NASA Technical Memorandum 88802

NASA-TM-88802 19860019179

Hardware Configuration for a Real-Time Multiprocessor Simulator

Richard A. Blech and Anthony D. Williams
Lewis Research Center
Cleveland, Ohio

July 1986

```
ENTER:
          DISPLAY 40/6/1
      86N28651*#   ISSUE 19   PAGE 3084   CATEGORY 62   RPT#: NASA-TM-88802
      E-3141 NAS 1.15:88802   86/07/00  21 PAGES   UNCLASSIFIED DOCUMENT
UTTL: Hardware configuration for a real-time multiprocessor simulator
AUTH: A/BLECH, R. A.;  B/WILLIAMS, A. D.
CORP: National Aeronautics and Space Administration.  Lewis Research Center,
      Cleveland, Ohio.    AVAIL.NTIS
SAP:  HC A02/MF A01
CIO:  UNITED STATES
MAJS: /*COMPUTERIZED SIMULATION/*MICROCOMPUTERS/*PARALLEL PROCESSING (COMPUTERS)
      /*REAL TIME OPERATION
MINS: / DATA PROCESSING/ PARALLEL COMPUTERS/ USER MANUALS (COMPUTER PROGRAMS)
ABA:  Author
```

HARDWARE CONFIGURATION FOR A REAL-TIME MULTIPROCESSOR SIMULATOR

Richard A. Blech and Anthony D. Williams
National Aeronautics and Space Administration
Lewis Research Center
Cleveland, Ohio 44135

SUMMARY

The Real-Time Multiprocessor Simulator (RTMPS) is a multiple microcomputer system used to investigate the application of parallelprocessing concepts to real-time simulation. This users manual describes the set-up and installation considerations for the RTMPS hardware. Any modifications or future improvements to the RTMPS hardware will be documented in an addendum to this manual.

INTRODUCTION

E-3141

The Real-Time Multiprocessor Simulator (RTMPS) project at the NASA Lewis Research Center was aimed at developing multiprocessor hardware and software technology for low-cost, portable, user-friendly real-time simulators. To accomplish this task, an experimental multiprocessor system was constructed. The design was intended to provide as many interprocessor communications paths as possible using off-the-shelf microcomputer boards and minimal customized interfacing. This approach allowed maximum emphasis to be placed on software development, which is a critical element in the development of parallel processing systems, such as the RTMPS. Using the experimental RTMPS, a number of software issues were addressed by the RTMPS project. These included high-level programming languages, scheduling of intercomputer data transfer, techniques for program partitioning and operating system support.

As currently configured, the RTMPS system may be viewed as a combination of three major components: the system hardware and firmware; an operating system; and a programming language. The Real-Time Multiprocessor Operating System (RTMPOS) (ref. 1) was developed to provide a user-friendly interface to all of the RTMPS hardware components. It resides on a main processor called the Front End Processor (FEP) and facilitates control of the simulation and user interface. The Real-Time Multiprocessor Programming Language (RTMPL) (ref. 2) provides high-order language (HOL) programming capabilities to the RTMPS system. It translates a HOL source input to a time-efficient assembler source program, and is designed to support a variety of microprocessor configurations and types.

The RTMPOS and RTMPL software was designed to minimize hardware dependencies. For example, the RTMPS system could be based on a number of microcomputer types. The hardware characteristics are accounted for in a system executive, a set of macros, and a target computer definition file. The system executive performs rudimentary functions such as initialization, data transfer and interrupt handling. The system macros define basic mathematical and data transfer operations for the target hardware. The target definition file provides information to the RTMPL about the microcomputer programming model. To retarget the RTMPS system to a different microcomputer, changes are made to the above software elements. The bulk of the RTMPS software, contained in RTMPL and RTMPOS, remains unchanged.

This report addresses the RTMPS system hardware and is intended to provide the reader with the information necessary to be able to assemble the hardware elements into a working system. First, a brief description of the RTMPS hardware components is given. Then the hardware configuration and assembly details are discussed. The report describes the RTMPS system in its present configuration. However, variations to this configuration are possible based on the intended application. The available options are documented in the various user's manuals listed in the references.

RTMPS HARDWARE DESCRIPTION

This section describes the hardware components of the RTMPS system. A block diagram of the RTMPS hardware is depicted in figure 1. The architecture is dual bus, with multiple microcomputers on each bus. Each microcomputer communicates with a corresponding microcomputer on the opposite bus through a dual-port interface memory. A Front-End Processor (FEP) on the lower bus functions as the user interface, while the analog I/O on the upper bus extension provides an interface to analog devices.

The Front-End Processor

The front-end processor serves as the user interface to the RTMPS. It provides the means for simulation run-time operations such as program loading into each of the simulation processors, simulation mode control, data handling, and data output and display. It also services the simulation peripherals including the user's terminal(s), disk, and printer, and handles files and special function calls within its resident operating system.

The FEP used for the NASA RTMPS is the Motorola Exormacs Development System, based on the M68000 family of microprocessors, microcomputers, and peripherals. The basic system consists of the Exormacs chassis, the Exorterm 155 Display Console (terminal), a hard disk drive, a floppy disk drive, and a Centronics model 703 printer. The Exormacs chassis contains the following modules:

DEBug Module - Contains firmware for debugging user programs plus three I/O ports -- two RS-232C ports for terminal connections and one printer port.

MPU - Contains the main processor and the Memory Management Unit (MMU).

FDC (Floppy Disk Controller) Module - contains a floppy disk controller that supports up to four disk drives.

Memory boards - Added as desired in various sizes.

UIPC (Universal Intelligent Peripheral Controller) and DIM (Disk Interface Module) - Together perform all important disk I/O operations (for hard and floppy disk).

MCCM (Multi-Channel Communications Module) - Optional module that adds four RS-232 serial communications ports and one extra printer port to the system.

The Exormacs System Operations Manual (ref. 3) gives information about the operation of each of the components listed above. It also gives complete details on how to connect these components together to get a working system, along with any specifications that must be met regarding connectors, cable types, and cable lengths.

In addition to the basic system described above, a 128 K-byte memory board is included in the Exormacs system to be used exclusively by the RTMPOS operating system. This auxiliary memory is not contiguous with the main system memory so that the Exormacs' MMU cannot detect its presence, and therefore will not allocate it to one of the system procedures. This allows absolute memory addressing techniques to be used within the RTMPOS operating system program.

The FEP also contains a GSI (Graphics Strategies, Inc.) VGM-1024 VersaGraphic Module graphics board which runs directly on the Versabus. It adds the flexibility of being able to generate plots of data obtained during real-time simulation execution. An interface from the system's RTMPOS operating system allows direct calling of graphics routines and utilities during interaction with the operating system, or from within a simulation.

Interactive Information Bus (IBUS)

The bus on which the FEP and its associated hardware communicate is called the Interactive Information Bus, or IBUS. This bus is implemented, as described later, using the Motorola Versabus. It is on this bus that all FEP related data transfers occur -- for example, terminal I/O, printer output and control, graphics display transfers, and floppy and hard disk data transfers. This bus also serves as the user's interface to the simulation processors, and provides the data paths for sampling simulation variables, monitoring the system status, and setting the simulation mode.

The simulation processors running on the IBUS are called the Interactive Bus Processors (IBPs). These processors may be used to run a specific part of a simulation, to collect data from other processors running the simulation (to be sent to the user), or to do both of these tasks depending on the application requirements. These were implemented in the Lewis RTMPS system using Motorola VM02 Microcomputer Boards (ref. 4). Each of these boards is based on an 8-mHz 68 000 microprocessor, and contains 128 K bytes of dual-ported random access memory (RAM) that can communicate with the microprocessor's on-board local bus as well as the system bus external to the board (the IBUS, implemented using Motorola's Versabus). Additional resources include three programmable timers, an interrupt controller and interrupt generating hardware, and a system bus interface. Each board also contains an I/O channel used to communicate with devices not on the board's local or system bus.

Each processor on the IBUS is coordinated by a system executive. The executive contains code which initializes the processor, handles interrupts and facilitates interprocessor communication. It also contains routines which interface the processor to the RTMPOS and RTMPL.

Real-Time Information Bus (RBUS)

The RTMPS system contains a second bus called the Real-Time Information Bus (RBUS). This bus is used for data transfers related to simulation calculations. Since system operations and user I/O are all done on the IBUS, the RBUS can be used exclusively for simulation variables that must be quickly sent to other processors on the bus for immediate use. This allows the Real-Time Bus Processors (RBPs) to communicate with each other at an optimum speed regardless of how much traffic is on the IBUS. Because of this fact, the RBUS is also used when two or more IBPs need to communicate simulation data with each other. Like the IBPs, the RBP processors are implemented using VM02 boards, and are set up in an almost identical configuration. Each of the RBPs contains a system executive similar to that in the IBPs.

The RBUS can also be expanded to accommodate components other than the RBPs. For example, the RTMPS at Lewis allows analog devices to be monitored and controlled by analog-to-digital converters (ADCs) and digital-to-analog converters (DACs) attached to the simulator via the RBUS. These devices are described in a following section.

RTMPS Channels

The basic RTMPS configuration consists two groups of microcomputers -- the RBPs and IBPs -- operating on two different buses. The two groups must be able to communicate with each other to run a simulation. For example, the RBPs may need to get new variable values and execution commands from the user and the FEP via the IBUS, while the IBPs may need updated simulation variables from the RBUS processors to send to the FEP. To provide for this communication, a dual-ported memory interface was added to the RTMPS system consisting of customized circuitry which allows two different VM02 boards to communicate through their I/O channels. This memory interface serves as the link between processors on the RBUS and those on the IBUS. It consists of three 1 K blocks of RAM, arbitration and control circuitry, and an interrupt generator.

The combination of an IBUS processor (IBP), interface memory and RBUS processor (RBP) is called a channel. In theory, the number of simulator channels is only limited by the electrical characteristics of the bus. In practice, considerations such as size, mechanical layout, and power requirements are dominant in determining the maximum number of channels. The RBP processor in channel 0 is called the real-time controller (abbreviated RTX). It is responsible for synchronizing the other channels, maintaining simulation timing, and performing the I/O to external hardware (i.e., the analog interface). The IBP processor in channel 0 controls the operating mode (RUN, HOLD, or STOP) and provides real-time analysis functions as commanded by the FEP. These functions include data collection, event triggering, rate-of-change monitoring and peak detection.

Analog I/O

The RTMPS system has an analog interface to allow connection of devices such as sensors, actuators or analog computers. The interface consists of a Versabus to Multibus converter, a Multibus chassis and standard Multibus analog I/O cards. This (the Multibus) is the bus extension depicted in figure 1. Multibus was selected for the analog interface because of the wide variety of Multibus peripheral devices available. In this manner, other devices besides analog I/O cards could also be used (i.e., digital serial or parallel I/O). The Multibus interface is implemented as an extension of the RBUS to allow fast transfer of data between the analog peripherals and the RTMPS processors.

RTMPS SYSTEM ASSEMBLY

This section describes how the RTMPS System hardware components are assembled and configured to create the final system architecture. In this context, configuration refers to the process of selecting hardware options on microcomputer or peripheral boards. Typically, these options are enabled by setting switches or connecting posts (via jumper clips or wire wrap).

Front-End Processor (FEP)

The FEP for the RTMPS System is a 15-slot Exormacs system which is based on Motorola's Versabus. Housed within the Exormacs chasis are the Exormacs Development System, the system power supply, cooling fans, and a front control panel. The Exormacs Development System consists of the DEbug, MPU, Memory, UIPC, DIM, and MCCM Modules described in the RTMPS Hardware Description section of this manual. The graphics card used in the RTMPS system also resides on this bus (because it serves as an additional interface to the RTMPS user, the graphics card may be regarded as an extension of the FEP). Figure 2 shows the positioning of the FEP components in the Exormacs chassis. The last three slots are used to hold the IBPs.

All of the boards in the FEP must be set up for operation as described in the "Exormacs System Hardware and Software Configuration Manual" (ref. 5). The set up procedure for the IBPs is discussed in a following section. The 128 K RAM board used by the RTMPOS is unique in that it must be set up differently than the normal FEP RAM. There are four switches on the RAM board which select the base address of that board on the Versabus. Table I shows the switch settings which must be used to allow RTMPOS to correctly address this auxiliary memory. The remaining configuration settings for this board are as described in reference 5.

Note in figure 2 that there are three additional 128 K RAM boards in the FEP. These boards can be used for extra FEP memory or removed to allow for the addition of more IBPs. The current configuration of the FEP has these boards installed. However, this additional memory is not utilized by any of the RTMPS software.

The VGM-1024 graphics board must be configured according to the default settings specified in the "Versagraphic Module Hardware Manual" (ref. 6). This involves setting the base address for the memory-mapped I/O used by the board, an interrupt level and interrupt vector. The default settings specified in reference 6 are the ones used for the current RTMPS configuration. Reference 6 also gives specifics on connecting the board's video outputs to a RGB color video monitor.

The peripheral devices connected to the FEP are illustrated in figure 3. There are six RS-232 serial ports available on the Exormacs system equipped with a MCCM module. Two of these ports originate from the debug module and four from the MCCM module. The two ports on the debug board are designated ports CN00 and CN01 by the operating system software. The ports on the MCCM module are designated CN10 through CN13. Of the six available ports, three are used by the FEP in the current RTMPS configuration. Ports CN00 and CN13 are connected to Motorola Exorterm 155 terminals. These ports can be configured (baud rate, stop bits, etc.) as described in reference 5. The Exorterm terminals can be used to operate the RTMPS or for software development. Serial port CN01 is used as a connection to another general purpose terminal and a modem. This is accomplished by using an RS-232 switch as shown in figure 3. Only one device connected to CN01 through the switch may be operational at any time. Either device can be selected by configuring the RS-232 switch box as outlined in reference 7.

The modem is used to connect the FEP to a mainframe computer. The terminal, an ADDS Viewpoint A1, can be used to operate the RTMPS or for any software development function. It should be noted that since the ADDS terminal lacks some of the features available on the Exorterm 155 terminals, the cursor oriented editing functions of the Motorola supplied editor software will not function with this terminal. The line editing mode must be used as discussed in reference 8. The ADDS terminal has an auxiliary serial port which can be used to attach a serial printer, as shown in figure 3. The printer can be used to obtain a copy of whatever appears on the ADDS terminal screen. A Texas Instruments 850 printer (ref. 9) is used in the RTMPS system. The printer must be set for serial operation as described in reference 9. The printer also has dot graphics capability, and is utilized by the RTMPS graphics software to obtain a hard copy of the color graphics monitor when desired. In the graphics mode, the ADDS terminal serves as a transparent link between the FEP and the printer, since it does not respond to the printer graphics commands.

Serial port CN01 must be set up to operate at 1200 baud (set up procedure is described in ref. 5) since the printer requires this rate for proper operation in the serial mode.

In addition to the terminals described above, there is an additional printer, dual floppy disk drives and a hard disk drive. These peripherals are connected to the Exormacs system as described in reference 3.

The system should also be configured with various software options including the Versados operating system, and a means of editing, assembling, and debugging user programs. The Lewis RTMPS system uses Motorola's software packages for all of these functions.

IBUS MPUs

The Interactive Information Bus (IBUS) is, in actuality, the FEP's system bus -- in this case the Exormacs Versabus. Therefore, some of the 15 card slots will be occupied by the FEP and its associated support devices as previously described. The remaining slots can be used for the Interactive Bus Processor (IBP) boards. These boards, once properly configured, are placed in the last vacant slots of the Versabus card cage so that the FEP and other boards that deal directly with the user's I/O devices have a higher positional bus priority, and therefore, will be serviced first. This arrangement can be changed to match the user's needs. For example, if the user does not need to have much user and I/O interaction capabilities during simulation run time, the IBP simulation boards can be positioned so as to give them a higher priority on the bus than the other devices, thus making their use in simulation calculations almost as efficient as using processors on the RBUS. Their interrupt priorities may also be set high or low depending on the application requirements.

The Motorola VM02 microcomputer boards as shipped from the factory must be configured by the user before they can be used in the RTMPS System. This involves the installation of jumper clips and user-programmed devices as described in the VM02 users manual (ref. 4). The jumper clips allow the user to select various hardware options and modes of operation. The VM02 boards are initially configured at the factory and have default jumpers installed. The user has the option of leaving the default jumpers or changing them to obtain different hardware options. The user programmed devices include the bootstrap EPROM and address decoding PROMs. The bootstrap EPROM contains the code which is executed at power on. This code usually initializes the board, performs a self test, and then procedes to load an applications program. The decoding PROMs contain information telling the microcomputer board which addresses it should respond to during various machine cycles.

Table II summarizes the microcomputer board jumpers which must be installed for RTMPS operation. The location of each of the jumpers can be found from figure 2-1 of the VM02 users manual (ref. 4). Jumper J11 enables selected bus interrupts to occur on the microcomputer board. Jumper J16 defines which board is the system controller board. The RAM base address jumper (J15) is set as specified in note 2 of table II. The settings for J15 were selected to place the VM02 dual-port RAM at addresses starting just after the last address used by the FEP. Any address may be selected as long as: (1) it starts on a 256 K byte boundary and (2) it is not an address assigned to the FEP or any of its peripherals. Each processor must have a unique RAM base address. Finally, the serial port jumpers J25 and J26 must have all clips removed for all IBUS processors. This effectively removes all serial I/O signals from the P2 connector of the board. These signals could interfere with the Exormacs development system if not removed.

After the correct jumpers have been installed, the bootstrap EPROMs must be installed. These EPROMs contain code for power-up initialization, self testing of the interface memory and the bootstrap loader for the system executive. The code for these EPROMs was developed at Lewis to interface the boards to the RTMPOS and RTMPL software. There are two sets of EPROMs, one for IBUS processors, the other for RBUS processors. Each set consists of one even byte EPROM and one odd byte EPROM. The even byte EPROM is installed in the socket labeled U12 on the VM02 board. The relative location of this socket is indicated in figure 4. Likewise, the odd byte EPROM is installed in the socket labeled U11. The index (notch indicating pin 1) on both EPROMs must be oriented as shown on figure 4. Intel 2732A EPROMs were used in the RTMPS system. However, any of a number of compatible devices may be used, as specified in the VM02 User's Manual (ref. 4).

The final step in preparing the IBUS VM02 boards is the replacement of two address decoding PROMs. These PROMs are used to monitor the microprocessor address bus and enable memory and/or peripheral circuitry if selected. The location of these PROMs is indicated in figure 4. They are ICs U138 and U144 (refer to VM02 schematic in the VM02 User's Manual) which are 82S129 bipolar PROMs. U138 decodes interrupt acknowledge cycles for interrupts initiated and received by the VM02 board. The standard U138 PROM provided with the VM02 board is programmed to respond to the Versabus standard interrupt acknowledge code (for a discussion of Versabus interrupt handling, see ref. 10). However, the Exormacs Versabus system uses a nonstandard interrupt acknowledge code. In order to generate interrupts on the Exormacs Versabus, the VM02 decoding PROM U138 must be changed to respond to this code. The necessary change is shown in table III. At the address listed, the device must be reprogrammed to the new data value shown. All other PROM locations remain unchanged. This change can be made using any PROM programming system that can handle the 82S129 device.

The U144 decoding PROM must likewise have one location modified as shown in table III. This change results from the write-protect decoding implemented by U144. The factory supplied PROM has locations $400 to $800 of RAM write protected from the Versabus. However, these locations must be written by the FEP via the Versabus to enable various executive functions. Therefore the write-protect mechanism must be disabled for these memory locations. The change to U144 is made using a PROM programming device as mentioned for the U138 changes.

RBUS MPUs

The Real-Time Interactive Bus (RBUS) is located within a completely separate Versabus card cage than that of the IBUS. Its main function is to provide a high speed data path for simulation calculations. It contains one processor -- the real-time controller RBP -- that synchronizes the channels, maintains simulation timing, and is the main processor used for communicating with the analog I/O boards. All of the other processors (the RBPs) have equal priority, and it is left to the Versabus bus arbitration circuitry to make sure that each processor gets an equal turn at the bus when needed. Apart from these distinctions, the RBUS is identical to the IBUS in almost every way.

The RBUS VM02 microcomputer boards must also be configured before they can be used in the RTMPS System. Table I can again be used to define the jumpers which must be installed for proper operation. The location of each of the jumpers can be found from figure 2-1 of the VM02 users manual. Note in table II that certain jumper settings are unique to the Channel 0 RBUS processor. The first is J11, which enables selected bus interrupts on the microcomputer board. There is a clip placed between pins 3 and 4 of J11 on RBUS processor 0 only. All other RBUS and IBUS processors have no clips placed on J11. The second jumper, J16, defines which board is the system bus controller. Since RBUS processor 0 is the controller for that bus, a clip must be placed between pins 2 and 3 of J16. All other RBUS and IBUS processors have a clip between pins 1 and 2 of J16.

The RAM base address jumper (J15) is set in exactly the same manner as for the IBUS processors. The settings in note 2 of table II can again be used. It is important to note that within a channel, both processors will have the same RAM base address. Thus the base address for the channel 0 RBP RAM (on the real-time bus) will be the same as the base address of the channel 0 IBP RAM (on the interactive bus).

Jumpers J25 and J26 enable serial port I/O operations on the board's P2 connector. For the RBUS processors, these clips can be left as they are since the serial port connections do not interfere with the bus operation. Finally, the bootstrap EPROMs are installed as described above for the IBUS processors. The only difference is that these EPROMs contain the RBUS processor version of the self-test and bootstrap loader code.

RTMPS Channel Interface

The dual-ported interface memory is housed in a rack mounted chassis. Inside the chassis are two wire-wrap panels which contain the interface memory circuitry. A diagram of the wire-wrap panel layout (as viewed from the top) is shown in figure 5. There are three identical interface memory circuits on the panels which are labeled I/F 1, I/F 2 and I/F 3. Each block of circuitry has two headers which connect to 2 50-pin flat ribbon cables. For channel 1 the headers are labeled 1J1 and 1J2, for channel 2 they are 2J1 and 2J2 and finally 3J1 and 3J2 for channel 3. The other ends of the flat ribbon cables connected to the "J1" headers of each interface memory channel are brought out to the Exormacs backplane. Here each cable is connected to its respective IBUS processor via the 50-pin I/O connector on the Exormacs backplane. This 50-pin I/O connector is the connection to the I/O channel of the VM02 IBUS processor which is resident in the Exormacs chassis slot. Likewise, the cables connected to the "J2" headers are attached to the 50-pin I/O channel connector on the back of the RBUS (Versabus) chassis. For more specific details on these connections, refer to the VM02 Users Manual (ref. 4) and the Exormacs Chassis User's Guide (ref. 11).

The interface memory circuitry requires a 5-V power source. The connections for the power source are two screw terminals as shown in figure 5. For the RTMPS, this power supply is provided separately from the system power supplies resident in the Exormacs and RBUS chasses. This is done to allow as many processor boards as possible in these chasses (without overloading the power supplies).

Analog I/O Boards

The analog I/O system for the RTMPS consists of analog-to-digital converter (ADC) boards, digital-to-analog converter (DAC) boards, a Multibus card cage (with power supply) and a Versabus to Multibus converter. The Versabus to Multibus converter system is illustrated in figure 6. It consists of one Versabus card, two Multibus cards and a 64-pin flat-ribbon cable. As shown, the large Versabus card is installed in the RBUS chassis, and the Multibus repeater card fits into connectors on the Versabus card. A 64-pin flat-ribbon cable is then connected between the Multibus repeater and the Multibus extender card. The extender card is then inserted into the first slot of a Multibus chassis. Any Multibus card may then be used in the chassis as if it were directly attached to the RBUS. This includes cards capable of being a bus master, such as intelligent controllers or microcomputer cards. In the RTMPS system, however, only analog I/O slave cards are used.

Before all of the cards can be installed in the system, there are several items which must be set up. The first is the large Versabus card of the Versabus to Multibus converter. Table IV lists the various jumper settings required. Particular attention should be made to the modification described in note 2 of table IV. This involves cutting one of the traces in the printed circuit board. The Multibus repeater card also requires that a trace be cut. This trace is between pins 1 and 2 of jumper block K1. Also jumper clips must be installed between pins 9 and 10, and between pins 11 and 12. Details about all of the above set up procedures can be found in the " Synergist Product Specification Manual " (ref. 12).

A nine-slot Multibus chassis with power supply was used in the RTMPS system. Most Multibus systems require the selection of a priority resolution scheme when multiple bus masters are resident in the system. Since only analog I/O slave cards are used in the RTMPS system, selection of a priority scheme was irrelevant. However, if it is desired to use devices capable of bus control, then the Multibus Chassis User's Guide (ref. 13) should be consulted regarding bus priority resolution selection.

The two types of Multibus analog boards used in the RTMPS are both manufactured by Data Translation, Inc. The first is a model DT711 analog input board. This board consists of a 12-bit analog-to-digital (ADC) converter and a 32 input multiplexer. It can be configured to operate as a 32 single-ended or 16 differential input system. The second board is a DT728 8 channel digital-to-analog (DAC) converter. As with the ADC board, each DAC is a 12-bit converter. Both the DAC and ADC boards operate as bipolar converters with a 10-V signal level, yielding a resolution of 4.88 µV per bit.

Table V lists the ADC board jumpering necessary to configure the board for use in the RTMPS. Note that the base address selection for each ADC board is incremented by $10 (16 decimal) starting at $20 000. This allows 16 memory-mapped I/O bytes for communication with each board. The jumper connections listed in the table are the only ones that should appear on the board. There should be no other connections. This implies that some factory installed default connections have to be removed. Note particularly the removal of the connection between jumper pins 78 and 87, which disables an unused interrupt option. The RTMPS system uses the differential input, 10-V bipolar configuration for the ADCs. The digital output is chosen to be in the 2's complement coding. These configuration options can be changed at the discretion of the user. For example, single-ended inputs with a 10-V unipolar range could be selected. For more details on these options, refer to the DT711 user's manual (ref. 14).

In a similar manner, the DT728 analog output board must also be configured. The jumpering for the configuration used in the RTMPS is listed in table VI. The same comments made regarding the base address selection on the ADC boards apply here. The only difference is the starting address for the first DAC board which is $20 100. Jumper connections are specified as pairs for the DAC boards - i.e., W-21 refers to 2 pins on the board which must be connected together. For further details, refer to the DT728 User's Manual (ref. 15).

Any combination of DAC and ADC boards may be used, up to the maximum of eight allowed by the nine-slot Multibus chassis (one slot is used for the Versabus-Multibus converter). The base addressing scheme outlined above must be adhered to, however. Therefore ADC boards must be addressed starting at $20 000 for the first board, with each additional board having an address that is $10 higher than the previous (i.e., $20 000 for the first board, $20 010 for the second, $20 020 for the third, etc.). DAC boards start at $20 100 for the first board and each additional board address is incremented by $10.

Chassis and Power Supply

The IBUS and RBUS electrical connections are provided by the Versabus card chassis. As mentioned previously, the IBUS chassis is actually the Exormacs chassis. The current RTMPS configuration uses the last three slots of this chassis for the IBPs. Power is supplied to these IBPs by the existing Exormacs power supply. The Exormacs power supply is sufficient for the existing FEP/IBUS configuration of the RTMPS. If additional or different IBUS processors are used, a larger power supply may be required. Up to three additional processors could be added by removing the three unused 128 K RAM cards. By using additional driver circuitry and card cages, the bus could be further expanded. Details on the Exormacs chassis can be found in reference 11.

The RBUS is implemented using Motorola Versabus card cages and a multiple output power supply. The base Versabus card cage has four slots available. This can be expanded to a maximum of 12 slots with two additional four-slot card cages and expansion connectors. The bus can be further expanded by using additional driver circuitry. It should be noted, however, that such additional circuitry adds to the bus propagation delay. This delay must be taken into account in the overall system design. The RBUS power supply is the Motorola M68KVMPS1 228 W switching power supply. This supply comes with a power monitor circuit which provides several Versabus power-fail and timing signals. Information on all of the RBUS components can be found in references 16 through 19.

CONCLUDING REMARKS

This report documents the set-up requirements and procedures for the RTMPS. These requirements are based on the current RTMPS configuration and its use as a research tool. The set-up requirements may change depending on the user's particular application. Most changes should be made after consulting the references in this report.

The current RTMPS system is in the process of being upgraded to accommodate new 32 bit microcomputer boards with hardware floating-point math and cache memory. This upgrade will require changes to the configuration just described. These changes will be documented in an addendum to this report.

REFERENCES

1. Arpasi, D.J.: RTMPL - A Structured Programming and Documentation Utility for Real-Time Multiprocessor Simulations. NASA TM-83606, 1984.

2. Cole, G.L.: Operating System for a Real-Time Multiprocessor Propulsion System Simulator. NASA TM-83605, 1984.

3. EXORmacs System, Operations Manual. 9th ed., Microsystems M68KMACS(D9), Motorola, Inc., 1982.

4. M68KVM02 Series VERSAmodule Monoboard Microcomputer, User's Manual. M68KVM02/D2, Motorola, Inc., 1983.

5. EXORmacs System, Hardware/Software Configuration Manual. M68KMACSCM(D1), Motorola, Inc., 1982.

6. VGM-1024, Versagraphic Module, Hardware Manual. Graphic Strategies, Inc., 1983.

7. 3R Universal Voltage and Current Loop Junction-Instructions. Digital Laboratories, Inc., Watertown, MS, 1977.

8. EXORmacs System, CRT Editor. 6th ed., Microsystems M68KEDIT(D6), Motorola, Inc., 1982.

9. Model 850 Printer, User's Manual. Texas Instruments, Inc., 1983.

10. VERSAbus Specification Manual. 4th ed., Microsystems M68KVBS(D4), Motorola, Inc., 1981.

11. EXORmacs System, EXORmacs Chassis User's Guide. M68KCHAS(D3), Motorola, Inc., 1982.

12. Synergist, Product Specification Manual. Hal-Versa Engineering, Inc., 1983.

13. ZX-609 Cardcage Hardware Reference Manual. Zendex, Inc., Dublin, CA, 1980.

14. User Manual for DT711/DT732 Analog Input and Analog I/O Boards. Document UM-00116-A-1224, Data Translation, Inc., Marlborough, MS, 1984.

15. User Manual for DT728 Multibus-Compatible D/A Board. Document UM-00476-1, Data Translation, Inc., Marlborough, MS, 1982.

16. M68KVMPM1 VERSAmodule System Power Monitor Module, User's Manual. M68KVMPM1/D1, Motorola, Inc., 1982.

17. M68KVMPS1 VERSAmodule System Switching Power Supply, User's Manual. M68KVMPS1/D1, Motorola, Inc., 1982.

18. Versamodule Chassis, Card Cage, Power Supply, and Power Monitor User's Guide. M68KVMESH(D1), Motorola, Inc., 1981.

19. Installation Instructions for VERSAmodule System Chassis Expansion Kit. M68KVMCHE(D1), Motorola, Inc., 1981.

TABLE I. - SWITCH SETTINGS FOR 128 K RTMPOS AUXILIARY MEMORY

	Position									
Switch	A15*	A16*	A17*	A18*	A19*	A20*	A21*	A22*	A23*	PRIM
S1	ON	ON	ON	ON	ON	ON	OFF	ON	ON	ON
S2	OFF	ON	ON	ON	ON	ON	OFF	ON	ON	ON
S3	ON	OFF	ON	ON	ON	ON	OFF	ON	ON	ON
S4	OFF	OFF	ON	ON	ON	ON	OFF	ON	ON	ON

TABLE II. - MICROCOMPUTER BOARD JUMPERING

Jumper	Conection	Description
J1	3-4	Microprocessor clock speed, 8 mHz
J2	1-2, 4-5	Microprocessor clock speed, 8 mHz
J3	5-6, 11-14, 18-21, 20-23	EPROM device type, 2732
J4	1-2	RAM parity check, disabled
J5	1-2, 3-4, 5-6, 7-8	RAM power source, Versabus
J6	2-3	Factory use
J7	1-2	RAM size, 128 K
J8	1-2, 4-5, 7-8, 10-11, 13-14, 16-17	User defined status bits
J9	4-5, 7-8, 10-11, 13-14	On board interrupt select
J10	5-6, 14-15	On board interrupt select
J11	[a]3-4	Bus interrupt select
J12	7-8	Local resource time out, 64 μs
J13	9-10	ROM access time, 450 ns
J14	7-8	Versabus time out, 64 μs
J15	(b)	RAM base address
J16	[c]2-3	System controller select
J17	1-2, 3-4, 5-6	Programmable timer set up
J18	2-4, 3-5	Programmable timer set up
J19	2-4	Programmable timer set up
J20	1-2, 4-5, 7-8	Serial port options
J21	7-8	↓
J22	2-3, 4-5, 8-9, 10-11	↓
J23	7-8	↓
J24	2-3, 4-5, 8-9, 10-11	↓
J25	(d)	↓
J26	(d)	↓
J27	2-3, 5-6, 7-8, 10-11, 14-15, 18-19	Bus priority level, level 1
J28	3-4	Bus priority level, level 1

[a]This jumper installed only on real-time controller (RBUS processor in channel ϕ). All other processors have no jumpers installed on J11.

[b]The jumpering here depends on which channel the board will reside in. For the three channel simulator the jumpers are:

Channel	Connection	Base address
0	1-2, 5-6, 8-9, 10-11, 14-15, 17-18	$24 0000
1	2-3, 4-5, 8-9, 10-11, 14-15, 17-18	$28 0000
2	1-2, 4-5, 8-9, 10-11, 14-15, 17-18	$20 0000

[c]Jumper 2-3 for all processors except the real-time controller. Jumper 1-2 on this board since it must be bus master on the real-time bus.

[d]These jumpers must be completely removed for all IBUS processors. For RBUS processors, the jumpers can be left as set of the factory.

TABLE III. - VMϕ2 DECODING

PROM CHANGES

Prom	Address	Old data	New data
U138	$C8	$0	$3
U144	$FE	$6	$4

TABLE IV. - VERSABUS TO MULTIBUS CONVERTER SET UP

Jumper/switch	Connection/setting	Description
SW1	3	Multibus block address select
SW2	F	Versabus block address select
SW3	0	Versabus priority level
K1	No connections	Multibus P2-Versabus I/O routing
K2	No connections	Unused Versabus and Multibus signals
K3	[a]1-2, 3-4, 5-6, 7-8, 17-18, 19-20	Master clock and strobe delays
K4	[b]1-2, 3-4, 6-8, 7-8, 9-10, 11-12	Interrupt enable and acknowledge
K5	7-8, 9-10, 11-12, [a]13-14	Various control signals
K6	1-2	Factory use
K7	5-12, 6-11, 7-10, [c]8-9	Address mode select
K8	1-2	

[a] Connected at factory via trace in PC board.

[b] Trace between 5-6, originally installed at factory as PCB trace, must be cut. Install jump clip between 6-8, 9-10, 11-12. Connections 1-2, 3-4, 7-8 are already implemented in PCB traces.

[c] Implemented using fusable links.

TABLE V. - ADC BOARD CONFIGURATION TABLE

Description	Connection
Base address select (B4)	(a)
Base address select (B3)	2, 6, 7-4, 3-8
A/D input type	23-24A, 25-27, 51-53B, 53A-56, 54-55B, 55A-57
A/D input range	58-71, 60-62A
A/D output code	67-70
Board interrupt function B	95-96
Systems bus interrupt level select B	[a]79-84
Transfer acknowledge delay	No jumpers installed
Pacer clock interval	40-38, 41-28
Reference jumpers	97-98, 48-49
Inhibit selection	No jumpers installed

[a] Jumpering is different for each board in the system. Addresses are selected starting at $20 000 and are incremented by $10. For a two-board ADC system the, the connections are:
Board 1 - 2, 3, 4, 5, 6, 9, 10, 11, 12, 13, 14, 15-8
Board 2 - 2, 3, 4, 5, 6, 9, 10, 11, 12, 13, 14, 15-8 and 4-16.

[b] The board comes from the factory with interrupt A connected to system level 1. This must be removed by disconnecting pin 78 from 87.

TABLE VI. - DAC BOARD CONFIGURATION TABLE

Description	Connection
Base address select	(a)
Addressing mode	W-21, W-22, W-23
Memory mapping select	W-42
Output range select	W-1, W-3, W-5, W-7, W-11, W-13, W-15, W-17
Multiplying DAC option	W-9, W-19
DAC input coding select	W-53, W-51
Transfer acknowledge delay	W-39
Bus priority control	W-55
Inhibit selection	No jumpers installed

[a] Jumpering is different for each board in the system. Addresses are selected starting at $20 100 and are incremented by $10. For a three-board DAC system, the connections are:
Board 1 - W-44, W-50
Board 2 - W-44, W-50, W-33
Board 3 - W-44, W-50, W-32

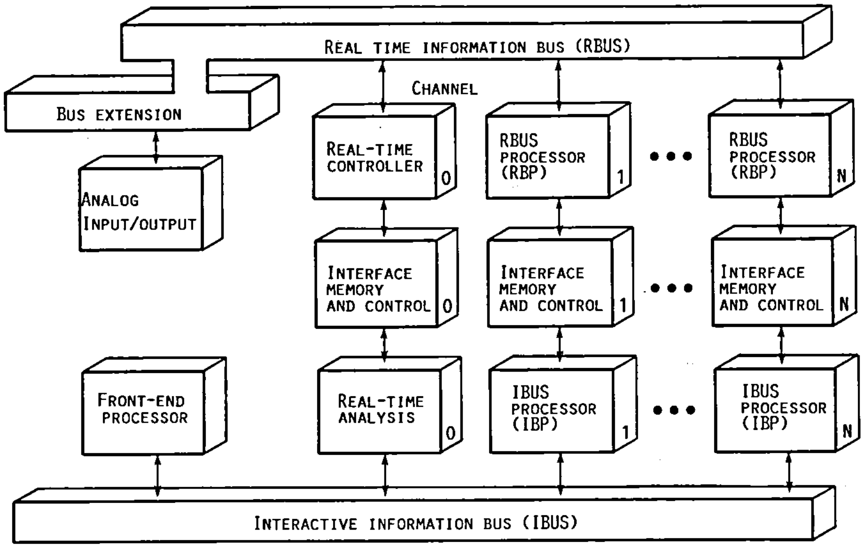

FIGURE 1.-GENERAL BLOCK DIAGRAM OF RTMPS HARDWARE.

SLOT ID	A1	A2	A3	A4	A5	A6	A7	A8	A9	A10	A11	A12	A13	A14	A15
	MPU	DEBUG	512K RAM	512K RAM	128K RAM	UDC IPC	UDC DIM	128K RAM	128K RAM	128K RAM	MCCM	GRAPHICS	IBP0	IBP1	IBP2

FIGURE 2.- FRONT-END PROCESSOR/IBUS CARD SLOT LAYOUT.

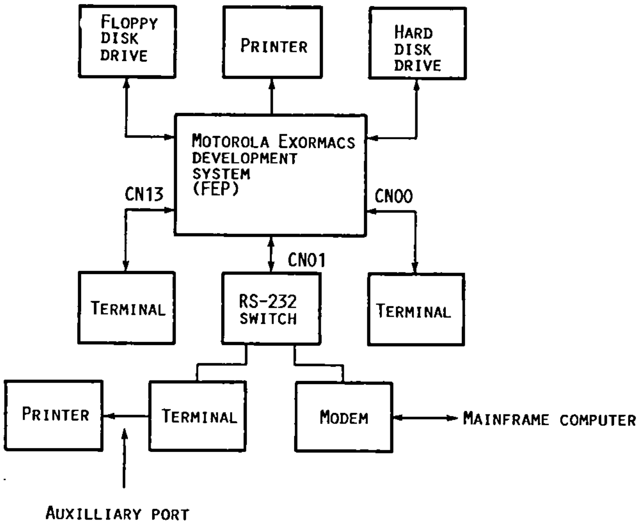

FIGURE 3.- FEP PERIPHERAL CONNECTIONS.

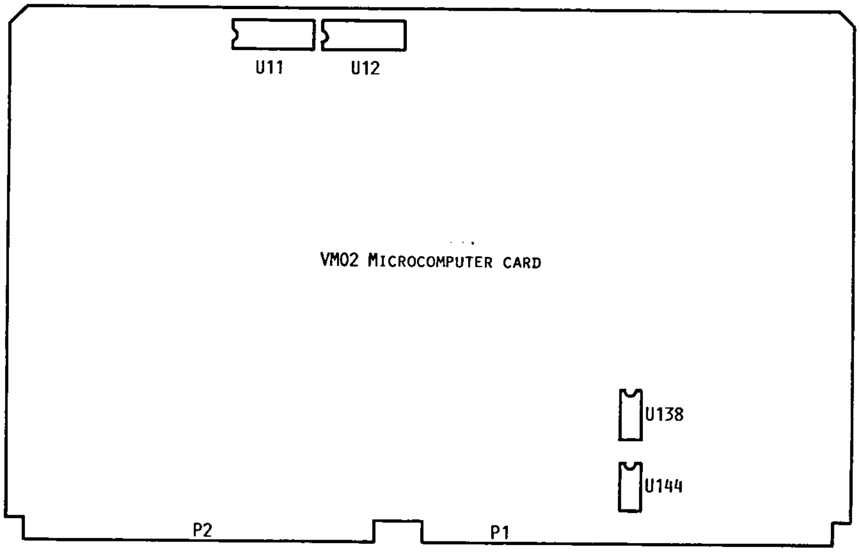

FIGURE 4.- BOOTSTRAP EPROM AND DECODING PROM LOCATIONS.

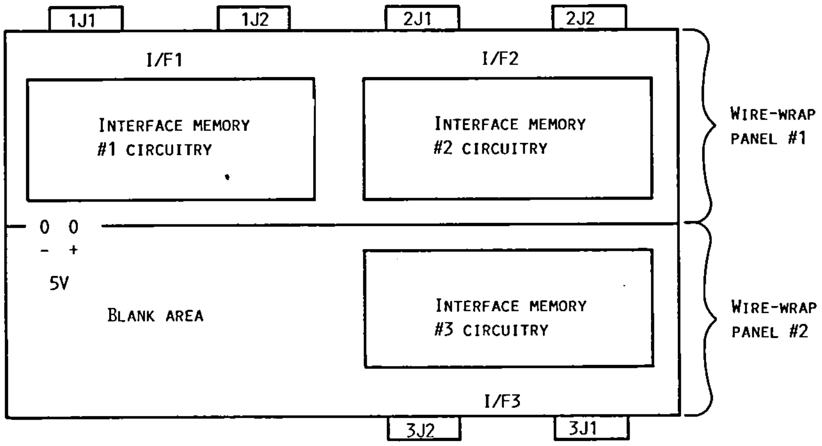
1J1
1J2
2J1
2J2
I/F1
I/F2
Interface memory #1 circuitry
Interface memory #2 circuitry
Wire-wrap panel #1
0 0
- +
5V
Blank area
Interface memory #3 circuitry
Wire-wrap panel #2
I/F3
3J2
3J1

Figure 5.- Interface memory chassis layout.

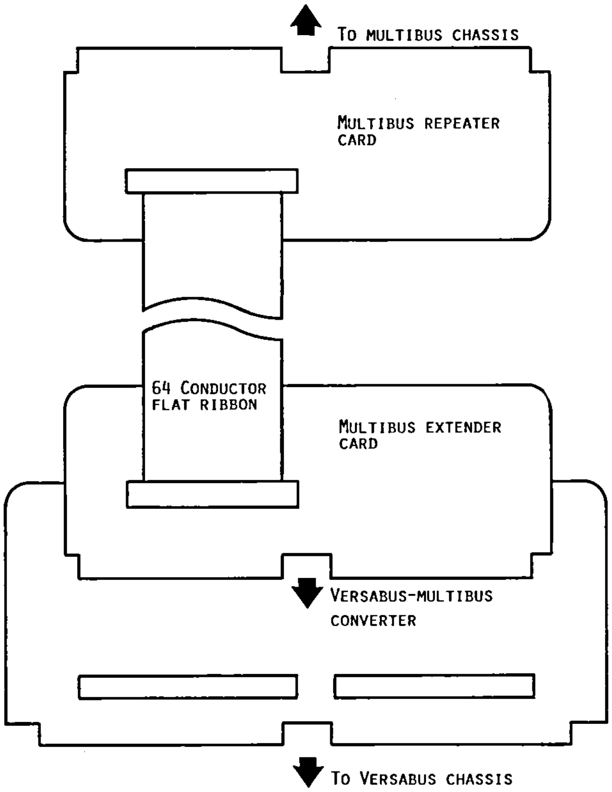

Figure 6.- Multibus to Versabus converter system.

1. Report No.
NASA TM-88802

2. Government Accession No.

3. Recipient's Catalog No.

4. Title and Subtitle

Hardware Configuration for a Real-Time
Multiprocessor Simulator

5. Report Date
July 1986

6. Performing Organization Code
505-62-21

7. Author(s)
Richard A. Blech and Anthony D. Williams

8. Performing Organization Report No.
E-3141

10. Work Unit No.

9. Performing Organization Name and Address

National Aeronautics and Space Administration
Lewis Research Center
Cleveland, Ohio 44135

11. Contract or Grant No.

13. Type of Report and Period Covered
Technical Memorandum

12. Sponsoring Agency Name and Address

National Aeronautics and Space Administration
Washington, D.C. 20546

14. Sponsoring Agency Code

15. Supplementary Notes

16. Abstract

The Real-Time Multiprocessor Simulator (RTMPS) is a multiple microcomputer system used to investigate the application of parallel- processing concepts to real-time simulation. This users manual describes the set-up and installation considerations for the RTMPS hardware. Any modifications or future improvements to the RTMPS hardware will be documented in an addendum to this manual.

17. Key Words (Suggested by Author(s))

Parallel processing
Real-time simulation
Microcomputer

18. Distribution Statement

Unclassified - unlimited
STAR Category 62

19. Security Classif. (of this report)
Unclassified

20. Security Classif. (of this page)
Unclassified

21. No. of pages

22. Price*

*For sale by the National Technical Information Service, Springfield, Virginia 22161