IMS D703

DSP Development System

Product Overview

Features

- Numerically accurate simulator of the IMS A100 Cascadable Signal Processor
- Software simulation of both 'raw' IMS A100 cascade and IMS B009
- Comprehensive IMS T212-based IMS A100 driver optimised for IMS B009 facilities
- Interactive mode providing direct access to all software models
- User definable application tasks, written in occam 2
- Capable of simulating any configuration of IMS A100 devices
- Wide range of examples supplied, including FIR, convolution, correlation, DFT, and algorithm partitioning techniques
- Simple and efficient software to hardware development route
- occam 2 language for data generation and control of user applications
- Multi-window graphics interface
- Range of DSP design aids, including FIR, DFT design tools
- Runs on any IBM PC (with CGA/EGA) hosted transputer/occam 2 system
- All occam software components supplied in both fully documented source and binary code forms

7.1 Introduction

The IMS D703 DSP Development System is a comprehensive software tool for developing DSP applications at a number of different levels, e.g.

- Optimal (Remez exchange) FIR filter design.
- Numerically accurate simulation of a filter on the IMS A100 software model with test data.
- Software emulation of an entire IMS A100/Transputer based DSP system.
- Evaluation of a filter on 'live' data in real time (using IMS A100s on an IMS B009 evaluation board).

The IMS D703 is based on a software harness which controls the data flow between a number of processes providing facilities such as:

- Accurate simulation of one or more IMS A100s.
- Communication with the filing system, I/O etc. of the host PC.
- Communication with other evaluation hardware (e.g. the IMS B009 or any transputer based system).
- Application software.

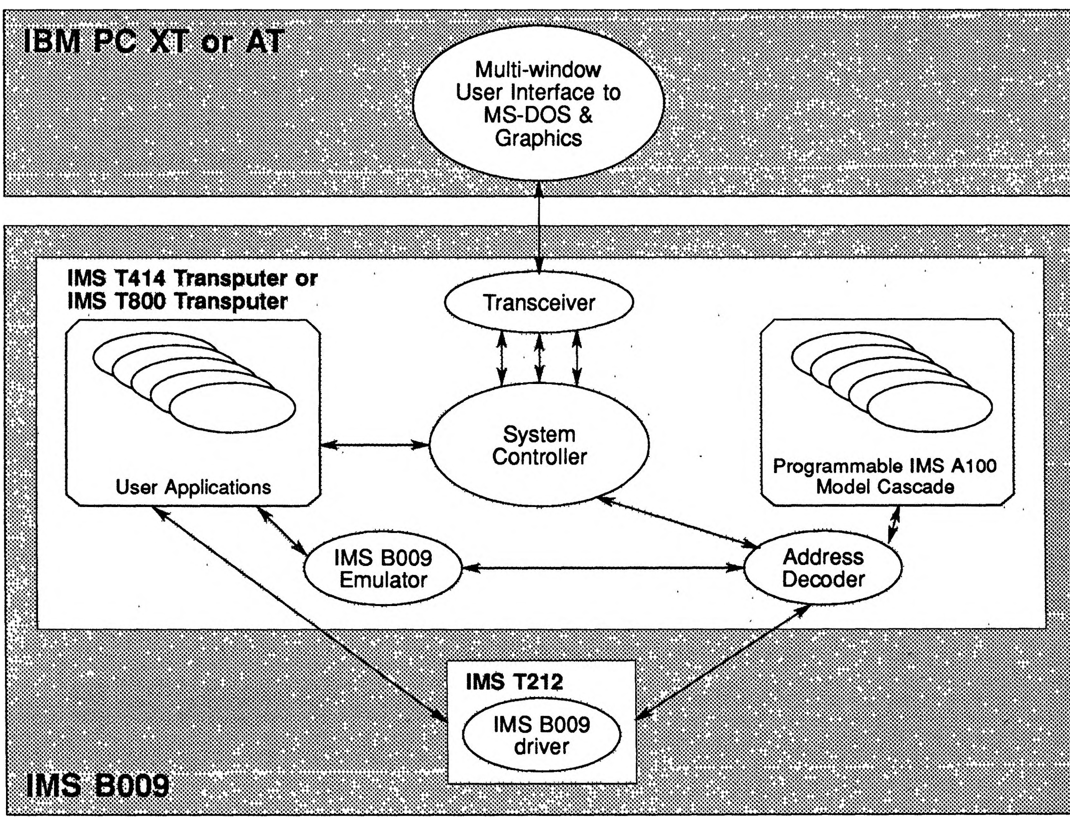

Figure 7.1 IMS D703 structure

An example of an application software package is the filter design and evaluation package provided in the IMS D703. This allows the user to interactively specify a filter which is then designed using the Remez exchange technique. Having designed a filter this application package then allows the user to apply test data patterns to either a simulation of the filter or to a hardware implementation of the filter (using the 4 IMS A100s on an IMS B009, if available). The input and output waveforms of the filter can be displayed in different graphics windows.

The application packages provided within the IMS D703 allow the user to interactively experiment in a number of DSP areas (e.g. filter design and using correlation to detect a signal in noise) without modifying the IMS D703 software.

Users can perform more elaborate DSP system simulations by writing their own application packages to work within the harness of the IMS D703. This allows the user to concentrate on developing algorithms and techniques while using the resources for file I/O, simulation etc. provided by the IMS D703.

7.2 Requirements

The IMS D703 is supplied as both (IMS T414) executable code and occam source code. If the user wants to modify the IMS D703 or to recompile it to execute on a different 32 bit transputer (e.g. IMS T800) the IMS D700 Transputer Development System is required. The IMS D703 and IMS D700 packages are designed to run on a transputer (T4 or T8 family) with at least 1 Mbyte of memory, hosted by an IBM PC XT, AT or compatible running MS-DOS. The IMS D703 also requires an EGA/CGA graphics display.

A suitable combination of products is:

IMS D703, IMS D700, IMS B009-1, IMS B404

Alternatively an IMS B008 can be substituted for the IMS B009-1; this only allows software simulation of IMS A100s.

7.3 Software Description

7.3.1 User Applications

A 'user application' is an occam process which can control any of the IMS A100 systems accessible to the IMS D703. By combining the services of the system controller with those of the IMS B009 driver, application programs can be written to perform any required task. In order to make these programs as simple a possible to write, a standard occam 'harness' is provided which includes a comprehensive set of standard procedures, constants, and interfaces to the rest of the IMS D703 environment. All the example applications supplied with the IMS D703 use this standard structure. They include code for convolutions, Discrete Fourier Transforms and FIR design using both hardware and software versions of the IMS A100.

Since the IMS D703 is implemented using the occam 2 language, user applications themselves are written in occam. Thus an advanced high level language is used for programming the system, rather than awkward text files or specialised macro languages. The range of standard procedures and examples supplied means that users can quickly modify an application to meet their specific requirements, whilst achieving a high level of performance. As users become more confident, they can develop their own optimised systems using parts of the IMS D703 as required. Indeed, because the IMS D703 executes entirely on the transputer host, dedicated applications can be developed which enable the IMS B009 to act as a high performance 'turbocharger' to an existing MS-DOS applications.

```
PROC FIR.example(CHAN OF ANY to.controller, from.controller)
  ... STANDARD     services
  ... USER_DEFINED data and PROCs

  VAL descriptor IS "128 tap FIR demo" :

  SEQ
    ... STANDARD initialisation
    WHILE TRUE -- loop forever
      SEQ
        ... evaluate coefficients
        -- Load coefficients into the A100s
        SEQ address=ccr.base FOR number.of.stages
          A100.write(address, coefft[i])

        -- Pass test data through A100s
        SEQ i = 0 FOR test.data.size
          SEQ
            A100.write(DIR, test.data[i])
            A100.read(DOL, result.data[i])
    stop()
:
```

Figure 7.2 Example user application for an FIR filter

A range of applications are provided to demonstrate the system, and provide a valuable means of evaluating the IMS A100. These include:

- 1D and 2D Convolution
- Correlation for Pulse Compression
- Discrete Fourier Transforms using the Prime Number Transform
- FIR Coefficient Design (using Remez Exchange method)

7.3.2 IMS A100 Model

At the heart of the IMS D703 is the system level occam 2 software model of the IMS A100. This provides a complete numerically accurate model of the IMS A100 in all modes of operation. Any configuration of cascaded IMS A100s can also be modelled, by joining the models together with occam channels. Also, since the model is written in occam, a model cascade can be distributed across an array of transputers for improved performance.

System simulations can be performed by connecting the Standard Microprocessor Interface (SMI) channels of one or more IMS A100 models to an address decoder model. This enables memory mapped systems to be modelled, and the address decoding scheme of a prototype system tested. Several examples of these are provided with the IMS D703, including a simple linear address space model, and an emulation of the IMS B009 address decoder. Alternatively, test data can be supplied to the data input channels, which emulate the external Data INput (DIN) port of the IMS A100. Likewise, the output can be received on the data output channel of the model (DOUT of the IMS A100), or read from the DOL/DOH registers via the SMI channels.

7.3.3 Address Decoder

To model the behaviour of a microprocessor-controlled IMS A100 based system, an address decoder process is provided. This receives all memory read and write requests for the IMS A100s, and decodes the addresses according to the selected decoding scheme.

Three modes of operation are provided:

- Raw Cascade Mode: Each device of the IMS A100 software model cascade is allocated 128 contiguous address locations. A special address is also provided so that a single write operation loads data into the Data Input Registers (DIRs) of all the devices in the cascade at the same time.
- IMS B009 Emulator Mode: A fixed cascade size of four IMS A100 devices is used, and the address decoder emulates the decoding used on the IMS B009 hardware.
- IMS B009 Hardware: All memory requests are passed directly to the IMS B009 driver. Additional commands are provided to reset and boot the IMS T2xx on the IMS B009 with a new driver if required.

7.3.4 IMS B009 driver

Another feature of the IMS D703 is the IMS B009 driver. The IMS B009 is a high performance DSP board, which contains four IMS A100s with a hardware optimised interface to the IMS T2xx controller. In order to control this board, and make full use of features such as address mapping and special block move modes, a software driver is provided which executes on the IMS T2xx of the IMS B009.

During startup, the IMS D703 will look for the IMS T2xx, and if present it will automatically bootstrap the IMS T2xx with the IMS B009 driver. This enables the user to access the resources of the IMS B009, and provides a means whereby user applications can perform complex operations at high speed with minimal knowledge of the IMS B009 hardware.

For certain applications demanding maximum performance, more optimised drivers may be required. The standard IMS T2xx-based driver provided with the IMS D703 gives users an excellent framework from which highly optimised drivers can be rapidly and reliably produced. Also, user applications can bootstrap the IMS T2xx directly, so that experienced users can create optimised drivers for each application if necessary, within a common environment.

7.3.5 IMS B009 Emulator

A software emulation of the IMS B009 loaded with the driver described above is also provided. IMS D703 users can thus develop code for the IMS B009 without having the hardware physically present. Using the standard procedures provided, users can produce applications that can execute with either the IMS B009 software emulator or the IMS B009 hardware without modification.

7.3.6 System Controller

The user environment is designed to enable DSP engineers to develop new algorithms quickly, and explore their performance without learning large command repertoires or understanding the software complexities of the IMS D703. The system controller controls this environment by providing a range of interactive services for debugging and monitoring of the system. 'Interactive' commands are entered using the keyboard, and are processed by a simple command line interpreter. Included are commands for reading and writing locations, plotting data both graphically and textually, monitoring messages from any part of the system, executing user applications, and controlling the various modes of operation. A full hierarchical online help facility is also provided, which can be customised to users' needs. A summary of commands is given in figure 7.3.

User application tasks can access additional functions via a special group of services, known as 'Binary' commands. These services are designed to execute far faster than the interactive commands, by processing commands in internal binary format directly rather than converting each command from text to binary prior to execution. Facilities that involve large amounts of data, such as plotting, are only provided with binary commands. The IMS B009 driver can also be accessed using binary commands. Both interactive and binary commands can be executed by user applications.

A keystroke file facility is provided, so that users can save all keyboard actions to file, and replay them. Keystroke files can also be created externally using any text editor or MS-DOS application, so that users can feed data gathered from external sources into the system, and capture the results for subsequent processing.

INTERACTIVE

- Application
- Continue
- Dump
- Execute
- Help
- Keep
- Quit
- Read
- Set
  - CascadeSize
  - Decoder
    - A100model
    - B009model
    - B009hardware
  - RecordFile
  - Graphics
- Write

BINARY

- BootT2
- FileIO
  - Open
  - Close
  - ReadByte
  - WriteByte
  - ReadBlock
  - WriteBlock
- ReadLocation
- WriteLocation
- PlotText
- Plotgraphics
- StopApplication

Figure 7.3 IMS D703 Command Summary

7.3.7 Multi-Window MS-DOS Interface

A multi-window user interface MS-DOS program is provided, which acts as a bridge between the resources of the IBM PC and the transputers. This program enables users to boot the host transputers, examine system status, plot data, access files, and manipulate the windows. In order to display the windows, and for plotting data graphically, an IBM PC fitted with CGA or EGA graphics hardware is required.

A comprehensive graphics suite is available to user applications, so that they can also make use of facilities such as automatic scaling, curve fitting, windows, and screen dumping. The graphics is based on the Turbo Pascal Graphics Toolbox, and access to most of the facilities of this package are provided.

7.4 Host Environment

The IMS D703 executes on any 32-bit transputer with at least 1Mbyte of RAM. If an IMS B009-1 and TRAM are being used, or the IMS D703 is being executed on an IMS B004 connected to an IMS B009-1, the standard IMS B009 driver will be automatically booted into the IMS T2xx. To modify the IMS D703 source code, users will require an IMS D700 Transputer Development System (TDS) for occam 2. The TDS runs on the same hardware as the IMS D703.

The IMS D703 graphics environment and run time support requires an IBM PC XT or AT with at least 512kbytes of RAM, a hard disk, MS-DOS version 3.1 or later, and an IBM compatible CGA or EGA graphics adaptor with monochrome or colour monitor.

7.5 Ordering Details

Product	Part number
B009 Evaluation board	IMS B009-1
Transputer Development System	IMS D700
DSP Development Software	IMS D703