #;
(All,c,COM)(All,s,SRC)
    LEVX5_O(c,s) # Quantities of Govt cons by com/src in f'cast #;
(All,c,COM)(All,s,SRC)
    LEVP5(c,s) # Price of Govt cons by commodity/source #;
(Parameter)(All,c,COM)(All,s,SRC)
    LEVP5B(c,s) # Price of Govt cons by commodity/source, base #;
(Parameter)(All,c,COM)(All,s,SRC)
    LEVP5B_O(c,s) # Price of Govt cons by com/src in f'cast, base #;
(All,c,COM)(All,s,SRC)
    LEVP5_O(c,s) # Price of Govt cons by com/src in f'cast #;
(All,c,COM)(All,s,SRC)
    B5SH(c,s) # Shares in Govt consumption #;
(All,c,COM)(All,s,SRC)
    B5SHO(c,s) # Shares in Govt consumption, forecast #;
(Parameter)(All,c,COM)(All,s,SRC)
    B5SHBO(c,s) # Shares in Govt consumption, forecast base #;
(Parameter)
    I_5QL # Govt consumption, Laspeyres quantity #;
(Parameter)
    I_5PL # Govt consumption, Laspeyres price #;
(Parameter)
    I_5QP # Govt consumption, Paasche quantity #;
(Parameter)
    I_5PP # Govt consumption, Paasche price #;
    V5_T # Govt consumption, value #;
    V5_T_O # Govt consumption, forecast value #;
(Parameter)
    V5_TB # Govt consumption, value, base #;
(Parameter)
    V5_TB_O # Govt consumption, forecast value, base #;
(Parameter)
    DEV_5V # Govt consumption, deviation value #;

```

Read

```

LEVX5 from file EXTRA Header "LVX5";
LEVX5_O from file EXTRA3 Header "LVX5";
B5SH from file EXTRA Header "B5SH";
B5SHO from file EXTRA3 Header "B5SH";
LEVP5 from file EXTRA Header "LVP5";
LEVP5_O from file EXTRA3 Header "LVP5";
V5_T from file EXTRA Header "V5TT";
V5_T_O from file EXTRA3 Header "V5TT";

```

Formula

```

(Initial)(All,c,COM)(All,s,SRC) LEVX5B(c,s) =LEVX5(c,s);
(Initial)(All,c,COM)(All,s,SRC) LEVX5B_O(c,s) = LEVX5_O(c,s);
(Initial)(All,c,COM)(All,s,SRC) B5SH(c,s) =
    (1-DUM_YEAR1)*{V5BAS(c,s)/V5TOT} + DUM_YEAR1*B5SH(c,s);
(Initial)(All,c,COM)(All,s,SRC) B5SHO(c,s) =
    (1-DUM_YEAR1)*{V5BAS(c,s)/V5TOT} + DUM_YEAR1*B5SHO(c,s);
(Initial)(All,c,COM)(All,s,SRC) B5SHBO(c,s) = B5SHO(c,s);
(Initial)(All,c,COM)(All,s,SRC) LEVP5B(c,s) =LEVX5(c,s);
(Initial)(All,c,COM)(All,s,SRC) LEVP5B_O(c,s) = LEVX5_O(c,s);
(Initial) I_5QL=Sum(c,COM,Sum(s,SRC,
    B5SHBO(c,s)*(LEVX5B(c,s)-LEVX5B_O(c,s))/LEVX5B_O(c,s)));
(Initial) I_5PL=Sum(c,COM,Sum(s,SRC,
    B5SHBO(c,s)*(LEVP5B(c,s)-LEVP5B_O(c,s))/LEVP5B_O(c,s)));
(Initial) V5_T = (1-DUM_YEAR1)*V5TOT + DUM_YEAR1*V5_T;
(Initial) V5_T_O = (1-DUM_YEAR1)*V5TOT + DUM_YEAR1*V5_T_O;
(Initial) V5_TB = V5_T;
(Initial) V5_TB_O = V5_T_O;
(Initial) DEV_5V= (V5_TB - V5_TB_O)/V5_TB_O;
(Initial) I_5PP = (1+DEV_5V)/(1+I_5QL)-1;
(Initial) I_5QP = (1+DEV_5V)/(1+I_5PL)-1;

```

Variable

```
(All,c,COM)(All,s,SRC)
    x5_o(c,s) # Govt cons by com/source in forecast simulation #;
(All,c,COM)(All,s,SRC)
    f_x5_o(c,s) # Allows exogenization of forecast Govt cons #;
(All,c,COM)(All,s,SRC)
    b5shr_o(c,s) # Government consumption shares, forecast # ;
(All,c,COM)(All,s,SRC)
    b5shr(c,s) # Government consumption shares # ;
(All,c,COM)(All,s,SRC)
    f_b5shr_o(c,s) # Allows exog. of forecast Govt cons shares #;
    x_5la # Laspeyres quantity index for Govt consumption #;
    x_5pa # Paasche quantity index for Govt consumption #;
(All,c,COM)(All,s,SRC)
    p5_o(c,s) # Govt cons. prices by commodity/source in forecast #;
(All,c,COM)(All,s,SRC)
    f_P5_o(c,s) # Allows exog. of forecast Govt cons prices #;
    p_5la # Laspeyres price index for Govt consumption #;
    p_5pa # Paasche price index for Govt consumption #;
    w5tot_o # Value of Govt consumption, forecast #;
    f_w5tot_o # Allows exog. of value of Govt consumption, f'cast #;
```

Update

```
(All,c,COM)(All,s,SRC) LEVX5(c,s) =x5(c,s);
(All,c,COM)(All,s,SRC) LEVX5_O(c,s) =x5_o(c,s);
(All,c,COM)(All,s,SRC) B5SH(c,s) =b5shr(c,s);
(All,c,COM)(All,s,SRC) B5SHO(c,s) =b5shr_o(c,s);
V5_T = w5tot;
V5_T_O = w5tot_o;
(All,c,COM)(All,s,SRC) LEVP5(c,s) =P5(c,s);
(All,c,COM)(All,s,SRC) LEVP5_O(c,s) =P5_o(c,s);
```

Equation E_x_5la

```
# Laspeyres quantity index for Govt consumption #
x_5la = 100*I_5QL*del_unity;
```

Equation E_x_5pa

```
# Paasche quantity index for Govt consumption #
x_5pa = 100*I_5QP*del_unity;
```

Equation E_p_5la

```
# Laspeyres price index for Govt consumption#
p_5la = 100*I_5PL*del_unity;
```

Equation E_p_5pa

```
# Paasche price index for Govt consumption #
p_5pa = 100*I_5PP*del_unity;
```

Equation E_b5shr # Budget shares in Govt consumption

```
(All,c,COM)(All,s,SRC) b5shr(c,s) = x5(c,s)+p5(c,s) - w5tot;
```

Equation E_x5cs_o

```
# Introduces forecasts of Govt consumption by com/source into policy sims. #
(All,c,COM)(All,s,SRC) x5_o(c,s) = x5(c,s) + f_x5_o(c,s);
```

Equation E_b5sh_o

```
# Introduces forecasts of Govt budget shares by com/source into policy sims. #
(All,c,COM)(All,s,SRC) b5shr_o(c,s) = b5shr(c,s)+ f_b5shr_o(c,s);
```

```

Equation E_P5cs_o
# Introduces forecasts of Govt prices into policy sims. #
(All,c,COM)(All,s,SRC) p5_o(c,s) = p5(c,s) + f_p5_o(c,s);

Equation E_w5tot_o
# Introduces forecasts of nominal Govt consumption into policy sims. #
w5tot_o = w5tot + f_w5tot_o;

! Export quantity and price indexes !
Coefficient
(All,c,COM) LEVX4(c) # Quantities of exports by commodity #;
(Parameter)(All,c,COM)
    LEVX4B(c) # Quantities of exports by commodity, base #;
(Parameter)(All,c,COM)
    LEVX4B_O(c) # Quantities of exports in f'cast, base #;
(All,c,COM)
    LEVX4_O(c) # Quantities of exports by commodity in f'cast #;
(All,c,COM)
    LEVP4(c) # Fob price of exports by commodity #;
(Parameter)(All,c,COM)
    LEVP4B(c) # Fob price of exports by com , base #;
(Parameter)(All,c,COM)
    LEVP4B_O(c) # Fob price of exports by com in f'cast, base #;
(All,c,COM)
    LEVP4_O(c) # Fob price of exports by commodity in f'cast #;
(All,c,COM)
    B4SH(c) # Shares in exports #;
(All,c,COM)
    B4SHO(c) # Shares in exports, forecast #;
(Parameter)(All,c,COM)
    B4SHBO(c) # Shares in exports, forecast base #;
(Parameter)
    I_4QL # Exports, Laspeyres quantity #;
(Parameter)
    I_4PL # Exports, Laspeyres price #;
(Parameter)
    I_4QP # Exports, Paasche quantity #;
(Parameter)
    I_4PP # Exports, Paasche price #;
    V4_T # Exports, value #;
    V4_T_O # Exports, forecast value #;
(Parameter)
    V4_TB # Exports, value, base #;
(Parameter)
    V4_TB_O # Exports, forecast value, base #;
(Parameter)
    DEV_4V # Exports, deviation value #;

Read
LEVX4 from file EXTRA Header "LVX4";
LEVX4_O from file EXTRA3 Header "LVX4";
B4SH from file EXTRA Header "B4SH";
B4SHO from file EXTRA3 Header "B4SH";
LEVP4 from file EXTRA Header "LVP4";
LEVP4_O from file EXTRA3 Header "LVP4";
V4_T from file EXTRA Header "V4TT";
V4_T_O from file EXTRA3 Header "V4TT";

Formula
(Initial)(All,c,COM) LEVX4B(c) = LEVX4(c);
(Initial)(All,c,COM) LEVX4B_O(c) = LEVX4_O(c);
(Initial)(All,c,COM) B4SH(c) =
    (1-DUM_YEAR1)*{V4PUR(c)/V4TOT} + DUM_YEAR1*B4SH(c);
(Initial)(All,c,COM) B4SHO(c) =
    (1-DUM_YEAR1)*{V4PUR(c)/V4TOT} + DUM_YEAR1*B4SHO(c);
(Initial)(All,c,COM) B4SHBO(c) = B4SHO(c);
(Initial)(All,c,COM) LEVP4B(c) = LEVP4(c);
(Initial)(All,c,COM) LEVP4B_O(c) = LEVP4_O(c);

```

```

(Initial) I_4QL=Sum(c,COM,B4SHBO(c)*(LEVX4B(c)-LEVX4B_O(c))/LEVX4B_O(c));
(Initial) I_4PL=Sum(c,COM,B4SHBO(c)*(LEVP4B(c)-LEVP4B_O(c))/LEVP4B_O(c));
(Initial) V4_T = (1-DUM_YEAR1)*V4TOT + DUM_YEAR1*V4_T;
(Initial) V4_T_O = (1-DUM_YEAR1)*V4TOT + DUM_YEAR1*V4_T_O;
(Initial) V4_TB = V4_T;
(Initial) V4_TB_O = V4_T_O;
(Initial) DEV_4V= (V4_TB - V4_TB_O)/V4_TB_O;
(Initial) I_4PP = (1+DEV_4V)/(1+I_4QL)-1;
(Initial) I_4QP = (1+DEV_4V)/(1+I_4PL)-1;

```

Variable

```

(All,c,COM)
    x4_o(c) # Exports by com in forecast simulation #;
(All,c,COM)
    f_x4_o(c) # Allows exogenization of forecast exports #;
(All,c,COM)
    b4shr_o(c) # Export shares, forecast # ;
(All,c,COM)
    b4shr(c) # Export shares # ;
(All,c,COM)
    f_b4shr_o(c) # Allows exog. of forecast export shares #;
    x_4la # Laspeyres quantity index for exports #;
    x_4pa # Paasche quantity index for exports #;
(All,c,COM)
    p4_o(c) # Fob export prices by commodity in forecast #;
(All,c,COM)
    f_p4_o(c) # Allows exog. of forecast fob export prices #;
    p_4la # Laspeyres fob price index for exports #;
    p_4pa # Paasche fob price index for exports #;
    w4tot_o # Value of exports, forecast #;
    f_w4tot_o # Allows exog. of value of exports, f'cast #;

```

Update

```

(All,c,COM) LEVX4(c) =x4(c);
(All,c,COM) LEVX4_O(c) =x4_o(c);
(All,c,COM) B4SH(c) =b4shr(c);
(All,c,COM) B4SHO(c) =b4shr_o(c);
V4_T = w4tot;
V4_T_O = w4tot_o;
(All,c,COM) LEVP4(c) =P4(c);
(All,c,COM) LEVP4_O(c) =P4_o(c);

```

Equation E_x_4la

```

# Laspeyres quantity index for exports #
x_4la = 100*I_4QL*del_unity;

```

Equation E_x_4pa

```

# Paasche quantity index for exports #
x_4pa = 100*I_4QP*del_unity;

```

Equation E_p_4la

```

# Laspeyres price index for exports #
p_4la = 100*I_4PL*del_unity;

```

Equation E_p_4pa

```

# Paasche price index for exports #
p_4pa = 100*I_4PP*del_unity;

```

Equation E_b4shr # Shares in exports

```

(All,c,COM) b4shr(c) = x4(c)+p4(c) - w4tot;

```

```

Equation E_x4cs_o
# Introduces forecasts of exports by com into policy sims. #
(All,c,COM) x4_o(c) = x4(c) + f_x4_o(c);

Equation E_b4shr_o
# Introduces forecasts of export shares by com into policy sims. #
(All,c,COM) b4shr_o(c) = b4shr(c)+ f_b4shr_o(c);

Equation E_P4cs_o
# Introduces forecasts of fob export prices into policy sims. #
(All,c,COM) p4_o(c) = p4(c) + f_p4_o(c);

Equation E_w4tot_o
# Introduces forecasts of nominal exports into policy sims. #
w4tot_o = w4tot + f_w4tot_o;

! Import quantity and price indexes !
Coefficient
(All,c,COM)
    LEVX7(c) # Quantities of imports by commodity #;
(Parameter)(All,c,COM)
    LEVX7B(c) # Quantities of imports by commodity, base #;
(Parameter)(All,c,COM)
    LEVX7B_O(c) # Quantities of imports in f'cast, base #;
(All,c,COM)
    LEVX7_O(c) # Quantities of imports by commodity in f'cast #;
(All,c,COM)
    LEVP7(c) # Cif price of imports by commodity #;
(Parameter)(All,c,COM)
    LEVP7B(c) # Cif price of imports by commodity, base #;
(Parameter)(All,c,COM)
    LEVP7B_O(c) # Cif price of imports by com in f'cast, base #;
(All,c,COM)
    LEVP7_O(c) # Cif price of imports by commodity in f'cast #;
(All,c,COM)
    B7SH(c) # Shares in imports #;
(All,c,COM)
    B7SHO(c) # Shares in imports, forecast #;
(Parameter)(All,c,COM)
    B7SHBO(c) # Shares in imports, forecast base #;
(Parameter)
    I_7QL # Imports, Laspeyres quantity #;
(Parameter)
    I_7PL # Imports, Laspeyres price #;
(Parameter)
    I_7QP # Imports, Paasche quantity #;
(Parameter)
    I_7PP # Imports, Paasche price #;
    V7_T # Imports, value #;
    V7_T_O # Imports, forecast value #;
(Parameter)
    V7_TB # Imports, value, base #;
(Parameter)
    V7_TB_O # Imports, forecast value, base #;
(Parameter)
    DEV_7V # Imports, deviation value #;

Read
LEVX7 from file EXTRA Header "LVX7";
LEVX7_O from file EXTRA3 Header "LVX7";
B7SH from file EXTRA Header "B7SH";
B7SHO from file EXTRA3 Header "B7SH";
LEVP7 from file EXTRA Header "LVP7";
LEVP7_O from file EXTRA3 Header "LVP7";
V7_T from file EXTRA Header "V7TT";
V7_T_O from file EXTRA3 Header "V7TT";

```

Formula

```
(Initial)(All,c,COM) LEVX7B(c) =LEVX7(c);
(Initial)(All,c,COM) LEVX7B_O(c) = LEVX7_O(c);
(Initial)(All,c,COM) B7SH(c) =
    (1-DUM_YEAR1)*{V0CIF(c)/V0CIF_C} + DUM_YEAR1*B7SH(c);
(Initial)(All,c,COM) B7SHO(c) =
    (1-DUM_YEAR1)*{V0CIF(c)/V0CIF_C} + DUM_YEAR1*B7SHO(c);
(Initial)(All,c,COM) B7SHBO(c) = B7SHO(c);
(Initial)(All,c,COM) LEVP7B(c) =LEV7(c);
(Initial)(All,c,COM) LEVP7B_O(c) = LEVP7_O(c);
(Initial) I_7QL=Sum(c,COM, B7SHBO(c)*(LEVX7B(c)-LEVX7B_O(c))/LEVX7B_O(c));
(Initial) I_7PL=Sum(c,COM, B7SHBO(c)*(LEVP7B(c)-LEVP7B_O(c))/LEVP7B_O(c));
(Initial) V7_T = (1-DUM_YEAR1)*V0CIF_C + DUM_YEAR1*V7_T;
(Initial) V7_T_O = (1-DUM_YEAR1)*V0CIF_C + DUM_YEAR1*V7_T_O;
(Initial) V7_TB = V7_T;
(Initial) V7_TB_O = V7_T_O;
(Initial) DEV_7V= (V7_TB - V7_TB_O)/V7_TB_O;
(Initial) I_7PP = (1+DEV_7V)/(1+I_7QL)-1;
(Initial) I_7QP = (1+DEV_7V)/(1+I_7PL)-1;
```

Variable

```
(All,c,COM)
    x7_o(c) # Imports by com in forecast simulation #;
(All,c,COM)
    f_x7_o(c) # Allows exogenization of forecast imports #;
(All,c,COM)
    b7shr_o(c) # Import shares, forecast # ;
(All,c,COM)
    b7shr(c) # Import shares # ;
(All,c,COM)
    f_b7shr_o(c) # Allows exog. of forecast import shares #;
    x_7la # Laspeyres quantity index for imports #;
    x_7pa # Paasche quantity index for imports #;
(All,c,COM)
    p0cif(c) # Cif import prices by commodity in forecast #;
(All,c,COM)
    p7_o(c) # Cif import prices by commodity in forecast #;
(All,c,COM)
    f_P7_o(c) # Allows exog. of forecast cif import prices #;
    p_7la # Laspeyres cif import index for exports #;
    p_7pa # Paasche cif import index for exports #;
    w7tot_o # Value of imports, forecast #;
    f_w7tot_o # Allows exog. of value of imports, f'cast #;
```

Update

```
(All,c,COM) LEVX7(c) =x0imp(c);
(All,c,COM) LEVX7_O(c) =x7_o(c);
(All,c,COM) B7SH(c) =b7shr(c);
(All,c,COM) B7SHO(c) =b7shr_o(c);
V7_T = w0cif_c;
V7_T_O = w7tot_o;
(All,c,COM) LEVP7(c) =P0cif(c);
(All,c,COM) LEVP7_O(c) =P7_o(c);
```

Equation E_x_7la

```
# Laspeyres quantity index for imports #
x_7la = 100*I_7QL*del_unity;
```

Equation E_x_7pa

```
# Paasche quantity index for imports #
x_7pa = 100*I_7QP*del_unity;
```

```

Equation E_p_7la
# Laspeyres price index for imports #
p_7la = 100*I_7PL*del_unity;

Equation E_p_7pa
# Paasche price index for imports #
p_7pa = 100*I_7PP*del_unity;

Equation E_b7shr # Shares in imports #
(All,c,COM) b7shr(c) = x0imp(c)+p0cif(c) - w0cif_c;

Equation E_x7cs_o
# Introduces forecasts of imports by com into policy sims. #
(All,c,COM) x7_o(c) = x0imp(c) + f_x7_o(c);

Equation E_b7sh_o
# Introduces forecasts of import shares by com into policy sims. #
(All,c,COM) b7shr_o(c) = b7shr(c)+ f_b7shr_o(c);

Equation E_P7cs_o
# Introduces forecasts of cif import prices into policy sims. #
(All,c,COM) p7_o(c) = p0cif(c) + f_p7_o(c);

Equation E_w7tot_o
# Introduces forecasts of nominal imports into policy sims. #
w7tot_o = w0cif_c + f_w7tot_o;

Equation E_p0cif
# Cif import price, domestic currency #
(All,c,COM)p0cif(c) = pf0cif(c) - phi;

! Capital creation quantity and price indexes !
Coefficient
(All,i,IND)
    LEVX2(i) # Quantities of investment by industry #;
(Parameter)(All,i,IND)
    LEVX2B(i) # Quantities of investment by industry, base #;
(Parameter)(All,i,IND)
    LEVX2B_O(i) # Quantities of investment in f'cast, base #;
(All,i,IND)
    LEVX2_O(i) # Quantities of investment in f'cast #;
(All,i,IND)
    LEVP2(i) # Price of investment by industry #;
(Parameter)(All,i,IND)
    LEVP2B(i) # Price of investment by industry, base #;
(Parameter)(All,i,IND)
    LEVP2B_O(i) # Price of investment by industry in f'cast, base #;
(All,i,IND)
    LEVP2_O(i) # Price of investment by industry in f'cast #;
(All,i,IND)
    B2SH(i) # Shares in investment #;
(All,i,IND)
    B2SHO(i) # Shares in investment, forecast #;
(Parameter)(All,i,IND)
    B2SHBO(i) # Shares in investment, forecast base #;
(Parameter)
    I_2QL # Investment, Laspeyres quantity #;
(Parameter)
    I_2PL # Investment, Laspeyres price #;
(Parameter)
    I_2QP # Investment, Paasche quantity #;
(Parameter)
    I_2PP # Investment, Paasche price #;
    V2_T # Investment, value #;

```

```

V2_T_O # Investment, forecast value #;
(Parameter) V2_TB # Investment, value, base #;
(Parameter) V2_TB_O # Investment, forecast value, base #;
(Parameter) DEV_2V # Investment, deviation value #;

```

Read

```

LEVX2 from file EXTRA Header "LVX2";
LEVX2_O from file EXTRA3 Header "LVX2";
B2SH from file EXTRA Header "B2SH";
B2SHO from file EXTRA3 Header "B2SH";
LEVP2 from file EXTRA Header "LVP2";
LEVP2_O from file EXTRA3 Header "LVP2";
V2_T from file EXTRA Header "V2TT";
V2_T_O from file EXTRA3 Header "V2TT";

```

Formula

```

(Initial)(All,i,IND) LEVX2B(i) =LEVX2(i);
(Initial)(All,i,IND) LEVX2B_O(i) = LEVX2_O(i);
(Initial)(All,i,IND) B2SH(i) =
(1-DUM_YEAR1)*{V2TOT(i)/V2TOT_I} + DUM_YEAR1*B2SH(i);
(Initial)(All,i,IND) B2SHO(i) =
(1-DUM_YEAR1)*{V2TOT(i)/V2TOT_I} + DUM_YEAR1*B2SHO(i);
(Initial)(All,i,IND) B2SHBO(i) = B2SHO(i);
(Initial)(All,i,IND) LEVP2B(i) =LEVP2(i);
(Initial)(All,i,IND) LEVP2B_O(i) = LEVP2_O(i);
(Initial) I_2QL=Sum(i,IND,B2SHBO(i)*(LEVX2B(i)-LEVX2B_O(i))/LEVX2B_O(i));
(Initial) I_2PL=Sum(i,IND,B2SHBO(i)*(LEVP2B(i)-LEVP2B_O(i))/LEVP2B_O(i));
(Initial) V2_T = (1-DUM_YEAR1)*V2TOT_I + DUM_YEAR1*V2_T;
(Initial) V2_T_O = (1-DUM_YEAR1)*V2TOT_I + DUM_YEAR1*V2_T_O;
(Initial) V2_TB = V2_T;
(Initial) V2_TB_O = V2_T_O;
(Initial)DEV_2V= (V2_TB - V2_TB_O)/V2_TB_O;
(Initial)I_2PP = (1+DEV_2V)/(1+I_2QL)-1;
(Initial)I_2QP = (1+DEV_2V)/(1+I_2PL)-1;

```

Variable

```

(All,i,IND)
x2tot_o(i) # Investment by industry in forecast simulation #;
(All,i,IND)
f_x2tot_o(i) # Allows exogenization of forecast investment #;
(All,i,IND)
b2shr_o(i) # Investment shares, forecast # ;
(All,i,IND)
b2shr(i) # Investment shares # ;
(All,i,IND)
f_b2shr_o(i) # Allows exog. of forecast investment shares #;
x_2la # Laspeyres quantity index for investment #;
x_2pa # Paasche quantity index for investment #;
(All,i,IND)
p2tot_o(i) # Investment prices by industry in forecast #;
(All,i,IND)
f_p2tot_o(i) # Allows exog. of forecast investment prices #;
p_2la # Laspeyres price index for investment #;
p_2pa # Paasche price index for investment #;
w2tot_o # Value of investment, forecast #;
f_w2tot_o # Allows exog. of value of investment, f'cast #;

```

Update

```

(All,i,IND) LEVX2(i) =x2tot(i);
(All,i,IND) LEVX2_O(i) =x2tot_o(i);
(All,i,IND) B2SH(i) =b2shr(i);

```



```

(All,i,IND) B2SHO(i) =b2shr_o(i);
V2_T = w2tot_i;
V2_T_O = w2tot_o;
(All,i,IND) LEVP2(i) =p2tot(i);
(All,i,IND) LEVP2_O(i) =p2tot_o(i);

Equation E_x_2la
# Laspeyres quantity index for investment #
x_2la = 100*I_2QL*del_unity;

Equation E_x_2pa
# Paasche quantity index for investment #
x_2pa = 100*I_2QP*del_unity;

Equation E_p_2la
# Laspeyres price index for investment #
p_2la = 100*I_2PL*del_unity;

Equation E_p_2pa
# Paasche price index for investment #
p_2pa = 100*I_2PP*del_unity;

Equation E_b2shr # Shares in investment #
(All,i,IND) b2shr(i) = x2tot(i)+p2tot(i) - w2tot_i;

Equation E_x2cs_o
# Introduces forecasts of investment by industry into policy sims. #
(All,i,IND) x2tot_o(i) = x2tot(i) + f_x2tot_o(i);

Equation E_b2shr_o
# Introduces forecasts of investement shares by industry into policy sims. #
(All,i,IND) b2shr_o(i) = b2shr(i)+ f_b2shr_o(i);

Equation E_P2cs_o
# Introduces forecasts of investment prices into policy sims. #
(All,i,IND) p2tot_o(i) = p2tot(i) + f_p2tot_o(i);

Equation E_w2tot_o
# Introduces forecasts of nominal investment into policy sims. #
w2tot_o = w2tot_i + f_w2tot_o;

! GDP quantity and price indexes !
Coefficient
SH2_GDP # Share of investment in GDP #;
SH3_GDP # Share of consumption in GDP #;
SH4_GDP # Share of exports in GDP #;
SH5_GDP # Share of Govt consumption in GDP #;
SH7_GDP # Share of imports in GDP #;
SH2_GDP_O # Share of investment in GDP, f'cast #;
SH3_GDP_O # Share of consumption in GDP, f'cast #;
SH4_GDP_O # Share of exports in GDP, f'cast #;
SH5_GDP_O # Share of Govt consumption in GDP, f'cast #;
SH7_GDP_O # Share of imports in GDP, f'cast #;
(Parameter) SH2_GDPB_O # Share of investment in GDP, f'cast, base #;
(Parameter) SH3_GDPB_O # Share of consumption in GDP, f'cast, base #;
(Parameter) SH4_GDPB_O # Share of exports in GDP, f'cast, base #;
(Parameter) SH5_GDPB_O # Share of Govt consumption in GDP, f'cast, base #;
(Parameter) SH7_GDPB_O # Share of imports in GDP, f'cast, base #;
VGDP_T # Investment, value #;
VGDP_T_O # Investment, forecast value #;
(Parameter) VGDP_TB # Investment, value, base #;

```

```

(Parameter)      VGDP_TB_O # Investment, forecast value, base #;
(Parameter)      DEV_GDPV # GDP, deviation value #;
(Parameter)      I_GDPQL # GDP, Laspeyres quantity #;
(Parameter)      I_GDPPL # GDP, Laspeyres price #;
(Parameter)      I_GDPQP # GDP, Paasche quantity #;
(Parameter)      I_GDPPP # GDP, Paasche price #;

```

Read

```

SH2_GDP from file EXTRA Header"BG2";
SH3_GDP from file EXTRA Header"BG3";
SH4_GDP from file EXTRA Header"BG4";
SH5_GDP from file EXTRA Header"BG5";
SH7_GDP from file EXTRA Header"BG7";
SH2_GDP_O from file EXTRA3 Header"BG2";
SH3_GDP_O from file EXTRA3 Header"BG3";
SH4_GDP_O from file EXTRA3 Header"BG4";
SH5_GDP_O from file EXTRA3 Header"BG5";
SH7_GDP_O from file EXTRA3 Header"BG7";
VGDP_T from file EXTRA Header "VGT";
VGDP_T_O from file EXTRA3 Header "VGT";

```

Formula

```

(Initial) SH2_GDP =(1-DUM_YEAR1)*{V2TOT_I/V0GDPEXP} + DUM_YEAR1*SH2_GDP ;
(Initial) SH3_GDP =(1-DUM_YEAR1)*{V3TOT/V0GDPEXP} + DUM_YEAR1*SH3_GDP ;
(Initial) SH4_GDP =(1-DUM_YEAR1)*{V4TOT/V0GDPEXP} + DUM_YEAR1*SH4_GDP ;
(Initial) SH5_GDP =(1-DUM_YEAR1)*{V5TOT/V0GDPEXP} + DUM_YEAR1*SH5_GDP ;
(Initial) SH7_GDP =(1-DUM_YEAR1)*{V0CIF_C/V0GDPEXP} + DUM_YEAR1*SH7_GDP ;
(Initial) SH2_GDP_O =(1-DUM_YEAR1)*{V2TOT_I/V0GDPEXP} + DUM_YEAR1*SH2_GDP_O ;
(Initial) SH3_GDP_O =(1-DUM_YEAR1)*{V3TOT/V0GDPEXP} + DUM_YEAR1*SH3_GDP_O ;
(Initial) SH4_GDP_O =(1-DUM_YEAR1)*{V4TOT/V0GDPEXP} + DUM_YEAR1*SH4_GDP_O ;
(Initial) SH5_GDP_O =(1-DUM_YEAR1)*{V5TOT/V0GDPEXP} + DUM_YEAR1*SH5_GDP_O ;
(Initial) SH7_GDP_O =(1-DUM_YEAR1)*{V0CIF_C/V0GDPEXP} + DUM_YEAR1*SH7_GDP_O ;
(Initial) SH2_GDPB_O =SH2_GDP_O ;
(Initial) SH3_GDPB_O =SH3_GDP_O ;
(Initial) SH4_GDPB_O =SH4_GDP_O ;
(Initial) SH5_GDPB_O =SH5_GDP_O ;
(Initial) SH7_GDPB_O =SH7_GDP_O ;

```

Formula

```

(Initial) I_GDPQL = SH2_GDPB_O*I_2QL + SH3_GDPB_O*I_3QL + SH4_GDPB_O*I_4QL
               + SH5_GDPB_O*I_5QL - SH7_GDPB_O*I_7QL;
(Initial) I_GDPPL = SH2_GDPB_O*I_2PL + SH3_GDPB_O*I_3PL + SH4_GDPB_O*I_4PL
               + SH5_GDPB_O*I_5PL - SH7_GDPB_O*I_7PL;

(Initial) VGDP_T = (1-DUM_YEAR1)*V0GDPEXP + DUM_YEAR1*VGDP_T;
(Initial) VGDP_T_O = (1-DUM_YEAR1)*V0GDPEXP + DUM_YEAR1*VGDP_T_O;
(Initial) VGDP_TB = VGDP_T;
(Initial) VGDP_TB_O = VGDP_T_O;
(Initial) DEV_GDPV=(VGDP_TB - VGDP_TB_O)/VGDP_TB_O;

(Initial)I_GDPPP = (1+DEV_GDPV)/(1+I_GDPQL)-1;
(Initial)I_GDPQP = (1+DEV_GDPV)/(1+I_GDPPL)-1;

```

Variable

```

bsh2 # Share of investment in GDP #;
bsh3 # Share of consumption in GDP #;
bsh4 # Share of exports in GDP #;
bsh5 # Share of Govt consumption in GDP #;
bsh7 # Share of imports in GDP #;
bsh2_o # Share of investment in GDP, f'cast #;
bsh3_o # Share of consumption in GDP, f'cast #;

```

```

bsh4_o # Share of exports in GDP, f'cast #;
bsh5_o # Share of Govt consumption in GDP, f'cast #;
bsh7_o # Share of imports in GDP, f'cast#;
f_bsh2_o # Allows exog. of investment/GDP, f'cast #;
f_bsh3_o # Allows exog. of consumption/GDP, f'cast #;
f_bsh4_o # Allows exog. of exports/GDP, f'cast #;
f_bsh5_o # Allows exog. of Govt-consumption/GDP, f'cast #;
f_bsh7_o # Allows exog. of imports/GDP, f'cast # ;
w0gdpexp_o # GDP, Forecast #;
f_w0gdpexp_o # Allows exog. of GDP, f'cast # ;
x_gdpla # Laspeyres quantity index for investment #;
x_gdppa # Paasche quantity index for investment #;
p_gdpla # Laspeyres quantity index for investment #;
p_gdppa # Paasche quantity index for investment #;

```

Update

```

SH2_GDP =bsh2;
SH3_GDP =bsh3;
H4_GDP =bsh4;
SH5_GDP =bsh5;
SH7_GDP =bsh7;
SH2_GDP_O =bsh2_o;
SH3_GDP_O =bsh3_o;
SH4_GDP_O =bsh4_o;
SH5_GDP_O =bsh5_o;
SH7_GDP_O =bsh7_o;
VGDP_T = w0gdpexp;
VGDP_T_O = w0gdpexp_o;

```

```

Equation E_bsh2 # Share of investment in GDP #
bsh2 = w2tot_i - w0gdpexp;

```

```

Equation E_bsh3 # Share of consumption in GDP #
bsh3 = w3tot - w0gdpexp;

```

```

Equation E_bsh4 # Share of exports in GDP #
bsh4 = w4tot - w0gdpexp;

```

```

Equation E_bsh5 # Share of Govt consumption in GDP #
bsh5 = w5tot - w0gdpexp;

```

```

Equation E_bsh7 # Share of imports in GDP #
bsh7 = w0cif_c - w0gdpexp;

```

```

Equation E_bsh2_o # Allows exog. of investment/GDP, f'cast #
bsh2_o = bsh2 + f_bsh2_o;

```

```

Equation E_bsh3_o # Allows exog. of consumption/GDP, f'cast #
bsh3_o = bsh3 + f_bsh3_o;

```

```

Equation E_bsh4_o # Allows exog. of export/GDP, f'cast #
bsh4_o = bsh4 + f_bsh4_o;

```

```

Equation E_bsh5_o # Allows exog. of Govt consumption/GDP, f'cast #
bsh5_o = bsh5 + f_bsh5_o;

```

```

Equation E_bsh7_o # Allows exog. of imports/GDP, f'cast #
bsh7_o = bsh7 + f_bsh7_o;

```

```

Equation E_w0gdpexp_o
# Introduces forecasts of nominal investment into policy sims. #
w0gdpexp_o = w0gdpexp + f_w0gdpexp_o;

Equation E_x_gdpla
# Upper bound on equivalent variation as % of h'hold expenditure #
x_gdpla= 100*[I_GDPQL]*del_unity;

Equation E_x_gdppa
# Upper bound on equivalent variation as % of h'hold expenditure #
x_gdppa = 100*[I_GDPQP]*del_unity;

Equation E_p_gdpla
# Upper bound on equivalent variation as % of h'hold expenditure #
p_gdpla= 100*[I_GDPPL]*del_unity;

Equation E_p_gdppa
# Upper bound on equivalent variation as % of h'hold expenditure #
p_gdppa = 100*[I_GDPPP]*del_unity;

! Capital input in current production !
Coefficient
(All,i,IND)
LEVXCAP(i) # Quantities of capital input by industry #;
(Parameter)(All,i,IND)
LEVXCAPB(i) # Quantities of capital input by industry, base #;
(Parameter)(All,i,IND)
LEVXCAPB_O(i) # Qty of capital input by industry, f'cast base #;
(All,i,IND)
LEVXCAP_O(i) # Qty of capital input by industry f'cast #;
(All,i,IND)
LEVPCAP(i) # Price of capital input by industry #;
(Parameter)(All,i,IND)
LEVPCAPB(i) # Price of capital input by industry, base #;
(Parameter)(All,i,IND)
LEVPCAPB_O(i) # Price of capital input by ind in f'cast, base #;
(All,i,IND)
LEVPCAP_O(i) # Price of capital input by industry in f'cast #;
(All,i,IND)
BCAPSH(i) # Share of capital in industry i #;
(All,i,IND)
BCAPSHO(i) # Share of capital in industry i, forecast #;
(Parameter)(All,i,IND)
BCAPSHBO(i) # Share of capital in industry i, forecast base #;
(Parameter)
I_CAPQL # Capital, Laspeyres quantity #;
(Parameter)
I_CAPPL # Capital, Laspeyres price #;
(Parameter)
I_CAPQP # Capital, Paasche quantity #;
(Parameter)
I_CAPPP # Capital, Paasche price #;
VCAP_T # Capital, value #;
VCAP_T_O # Capital, forecast value #;
(Parameter)
VCAP_TB # Capital, value, base #;
(Parameter)
VCAP_TB_O # Capital, forecast value, base #;
(Parameter)
DEV_CAPV # Capital, deviation value #;

Read
LEVXCAP from file EXTRA Header"LXCP";
LEVXCAP_O from file EXTRA3 Header"LXCP";
BCAPSH from file EXTRA Header "CPSH";
BCAPSHO from file EXTRA3 Header "CPSH";
LEVPCAP from file EXTRA Header"LPCP";
LEVPCAP_O from file EXTRA3 Header"LPCP";

```

VCAP_T from file EXTRA Header "VCTT";
VCAP_T_O from file EXTRA3 Header "VCTT";

Formula

```
(Initial)(All,i,IND) LEVXCAPB(i) =LEVXCAP(i);
(Initial)(All,i,IND) LEVXCAPB_O(i) = LEVXCAP_O(i);
(Initial)(All,i,IND) BCAPSH(i) =
(1-DUM_YEAR1)*{V1CAP(i)/V1CAP_I} + DUM_YEAR1*BCAPSH(i);
(Initial)(All,i,IND) BCAPSHO(i) =
(1-DUM_YEAR1)*{V1CAP(i)/V1CAP_I} + DUM_YEAR1*BCAPSHO(i);
(Initial)(All,i,IND) BCAPSHBO(i) = BCAPSHO(i);
(Initial)(All,i,IND) LEVPCAPB(i) =LEVPCAP(i);
(Initial)(All,i,IND) LEVPCAPB_O(i) = LEVPCAP_O(i);
(Initial) I_CAPQL=Sum(i,IND,
BCAPSHBO(i)*(LEVXCAPB(i)-LEVXCAPB_O(i))/LEVXCAPB_O(i));
(Initial) I_CAPPL=Sum(i,IND,
BCAPSHBO(i)*(LEVPCAPB(i)-LEVPCAPB_O(i))/LEVPCAPB_O(i));
(Initial) VCAP_T = (1-DUM_YEAR1)*V1CAP_I + DUM_YEAR1*VCAP_T;
(Initial) VCAP_T_O = (1-DUM_YEAR1)*V1CAP_I + DUM_YEAR1*VCAP_T_O;
(Initial) VCAP_TB = VCAP_T;
(Initial) VCAP_TB_O = VCAP_T_O;
(Initial)DEV_CAPV= (VCAP_TB - VCAP_TB_O)/VCAP_TB_O;
(Initial)I_CAPPP = (1+DEV_CAPV)/(1+I_CAPQL)-1;
(Initial)I_CAPQP = (1+DEV_CAPV)/(1+I_CAPPL)-1;
```

Variable

```
(All,i,IND) xCAPtot_o(i) # Qty of capital input in forecast simulation #;
(All,i,IND) f_xCAPtot_o(i) # Allows exog.of forecast capital input #;
(All,i,IND) bCAPshr_o(i) # Capital shares by ind , forecast # ;
(All,i,IND) bCAPshr(i) # Capital shares by ind # ;
(All,i,IND) f_bCAPshr_o(i) # Allows exog. of forecast capital shares #;
x_CAPla # Laspeyres quantity index for capital input #;
x_CAPpa # Paasche quantity index for capital input #;
(All,i,IND) pCAPtot_o(i) # Capital rental price by industry in forecast #;
(All,i,IND) f_pCAPtot_o(i) # Allows exog. of forecast capital rentals #;
p_CAPla # Laspeyres price index for capital input #;
p_CAPpa # Paasche price index for capital input #;
wCAPtot_o # Value of capital input, forecast #;
f_wCAPtot_o # Allows exog. of value of capital input, f'cast #;
```

Update

```
(All,i,IND) LEVXCAP(i) =x1CAP(i);
(All,i,IND) LEVXCAP_O(i) =xCAPTtot_o(i);
(All,i,IND) BCAPSH(i) =bCAPshr(i);
(All,i,IND) BCAPSHO(i) =bCAPshr_o(i);
VCAP_T = wlcap_i;
VCAP_T_O = wCAPtot_o;
(All,i,IND) LEVPCAP(i) =p1CAP(i);
(All,i,IND) LEVPCAP_O(i) =pCAPtot_o(i);
```

Equation E_x_CAPla # Laspeyres quantity index for capital input #
x_CAPla = 100*I_CAPQL*del_unity;

Equation E_x_CAPpa # Paasche quantity index for capital input #
x_CAPpa = 100*I_CAPQP*del_unity;

Equation E_p_CAPla # Laspeyres price index for capital input #
p_CAPla = 100*I_CAPPL*del_unity;

Equation E_p_CAPpa # Paasche price index for capital input #
p_CAPpa = 100*I_CAPPP*del_unity;

```

Equation E_bCAPshr # Capital shares by ind #
  (All,i,IND) bCAPshr(i) = xlcap(i)+plcap(i) - wlcap_i;

Equation E_xCAPcs_o
  # Introduces forecasts of capital by industry into policy sims. #
  (All,i,IND) xcaptot_o(i) = xlcap(i) + f_xcaptot_o(i);

Equation E_bCAPsh_o
  # Introduces forecasts of capital shares by industry into policy sims. #
  (All,i,IND) bCAPshr_o(i) = bCAPshr(i)+ f_bCAPshr_o(i);

Equation E_PCAPcs_o
  # Introduces forecasts of capital rental price into policy sims. #
  (All,i,IND) pCAPtot_o(i) = plcap(i) + f_pCAPtot_o(i);

Equation E_wCAPtot_o
  # Introduces forecasts of nominal investment into policy sims. #
  wCAPtot_o = wlcap_i + f_wCAPtot_o;

! Labour input in current production !
Coefficient
(All,i,IND) LEVXLAB(i) # Quantities of labour input by industry #;
(Parameter)(All,i,IND)
  LEVXLABB(i) # Quantities of labour input by industry, base #;
(Parameter)(All,i,IND)
  LEVXLABB_O(i) # Qty of labour input by industry, f'cast base #;
(All,i,IND) LEVXLAB_O(i) # Qty of labour input by industry f'cast #;
(All,i,IND) LEVPLAB(i) # Price of labour input by industry #;
(Parameter)(All,i,IND)
  LEVPLABB(i) # Price of labour input by industry, base #;
(Parameter)(All,i,IND)
  LEVPLABB_O(i) # Price of labour input by ind in f'cast, base #;
(All,i,IND) LEVPLAB_O(i) # Price of labour input by industry in f'cast #;
(All,i,IND) BLABSH(i) # Share of labour in industry i #;
(All,i,IND) BLABSHO(i) # Share of labour in industry i, forecast #;
(Parameter)(All,i,IND)
  BLABSHBO(i) # Share of labour in industry i, forecast base #;
(Parameter) I_LABQL # Labour, Laspeyres quantity #;
(Parameter) I_LABPL # Labour, Laspeyres price #;
(Parameter) I_LABQP # Labour, Paasche quantity #;
(Parameter) I_LABPP # Labour, Paasche price #;
  VLAB_T # Labour, value #;
  VLAB_T_O # Capital, forecast value #;
(Parameter) VLAB_TB # Labour, value, base #;
(Parameter) VLAB_TB_O # Labour, forecast value, base #;
(Parameter) DEV_LABV # Labour, deviation value #;

Read
LEVXLAB from file EXTRA Header "LXLB";
LEVXLAB_O from file EXTRA3 Header "LXLB";
BLABSH from file EXTRA Header "LBSH";
BLABSHO from file EXTRA3 Header "LBSH";
LEVPLAB from file EXTRA Header "LPLB";
LEVPLAB_O from file EXTRA3 Header "LPLB";
VLAB_T from file EXTRA Header "VLT";
VLAB_T_O from file EXTRA3 Header "VLT";

Formula
(Initial)(All,i,IND) LEVXLABB(i) =LEVXLAB(i);
(Initial)(All,i,IND) LEVXLABB_O(i) = LEVXLAB_O(i);

```

```

(Initial)(All,i,IND) BLABSH(i) =
    (1-DUM_YEAR1)*{VLAB(i)/VLAB_I} + DUM_YEAR1*BLABSH(i);
(Initial)(All,i,IND) BLABSHO(i) =
    (1-DUM_YEAR1)*{VLAB(i)/VLAB_I} + DUM_YEAR1*BLABSHO(i);
(Initial)(All,i,IND) BLABSHBO(i) = BLABSHO(i);
(Initial)(All,i,IND) LEVPLABB(i) = LEVPLAB(i);
(Initial)(All,i,IND) LEVPLABB_O(i) = LEVPLAB_O(i);
(Initial) I_LABQL=Sum(i,IND,
    BLABSHBO(i)*(LEVXLABB(i)-LEVXLABB_O(i))/LEVXLABB_O(i));
(Initial) I_LABPL=Sum(i,IND,
    BLABSHBO(i)*(LEVPLABB(i)-LEVPLABB_O(i))/LEVPLABB_O(i));
(Initial) VLAB_T = (1-DUM_YEAR1)*VLAB_I + DUM_YEAR1*VLAB_T;
(Initial) VLAB_T_O = (1-DUM_YEAR1)*VLAB_I + DUM_YEAR1*VLAB_T_O;
(Initial) VLAB_TB = VLAB_T;
(Initial) VLAB_TB_O = VLAB_T_O;
(Initial) DEV_LABV= (VLAB_TB - VLAB_TB_O)/VLAB_TB_O;
(Initial) I_LABPP = (1+DEV_LABV)/(1+I_LABQL)-1;
(Initial) I_LABQP = (1+DEV_LABV)/(1+I_LABPL)-1;

```

Variable

```

(All,i,IND) xlabtot_o(i) # Qty of labour input in forecast simulation #;
(All,i,IND) f_xlabtot_o(i) # Allows exog.of forecast labour input #;
(All,i,IND) blabshr_o(i) # Labour shares by ind , forecast # ;
(All,i,IND) blabshr(i) # Labour shares by ind # ;
(All,i,IND) f_bLABshr_o(i) # Allows exog. of forecast labour shares #;
x_LABla # Laspeyres quantity index for labour input #;
x_LABpa # Paasche quantity index for labour input #;
(All,i,IND) pLABtot_o(i) # Labour rental price by industry in forecast #;
(All,i,IND) f_pLABtot_o(i) # Allows exog. of forecast labour rentals #;
p_LABla # Laspeyres price index for labour input #;
p_LABpa # Paasche price index for labour input #;
wLABtot_o # Value of labour input, forecast #;
f_wLABtot_o # Allows exog. of value of labour input, f'cast #;

```

Update

```

(All,i,IND) LEVXLAB(i) =xLAB(i);
(All,i,IND) LEVXLAB_O(i) =xLABtot_o(i);
(All,i,IND) BLABSH(i) =bLABshr(i);
(All,i,IND) BLABSHO(i) =bLABshr_o(i);
VLAB_T = wllab_i;
VLAB_T_O = wLABtot_o;
(All,i,IND) LEVPLAB(i) =pLAB(i);
(All,i,IND) LEVPLAB_O(i) =pLABtot_o(i);

```

Equation E_x_LABla # Laspeyres quantity index for labour input #
x_LABla = 100*I_LABQL*del_unity;

Equation E_x_LABpa # Paasche quantity index for labour input #
x_LABpa = 100*I_LABQP*del_unity;

Equation E_p_LABla # Laspeyres price index for labour input #
p_LABla = 100*I_LABPL*del_unity;

Equation E_p_LABpa # Paasche price index for labour input #
p_LABpa = 100*I_LABPP*del_unity;

Equation E_bLABshr # Labour shares by ind #
(All,i,IND) bLABshr(i) = xllab (i)+pllab (i) - wllab_i;

Equation E_xLABcs_o

Introduces forecasts of labour by industry into policy sims.
(All,i,IND) xlabtot_o(i) = xllab (i) + f_xlabtot_o(i);

Equation E_bLABshr_o

Introduces forecasts of labour shares by industry into policy sims.
(All,i,IND) bLABshr_o(i) = blabshr(i)+ f_blabshr_o(i);

Equation E_PLABcs_o

Introduces forecasts of labour rental price into policy sims.
(All,i,IND) plabtot_o(i) = pllabor (i) + f_plabtot_o(i);

Equation E_wLABtot_o

Introduces forecasts of nominal investment into policy sims.
wlabtot_o = wllab_i + f_wlabtot_o;

!***** END OF FILE *****!