Data
Systems

PHILIPS

Service Manual

P800M
Interface and Installation Manual

Preliminary

A PUBLICATION OF
PHILIPS DATA SYSTEMS
APELDOORN, THE NETHERLANDS.

PUB. NO. 5122 991 26942

DATE April 1976

Great care has been taken to ensure that the information contained in this handbook is accurate and complete. Should any errors or omissions be discovered, however, or should any user wish to make a suggestion for improving this handbook, he is invited to send the relevant details to:

PHILIPS DATA SYSTEMS
SERV. DOC. AND TRAINING DEPT.
P.O. BOX 245, APELDOORN,
THE NETHERLANDS.

This Interface and Installation Manual is provided to aid a user wishing either to extend or modify an existing minicomputer system in the P800M family (P852M,P856M,or P857M),or to connect non-standard peripheral equipment to a system,or to integrate a P852M,P856M,or P857M into his own data processing system.

The manual provides technical data-such as functions, levels,and timings-for signals at the CPU General Purpose Bus and other interfaces,including interrupt and break request connections. It also describes,in general terms,the operation of the CPU when using programmed channel,I/O processor channel,or DMA controllers and gives examples of data exchanges which can take place.

A general description of control unit operation and I/O interface logic is given and information is provided to aid a user wishing to construct his own control units on printed wiring boards. Full details of all standard control units can be found in the appropriate manuals listed in the Appendix.

Section 2 of the manual provides installation information. The compact,yet versatile,nature of the P800M range creates no special difficulties for installation but the system possibilities are so diverse that it is not practical to give specific instructions on how to install each particular system configuration. Therefore information is given about the standard items of equipment available,to enable a suitable system arrangement to be determined and installed,and also general installation details which, it is considered,would be of use when installing any of the possible configurations. Reference should also be made to the documents issued by the peripheral equipment suppliers and to the other relevant manuals listed in the Appendix.

Great care has been taken to ensure that the information contained in this manual is accurate and complete. However,should a user find any errors or omissions, or wish to suggest improvements,he is invited to write his comments on the sheet provided at the end of the manual and send it to:

MANUAL WRITING SMALL COMPUTERS
at the address on the opposite page.

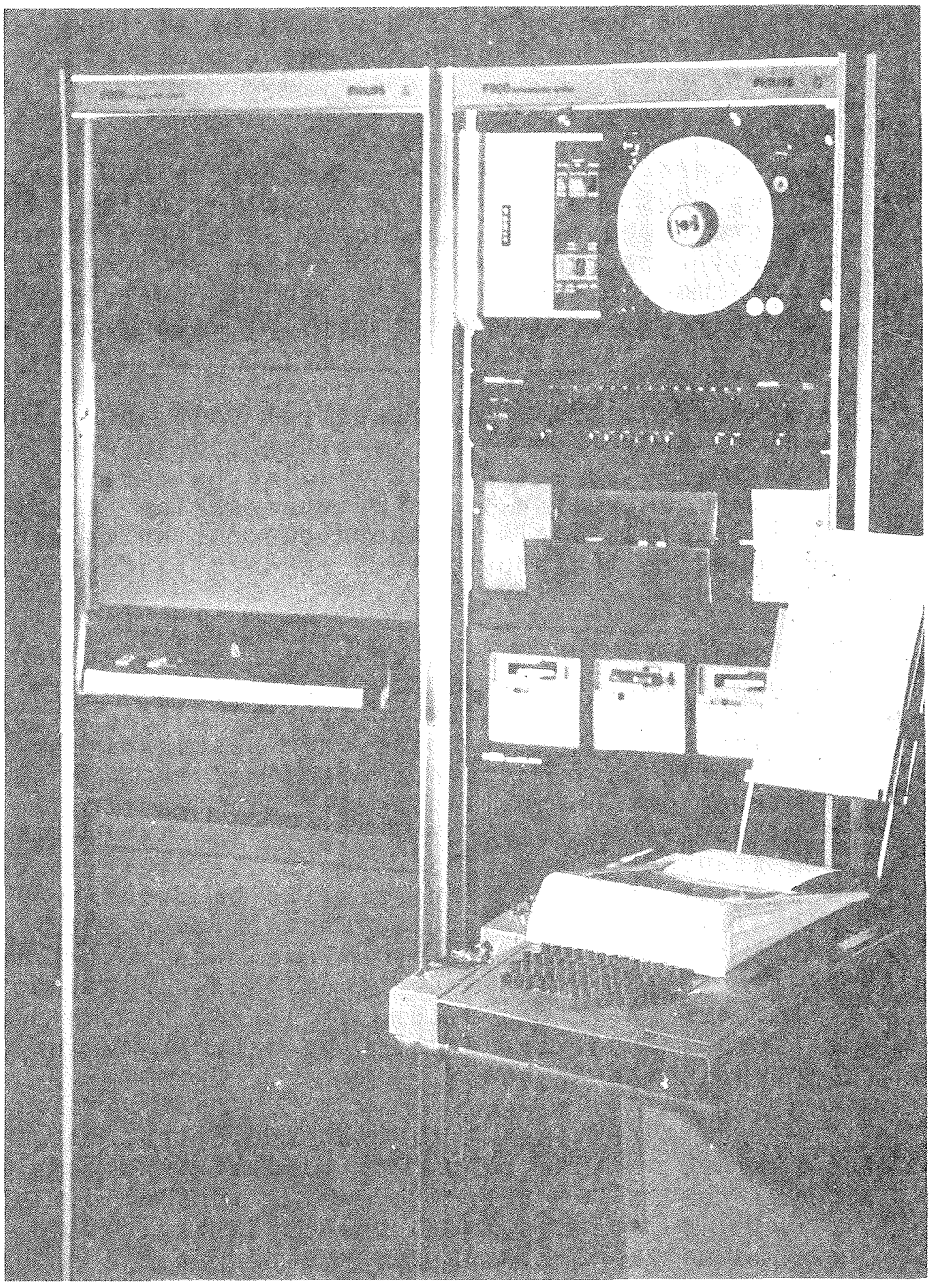


Table of Contents

	Page
Preface	III
List of illustrations	X
List of Tables	XIII
Summary	XIV
Definitions and Abbreviations	XVI
SECTION 1 - INTERFACING	
PART 1 INPUT/OUTPUT ORGANISATION	1-1
Chapter 1 General Description	1-3
Data Handling Unit P852M	1-4
Data Handling Unit P856M/P857M	1-8
Memory Management Unit (P857M)	1-11
Extra Facilities (P856M and P857M)	1-12
Memory Interleaving	1-12
Micro-diagnostics	1-12
Extended Control Panel	1-12
Input/Output Channels	1-13
General Purpose Bus	1-13
Master Slave Relationship	1-14
Priority Structure	1-14
External Connection	1-14
Standard Control Units	1-14
User Designed Control Units	1-16
Control Unit Operation	1-16
Interrupt System	1-16
Internal Interrupts	1-17
Timing Control Signals	1-18
Chapter 2 General Purpose Bus	1-19
GP Bus Signals	1-19
Chapter 3 Programmed Channel	1-24
I/O Instructions	1-27
Programming	1-27
I/O Processor and DMA Role	1-28

Page

Exchange Timings 1-28
Exchanges - Write External Register and Write in Memory 1-29
Exchange - Read External Register or Memory 1-32
Exchange - Master to Control Unit 1-34
Exchange - Control Unit to Master 1-35
Guard Times 1-41

Chapter 4 I/O Processor Channel 1-43
Operation 1-43
Initialization Operation 1-45
Exchange Operation 1-46
End of Block Transfer 1-47
Exchange of Variable Length Blocks 1-50
Relationship between Break Signal and Control Unit Address 1-50
System Organization Rules 1-51
Programming 1-51
Performance 1-51

Chapter 5 Priority Structure 1-53
CPU Priority 1-53
Bus Allocation System 1-53
System Operation 1-55
Timing of the Priority Chain 1-57
Overall Performance 1-58
Chaining of the Exchange 1-58

Chapter 6 Interrupt System 1-61
Interrupt System 1-61
Interrupt Handling 1-63
Interrupt Signal Encoding for a System 1-65
Interrupt Selection on BIECO-5 Lines 1-66
Timing of SCEIN Signal 1-67

Chapter 7 Digital Input/Output System (DIOS) 1-69
Configurations 1-69
Signals between DIOD Board and CPU 1-69
Signals between device and DIOD board 1-71
Operation 1-71
Initiation by a Call signal 1-73
Initiation by Software 1-75

Page

Timing for Control Lines 1-75
Sequences 1-77
Start Mode or Stop Mode 1-77

PART 2 INTERFACE CIRCUITS AND HARDWARE 2-1

Chapter 1 Introduction 2-3
Basic Mounting Boxes M1. M2, M4, M5 and Equipment Shelves E1 and E2 2-3
Printed Wiring Boards 2-3
Connector Details 2-5
Mounting Boxes Configuration 2-6
Connectors IOM and IOB 2-6
Method of Connection 2-8

Chapter 2 Bus - Programmed and I/O Processor Channel . . 2-11
Transmission Medium 2-11
Termination Resistors 2-11
Connection Details 2-12
Backpanel Addresses 2-12
GP Bus - Extension Connections 2-14
Bus Translator Board Connections 2-16
Break Request Connections on I/O Processor 2-19
GP Bus Interface Circuits 2-20
Receiver 0612 2-20
Emitter 1801 2-20
Equipment Shelf Division Rules 2-21
Division Rules 2-22
Break Request Connections 2-23

Chapter 3 control Units 2-25
CPU Commands 2-25
Address 2-26
Function Codes 2-26
Control Unit Sequencing Modes 2-26
Typical Control Unit - Logic Circuits 2-28
Address Decoding 2-30
Function Decoding 2-30
Sequencing 2-30
Control Flip-Flops 2-33

Page

Loading the CU Register 2-33
Reading from CU Register 2-34
Programmed Channel or I/O Processor Selection . . . 2-35
Circuit Boards 2-36
Physical Details 2-36
General Purpose Wiring Board 2-36
Wiring Board Rules 2-36
General Purpose Interface Board for Type 1 Control Unit 2-39
General Purpose Interface Board - Circuit Details . . 2-39

Chapter 4 Wiring Board Rules 2-43
Rules for Signals Remaining within a Board 2-43
Circuits with open Collector or a 3 State Output . . 2-44
Clock Circuits 2-45
Monostable Delay Circuit 2-45
Memory Circuit 2-45
Unused Inputs 2-46
Bus Interface Signals - Wiring Board Layout Rules . . . 2-47
Layout Rules - Double Face Board 2-47
Layout Rules - Multilayer Circuit 2-49
Other Signals - Wiring Board Layout Rules 2-49
Integrated Circuits Power Supply and Power Supply Decoupling . 2-50
Bus Interface ICs 2-50
Power Supply and Decoupling of other Circuits . . . 2-50
Power Supply Distribution on Logic Boards 2-51
Integrated Circuit Equivalence Table 2-51
Back Panel (Motherboard) 2-51

Chapter 5 CPU Power Supply Interface 2-55
Logic Signal and d.c. Supply Sequencing 2-57
Logic Signals 2-57
Real Time Clock 2-58
Power Control when using Equipment Shelves 2-59

Chapter 6 DIOS 2-61
Basic DIOS Connections 2-61
CALL Signals 2-62
Example of Input Circuit 2-62

Page

OK Signals 2- 63
Interrupt Level Selection 2- 64
Address Recognition 2- 64
Data Words Interface 2- 65
  Special Output Circuit 2- 65
  Input Data Circuit 2- 65
Pin Connections to DIOD Boards 2- 67

PART 3 INPUT/OUTPUT PROGRAMMING 3-1

Chapter 1 Introduction 3- 3
Peripheral Devices 3- 4
Basic Instruction Sequence 3- 4

Chapter 2 Input/Output Instructions 3- 7
  CIO - Control Input/Output 3- 7
  TST - Test Status 3- 8
  OTR - Output from Register 3- 9
  INR - Input to Register 3- 10
  SST - Send Status 3- 11
  WER - Write External Register 3- 12
  RER - Read External Register 3- 13

Chapter 3 Modes of Data Transfer 3- 15
Initiation of a Data Exchange 3- 15
Data Transfer in Wait Mode (Programmed Channel) . . . 3- 15
Data Transfer in Interrupt Mode (Programmed Channel) . . 3- 19
Data Transfer in Wait and Interrupt Mode (I/O Processor Channel . . . 3- 19
Special Device Handling 3- 19
Control Unit/Device Addressing 3- 23

Chapter 4 Coding of Input/Output Drivers 3- 27
Input/Output System 3- 27
M:IORM 3- 28
ENDIO 3- 29
COMIO 3- 30
M:RETR 3- 32

Chapter 5 Program Examples 3- 35
Example of an I/O Program in Interrupt Mode 3- 35
Example of a Program to read 20 characters via I/O Processor . . . 3- 37

SECTION 1 INTERFACING

PART 1

1.1 General Structure of CPU's 1-4
1.2 Block Diagram of P852M Data Handling Unit 1-5
1.3 Block Diagram of Data Handling Unit P856M/P857M . . 1-9
1.4 System Structure - Block Schematic 1-15

3.1 Instruction Flow P852M 1-25
3.2 Instruction Flow P856M/P857M 1-26
3.3 Instruction Format 1-27
3.4 Timing-Writing in Memory (or External Register) . . 1-31
3.5 Timing-Read Memory (or External Register) 1-33
3.6 Timing for an OTR Exchange 1-36
3.7 Timing for a CIO Exchange 1-37
3.8 Timing for INR, SST and TST Exchanges 1-40

4.1 I/O Processor Channel 1-44
4.2 Memory to CU Exchange-Timing Diagram 1-48
4.3 CU to Memory Exchange-Timing Diagram 1-49

5.1 Bus Control Priority Structure 1-54
5.2 Bus Request Timing 1-55
5.3 Typical Logic Assembly 1-56
5.4 Priority Chain Timing 1-57
5.5 Priority Chain Performance 1-58
5.6 Exchange Timing 1-59
5.7 Exchange Chaining-Master 1-CU, Master 2-RAM . . . 1-60

6.1 Interrupt System 1-62
6.2 Interrupt and Stacking System 1-64
6.3 Interrupt System Installation 1-65
6.4 Simplified Interrupt Encoding 1-66
6.5 Signal SCEIN Timing 1-67

7.1 DIOD Interfacing Signals 1-70
7.2 DIOD Board Logic-Block Diagram 1-72
7.3 GP Bus- Timing Diagrams 1-74
7.4 Timings for Control Lines 1-76

PART 2

1.1 Interface Connections 2-4

1.2 Printed Wiring Board-Connecter Layout 2-5

1.3 Layout of Type M1 Mounting Box Slots (P852M/P856M-100 Series) . . . 2-7

1.4 Layout of Type M2 Mounting Box Slots (P852M-200 Series) 2-7

1.5 Layout of Type M4 (and M4M) Mounting Box Slots (P852M/P856M/P857M-400 Series) . 2-7

1.6 Layout of Type M5 Mounting Box Slots (P857M-500 Series) 2-8

1.7 Break Request on I/O Processor Board 2-9

2.1 Terminator Bridge Adaption Networks 2-12

2.2 Emitter 1801 and Receiver 0612 Pin Layout 2-21

3.1 Control Unit Mode Sequences 2-27

3.2 Block Diagram of Typical Control Unit (P852M, P856M or P857M) . 2-29

3.3 Address Decode Logic 2-30

3.4 Function Decode Logic 2-31

3.5 Sequencing Logic 2-32

3.6 Control Flip-Flops 2-33

3.7 Loading the CU Data Register 2-34

3.8 Reading the CU Data Register 2-35

3.9 Control Unit Board Dimensions (Double CU Board) . . 2-37

3.10 General Purpose Wiring Board 2-38

3.11 General Purpose Interface Board (P852M/P856M/P857M) 2-40

3.12 General Purpose Interface Board-Logic 2-41

4.1 Power Supply Filter Circuit 2-51

5.1 CPU Power Supply 2-55

5.2 CPU Power Supply Interface Timing 2-57

5.3 Real Time Clock Signal 2-58

5.4 System Using External Power Supplies 2-59

6.1 CALL Signal Input Circuit 2-62

6.2 Circuit for Generation of OK Signal 2-63

6.3 Output Driver Circuit 2-65

6.4 Input Receiver Circuit 2-64

PART 3

3.1 Data Exchange Initiation-with TST 3-16
3.2 Data Exchange Initiation-without TST 3-17
3.3 Data Transfer in Wait Mode (Programmed Channel) . . 3-18
3.4 Data Transfer in Interrupt Mode (Programmed Channel) 3-20
3.5 Data Transfer in Wait Mode (I/O Processor) . . . 3-21
3.6 Data Transfer in Interrupt Mode (I/O Processor) . . 3-21

SECTION 1 INTERFACING

PART 1

3.1 Write in Memory CHA and MAD 15 bits 1-30
3.2 Read Memory CHA and MAD 15 bits 1-32
7.1 Command Instructions 1-73

PART 2

2.1 Backpanel Addresses-Connector 3 2-13
2.2 Connector IOM Connections 2-14
2.3 Connector IOB Connections 2-15
2.4 I/O Bus Connections to Connector 1 2-16
2.5 I/O Bus Connections to Connector 2 2-17
2.6 Break Request Connections to Connector 7 2-18
2.7 Break Request Connections to Connector 4 and 5 . . 2-19
4.1 Maximum Emitter Lengths 2-43
4.2 Integrated Circuit Substitutions 2-52
5.1 Output Supply Characteristics 2-56
5.2 Logic Signals Characteristics 2-56
6.1 Pin Connections for DIOD1 and DIOD2 Boards . . . 2-68
6.2 Additional Pin Connections for DIOD2 Board (P837-002) 2-69

PART 3

1.1 Connection of Peripheral Devices 3-3
3.1 Cassette Tape Operation 3-22
3.2 Moving Head Disc Operation 3-23
3.3 Example of CU/Device Addresses 3-24

This manual is divided into the sections and parts summarized below, followed by an Appendix containing reference information.

SECTION 1 INTERFACING

Part 1 Input/Output Organization

Describes in general terms the theory of operation and the structure of the computers and the integration of the input and output channels into the central processing functions. Examples of data exchanges, including timing diagrams, are given.

Part 2 Interface Circuits and Hardware

Provides technical details regarding the locations and signal connections to connectors, interface circuit characteristics, signal levels and input/output loads and control unit design information.

Part 3 Input/Output Programming

Contains the necessary information for the user to produce I/O drivers other than those already existing for standard equipment.

SECTION 2 INSTALLATION

Part 1 Site Considerations

Surveys various aspects which may require consideration when planning a site for a large configuration but which are not critical for the mini-computers themselves. Also provided in this part is information dealing with electrical supplies and interconnecting cables.

Part 2 Central Processors and Integral Equipment

Describes the various types of central processor and also the equipment which is either located within the actual processor main-frame or constitutes an extension of it. As well as physical details and environmental requirements information is also given concerning equipment interdependent relationship, d.c. requirements and sources, control interconnections etc., to aid in the determination and installation of a suitable system arrangment. To allow for the possibility of future reinstallation at another site one chapter in this part gives details for adapting the various d.c. power supply units to operate from different main voltages.

Part 3 Peripherals

Provides planning and installation data for each input and output device capable of being connected to the processing equipment described in Part 2. The data includes physical details, environmental and electrical power requirements and the standard lengths of the signal cables supplied with the equipment.

The method used to convert each device to operate from different mains voltages is described briefly; full particulars can be obtained from individual supplier manuals.

Part 4 Cabinetry

Gives details of the basic and extension racks, panels, mounting fixtures etc., which hold the equipment defined in Parts 2 and 3. The method of construction and the configuration rules governing the arrangement of the equipment in the racks are also described.

Part 5 Installation Procedure

Is a general guide which outlines the unpacking and installation procedures for the equipment and the steps required to prepare the system for commissioning.

Appendices

Contains a list of useful reference manuals, some of which are referred to in the text and an example of a typical control unit logic circuit. Information is also given on the method used to translate signals from an I/O bus into GP bus signals when using, small format, control unit boards.

1 General Description

The P852M/P856M and P857M are advanced, general purpose, word oriented minicomputers which employ an asynchronous, general purpose bus to transfer information between the various elements connected in the systems. A powerful and flexible set of instructions are provided - including facilities for bit and character handling, long and short constant handling, stack handling, and FORTRAN facilities - coupled with a complete range of peripherals which enable the computers to be used in industrial, scientific, or general data handling applications; whether operating in real time or not.

Uncomplicated device control unit busing, straightforward device addressing, and unrestricted interfacing make the computers easy to integrate into a user's system - a prime consideration for the system engineer the programmer and, of course, the purchaser. The design of the CPU, employing microprogram control using a read-only memory, is easy to understand thus benefitting both the system and the service engineer, and the general purpose bus, which is used for both programmed data transfers and for block transfers is not subject to any critical timing or addressing restrictions - an advantage to the system engineer whether using the many available standard control units or designing control units to meet his own system requirements.

The functional block diagram given in Figure 1.1 shows the main units which form the hardware of the P852M, P856M, or P857M CPU structure. As the data handling unit of the P852M and the P856M/P857M differ in certain respects they are described, for convenience, under separate headings.

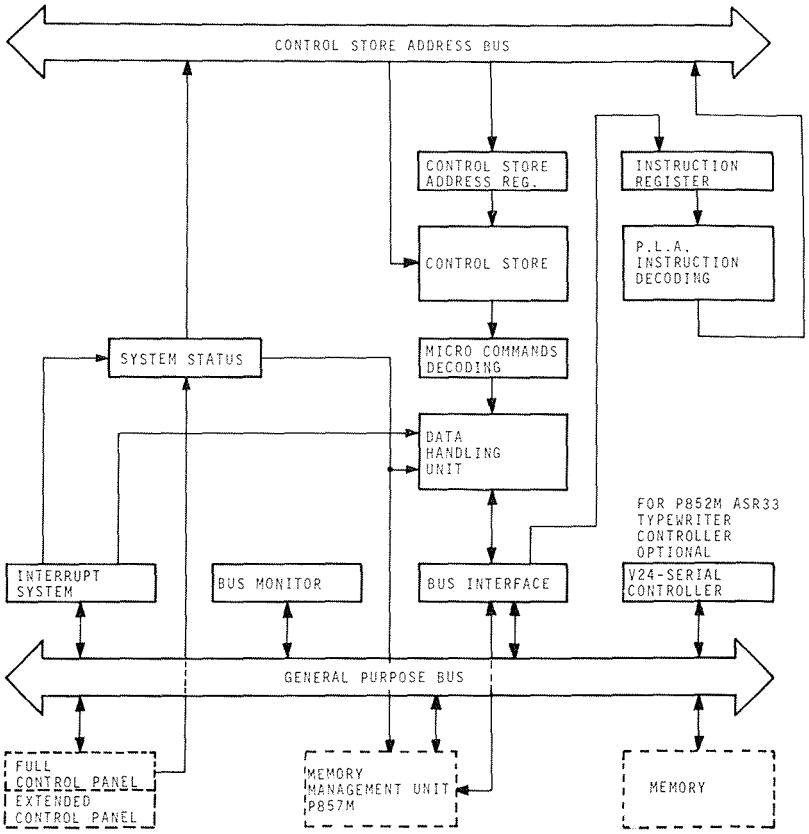


Figure 1.1. General Structure of P856M/P857M CPU's

DATA HANDLING UNIT

The relationship of the General Purpose Bus to the data handling unit components and the data paths through the unit are shown in Figure 1.2.

Data which is input from the General Purpose Bus(hereafter referred to as the GP bus)is routed,via multiplexer D,to shift register E which provides an output to input 'B' of the arithmetic and logic unit(ALU). Input 'A' of the ALU is provided from any one of four sources as follows:

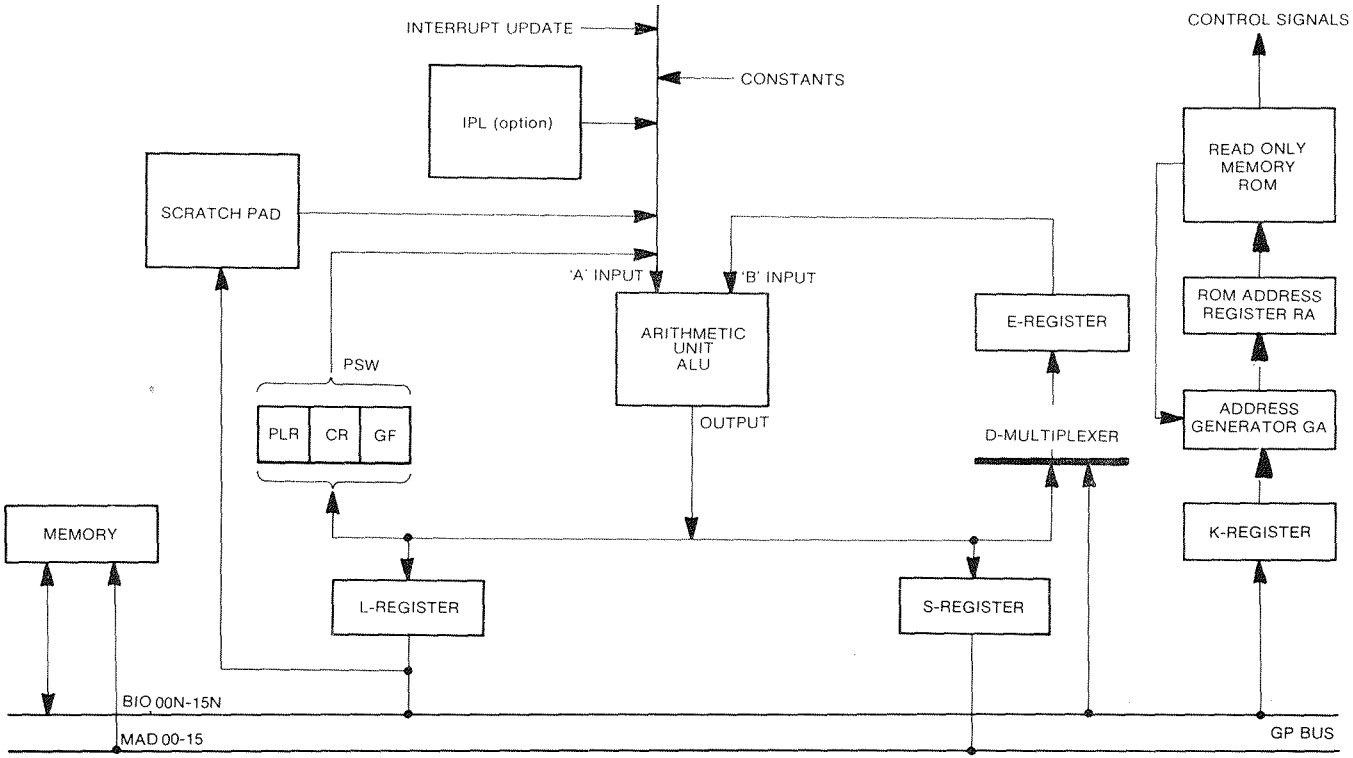


Figure 1.2. Block Diagram of P852M Data Handling Unit

1. The contents of any of the scratch pad registers.
2. The contents of the Initial Program Loader (IPL) — An optional extra.
3. A program status word (PSW) — a word holding information concerning the current status of the running program.
4. Any constants which may be required in the execution of the current instruction.

The output from ALU may be transmitted to:

1. Register L and from there to either the scratch pad or the GP bus.
2. Memory address register S.
3. The E register, via multiplexer D, and from there to the ALU for further processing.
4. The PSW register for updating the PSW.

The instruction to be carried out is derived from the GP bus by the K register and is output to address generator GA. This generator produces the appropriate microprogram addresses and transmits them via ROM address register RA to the read-only memory ROM. The ROM then generates micro-instructions to the CPU hardware units to control the execution of the instruction.

Priority level register PLR carries the priority level of the running program and condition register CR records the current program state of the CPU. The block defined on the diagram as GF contains six flip-flops and this block, together with the PLR and CR blocks, are all considered to be functionally part of the PSW whose states both influence and indicate the status of the CPU.

The hardware components shown in Figure 1.2 are:

MEMORY UNIT — A 16-bit word 1.2 µsec core memory expandable from 4k up to 32k words. Connection of a MOS memory is possible. The core memory is available in modules of 4k, 8k, and 16k. The memory is addressed directly via the GP bus and receives or provides data also directly to the GP bus.

SCRATCH PAD — A block of 16, 16-bit program accessible registers. A0 is the program instruction counter (P register) and A15 the stack pointer for the interrupt system. The remainder are available as accumulators. The registers are addressed by the instruction word received in the K register or by the scratch pad address keys on the control panel. Data from the scratch pad is loaded into the ALU for processing and is then output to the GP bus via the L register. The scratch pad registers can accept data from memory via the D multiplexer and E register, ALU, and L register.

ARITHMETIC UNIT ALU — A 16-bit parallel processing unit operating in true binary, two's complement. It performs arithmetic and logic operations on the data provided at input "A", from one of four sources and at input "B",

	from the E register. Data from the arithmetic unit can be output to register L, register S, to the E register via multiplexer D, or to the PSW register.
D MULTIPLEXER	A 16-bit multiplexer which can be operated as a 2 × 8-bit multiplexer. The inputs to the multiplexer are either from the GP bus or from the ALU output and it provides inputs to the E register as follows: 1. Data read from the GP bus. 2. Intermediate results from the ALU to be stored in E. 3. A short constant from the ALU. 4. The values in the PLR register during interrupt procedures.
E REGISTER	A 16-bit register used in shift operations. The register also acts as an input and output buffer for the ALU and as a buffer for signals read from the GP bus via the D multiplexer.
K REGISTER	Instruction register loaded from the GP bus lines with the instruction to be executed. Output from the register is to the ROM address generator GA and is also used to address the scratch pad.
L REGISTER	A 16-bit register used for temporary storage for results from the ALU which are to be passed to the scratch pad or to the GP bus.
S REGISTER	A 16-bit register arranged as a synchronous down counter. It performs 3 functions — acts as the address register for the addresses of the memory, external register, and control units; elicits from scratch pad register A15 the current value of the stack pointer; acts as a loop counter for the microprograms.
PLR/CR/GF REGISTER	A 16-bit program status word register holding the priority level of the running program, the 2-bit condition register CR and 8 status bits.
ROM ADDRESS GENERATOR GA AND REGISTER RA	A generator which derives the appropriate ROM addresses from the instructions of the running program, the current state of the CPU, and from the ROM output, and a register which provides the address of a single microprogram word to the ROM input.
READ-ONLY MEMORY (ROM)	Holds 512, fourty eight-bit words and contains a series of micro-instructions which control the CPU data processing elements in order to carry out the execution of the instruction set.

INTERRUPT SYSTEM — A system which compares the highest priority level interrupt received, either externally or internally, with the priority level of the running program, contained in the PLR register, and provides an interrupt output signal, if interrupts are allowed and if the received interrupt is of higher priority. The system also provides an output to update the PLR register with the priority level of the new interrupt.

I/O PROCESSOR — A unit containing the logic to operate peripheral control units on the GP bus in multiplex mode. In this mode of operation blocks of data are transferred between a control unit and memory via the GP bus, each character or word transfer being initiated by a break request BR from the control unit. The transfers are controlled by the I/O processor independantly of the CPU and simultaneously with the running of the program.

DATA HANDLING UNIT P856M/P857M

The data handling unit components for the P856M and P857M and the data paths through the unit are shown in Figure 1.3. Data which is input from the GP bus is routed to multiplexer C (via multiplexer D) and then to register M which provides an output to input B of the ALU, or to register L or S. Input A of the ALU is provided from the same sources previously described for the P852M.

The output from the ALU is switched via the D-multiplexer as follows:

1. As a direct or character swap output to the L-register, P-register, PSW register (GF bits) and C multiplexer. The L-register output is taken to the scratch pad and V24 serial controller and (if gated) to the GP bus. From the C-multiplexer the ALU output can be returned, via register M, to the B input of the ALU for further processing.

2. As an output to memory address register S the output of which (if gated) is taken to the address lines of the GP bus.

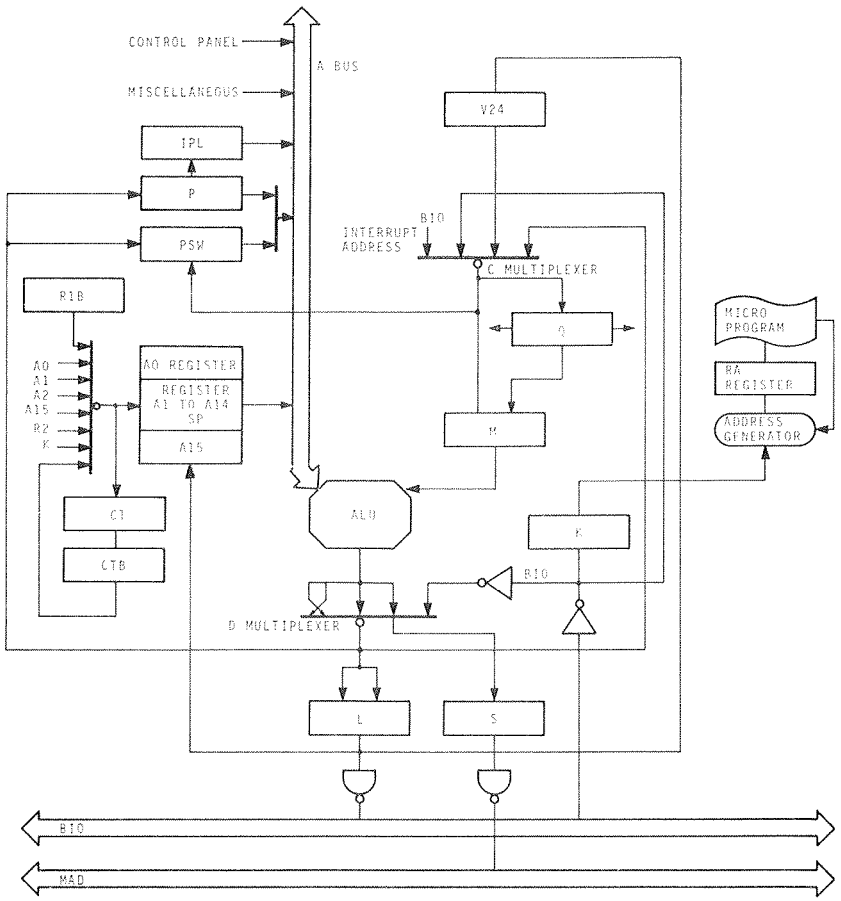


Figure 1.3. Block Diagram of Data Handling Unit P856M/P857M

The micro-instructions to control the CPU hardware are derived as previously described for the P852M data handling unit.
The main differences for the hardware components of the P852M and P856M/P857M data handling units are given below.

Note: A description is not given for a component which functions in a manner previously described for the P852M.

MEMORY UNIT — A 16-bit word core memory (cycle time 1.2 µs or 0.7 µs for P856M and 0.7 µs for P857M) expandable in 8K or 16K modules (1.2 µs) or 16K modules (0.7 µs) up to 32K for P856M and 128K for P857M. Connection of MOS, ROM and/or PROM memory is possible.

SCRATCH PAD

A block of 15, 16-bit program addressable registers A1 to A14 are available as accumulators and A15 functions as a stack pointer for the interrupt system; it may also be addressed from the instruction format. The registers are addressed by the instruction word received in the K register. A counter and counter buffer (CT and CTB) are used to address sequential scratch pad registers during multiple Load or Store instructions or to count repeat cycles for sequensor control.

D MULTIPLEXER

A 16-bit selector or 2x8-bit multiplexer used as a control element in the arithmetic loop. The inputs to the multiplexer are from the ALU or from the GP bus and it provides outputs in 4 modes as follows:

ALU output- direct(inverted output)
-shift right-input to S register(true output)
-character swap(inverted output)
GP bus input(inverted output)

The inverted output is sent to the L register, PSW register, instruction counter P, and to the C multiplexer. The true output is sent to S register - this mode is used for shift right and multiplication operations.

C MULTIPLEXER

A 4 input selector in the operand 'B' input data path of the ALU for two 16-bit sources which can be selected as long or short constants(D multiplexer output, BIO lines output)and two 8-bit sources (Interrupt address,V24 serializer output). The selected outputs are switched to the Q and M registers and the six least significant bits of the interrupt address are also sent to the PLR register(part of PSW register)during an interrupt routine.

M REGISTER

A 16-bit multiple input register. Used as a working register and buffer in the operand 'B' input data path to the ALU for inputs provided via the C multiplexer or from the Q shift register.

P REGISTER — A 16-bit synchronous register/up-down counter used to hold the address of the next instructions to be executed. The register is loaded(via the D multiplexer) with the 14-bit word address of the program; the register is also used as an internal counter during multiple word instructions.

Q REGISTER — A 16-bit left/right shift register. Used during double length instruction operations.

IPL — A 64-word Initial Program Loader bootstrap (ROM) of 256 four bit words(regrouped into 16-bit words when loaded into memory),which provides the system with the ability to automatically load and run an initial program loader or similar program from devices on either the programmed or I/O processor channels. The IPL is standard on the P857M and pptional on the P856M.

MEMORY MANAGEMENT UNIT (P857M)

A memory management unit (MMU),provided as standard with the P857M computer,gives the facility to access primary memory over 32Kw up to 128Kw (and practically unlimited space in secondary storage, such as high speed discs), permits dynamic program relocation,and offers a memory protection facility. Special instructions are provided for this unit.

Using the MMU gives the user a virtual memory system, with the advantage of working in a real time multi-programming or multi, tasking environment. The system is transparent to the user,who does not need to know where in primary or backing store,his program is located. A user program may not exceed 32Kw.

In operation the MMU dynamically translates a normal 16-bit program address(logical address) into a 18-bit physical address (real address) which is used to address one of 64 pages,each of 2Kw,in primary memory(the pages do not need to be consecutive). When used in system mode the system uses the logical 16-bit address-no translation is performed-as it always resides within the first 32Kw of primary memory. In user mode however the logical addresses are always translated by the MMU into physical addresses;to address the pages in memory a segment table(16x16 bits)is built for each program called. The bits 0-3 of the logical address contain a segment address and bits 4-15 the relative address from the beginning of the page. The segment address

locates a register in the segment table which then outputs a 6-bit physical page number and this, together with bits 4-15 from the logical address is used to produce the 18-bit physical address.

Memory protection is provided by a special interrupt 'Page Fault' which is given when an attempt is made to write into a protected page or when a missing or wrong page is tried to be accessed. The interrupt has priority over other internal or external interrupts.

EXTRA FACILITIES(P856M and P857M)

Memory Interleaving

Memory interleaving is possible, using 0.7 µs read/write core memory, which permits very fast execution of instructions. The access time comprises one read cycle after which the execution of the instruction starts immediately. To make full use of the capability the minimum memory size is 32Kw i.e. 2 modules of 16Kw each(one module will contain even addresses and the other the odd addresses; the CPU automatically applies interleaving(if the links have been made on the memory boards).

Micro-diagnostics

To aid in fault location a 100-word microdiagnostic program (contained in the micro-program ROM)is available which permits testing of the data paths, control unit-CPU dialogue, and memory. Automatic or step by step of the CPU-CU dialogue and the first 4K words(P856M)or 16K words(P857M)of memory is possible; the test program is made available by simply setting a key on the control panel to a position marked 'TEST'.

Extended Control Panel

A dual display control panel is included as standard on the P857M(optional on the P856M)which enables the entire memory to be addressed and also provides facilities for a quick and easy inspection of the memory contents- a valuable aid for software/system testing. A'stop on preset' facility is provided, for program debugging purposes.

INPUT/OUTPUT CHANNELS

Dialogue in the systems is in asynchronous conversational mode which is easily adaptable to the system elements involved. The exchanges which can take place and the channels used are summarized as follows:

Exchange	Channel Used
CPU and Control Unit	Programmed
Control Unit and Memory	I/O Processor
CPU and External Memory	

One of the most important system features is that they permit peripheral control units to make direct transfers with memory under control of an I/O processor which has an interface identical to that required for the programmed channel.

The choice of which channel to use depends on the type of device to be connected and the application for which it is to be used. *Programmed channel,* where the transfer is word or character oriented is used only for low speed applications such as the operator's typewriter or punched tape or card devices. When using the I/O processor, however, the programmed channel is also used to initially transfer the contents of two control words, which govern a block transfer, into the I/O processor and to activate the control unit to be used during the transfer. The execution of a program, when using the programmed channel, may be interrupted either by the generation of an internal interrupt signal or by an exchange request from a system element which is conveyed to the CPU on the interrupt request lines of the GP Bus.

The I/O processor provides for the control and automatic execution of direct transfers of variable-length blocks of data in either direction between control units and the block or blocks of central memory. The channels are used normally by high speed devices but can be used also by low speed devices. As many as 64 control units, operating on a priority basis, can be controlled via the channels. Requests for data transfers are made from control units using a system of break requests signal lines connected between the I/O processors and the control units. These signal lines are carried on separate cables which do not form part of the GP Bus.

A *direct memory access* facility can be used with the standard systems by designing a control unit which contains its own logic for obtaining access to memory and registers containing control words to govern the block transfer. A data transfer using this facility would then occur at memory speed.

GENERAL PURPOSE BUS

All transfers of information between elements of the system take place via the lines of the GP Bus. The bus lines, which are described in more detail in chapter 2, can be subdivided into groups as follows:

Data and Control	Address	(18 lines)	CLEARN
	Data	(16 lines)	ALN
	Timing	(5 lines)	CHA
	Others	(1 control unit response, 4 internal signals, 1 clear)	WR
Master Selection	5 lines	SPVC, ORO, MSN, BSYN, BUSRN total	
Interrupt Request	7 lines	(including 1 scanning line)	
Others	2 lines	(power control and power failure)	

With such a bus, which is asynchronous and multiprocedural, peripherals have individual address and can be addressed directly, via the bus. In addition the dialogue over the bus is more efficient as, with each type of dialogue, a procedure is associated which is specifically suited to the function being carried out.
The system concept is illustrated in Figure 1.4 which shows how all data handling system components are connected. Some control units (defined as Type 1) are connected directly to the GP Bus and others (defined as Type 2) are connected to an I/O bus and, via a bus translator, to the GP Bus.

MASTER-SLAVE RELATIONSHIP

Communication between two devices on the GP bus employs a master-slave relationship and during any bus exchange operation one device, called the master, has control of the bus when communicating with another device, called the slave. For this relationship a system element which can receive control of the bus to address memory is defined as a master and a system element which has not the possibility to address memory is called a slave. I/O processors and DMA controllers (if used) are examples of masters. The memory, the registers, and the device control units are always slaves.

The master of an exchange may address a slave or another master (which then functions as a slave); a priority control system determines which master obtains control of the bus. In any exchange involving the CPU, however, the CPU is always master of the bus.

PRIORITY STRUCTURE

A number of masters may be connected in the system and a bus controller is provided to regulate the access of the masters to the bus in order of priority of their requests. The CPU has automatically direct access to the memory (via the bus) at the end of the execution of each instruction if no other device is requesting or using the bus.

EXTERNAL CONNECTION

All peripheral units and data input or output devices are connected to the GP bus by means of control units. These can be single boards located in slots in the basic mounting box, or in an equipment shelf, or may consist of several boards housed in an equipment shelf. For P852M computers the control unit for an ASR typewriter may be constructed on the CPU board or may be a separate control unit. For P856M and P857M computers a V24 serial control unit is provided as standard on the CPU board.

Addressing facilities for up to 256 external registers are provided on all computers.

Standard Control Units

Peripheral control units are available to suit many types of peripherals including I/O typewriters, punched tape readers and punches, line printers, punched card readers, magnetic disc and tape readers, and cassette tape units. The control units are either connected directly to the GP Bus or are connected to the GP Bus via bus translator boards.

For general purpose digital input and output of up to 16 bits per channel the digital input/output (DIOS) system of control units is available Using Dual Input Output boards (DIOD).

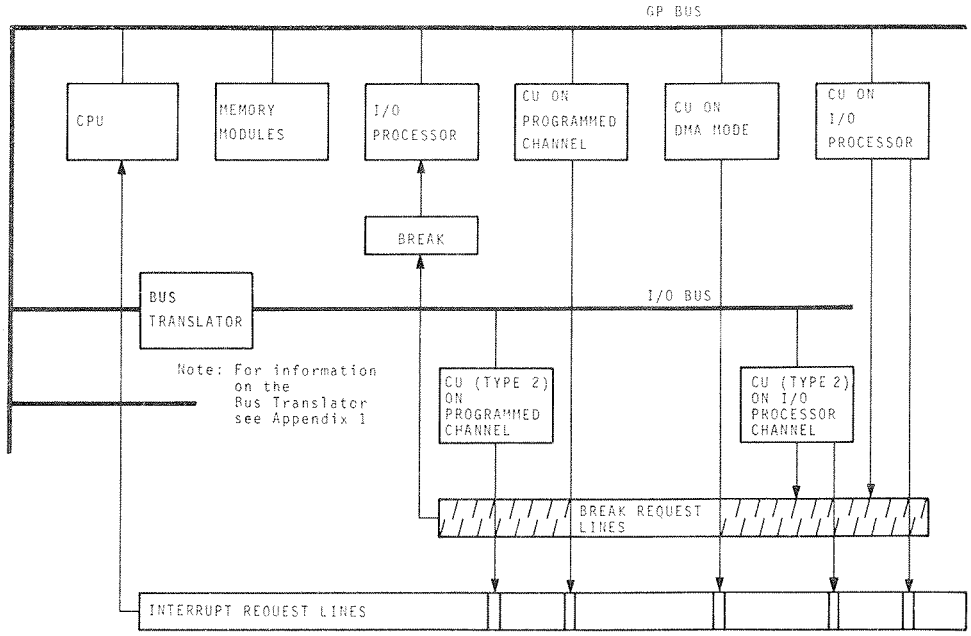


Figure 1.4. System Structure - Block Schematic

Analog or digital acquisition is handled by a modular input/output control(MIOS)system,which includes output units for driving display equipment,and an analogue/output (AIOS)system can be provided as an analogue data channel For further details refer to the MIOS manuals listed in the Appendix.

Data Communication in the systems is accomplished by the use of multiple line control units,both synchronous and asynchronous,at speeds up to 200,000 baud, and asynchronous line multiplexers at speeds up to 9600 baud. For further details refer to the P852M/P856M/P857M Data Communication Service Manual listed in the Appendix.

User Designed Control Units

To aid the user wishing to build his own control units for special applications there are two styles of general purpose board;one is already equipped with I/O control logic while the other is completely blank but has a hole pattern for accepting 226(14 or 16 pins) IC's and 15 MSI LSI packages with 24 or 40 pins as well as components or wire wrap pins.

Control Unit Operation

Each control unit has a unique address that is decoded from the lines of the GP bus which carry the address, thus any control unit can be addressed by software I/O instructions. Similarly the function lines of the GP bus are decoded to permit control of the associated peripheral device by software instructions. Recognizing its address the control unit decodes the function and then provides the necessary control signals or data synchronisation pulses to its peripheral.

INTERRUPT SYSTEM

When a control unit operating on programmed channel has been switched into service by a software I/O instruction it will give an interrupt request to the CPU when it is ready to send or receive data. The CPU may then service the interrupt request and interrupt the program according to the priority level assigned to the interrupt. Eight interrupt levels, 0 to 7, are included in the minimum configuration all of which arise within the basic mounting box; of these eight, four are interrupts generated inside the CPU. The interrupts may be increased to a maximum of 63 with the addition of levels 8 to 62 which may originate inside or outside the basic mounting box. In this case the interrupts are conveyed to the CPU on 6 separate lines of the GP Bus using a bused system which transmits a coded signal representing the interrupt level having the highest priority, which has been raised.

The state of all interrupt lines is sampled upon the completion of each one or sometimes two program instructions (fetch cycle) and compared with the priority level of the running program. If there is an interrupt request having a higher priority than that of the running program an interrupt routine is initiated by the interrupt, provided that the CPU is in the enable interrupt state. Return to the original program is made possible by storing the memory address at which it must restart.
The interrupt system makes use of automatic stack handling to service both internal and external interrupts. This means that, for each interrupt the contents of the P register (this is the address at which the program must restart) and a status word are stored in a part of the memory allocated for interrupt stacking. Access to the stack is made from scratch pad register 15 which always points to the next location in the stack. The stack pointer is automatically decremented each time program information is loaded into the stack and is incremented when information is removed from the stack to run a program. Automatic indication of stack overflow is given to warn the programmer that the stack is nearly full and unable to handle further interrupts.

Internal Interrupts
The following four interrupts are generated inside the CPU:

- Power Failure
- Program Interrupt
- Real Time Clock
- Control Panel

of these interrupts two are used in all configurations and these are:

Program Interrupt	This occurs if the stack overflow condition arises or if a Link to Monitor instruction is executed.
Control Panel	This is instigated by pressing the INT button.

The other two interrupts are generated by power failure or real time clock.

Internal interrupts within the CPU are handled in exactly the same manner as external interrupts except that, to be effective, the power failure interrupt must be assigned the highest priority level (lowest number) above all other internal and external interrupts.

TIMING CONTROL SIGNALS

Exchanges between masters and slave units are controlled by timing signals (from master) and timing response signals (from slave). The timing signals are used to validate addresses and data and to control the overall timing of an exchange. Use of such an asynchronous timing system permits either standard or non-standard peripheral devices to be used without the need for special timing circuits.

Further details of the timing signals used are given in Chapters 3 and 4 of this Part where timing diagrams are provided for the various exchanges possible.

2 General Purpose (GP) Bus

All transfers of information between elements of the P852M/P856M or P857M system take place via the GP bus, and the lines of the bus comprise the input and output signal and address lines necessary for the data transfer requirements of the system. The GP bus can be extended outside the basic mounting box to further equipment shelves by using 125 ohm, flexible, plug-in, transmission lines. The lines can be extended in convenient lengths between equipment shelves up to a total maximum length of 14.5 metres. Line termination facilities are provided in the basic mounting box and, if required, in the equipment shelves .

Termination details of the GP bus lines are given in Part 2 of this Section of the manual together with the characteristics of the emitters and receivers required to interface with the GP bus lines.

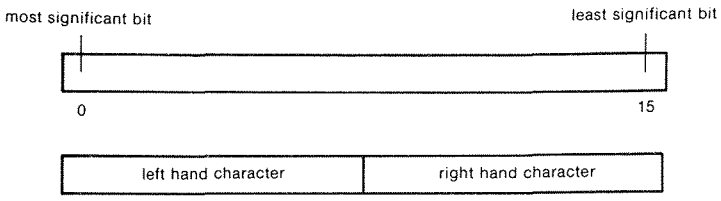
Two types of signal are used on the bus: command signals and data signals. Command signals are those which will cause an immediate action according to their change of state; these signals have no unknown state but are always either logic "1" or "0". Data signals carry the actual information exchanged amongst the system elements; these signals are permitted to adopt indeterminate values except when the information is actually being used in the processing.

The signals carried by the GP bus lines are described below. When a mnemonic ends in "N" it means, in the case of a data signal, that the signal transmitted on the GP bus is the complementary value of the true signal. In the case of the control signal the "N" means that the signal performs its function on being set to "0" (active low). Most of the GP bus lines are used both inside and outside the basic mounting box. Where this does not apply it is indicated in the signal description given below.

GP Bus Signals

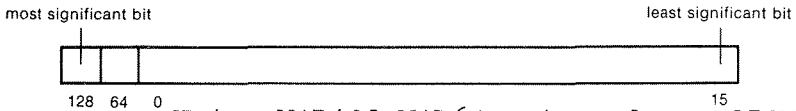
BI0 00N to BI0 15N — 16 data lines which are used to carry data information between all system elements concerned with the transmission or reception of data signals.

The bit location is as follows:



MAD128, 64 and 00 to 15 (only MAD03, 04,08-15 used externally also)

18 address lines which carry different information according to the type of exchange. The bit location is as follows:



Note: MAD128,MAD64 not used on P852M or P856M

For a master-memory exchange the MAD lines carry the memory address and MAD15 is used as a character indicator. When set to 1 it indicates the right (least significant) character and when set to 0 the left (most significant) character.

For a master-control unit exchange the lines MAD10-15 carry the address of the control unit and lines MAD 04, 08 and 09 the function to be performed. MAD03 indicates whether or not the exchange in progress is the last. The functions are as listed below:

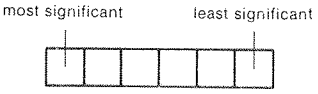
MAD04 = 0	exchange to control unit
MAD04 = 1	exchange to master
MAD08 = 0	data exchange (INR, OTR)
MAD08 = 1	command or status exchange
MAD09	special functions
MAD03 = 0	exchange not the last
MAD03 = 1	last exchange.

For a master-external register exchange the lines MAD08 to 15 carry the address of the external register. MAD04 is used to indicate whether it is a read or write operation as follows:

MAD04 = 0 write operation
MAD04 = 1 read operation.

BIEC 0 to BIEC 5

Six signal lines which represent in encoded form the interrupt raised (other than internal interrupts) having the highest priority. The format is as follows:



·SCEIN

A signal *Scan External Interrupts* sent by the CPU to control units at the end of each instruction which places on the BIEC 0 to 5 lines the 6 bits representing the highest priority external interrupt detected.

ACN

A signal *function accepted* which is sent by a control unit to the CPU to indicate that the requested function is accepted by the control unit.

BUSRN

A signal *bus request* which is sent to the bus controller in the CPU by a master which requires control of the bus to effect an exchange.

BSYN

A signal *bus busy* which is shared by all masters. It is set to "0" by the master which has been given control of the bus so that the exchange can commence without interruption.

MSN

Internal use only

A signal *master selected* which is transmitted to all other masters by the master which has become master of the bus to block all activity of the priority selection chain. The signal is released when the master is ready for the next priority transaction.

SPYC

A signal *scan priority chain* sent by the bus controller to all masters in response to a BUSRN signal. The signal enables the highest priority master which has transmitted BUSRN to block the priority chain at its level.

OKO (internal use only)	A signal generated by the bus controller after all masters have been alerted by SPYC It is sent to the master having the highest priority (determined by hard wiring in the priority selection chain).
OKI (internal use only)	A master which receives signal OKO regards the signal as OKI. It then retransmits a further OKO to the next master in the priority chain. The first master to receive OKI set to '1' and to retransmit OKO reset to '0' is next master of the bus.
CHA (internal use only)	A signal *character* transmitted to the memory by the master which has control of the bus to indicate whether the exchange is to be by character or by word as follows: CHA = 1 character operation CHA = 0 word operation.
WRITE (internal use only)	A signal *write* transmitted to the memory by the master which has control of the bus to write information into memory or to read information from memory as follows: WRITE = 1 write into memory WRITE = 0 read from memory.
CLEARN	A signal *clear* transmitted by the CPU to all elements connected to the GP bus to cause a general reset to zero.
TMRN (internal use only)	A signal *master to memory* transmitted by a master to memory to validate the data on the BIO and MAD lines during an exchange. The signal also controls the timing of the exchange.
TMPN	A signal *master to peripheral* transmitted by a master to a peripheral CU to validate the address of the peripheral CU and to initialize the exchange.
TMEN	A signal *master to external register* transmitted by a master to a unit containing the addressed register to validate the address and data of the register and to control the timing of the exchange.
TRMN	A signal *register or memory to master* transmitted by a unit controlling a register or by memory in reply to TMEN or TMRN to indicate that the unit

	transmitting the signal is in a condition to be read. The signal is also used to terminate the exchange.
TPMN	A signal *peripheral to master* transmitted by the peripheral control unit concerned in reply to TMPN. It is also used to validate the response of the control unit and to terminate the exchange.
RSLN	An Earth signal *reset line* transmitted by the CPU power supply (or external rack power supplies) and used to protect the peripherals during the switching on and switching off power sequence. The signal is also used to generate CLEARN when switching on.
PWFN	A signal *power fail* transmitted by the CPU power supply (or external rack power supplies) to warn the CPU that power failure has been detected. The signal is also used as a facility to restart the system at the point where it has been stopped.
4 spare lines	There are 4 spare lines provided on the GP bus extension cable outside the CPU cabinet.

3 Programmed Channel

The programmed channel is the basic I/O transfer channel used to transfer data between the CPU and the peripherals or external registers one word or character at a time. The data are transferred, via the GP bus, under the control of a set of instructions and the channel operates at rates of up to 20,000 16-bit words per second. Details of all connections to the channel are given in Part 2 of this Section of the manual.

The channel can be used in Wait Mode, where each word or character is exchanged separately and the complete program is held up in a waiting loop between each exchange, or in Interrupt Mode where each word or character is still exchanged separately but the necessary instructions form part of an interrupt routine which allows the main program to continue whilst an exchanged is in progress. The wait mode (the slowest form of transfer) is rarely used.

Figures 3.1 and 3.2 illustrate the Instructions flow for the P852M and the P856M/P857M computers respectively.

For any computer the instruction to be executed is received by the K register, on the BIO lines of the GP bus, and is passed, via the address generator GA and register RA, to the ROM. The ROM then generates micro-instructions to affect control of the data processing elements in order to carry out the instruction. For exchanges between the CPU and peripheral control units, or external registers, the appropriate addresses are output to the GP bus from register S on MAD lines 00 to 15. The S register is loaded from the ALU output.

The instruction format used for transfers between the CPU and control units is illustrated in Figure 3.3. Included in the instruction are three function bits (F in Figure 3.2) which are used by both the control units and the CPU. Bit 4 for example, is part of the operation code, and determines whether a control unit and the CPU are to operate in input mode or output mode. Bits 5, 6 and 7 (R in Figure 3.2) are used only by the CPU to determine which of the scratch pad registers A0 to A7, is to be used for the data transfer. Six address bits are included in the instruction word which give the device address. An I/O instruction always includes operation code (OPC in Figure 3.3) 1001 or 1000 to signify that it is an I/O instruction.

Information on the instruction format used for transfers between the CPU and external registers (WER and RER instructions) will be found in Chapter 4 of Part 1 of this Section of the manual.

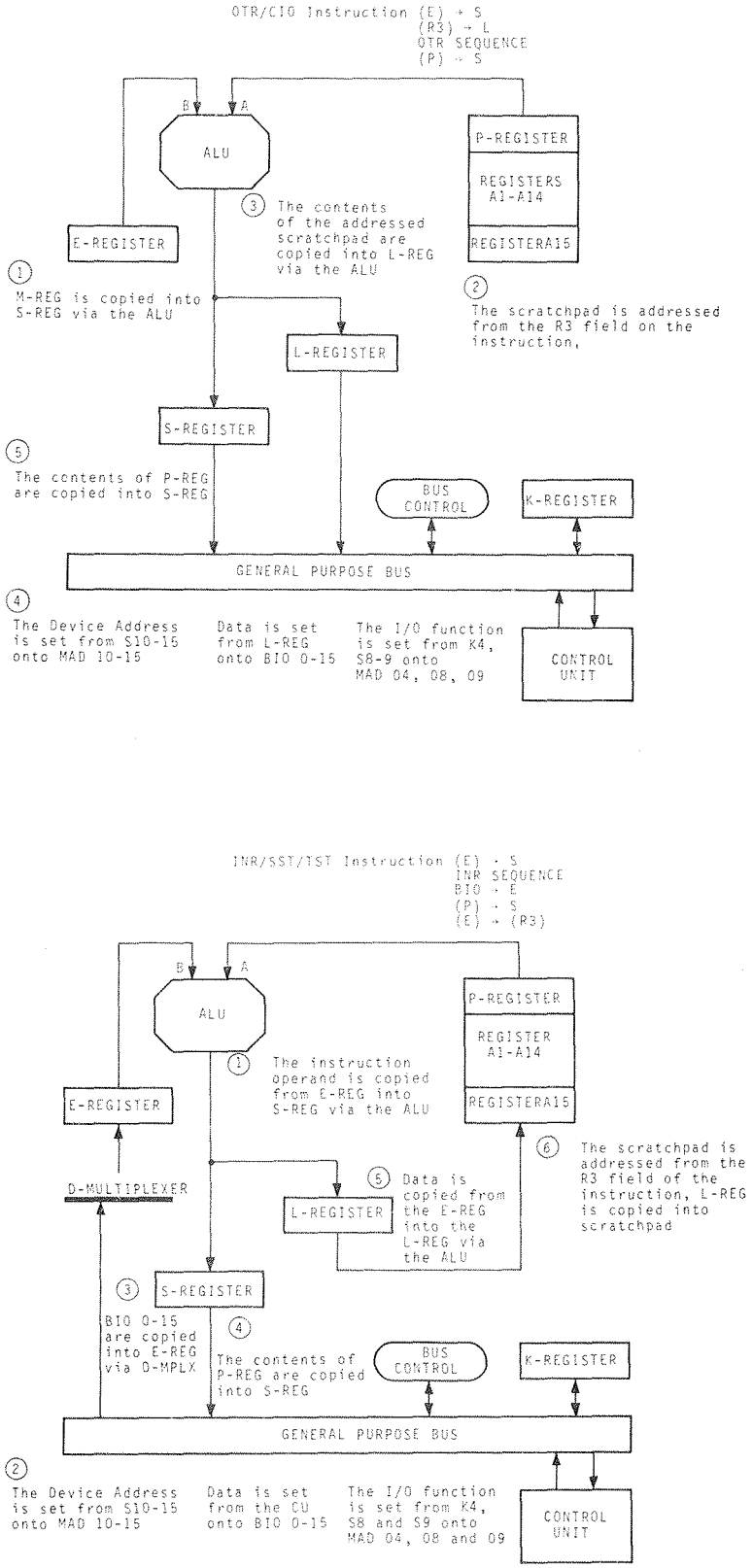


Figure 3.1. Instruction Flow P852M

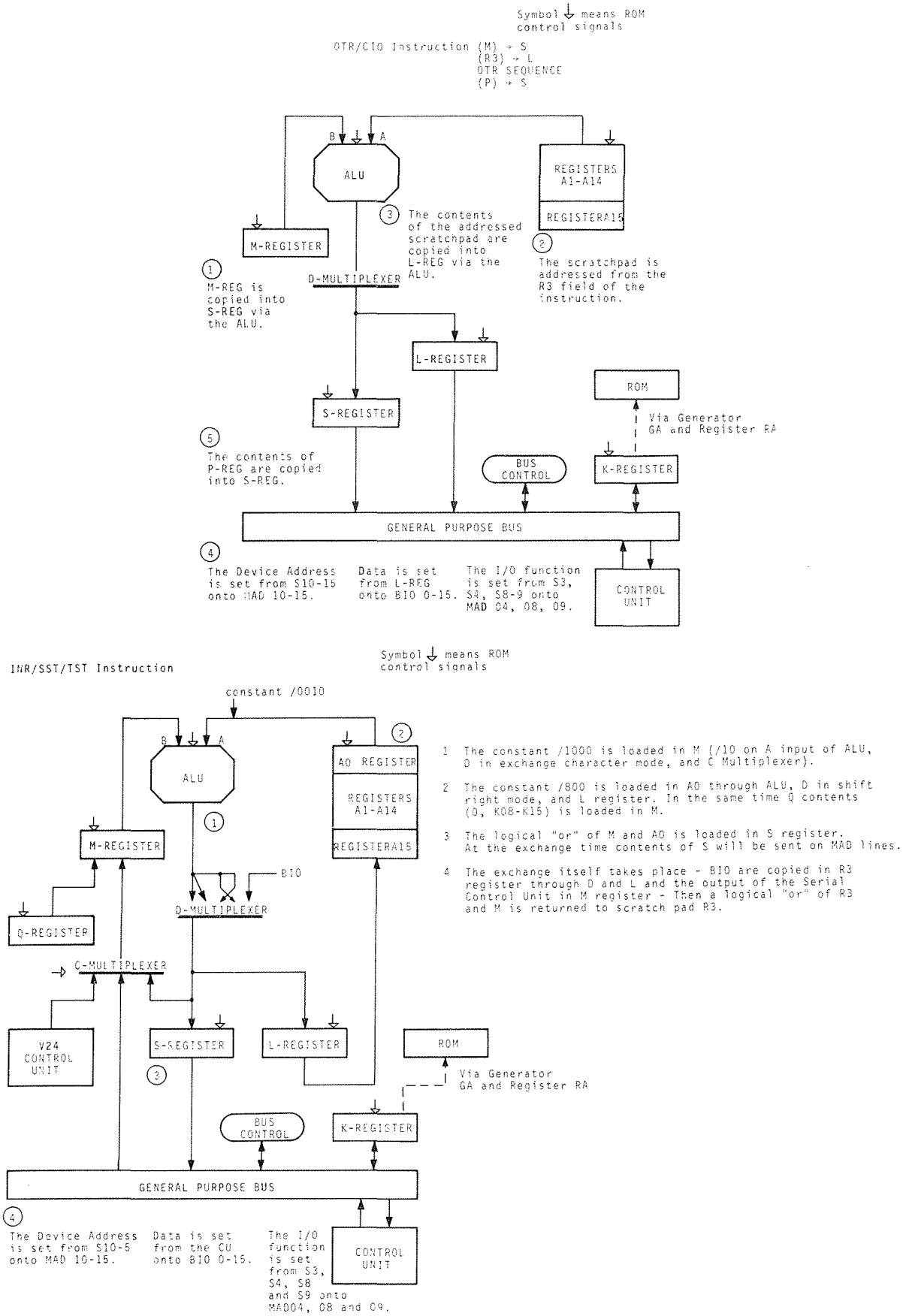


Figure 3.2. Instruction Flow P856M/P857M

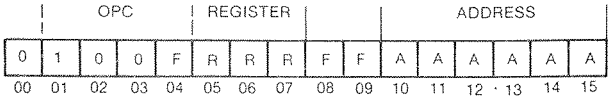


Figure 3.3 Instruction format

I/O INSTRUCTIONS

Five I/O instructions are used to program transfers between the CPU and control units. Control input/output (CIO) stops or starts an I/O operation depending on the state of instruction bit 9. Input to register (INR) transfers a data word from the addressed control unit to the scratch pad register specified in the instruction. The number of data bits transferred depends on the type of peripheral device involved. Output from register (OTR) transfers a data word from the scratch pad register specified in the instruction to the addressed control unit. Again the number of bits transferred depends on the type of peripheral device. Send status (SST) transfers the status word from the addressed control unit to the scratch pad register specified in the instruction. The status word contains information defining the status of the peripheral control unit. Test status (TST), which is always accepted by a control unit, is used to find out whether or not a control unit is busy. A status word is sent in reply from the addressed control unit, to the scratch pad register specified in the instruction, in which one bit indicates whether the control is busy or not.

Two I/O instructions are used for transfers between the CPU and external registers. Write External Register (WER) transfers a data word or character from the scratch pad register specified in the instruction to an addressed external register, and Read External Register (RER) transfers a word or character from an addressed external register to the scratch pad register specified in the instruction. These instructions are used when block transfers of data between control units and memory, under the control of an I/O processor, are required. Further details will be found in Chapter 4 of Part 1 of this manual.

Full details of all I/O instructions are given in Part 3 of this Section. The detailed operation of the CPU during the I/O and other instruction cycles can be found in the P852M/P856M/P857M Service Manual.

PROGRAMMING

Programming for data transfer on programmed channel must include a routine to call up the required control unit and place it in the *busy* state with a CIO start instruction, a data transfer routine, and a routine to shut down the control unit when it is no longer required.

The data transfer routine, which is initiated by an interrupt request from the control unit includes subroutines for checking or assigning interrupt and program priority and stacking control words to allow the interrupted program to be restarted. The routine also checks the status of the control unit to ensure that the data transfer is completed (this action is also initiated by an interrupt after a CIO halt instruction), and loads the data word into the scratch pad (for an output transfer) or takes the word from the scratch pad (for an input transfer).

The Basic Operating Monitor contains a complete I/O routine and its use is described in the manual Programmers Guide 1, Volume 1.

I/O PROCESSOR AND DMA ROLE

In addition to the data transfer of single words or characters, the programmed channel is used to start a data transfer in the I/O processor or direct memory access channels and to perform status exchanges. The control unit, wired for I/O processor or direct memory access operation, will then operate from the block transfer logic.

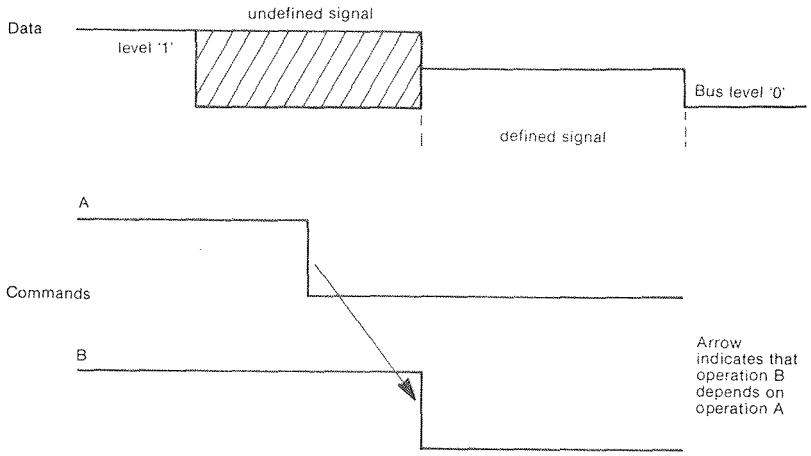
EXCHANGE TIMINGS

The following timing details are given for the exchanges Write/Read memory or external register and master To/From control unit which can take place using the programmed channel. When considering the exchanges the following should be borne in mind.

1. An exchange is always requested by the master.
2. The receiving station or the nature of the exchange may change.
3. The CPU is the master of an exchange between the CPU and another unit.
4. A memory block is always slave.
5. A peripheral control unit is always slave.

The timing diagrams included are measured at the connector of any element on the GP bus. All exchanges with memory or external registers are made with very short deskewing times to secure maximum advantage from the memory performance.

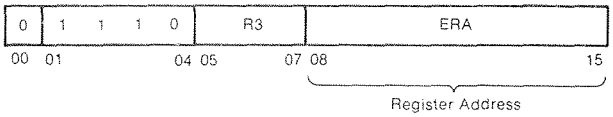
The following writing convention is employed for the timing diagrams given.



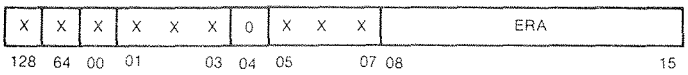
Exchanges — Write External Register and Write in Memory

WRITE EXTERNAL REGISTER Instruction: WER

Instruction Format.



This produces the following signals on the MAD lines.



MAD 08-15 — Carry the address of the addressed register
MAD 04 — "0" indicates it is a Write operation.

Bits marked X indicate that they are not significant for the exchange. Signal lines WRITE and CHA are both X for this exchange. (These signals do not leave the basic cabinet).

In this exchange the R3 field of the instruction addresses a register (1 to 7) of the CPU scratch pad and the contents of this register is sent to the external register on the BIO 00N to 15N lines of the GP bus .

Signal TMEN is used for the dialogue with the external register and the reply signal from the register is TRMN. The exchange transfer timing is the same as

that for writing in memory as shown in Figure 3.3 for external registers connected inside the basic cabinet (e.g. I/O processor).

Note: For external registers located in extension boxes a degradation equivalent to 60 ns delay must be allowed in the deskewing time due to the bus connection.

WRITE IN MEMORY

For this transfer WRITE = 1. The operations required are then controlled by CHA and MAD15 bits which define the nature of the exchange as follows:

CHA = 1 Character operation
CHA = 0 Word operation

MAD15 = 1 Right hand character
MAD15 = 0 Left hand character.

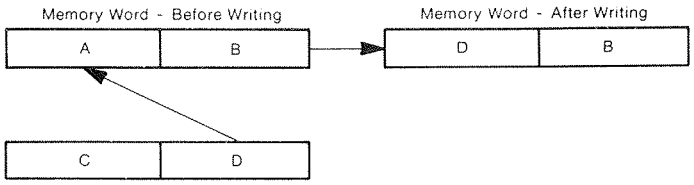
Table 3.1 illustrates the various combinations possible.

Table 3.1 Write in Memory CHA and MAD 15 bits

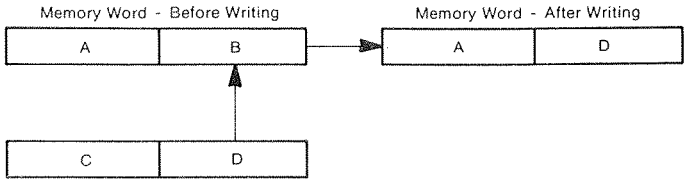
CHA	MAD15	OPERATION
0	0 1	The 16-bit word present on the BIO lines is written at the given address
1	0	The righthand character (bits 8-15) is written on the left
1	1	The righthand character (bits 8-15) is written on the right

P852M memories are designed to facilitate character operation and the two examples below illustrate the two character transfers shown in Table 3.1.

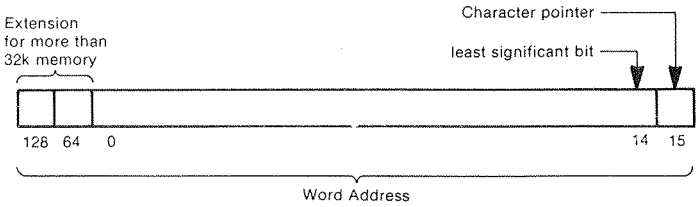
Example 1: MAD15 = 0 CHA = 1



Example 2: MAD15 = 1 CHA = 1



The address format for the exchange is shown below:



The timing for writing in memory (or external register) is shown in Figure 3.3. The timing is for masters and memories (or external registers) located inside the basic mounting box and is defined at bus connector level.

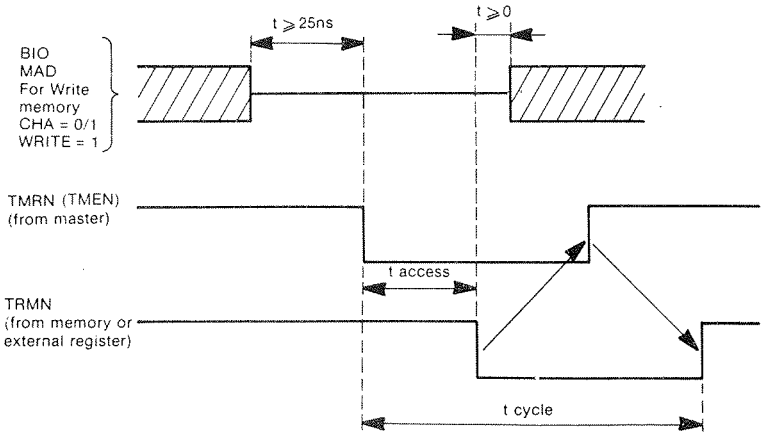


Figure 3.4. Timing-Writing in Memory (or External Register)

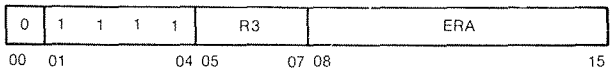
The master sets TMRN to 0 after a guard time of not less than 25 ns when the addresses and information required for the exchange are stabilized.

A memory sends back TRMN when it no longer needs the data for the exchange. The master may then suppress this data.

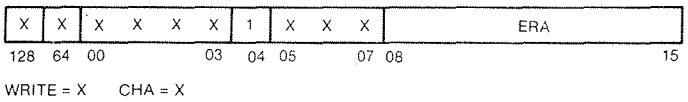
Exchange-Read External Register or Memory

READ EXTERNAL REGISTER Instruction: RER

Instruction Format:



This produces the following MAD signals from the S register.



The R3 field of this instruction addresses a scratch pad register (1 to 7) of the CPU. This register will be loaded with the data on BIO 00N-BIO 15N. The master uses TMEN for the exchange and the register replies with TRMN. The exchange timing is the same as shown for read memory in Figure 3.4 for registers located inside the basic mounting box. For registers outside the basic box 60 ns degradation time must be allowed for.

READ MEMORY

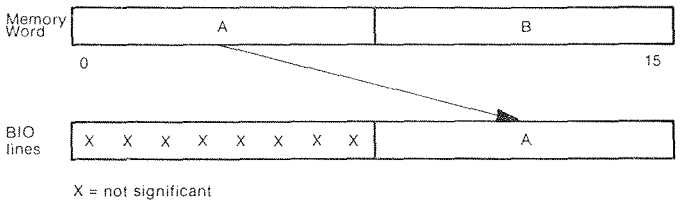
For this transfer WRITE = 0. The operations required are then controlled by CHA and MAD15 bits as previously described for the write operation. The various combinations are listed in Table 3.2.

Table 3.2 Read Memory CHA and MAD15 bits

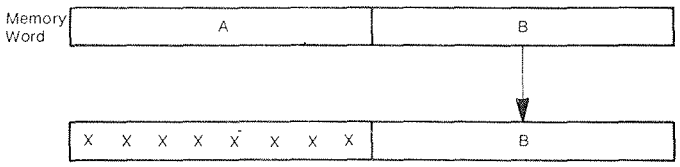
CHA	MAD15	OPERATION
0	n.s.	The 16-bit memory word is read on the BIO lines
1	0	The lefthand character (bits 0-7) is read on the righthand BIO lines
1	1	The righthand character (bits 8-15) is read on the righthand BIO lines

The examples given below illustrate the two character transfers shown in Table 3.2.

Example 1: MAD15 = 0 CHA = 1



Example 2: MAD15 = 1 CHA = 1



The following timing diagram illustrates a read memory (or read external register) exchange.

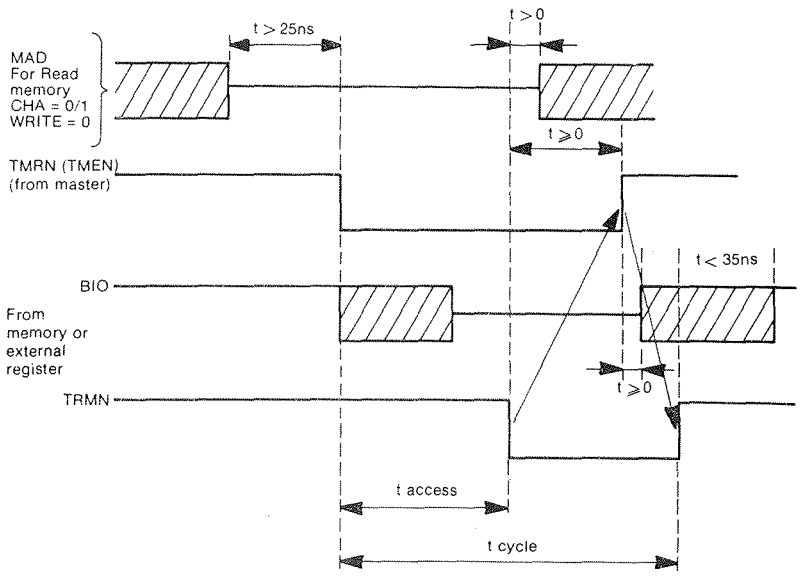


Figure 3.5. Timing-Read Memory (or External Register)

If the memory location addressed has its data ready and no longer requires the contents of the MAD lines it responds with TRMN which ensures that the exchange will terminate:

1. By suppressing the address
2. By suppressing TMRN (TMEN) when the master has read the data and has indicated in its turn that the data may be suppressed.

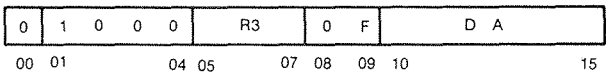
The data signals must become high again within 35 ns after TRMN goes high.

Exchange-Master to Control Unit

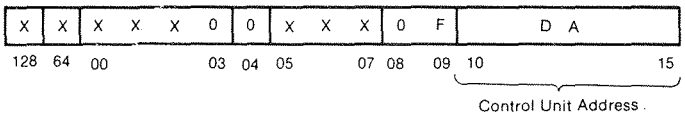
This exchange is used to send a word of maximum 16 bits to a peripheral control unit. The instructions OTR (Output Transfer) and CIO (Control Input/Output) are used during an exchange of this type.

OUTPUT TRANSFER Instruction: OTR

Instruction format



This produces the following MAD signals from the S register.



MAD03 = 0 (exchange not the last)
MAD04 = 0 (exchange to control unit)
MAD08 = 0 (data exchange)

Note: Bit F permits the coding of two different OTR instructions, as required for certain CU's.

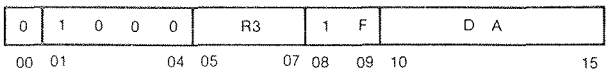
During the execution of this instruction the inverted contents of the scratch pad register addressed by the r3 field of the instruction are sent on the BIO lines to the addressed control unit. The control unit sends replies on line TPMN (Address recognized) and ACN (Accept) which are copied into the 2-bit condition register CR of the CPU. If time out occurs the condition register is set to 11. Examples of typical logic circuits used in control units are given in Part 2 of this Section of the manual.

The CR register functions are as follows:

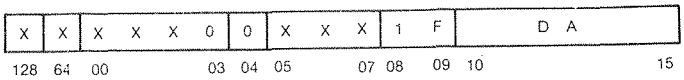
00 — Address recognized, function accepted
01 — Address recognized, function refused
11 — Address unknown.

CONTROL INPUT/OUTPUT Instruction: CIO

Instruction Format.



This produces the following MAD signals from the S register.



MAD08 = 1 (command status signal exchange)

During the execution of this instruction the contents of the scratch pad register addrssed by the R3 field of the instruction are sent on the BIO lines to the control unit. The information on lines TPMN and ACN is also copied into the CR register of the CPU.

The CIO instruction may be used to start an operation (F = 1) or to stop an operation (F = 0).

The content of R3 is used to specify the parameters which are necessary for the control unit to perform the operation.

Note: CIO HALT (F = 0) must always be accepted by the control unit.

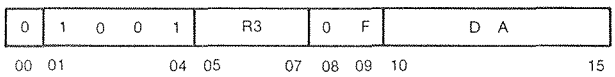
The timings for OTR and CIO exchanges are shown in Figures 3.5 and 3.6.

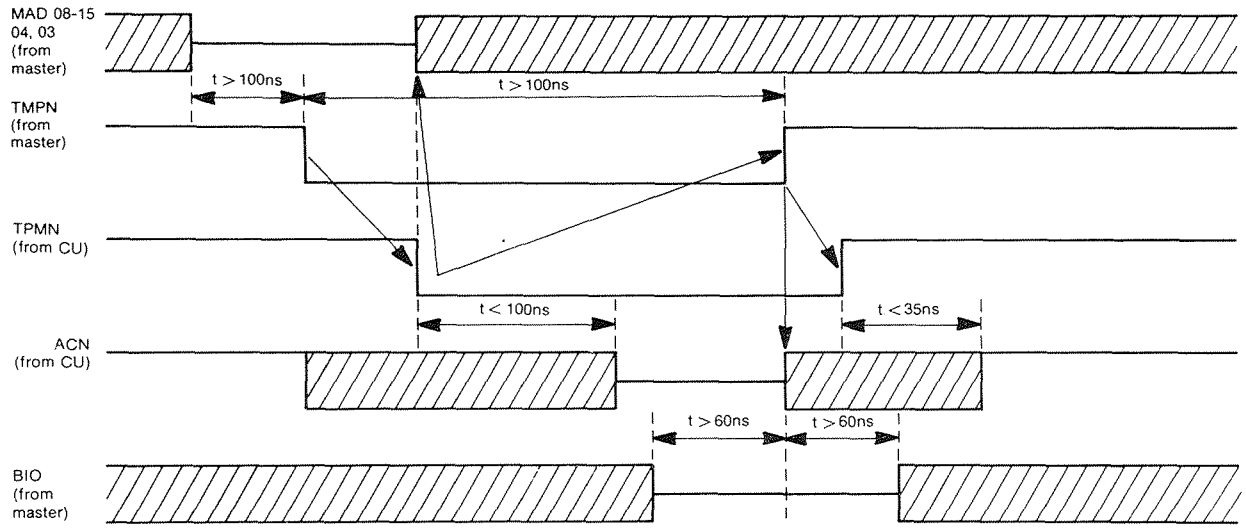
Exchange-Control Unit to Master

This exchange is used to send a word of maximum 16 bits from a peripheral control unit to a master. The instructions INR, SST and TST are executed during an exchange of this type.

INPUT TO REGISTER Instruction: INR

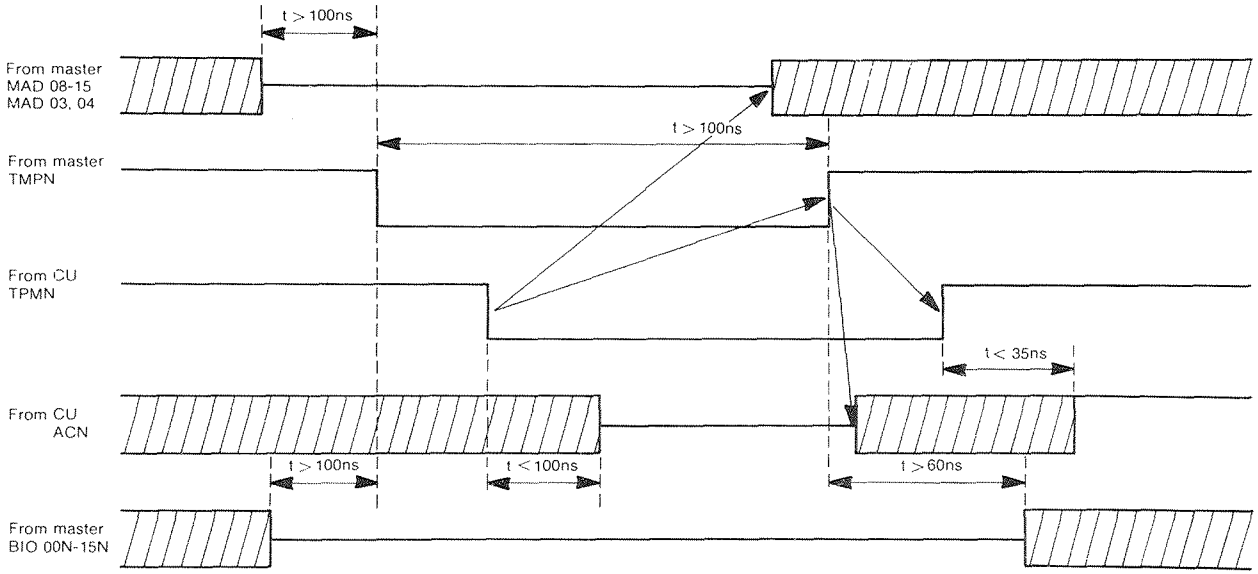
Instruction Format.





Note: CU which recognises its address specifies accept or not accept function using ACN. If function accepted CU must statisize data presented at least 60 ns before TMPN goes high again. Data must remain present at least 60 ns after TMPN goes high.

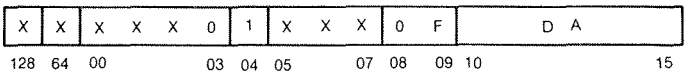
Figure 3.6. Timing for an OTR Exchange



This timing guaranteed by the master at any CU connection. The master must allow for 60 ns degradation due to propagation on the bus.
Control unit which has recognised its address answers TPMN and specifies accept or not accept I/0 command. CU may take into account the register contents either for answer ACN or for additional information.

Figure 3.7. Timing for a CIO Exchange

This produces the following MAD signals from the S register.



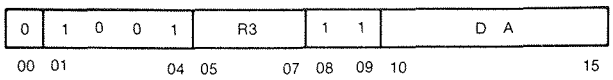
Note: Bit F permits the coding of two INR instructions, as required by certain CU's.

During the execution of this instruction the content of the BIO lines are loaded into the scratch pad register addressed by the r3 field of the instruction.

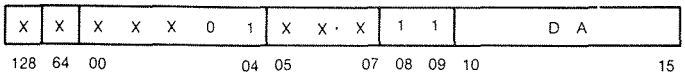
The information on lines TPMN and ACN from the control unit is copied by the CR register in the CPU as described for the master to control unit exchange.

SEND STATUS Instruction: SST

Instruction Format.



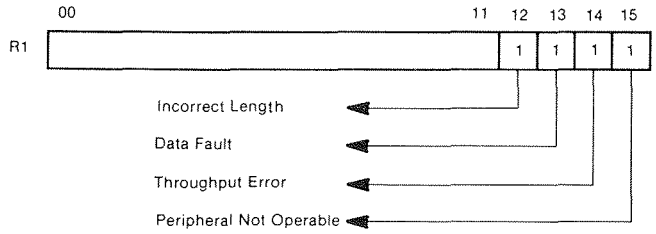
This produces the following MAD signals from the S register.



During the execution of this instruction the contents of the BIO lines are loaded into the scratch pad register addressed by the R 3 field of the instruction.

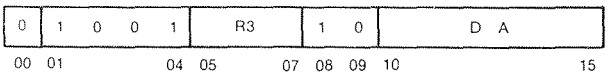
The information on bus lines TPMN and ACN from the control unit are copied by the condition register CR of the CPU.

This instruction allows the programmer to read the Status Word of the control unit; the length of this word may vary. Some bits in the word have a fixed position and meaning as indicated below:

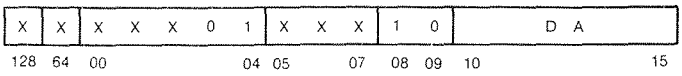


In general a 1-bit in the status word indicates a special condition.

Instruction Format.



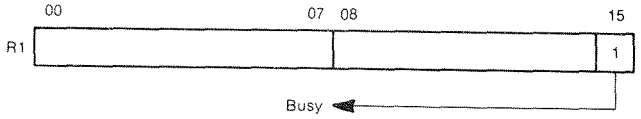
This produces the following MAD signals from the S register.



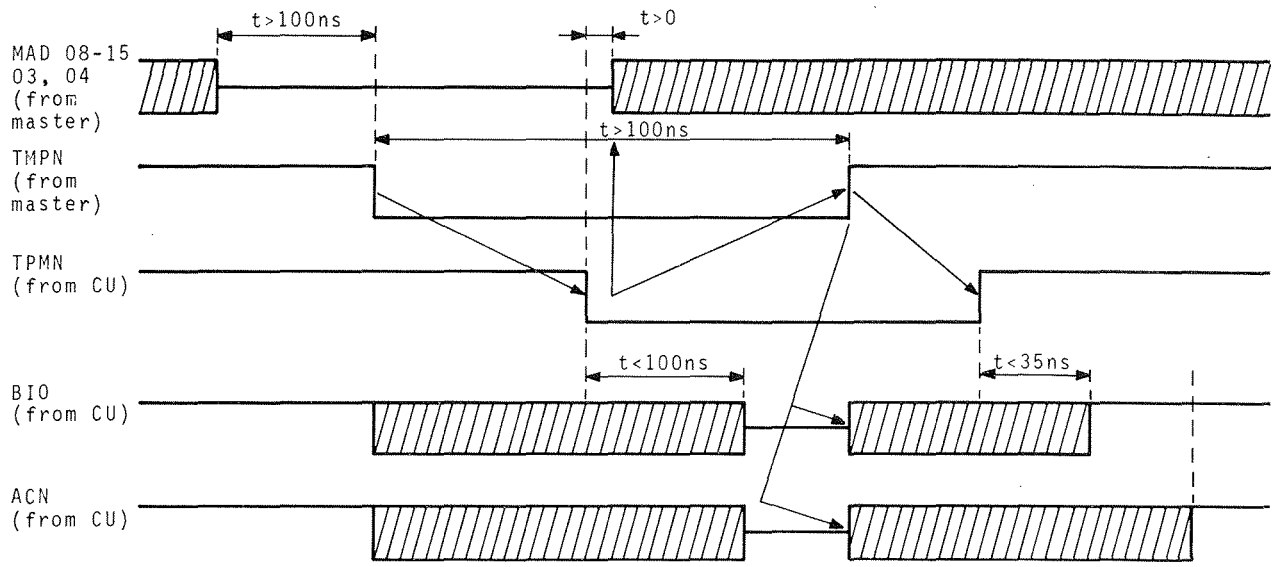
During this instruction the contents of the BIO lines are loaded into the scratch pad register addressed by the R3 field of the instruction.

The information on bus lines TPMN and ACN from the control unit are copied by the condition register CR of the CPU.

The control unit always accepts the TST instruction. Bit 15 has a specific meaning as indicated below:



The timing for the INR, SST, and TST instructions is shown in Figure 3.7.



Note: CU which recognises its address answers with TPMN and specifies accept or not accept function using ACN. CU writes data on the BIO lines within 100ns after TPMN - master may then suppress the address.
Master accepts the data then informs CU, by making TMPN high, that CU must release the BIO lines within 35ns after TMPN goes high.

Figure 3.8 Timing for INR, SST and TST Exchange

GUARD TIMES

In exchanges involving memory (or external registers), very short guard times have been adopted to exploit to the full the increased memory performances which arise when using fast logic circuitry. In this respect the ratio

$$\frac{\text{total exchange time}}{\text{minimum possible exchange time}}$$

is kept as small as possible.

To permit very simple designs to be used for control units rather longer guard times are used in order to give the CU more time to decode addresses or to load data.

Note: Guard time is defined as the time which is allowed to elapse between the initiation of an exchange and the validation of the data: it is necessary to make such an allowance because of the settling or *deskewing* time required for the logic circuitry associated with the memory to reach a stable state.

4 I/O Processor Channel

The I/O processor channel permits any peripheral equipped with a suitable control unit to take part in the high speed direct transfer of variable-length blocks of data, using the GP bus data lines, between the control unit and the block or blocks of the central memory. The same control unit as used on the programmed channel can be used as all control units have a standard interface connection.

The direct transfer method considerably reduces the I/O occupation of the CPU and permits data transfer rates of up to 1.1 million words per sec. to be achieved when the fast memories are used. Details of the connections required when using I/O processor channel are given in Part 2 of this Section of the manual.

OPERATION

The block diagram in Figure 4.1 shows the control signals provided by one I/O processor when controlling one control unit via a subchannel of the I/O processor which employs two program addressable external working registers. Each I/O processor may control up to 8 control units, each via a subchannel, (8 x 2 external working registers total) and a maximum of 8 I/O processors may be connected to the GP Bus (this maximum is only possible when using 400 or 500 series mounting boxes). The control units are given a preallocated priority level, which is sampled by the I/O processor during an operation, and each I/O processor in the system is given a bus priority level which is decided by hard wiring.

During an exchange the I/O processor is involved in three types of operation: initialization, exchange, and sampling. For the initialization operation the CPU is master and the I/O processor a slave. I/O instructions are used to effect transfers between the CPU internal registers and the external working registers of the I/O processor and to initialize the control unit. During data exchanges the I/O processor is the master and controls the dialogue between the memory and the control unit which are both slaves. When carrying out sampling operations the I/O processor is sampling break requests from its associated control units; the I/O processor does not occupy the bus for this type of operation.

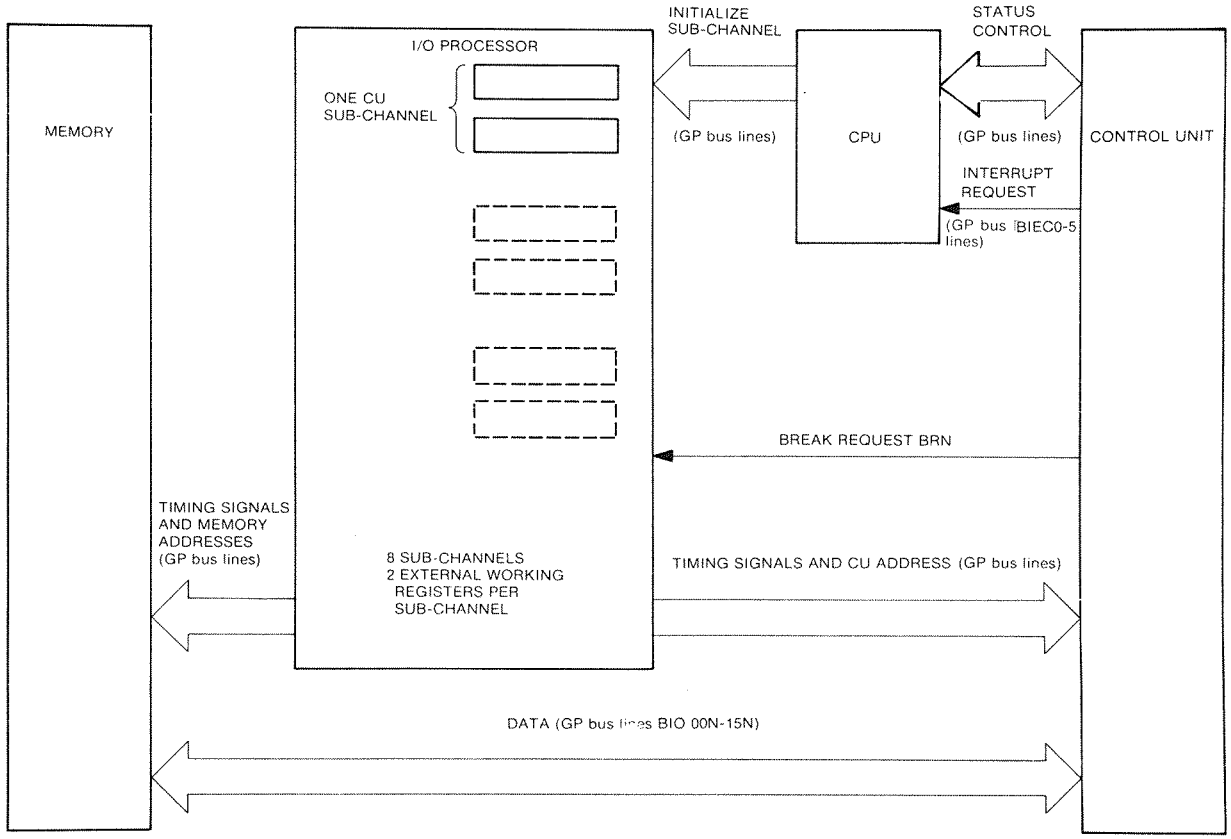
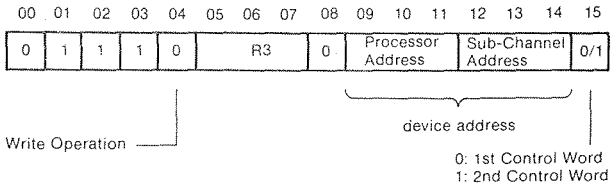


Figure 4.1 I/O Processor Channel

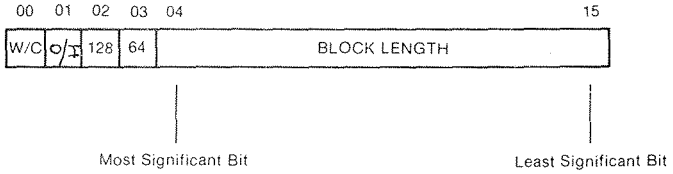
Initialization Operation

This operation is applied before commencing data transfer and the first part is controlled by the use of two write external register instructions WER to transfer two control words to the two working registers of the I/O processor subchannel. The instruction format is as shown below:



This instruction transfers the contents of the R3 field, previously loaded with a control word, to the external working register of the I/O processor specified by the device address bits 09 to 14.

The format of the first control word loaded is shown below:

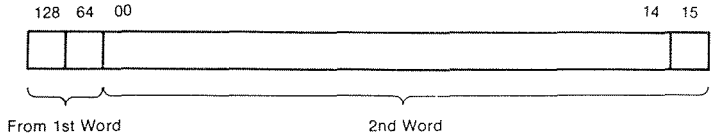


Bit 00 = 1 Exchange is in word mode.
= 0 Exchange is in character mode.
Bit 01 = 1 Exchange is from memory to control unit.
= 0 Exchange is from control unit to memory.
Bits 04 to 15 specify the number of characters/words to be transferred.
Bits 02, 03 are positioned to become the two most significant bits of the second control word.

The format of the second control word loaded is as follows:

Bits 00 to 15 specify the starting address in memory.
Bit 15 = 1 Right character is addressed } If transfer is
= 0 Left character is addressed } in character mode.

This address permits memory addressing up to 32k but with the addition of the two bits from the first control word the memory addressing range can be extended up to 128k. The 18-bit memory address, which is updated at every transfer as the current memory address, then has the format shown below:



After this instruction the control unit is also then initialized using an I/O instruction of the type CIO start (described in Chapter 3). The contents of the device address field (bits 10 — 15) should be the same as bits 09 — 14 of the WER instructions already used for initializing the control unit subchannel of the I/O processor.

Exchange Operation

After initialization the control unit passes to the *exchange* state in which it generates break requests, at the transfer frequency of the peripheral, to the I/O processor to indicate that it is ready to effect a data transfer. These BR's are attended to by the I/O processor which is directing the exchange thus ensuring automatic execution simultaneously with the running of the program and without any program intervention.

The I/O processor detects a Break Request and then requests control of the bus — the bus priority and request procedure is described in Chapter 5. Having obtained the bus the I/O processor resamples the incoming Break Requests — the exchange actually executed will be that corresponding to the highest priority even though the bus request may have been made in response to another, lower priority, BR. This method reduces the waiting time of the high speed peripherals connected to the higher priority BR lines.

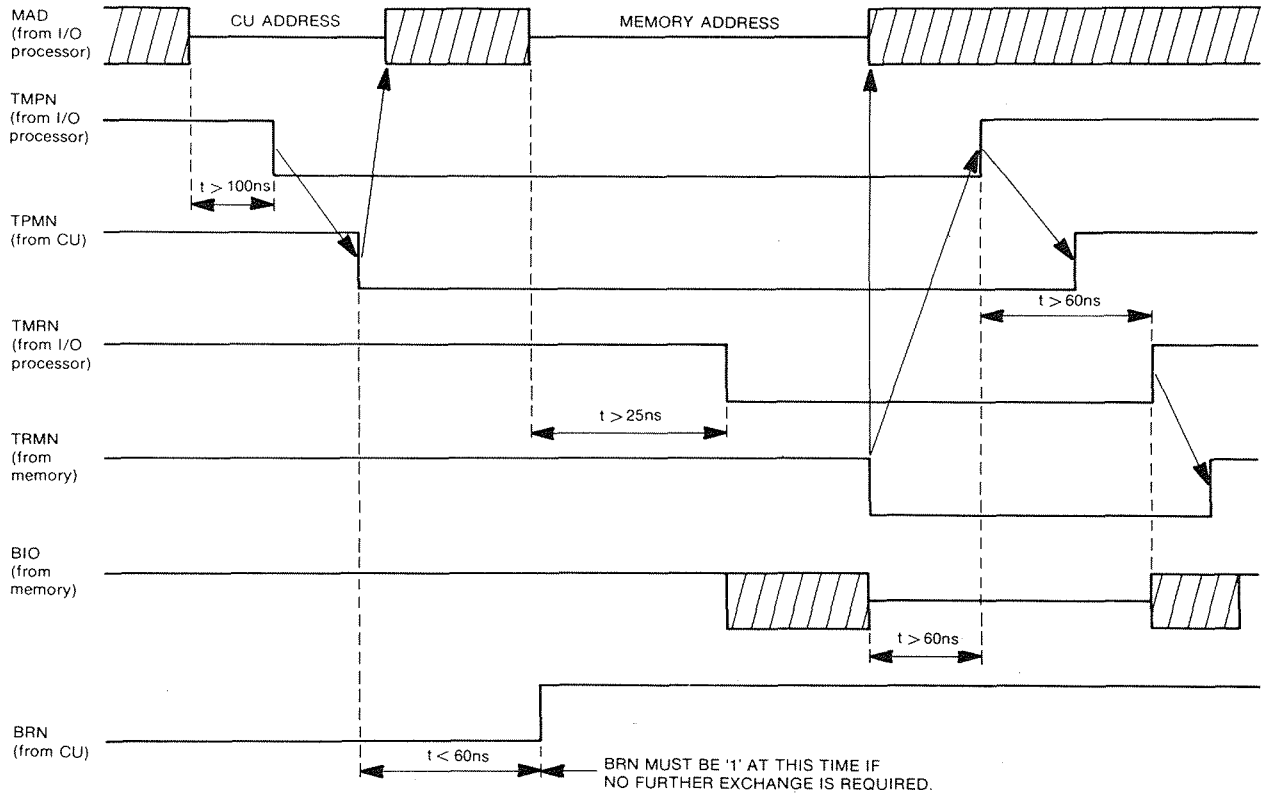
Having selected the subchannel corresponding to the Break Request with the highest priority the I/O processor then reads and updates the first control word in that subchannel and then addresses the associated control unit using GP bus lines MAD10-15 with the function required sent on MAD04, 08 and 09. After recognising its address the control unit responds with TPMN. The contents of MAD lines 10-15 may then be changed.

The I/O processor then reads and updates the second control word for the subchannel selected and sends the address to be used for the exchange to memory on MAD 0-15. The data exchange then takes place between the control unit and memory, using GP bus lines BIO 00 — 15, under the control of the I/O processor.

It should be noted that the I/O processor uses during the exchange the same address and data lines as used on the programmed channel but simulates only the OTR and INR instructions; CIO operations are performed by the CPU only. The I/O processor ignores signal ACN. The timing sequence for a memory to control unit exchange is shown in Figure 4.2 and for a control unit to memory exchange in Figure 4.3.

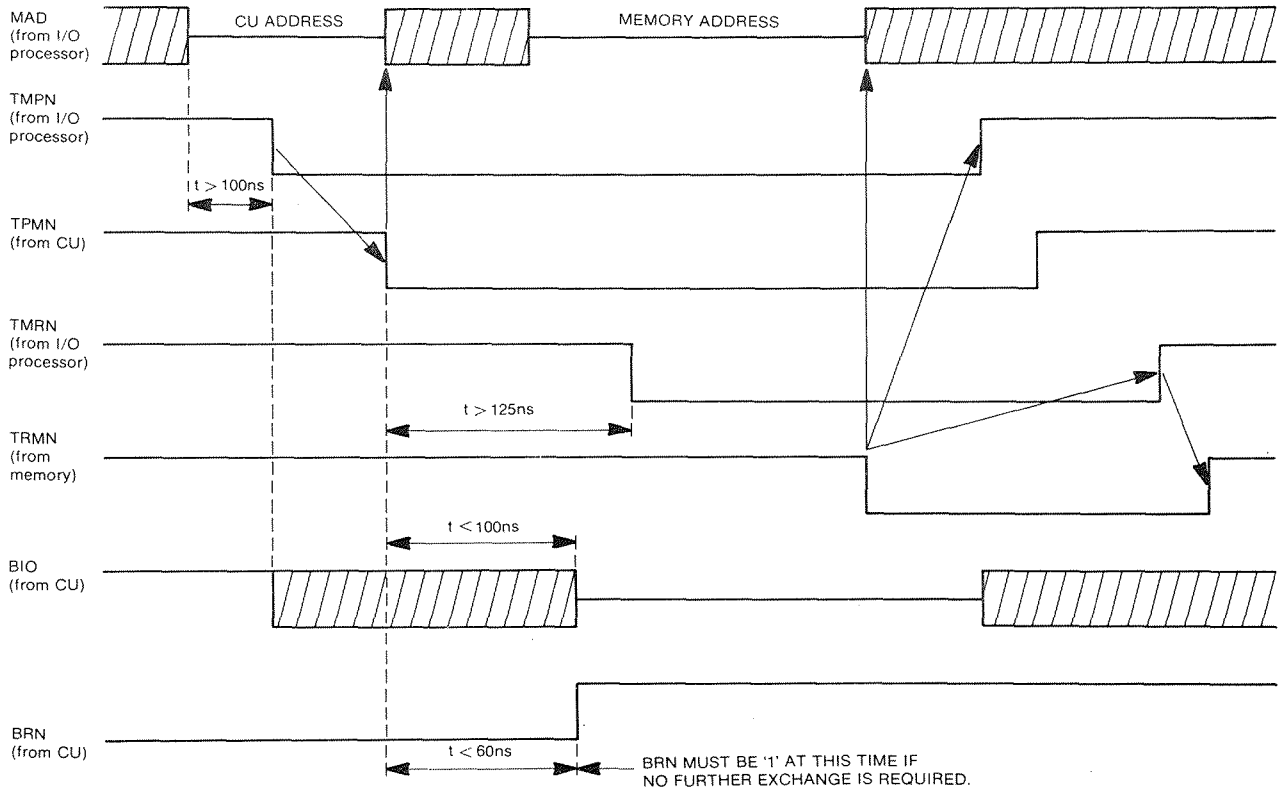
A further exchange with respect to overall transfer would be carried out immediately if the control unit activates its Break Request line again during the execution of the current exchange and provided no other higher priority break requests or bus master requests are outstanding.

End of Block Transfer
When the length of the block becomes zero in the I/O processor a signal from the I/O processor (MAD 03 = 1) to the control unit becomes active at the beginning of the last exchange to indicate that the control unit should not return to the exchange state at the end of the transfer. The control unit then behaves as if it had received a CIO halt instruction and passes into a condition in which it is awaiting an SST (Send Status instruction). The control unit does not then generate BR's but instead sends an Interrupt Request IRN to the CPU, using the 6 interrupt encoded lines of the GP Bus, which is dealt with according to its priority level. The CPU then executes an SST instruction to read the status of the CU and, following the execution of this instruction, the control unit becomes inactive, ready to be re-initialized for a new transfer.
In the interrupt Subroutine at the end of the exchange the programmer can check the length remaining to be transferred by re-reading the field of the first control word which contains the *length-of-block* parameter. An instruction of the type Read External Register RER is used — described in Chapter 2 of Part 3 of this manual — and as a result it is either verified that the transfer has been completed or, if the exchange was terminated not by a command from the I/O processor but by reason of special conditions arising in the peripheral (not operable, throughput error etc.), the length of block remaining to be transferred is determined.



Note: This timing diagram is a combination of the timings for OTR and read memory.

Figure 4.2. Memory to CU exchange — timing diagram

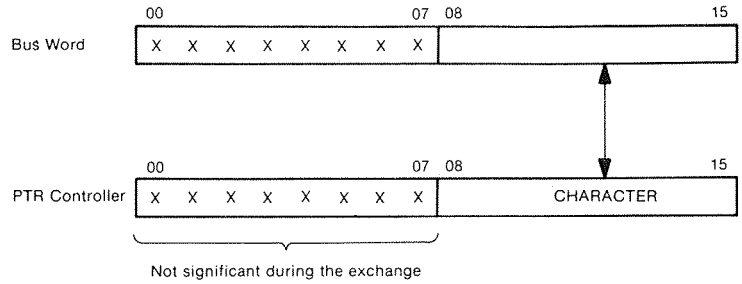


Note: This timing diagram is a combination of the timing diagrams for INR and write memory.

Figure 4.3 CU to Memory exchange — timing diagram

EXCHANGE OF VARIABLE LENGTH BLOCKS

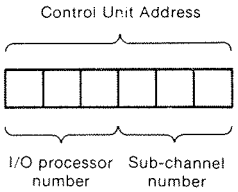
When the control unit exchanges characters (e.g. for a paper tape reader) or words of less than 16 bits (card reader) the connection to the GP bus is such that the right hand bits are transferred in output mode as well as input mode as shown in the following example.



This convention together with the ability of the memory to handle characters allows optimum use of the memory.

RELATIONSHIP BETWEEN BREAK SIGNAL AND CONTROL UNIT ADDRESS

An I/O processor is coded with a 3-bit number which may range from 0 to 7 (coded by straps on the I/O processor) and each of the 8 subchannels associated with an I/O processor is also coded with a 3-bit number from 0 to 7. The combined 6-bit number gives the address of the peripheral control unit as shown below:



This address is used by the WER instruction to load the 2 control words into the working register of the I/O processor.

The 8 incoming break request signals BR00 to BR07 to each I/O processor correspond to each subchannel in the I/O processor as follows:

BR00N corresponds to subchannel 0
BR07N corresponds to subchannel 7

BR00N has the highest priority and BR07N the lowest.

SYSTEM ORGANIZATION RULES

The I/O processors are always located in the basic mounting box and the user may connect a maximum of 8 control units to each I/O processor which can be located either in the basic mounting box or in an equipment shelf. Control units belonging to separate equipment shelves may also be connected to the same I/O processor in the basic mounting box.

PROGRAMMING

Programming for I/O processor data transfer must include the programmed channel routine for calling up the required control unit with CIO instructions and for shutting down after a data block has been transferred. Routines must also be included for loading the control words with WER instructions and for reading the I/O processor registers with RER instructions. The Basic Operating Monitor for I/O processor contains complete routines for I/O processor operation.

PERFORMANCE

The examples given show how simple the interfaces can be. In the more complex cases the interfaces are still kept simple by a suitable choice of guard times.

High performance is obtained by minimizing bus occupation time. This is relatively easy to realise in the design of the control unit.

On the I/O processor channel with a memory having an access time and cycle time of 300 ns and 0.7 µs respectively the transfer rate can approach the memory speed provided that the control unit is situated close to the basic cabinet.

Only masters (defined in Chapter 1) may obtain control of the GP bus and where several masters try to obtain control, a system of priority is used. The order of priority is decided by hard wiring (chaining system) and priority is given first to input/output units and then to the CPU. The following order of priority would be typical for a system in which a DMA controller and three I/O processors were fitted:

PRIORITY	UNIT
1	DMA controller
2	I/O processor No. 1
3	I/O processor No. 2
4	I/O processor No. 3
5	CPU

CPU PRIORITY

Although the CPU has the lowest priority in the system, as determined by the hard wiring, it must be given control of the bus at the earliest opportunity in order to process the instruction. The CPU is therefore given control of the bus automatically at the end of an instruction to access to the new one and is also given control when it makes a request provided:

1 There is no other request for bus control present.
2 There is no master already selected for the coming exchange.
3 There is no exchange taking place on the bus.

The CPU is given the highest priority for bus allocation in the case of power failure.

BUS ALLOCATION SYSTEM

The organization of the bus allocation system is shown in the block diagram in Figure 5.1. The six signals which control the system are carried on the GP bus, but in the basic cabinet only, and are as follows:

BUSRN — Bus request (active '0')
BSYN — Bus busy (active '0')
MSN — Master selected (active '0')
SPYC — Scan priority chain (active '0')

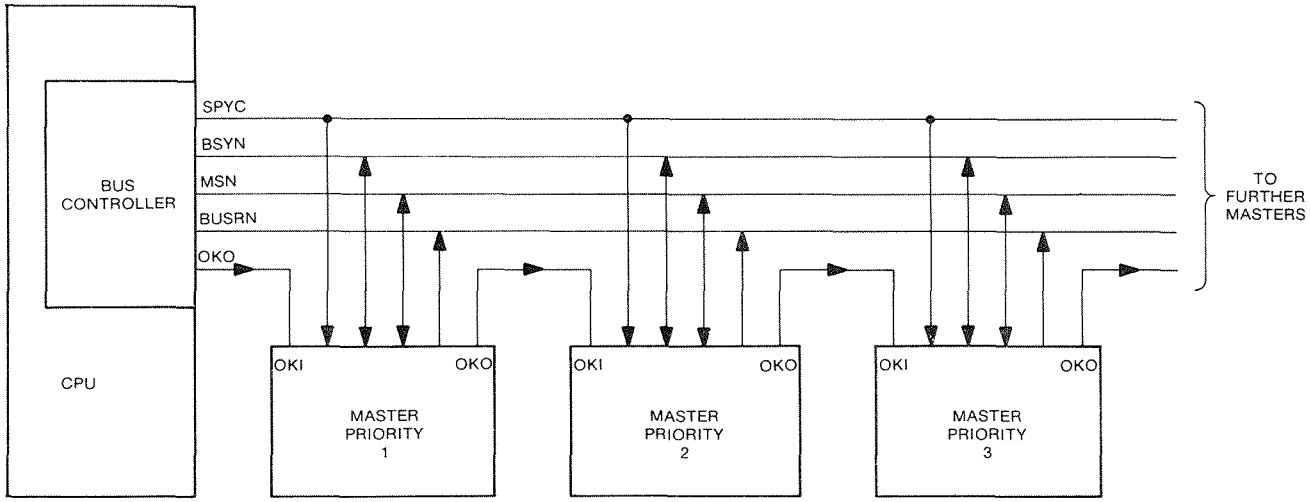


Figure 5.1 Bus control priority structure

OKO — OK output (active '1' — for selecting master)
OKI — OK input (active '0' — to block lower priority masters)

The OKO signal from the bus controller is connected as the OKI signal to the master having the highest priority and is then passed on as OKO from this master to become OKI for the next master in order of priority. This wiring arrangement is continued down the chain, as shown in the block diagram, and is used to ensure that only the highest priority master which is making a bus request is given access to the GP bus, as explained in the system operation description.

SYSTEM OPERATION

The following description should be read with reference to the timing diagram shown in Figure 5.2 and to the typical logic assembly which is contained in each master, shown in Figure 5.3. Note that all bus request operations are asynchonous and that timings are measured at the masters connector level.

In operation a master requiring access to the bus raises BUSRN (logic '0'). This is only possible if MSN is '1'. In response to BUSRN the bus controller raises SPYCN (logic '0') and this signal clocks a flip-flop in all the masters causing the signal RAN to be produced in the master or masters which have raised a request.

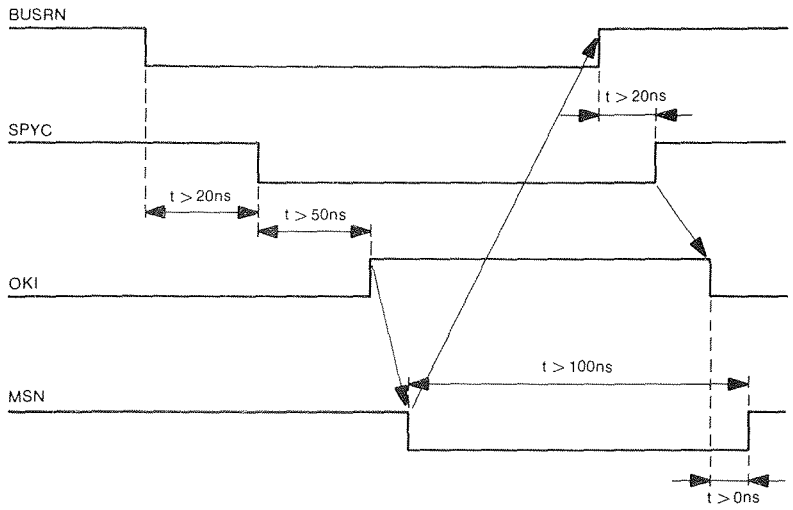


Figure 5.2 Bus request timing

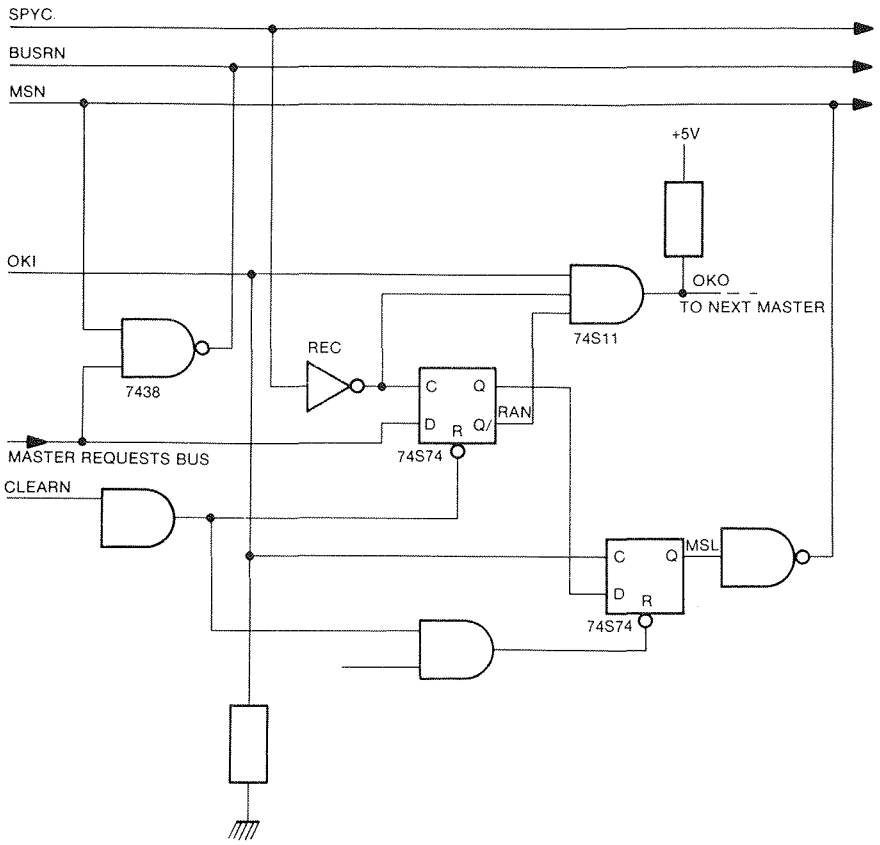


Figure 5.3 Typical logic assembly

The bus controller raises OKO (logic '1') after signal SPYCN, thus allowing time for the RAN signal to be raised by the master or masters. This signal is passed down the priority chain (OKI to each master) as logic 1 until it is blocked at the highest priority master in which signal RAN has been produced. The OKO signal is then passed on as logic 0 to the remaining masters of lower priority in the chain to prevent them from acquiring access to the bus.

Note: Should there be a power failure the bus controller suppresses OKO which allows the CPU to take control of the bus as soon as the current exchange has finished.

At the master which has requested access to the bus signal OKI clocks a further flip-flop which raises MSL and causes bus line MSN to be raised to '0'. This signal now blocks all activity of the priority chain (BUSRN can not be set to 0 at other masters) and the master which sets MSN to 0 now will become the new master of the bus when the bus is free.

When MSN is raised BUSRN is reset to the non-active state, (logic 1), and this causes the bus controller to suppress signal SPYCN and then OKO. If the bus busy line BSYN is at logic 1 the selected master concludes that the bus is available for it to access and MSN therefore raises BSYN (logic 0). This signal indicates to the CPU, and to all other masters, that an exchange is now in progress. MSN also sets the masters request signal to '0' if another exchange has not been requested. After BSYN has been raised MSN is reset to logic 1 to release the priority chain and allow the selection of the next master. The master maintains control of the bus, however, until it releases the bus by re-setting BSYN (logic '1').

Timing of the Priority Chain

The timing diagram in Figure 5.4 shows the relationship between the OKO signals generated by the bus controller and the logic signals generated by a

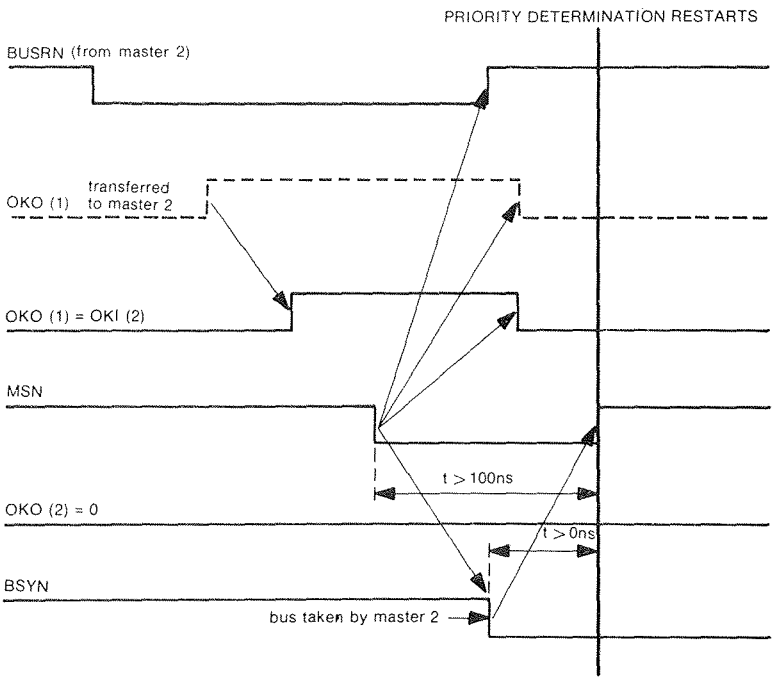


Figure 5.4 Priority chain timing

master to control the priority chain. The diagram is drawn for a master which is not the CPU and illustrates the timing for two situations as follows:

1. Master not controlling the bus.
2. Master having requested the bus.

Overall Performance
Figure 5.5 illustrates the overall performance of the priority chain. In the example given the time between BUSRN being set to 0, to make a request for access to the bus, and the transition of signal MSN is typically 150 ns. The example illustrates the worst case situation which is for a master connected in position 8 (least priority) of the priority chain. Note from the diagram that a new priority determination may occur after every 250 ns — when MSN is returned to logic 1.

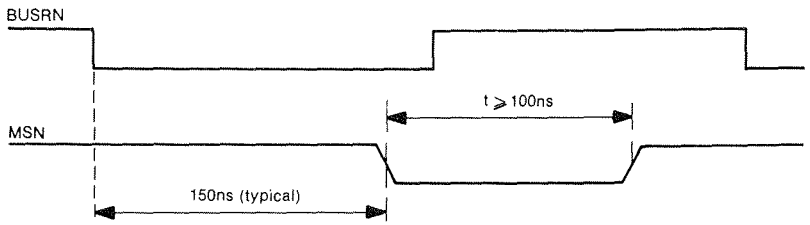


Figure 5.5 Priority chain performance

CHAINING OF THE EXCHANGE

In addition to the signals which control the allocation of the GP bus to a master there are other signals, also carried on the GP bus, which control the timing of an exchange. These signals are:

TMPN	—	Timing, master to peripheral
TMRN	—	Timing, master to memory
TMEN	—	Timing, master to external register
TPMN	—	Timing, peripheral to master
TRMN	—	Timing, external register or memory to master

All these signals are active low.

Signals TMPN, TMRN or TMEN are used depending on whether the master is exchanging with a peripheral unit or with a memory or an external register. The only restrictions for setting the signals are:

1. The GP bus must be free
2. The GP bus must be allocated to the master which sends the signals
3. The timing for the exchange must be respected.

Signals TPMN or TRMN are reply signals to TMPN, TMRN or TMEN and are used in conjunction with BSYN to indicate that the new master may start an exchange. The signals also terminate the exchange. Signal BSYN must continue to remain low after the transition of TRMN or TPMN to logic 0. The time relationships are illustrated in Figure 5.6.

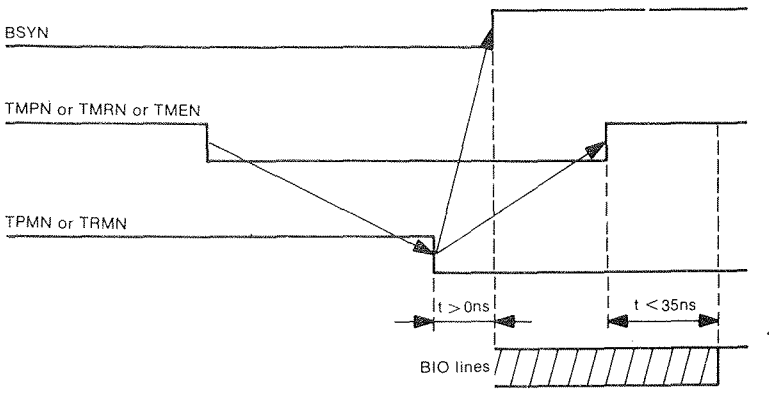


Figure 5.6 Exchange timing

The last of the signals, TMPN, TMRN or BSYN to arrive ensures that the bus lines are set to 1 by the master within 35 ns after this signal went to logic 1.

If TPMN is the last signal to go up, the next master before asserting its BIO lines must take care of a 60 ns degradation time due to the transmission on the bus lines. This means that the BIO lines will be freed at the master level 60 + 35 = 95 ns after TPMN goes high.

If reply signals TPMN or TRMN are not received at the bus controller in response to TMPN or TMRN thus causing either of these signals to last for longer than 6.4 μs then the bus controller generates a *time-out* signal by setting TRMN or TPMN bus lines to 0. The signal generated then clears the priority logic system.

An example of an exchange between a master and a control unit followed by an exchange between another master and a RAM is given in Figure 5.7. The time relationship given illustrates the situation where one master requests access to the bus shortly after another master has requested access.

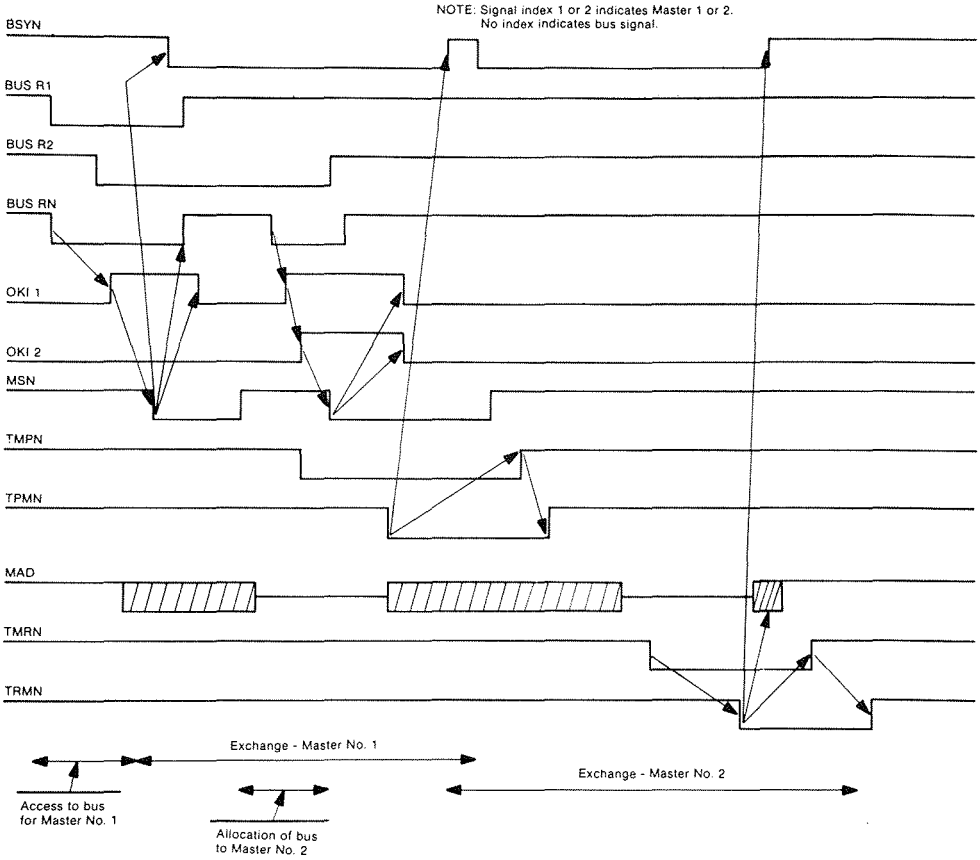


Figure 5.7 Exchange chaining — Master 1 — CU, Master 2 — RAM

6 Interrupt System

The interrupt system is used generally for all peripheral operations and for handling internally generated interrupts. It can deal with up to 63 interrupt request levels. Interrupts are handled according to their priority, and the highest priority (lowest number) interrupt request is accepted and compared with the priority level of the running program. If the priority level of the interrupt is higher than that of the running program and provided that the CPU is in the *enable interrupt* state then the program is interrupted. The contents of register P (program counter) which contains the address of the next instruction in the running program, and the program status word (containing the priority level and other information) are then stored in a memory stack. The CPU goes into the *inhibit interrupt* state and the priority level of the new program is stored in the PLR register. An indirect branch is made to the memory location pointed to in the PLR and the new interrupt routine is then executed. During the execution of the interrupt routine the CPU may be returned, by means of an ENB instruction, to the *enable interrupt* state and in this case the routine may in its turn be interrupted by another program of still higher priority.

Instructions for programming the interrupt and stack system are given in **Part** 3 of this Section of the manual and details of the interrupt request line connections are given in Part 2 (also of this Section).

INTERRUPT SYSTEM

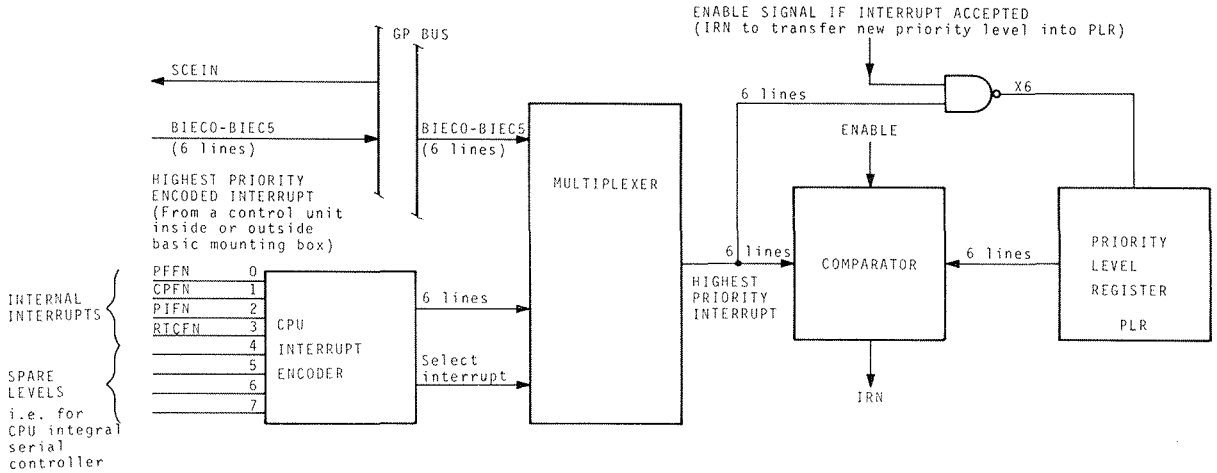
The system can handle a total of 63 interrupts. Of these interrupts eight, numbered 0 to 7, are encoded by an encoder mounted in the CPU; level 0 has the highest priority. The remaining interrupts, which are used by control units mounted either inside the basic mounting box or in external equipment shelves are encoded onto the BIEC0-5 lines of the GP bus. The system arrangement is shown in the block diagram in Figure 6.1.

The eight interrupts encoded by the CPU encoder include four internal interrupts pre-wired from CPU logic functions which are as follows:

Level 0 - PFFN Power Failure/Automatic Restart.
Level 1 - CPFN Control Panel (Operators interrupt)
Level 2 - PIFN Program Interrupt (Stack overflow, Link to Monitor)
Level 3 - RTCFN Real Time Clock

The remaining four interrupts may be connected to other units in the basic mounting box but are not used for C.U. I/O channel operations,

A 6-bit binary number is output from the CPU encoder representing the



Note: Comparison is only done if ENABLE is true.
ENABLE: -ENB instruction or RTNA15 instruction.
ENABLE: -Interruption or inhibit instruction

Figure 6.1 Interrupt system

highest priority interrupt which exists on the eight levels and a *Select Interrupt* signal is also output from the encoder which is set to active only if an interrupt is present on one of the levels.

The remaining interrupts which are encoded onto the BIEC0-5 lines of the GP bus are scanned by SCEIN — scan external interrupts — which is sent by the CPU at the beginning of each two instructions (fetch cycle) and a 6-bit binary number representing the interrupt having the highest priority is then placed on lines BIEC0-BIEC5 and conveyed via the GP bus to the CPU.

Both the CPU encoder output and the BIEC0-5 lines are input to a multiplexer in the CPU, together with the *Select Interrupt* signal. If an interrupt exists on the CPU encoder output then the *Select Interrupt* signal is active and is used in the multiplexer to gate the 6 lines containing the encoded interrupt signal from the CPU encoder to the multiplexer output. If no interrupts exists on the CPU encoder output then the *Select Interrupt* signal is inactive and the coded signal on the BIEC0-5 lines then appears at the multiplexer output. This arrangement ensures that the 8 interrupts connected to the CPU encoder always have priority over the interrupts connected via the BIEC0-5 lines.

The coded signal at the multiplexer output, whether from the CPU encoder output or from the signal on the BIEC0-5 lines is input to a comparator where, if an ENABLE signal is true, it is compared with the contents of the PLR register which holds a 6-bit code representing the priority level of the current running program. The comparator gives a program interrupt signal IRN only if the priority level of the interrupt is higher (lower number) than the priority level of the running program. The encoded signal from the multiplexer output is also taken via six gates to the input of the PLR register. The signal is gated into the register if the new interrupt is accepted as explained in the Interrupt Handling description.

INTERRUPT HANDLING

The operation of the interrupt and stacking system is shown in schematic form in Figure 6.2. A hardware routine is started by the program interrupt signal IRN from the comparator, after the current program instruction is completed, which commences by inhibiting further interrupts thus allowing the present interrupt to be serviced. The contents of P register and PSW register are stored in a memory stack addressed from the stack pointer (scratch pad register A15) and the stack pointer register is decremented by 2 to point to the next location in the stack.

On completion of the stacking operation the 6-bit number fed into the comparator, representing the priority level of the new program, is loaded into register PLR (Part of PSW register). An indirect branch is then made to the

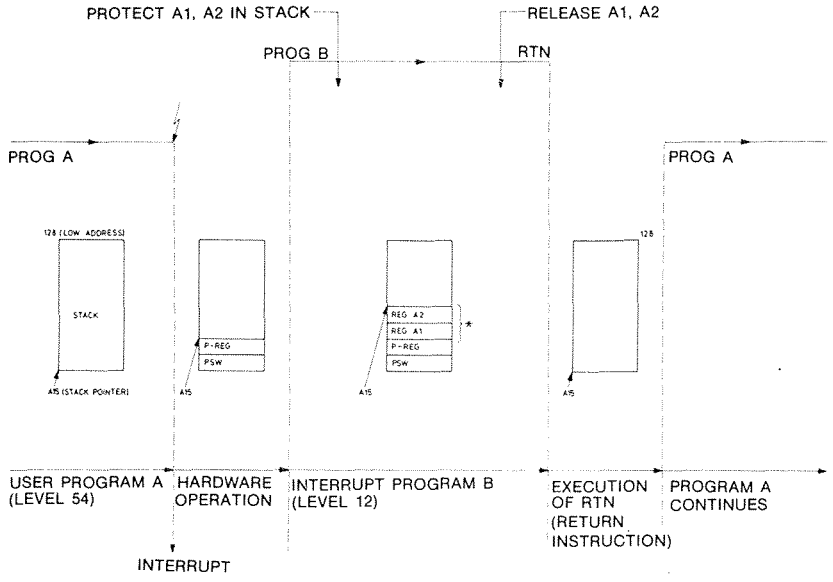


Figure 6.2 Interrupt and stacking system

memory location, pointed to by PLR. This location is the hardware interrupt location which is determined by the level of the interrupt and contains the starting address of the interrupt routine. The start address contained in the address location is then loaded into the P register.

The P and PSW register now contain information which is relative to the new level of program. The address at which the interrupted level is to restart, and the PSW for this level, are held in a memory stack. Instructions are now carried out from the new program level and it is these instructions which action the interrupt and define the software interrupt action to be taken. The new program started by the interrupt will normally contain routines to save the contents of registers for the old program. The program may also include an ENB instruction, to enable the interrupt system to accept new interrupt requests. If there is no ENB instruction in the interrupt routine any interrupt occurring, even with a higher priority, will not interrupt the first interrupt routine. An RTN instruction will then act as the ENB instruction to authorize again the interrupt system to accept a new interrupt request. These routines are included in the Basic Operating Monitors provided with the P852M/P856M/P857M systems.

Return to the interrupted program is initiated by a return instruction RTN which specifies register A15 as the stack pointer within this instruction. The

program status word and program address for the interrupted program which are contained in the address in the memory stack pointed to by the stack pointer are loaded back into the PSW and P registers. The stack pointer register A15 is incremented to point to the previous location in the stack. Before program action resumes on the interrupted program any outstanding interrupts are checked for priority against the 6-bit value in the PLR part of the PSW register. If a higher level interrupt exists then it is serviced in the manner previously described. If no level interrupt exists then the original program level continues.

INTERRUPT SIGNAL ENCODING FOR A SYSTEM

The block diagram in Figure 6.3 shows a typical system installation using Type 1 control units (GP Bus connected). The 63 interrupt priority levels are encoded into a 6-bit number from 0 to 62 with 0 representing the highest level priority and 62 the lowest, the format is as shown below:

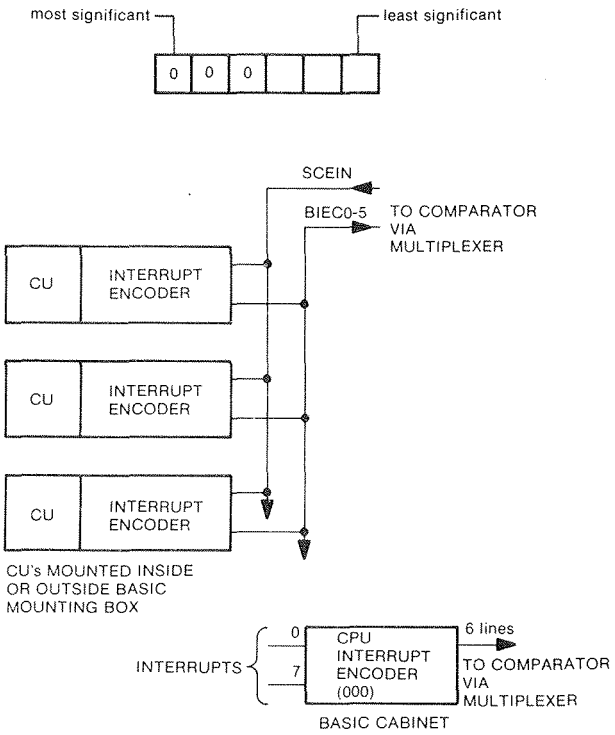


Figure 6.3 Interrupt system installation

Each Type 1 control unit mounted inside or outside the basic cabinet contains its own encoding logic and is prewired to give a 6-bit number on the BIEC lines corresponding to its priority level in the system.

Note: Details of the encoding facilities provided when using Type 2 control units (I/O Bus connected) in equipment shelves are given in Appendix 1 of this manual.

Interrupt Selection on BIEC0-5 lines

At the beginning of each instruction the scan external interrupt signal SCEIN is sent from the CPU to the encoder on each control unit. This signal permits any interrupts currently present to be actioned, and samples the condition of the lines BIEC0-5 to determine, in accordance with a priority algorithm, whether or not to maintain on these lines the information corresponding to the highest level of priority that it has detected.

Algorithm

Every interrupt level is coded with a 6-bit number ABCDEF with A the most significant bit.

If at least one interrupt is present at the moment of appearance of SCEIN then ABCDEF is written on the BIEC lines. Figure 6.4 shows the arrangement in simplified form for the three most significant bits ABC for two encoders connected to the common BIEC lines 0, 1 and 2

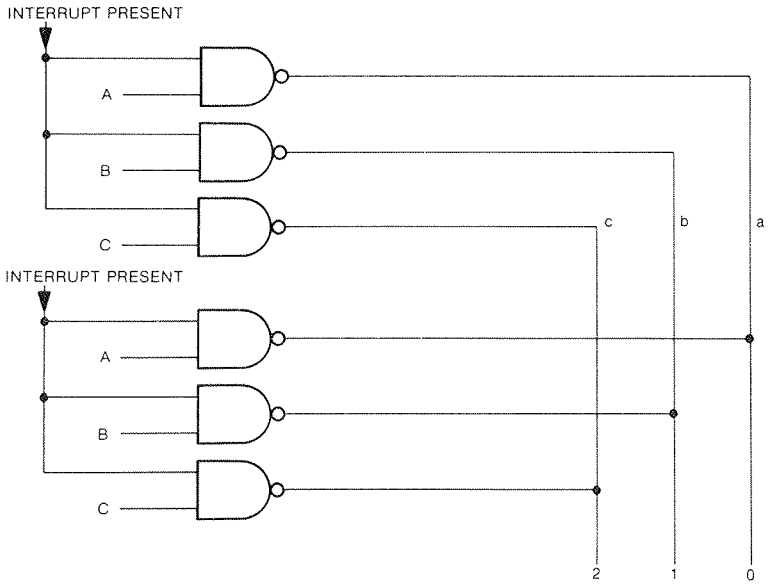


Figure 6.4 Simplified interrupt encoding

If, the state of the bus line 'a' is already 0 but number A is strapped to output 1 then 'a' remains at 0 (higher priority) and the output for B on line 'b' is replaced by a '1' (via feedback from the output lines on the board encoding logic). Otherwise the output for which B is strapped is maintained.

Similarly if the bus line 'b' is already 0 but the number B is strapped to output 1 then 'b' remains at 0 (higher priority) and the output C on line 'c' is replaced by a '1' as before. Otherwise the output for which C is strapped is maintained.

The sequence is continued in this manner for each BIEC line at each encoder until the BIEC lines contain, in encoded form, the number representing the highest priority interrupt present.

Timing of SCEIN Signal

The diagram in Figure 6.5 shows the BIEC line signals in relationship to SCEIN. The BIEC lines are required to be stabilized within 1800 ns from the appearance of SCEIN.

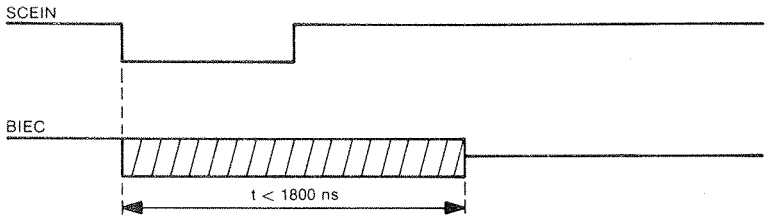


Figure 6.5 Signal SCEIN timing

7 Digital Input/Output System (DIOS)

The DIOS system provides the interface and control logic required to transfer digital information between user equipment and the P852M/P856M or P857M computers. The system connects directly to the GP bus and operates on programmed channel under software control at a maximum transfer rate of 10k words/sec inclusive of the I/O routine of the CPU with interrupt handling(calling mode)and at 50k words/sec without interrupt handling(sampling mode).

The programming rules for DIOS are given in Part 3 of this Section and full details of the hardware and connections in Part 2.

Configurations

There are two basic elements used in the DIOS system:

DIOD board 1	Dual Input/Output Digital with one input and one output channel for one 16-bit word.
DIOD board 2	Dual Input/Output Digital as for board 1 but with an additional input and output channel for a second 16-bit word.

A diagram showing the interfacing signals to a DIOD board 2 is given in Figure 7.1(for a DIOD board 1 Digital I/O word B would not be used). The interface signals are as follows:

Signals between DIOD board and CPU

GP bus signals which function as described in Chapter 2 of this part(General Purpose GP bus)

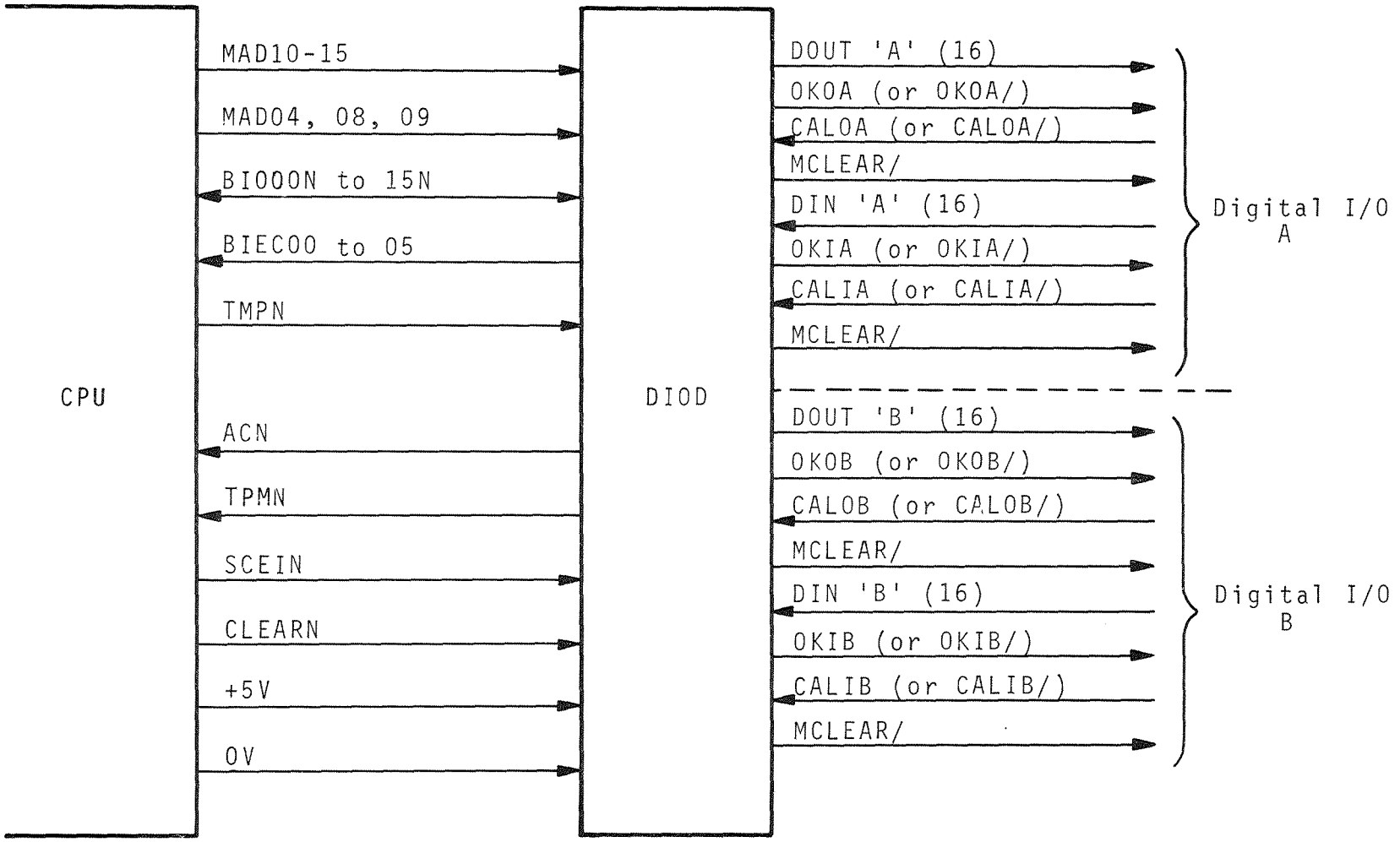


Figure 7.1. DIOD Interfacing Signals

Signals between device and DIOD board

Note: The signals are described for word A but similar signals are used for channel B.

DOUT A	16 lines which carry the data words from the DIOD board to the device.
DIN A	16 lines which carry the data words from the device to the DIOD board.
OKOA(or OKOA/)	Response from DIOD which indicates that a word(A) has been output from the DIOD board to the device.
OKIA(or OKIA/)	Response from DIOD board which indicates that a word (A) has been input to the DIOD board from the device.
CALOA (or CALOA/)	CALL signal from the device requesting a data word (A) from the DIOD board.
CALIA (or CALIA/)	CALL signal from the device requesting to input a data word (A) into the DIOD board.
M CLEAR/	A general reset signal for the devices.

Details of the link connections used when selecting CALL and OK signals can be found in Part 2 Chapter 6 of this Section.

Operation

A block diagram of the DIOD board is shown in Figure 7.2 and bus timing diagrams are given in Figure 7.3. The unit is connected to the bus and is controlled via the programmed channel using the following instructions shown in Table 7.1.

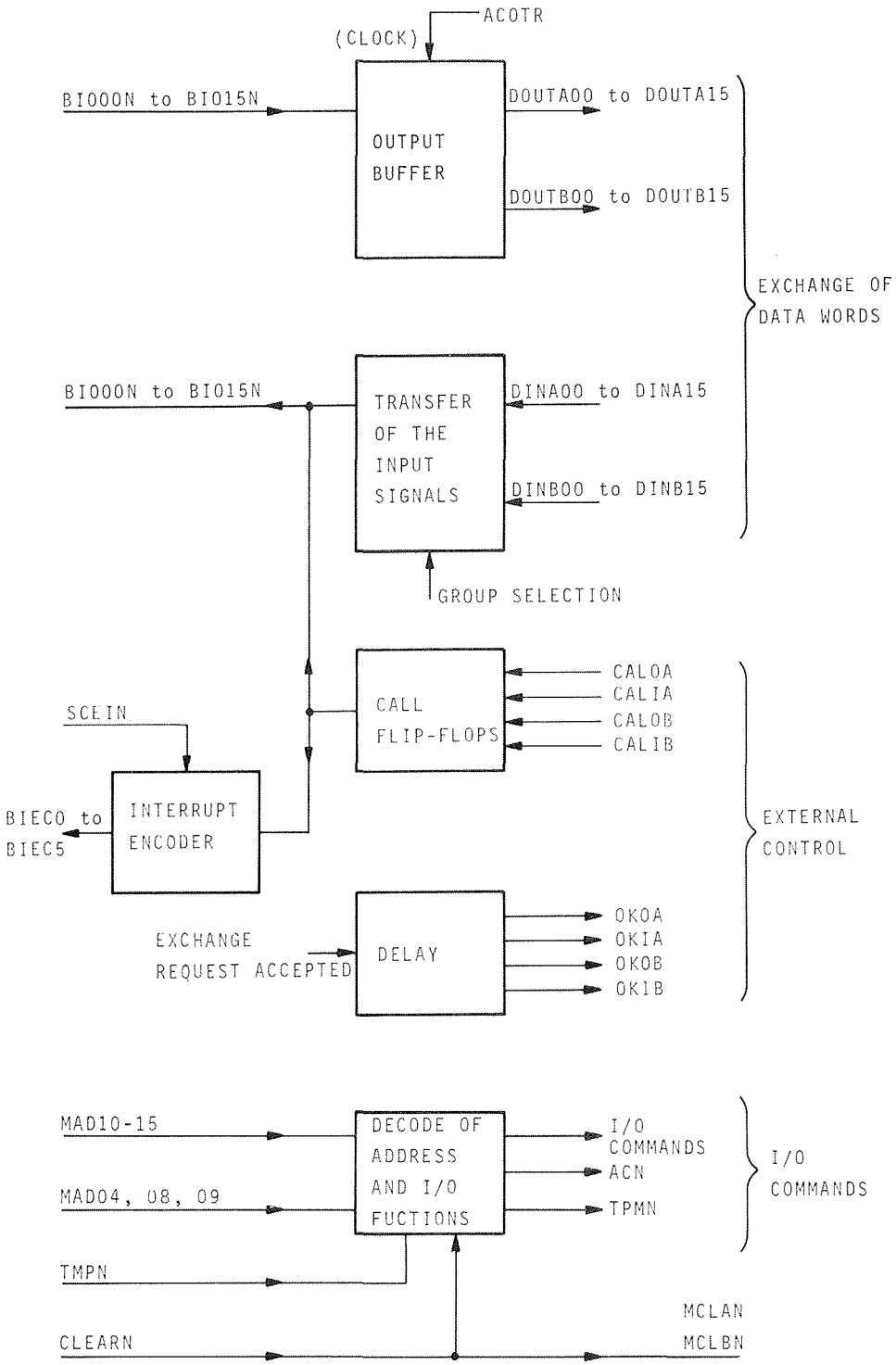


Figure 7.2. DIOD Board Logic-Block Diagram

Table 7.1 Command Instructions

MAD 04	MAD 08	MAD 09	Command
0	0	0	OTR Data
1	0	0	INR Data
1	0	1	INR Address
0	1	1	CIO Start
0	1	0	CIO Stop

Except for CIO Start and CIO Stop commands, which are always accepted, the other commands are gated before being stored in the command accept register.

Initiation by a Call Signal

Where data transfer is initiated by the user device a CALL signal is sent to the DIOD board to request each 16-bit transfer. The call signal is stored in a flip flop and an interrupt request is then sent to the CPU. The interrupt encoder on the DIOD board sends an interrupt request for either word A or word B on the BIEC 0 to 5 lines of the GP bus, if the request is of a higher priority than the existing request on the line as described previously in Chapter 6 Interrupt System of this Part. Information on the interrupt level selection links on the DIOD board can be found in Part 2 Chapter 6 of this Section.

The computer responds to the interrupt with an INR Address instruction addressed to the DIOD board which is decoded in the Address Decode and I/O functions logic to gate the contents of the CALL flip flop onto the BIO lines. A one bit in BIO 14N or BIO 15N indicates that the corresponding CALL flip flop is set as follows:

BIO 14N = CALI (Aor B) input word

BIO 15N = CALO (A or B) output word

The state of the calling lines are loaded into the R3 register of the CPU specified in the Address instruction. Note that the

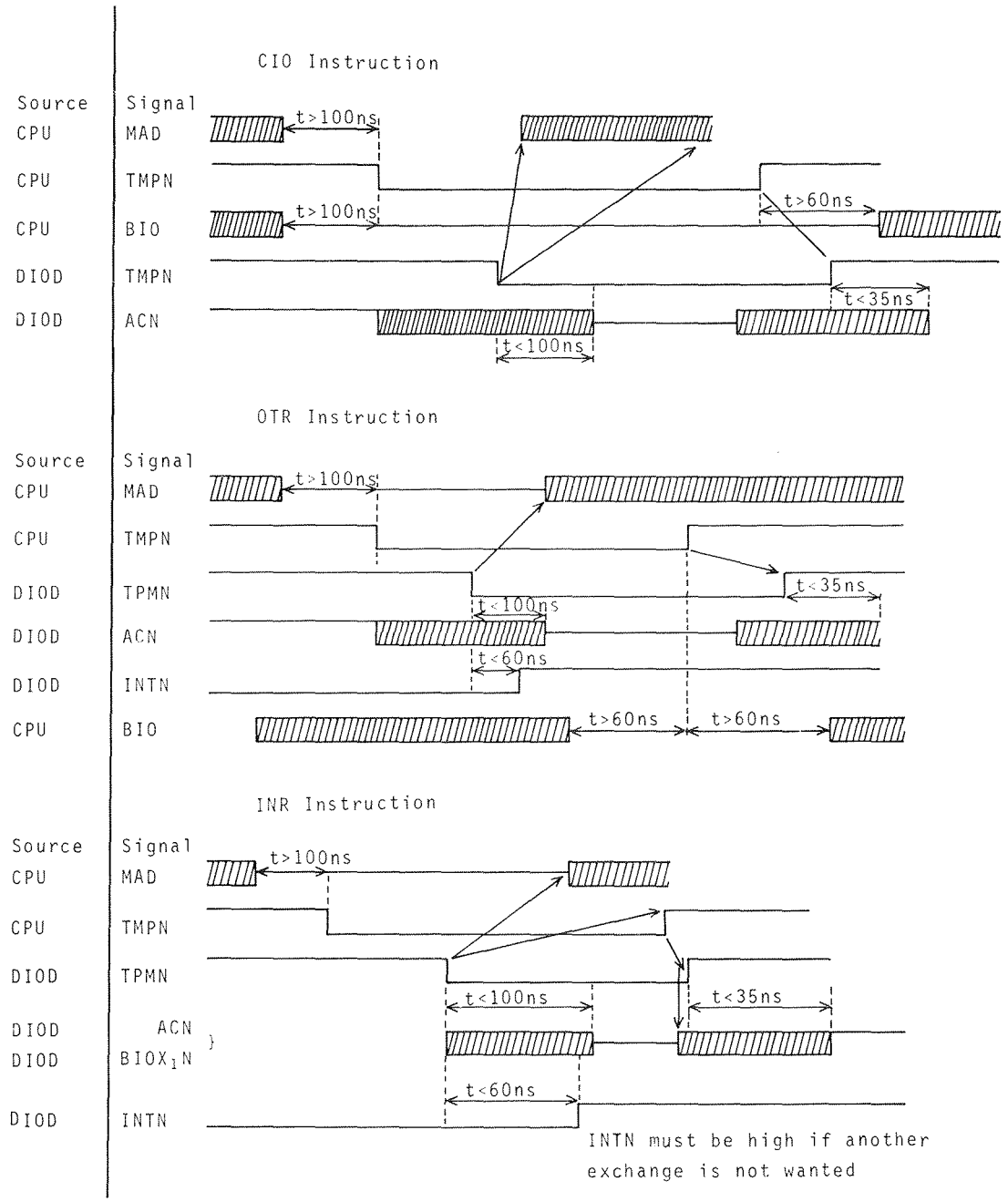


Figure 7.3. GP Bus-Timing Diagrams

interrupt from the DIOD board is not reset by this instruction.

The call information is then examined by the computer to determine whether an output or an input transfer is required and to program the data handling within the CPU. A second instruction is then sent to the DIOD board- INR Data, for an input, or OTR Data, for an output. This instruction is decoded in the Address Decode and I/O functions logic to either gate a 16-bit input word from the device onto the BIO lines or to strobe a 16-bit word on the BIO lines into an output buffer connected to the device input. The instruction also cancels the interrupt request from the DIOD board.

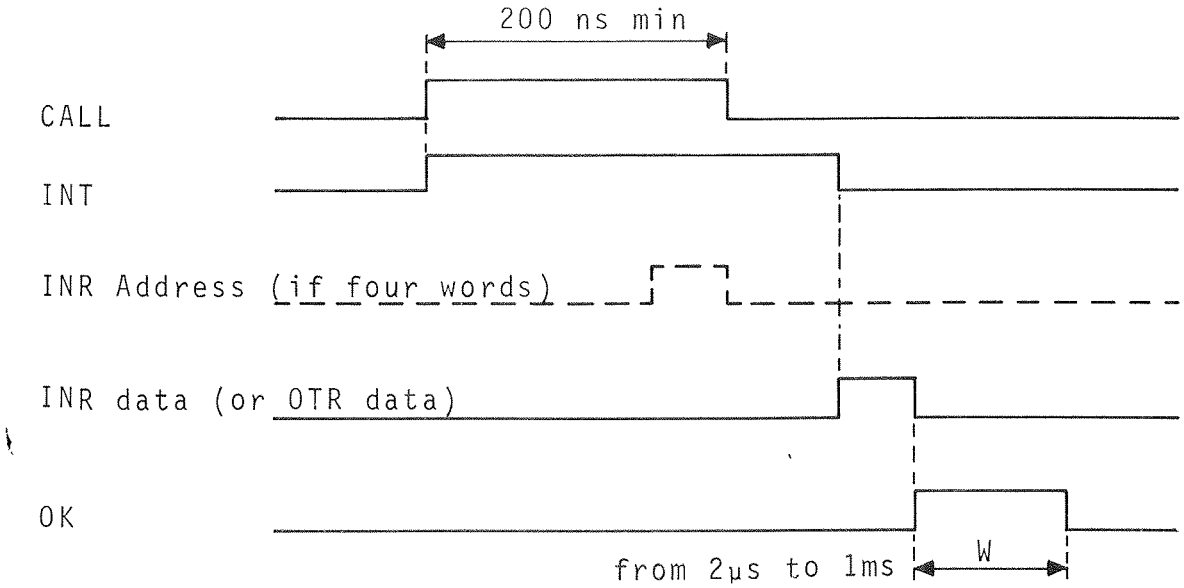
At the end of the data transfer an OK signal is sent to the device according to the selected channel.

Initiation by Software

Data may also be transferred under software control without a call signal from the device and a consequent interrupt from the DIOD board. In this mode of operation INR Data instructions are used to sample the input lines at intervals controlled by the software program and an OK signal is sent to the device each time that the lines are sampled. Output data is loaded into the DIOD output buffers by OTR Data instructions and an OK signal again sent to the device.

Timing for Control Lines

The timings for the control lines in calling mode and sampling mode are given in Figure 7.4

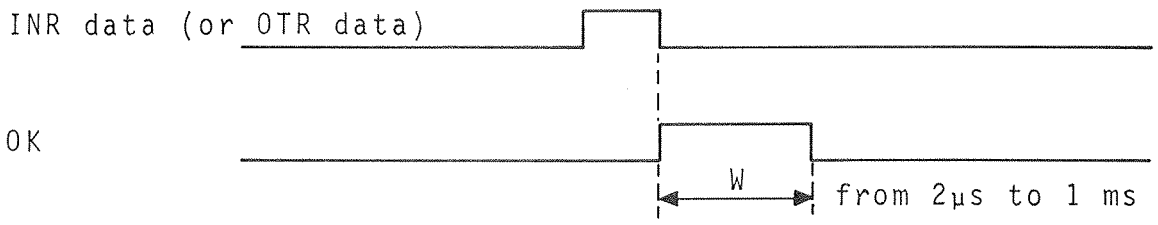


Output: When external equipment requests an exchange (via CALO lines-first positive transition)

1. INT sent to CPU
2. On next OTR data from CPU loaded into DIOD output buffer.
3. When next OKO signal occurs data(DOUT)are present and valid for external equipment.

Input: When external equipment requests an exchange (via CALI lines-first positive transition):

1. INT sent to CPU
2. On next INR data on 16 DIN lines from device are transferred to CPU.
3. An OKI signal is generated,at end of INR command.



On sampling mode(software program)data sent or received under software control. Each time that a command is performed data are exchanged and an OK signal then generated on the trailing edge of the command.

Figure 7.4. Timings for Control Lines

Sequences

Start Mode or Stop Mode.

In the Start Mode the DIOD Controller accepts all data exchanges (INR Address, INR Data, OTR Data)for one(or two)words.

The Controller can accept all CALL signals(in calling Mode) or direct exchanges required by software(in sampling Mode). The Start Mode is entered by the Controller after a CIO Start command.

In Stop Mode the DIOD Controller refuses all exchange requests. This Mode occurs after CIO Stop or after CLEAR.

SECTION 1 — INTERFACING

PART 2 INTERFACE CIRCUITS AND HARDWARE

The P852M, P856M, and P857M are of modular construction and are designed to be accommodated in standard 19 inch racks. In the basic system the functional modules are constructed on pluggable printed circuit boards, referred to hereafter as sub-assemblies, each provided with mechanical strengthening and with extractor handles. The sub-assemblies are inserted horizontally into slots in a basic mounting box, containing its own d.c. power supply and blower, which is slide mountable in a rack. Four types of basic mounting box are available:

Type M1 — Contains slots for 4 sub-assemblies
Type M2 — Contains slots for 6 sub-assemblies (P852M only)
Type M4/M4M - Contains slots for 10 sub-assemblies
Type M5 - Contains slots for 17 sub-assemblies

To further extend the facilities of the systems, equipment shelves are also available which contain additional sub-assembly slots. Each shelf, which is slide mountable in a 19 inch rack, contains its own d.c. power supply and blower.
The equipment shelves are as follows:

Type E1 — Contains slots for 8 sub-assemblies (Type 2 control units)
Type E2 — Contains slots for 6 sub-assemblies (Type 1 control units)

An example of the possible types of interface connection is given in Figure 1.1.

BASIC MOUNTING BOXES M1, M2, M4, M5 EQUIPMENT SHELVES E1 AND E2

The basic mounting boxes are so constructed that the sub-assemblies are inserted into their slots from the right-hand side of the box to plug into sockets mounted on a printed wiring board (motherboard) which forms the left-hand side of the box. Full construction details of the mounting boxes and equipment shelves are given in Section 2 of this manual.

PRINTED WIRING BOARDS

The P852M, P856M, and P857M printed wiring boards which are connected directly to the GP bus are 420 mm wide, 372 mm long (plus connectors) and 1.66 mm thick and are provided with connector plugs having gold plated pins. Four types of board are available:

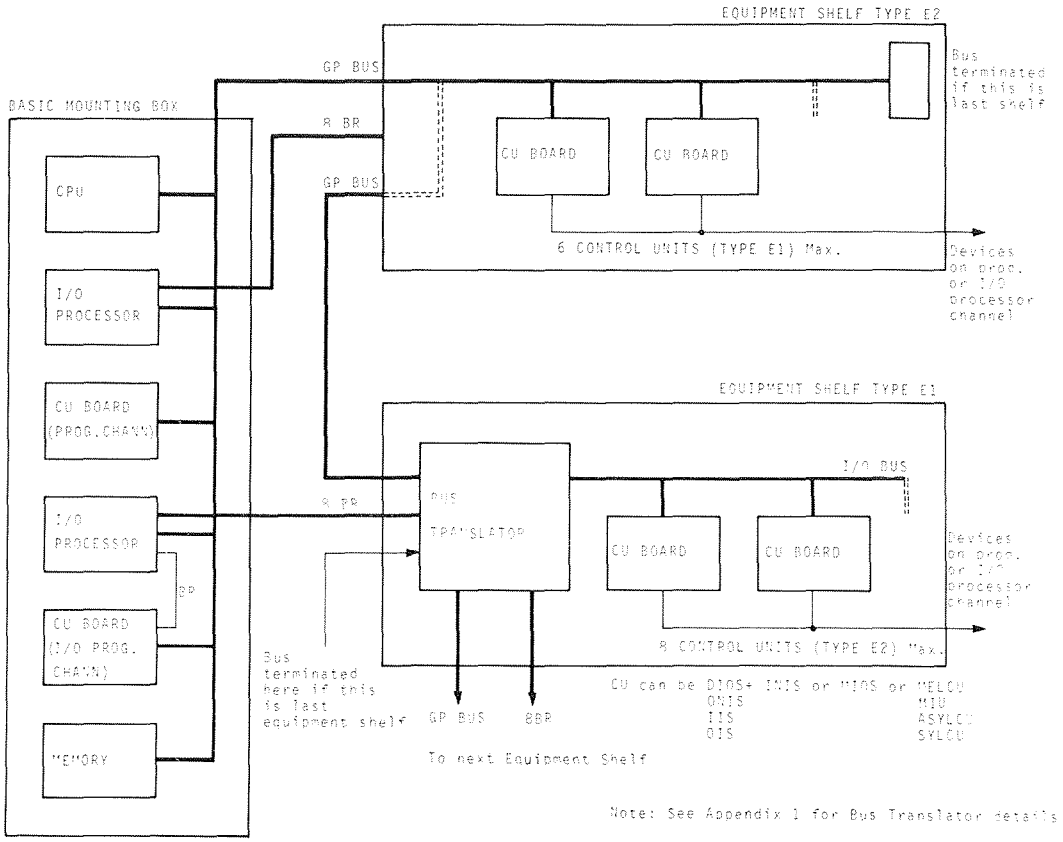


Figure 1.1. Interface Connections

Memory Board
CPU Board (also used for Double Control Unit)
Quadruple Control Unit Board
I/O Processor Board (or MMU board)

Connector Details
The boards are indentified by the connector layout as shown in Figure 1.2.

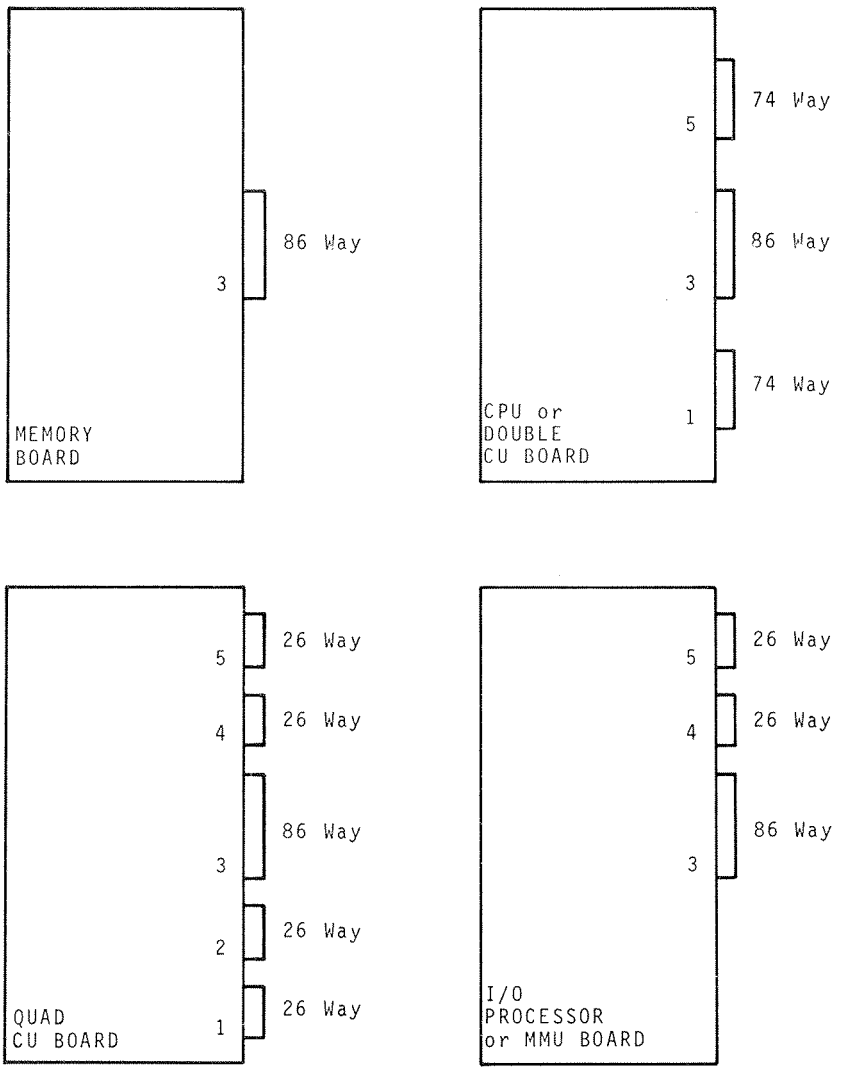
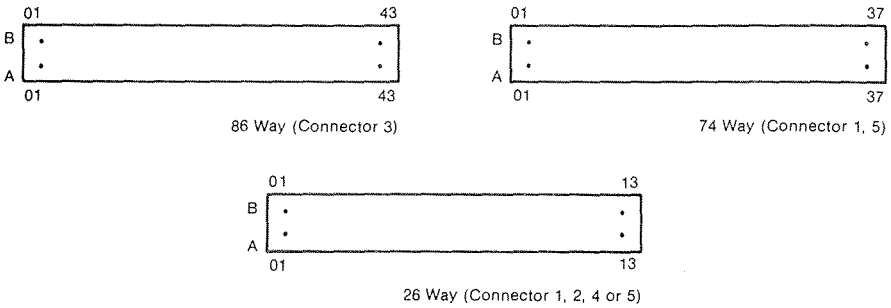


Figure 1.2. Printed Wiring Board-Connector Layout

The connectors on each board are numbered as shown in the diagram; connector 3 is always the bus and power supply connector and connectors 2 and 4 always 26-way connectors.

There are two rows of pins on each connector, the top row is identified as B (component side of board) and the bottom row as A (wiring side of board). The pins in each row are numbered from left to right when viewing the connector end of the board as shown below:



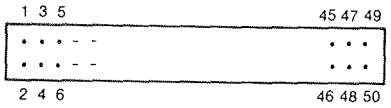
MOUNTING BOX CONFIGURATIONS

The physical layout for each mounting box sub-assembly slots is shown in Figure 1.3 to 1.6, when viewing the printed wiring board side of the box. The signal connections are given in Chapter 2 of this Part.

Note: When using a type M1, M2, M4 or M5 basic mounting box the bus cables, peripheral cables, and break request cables (if used) all leave the cabinet at the left rear area.

Connectors IOM and IOB

These connectors are each 50-pin connectors which are used to extend the GP bus outside the basic mounting box. The signal connections are given in chapter 2; the pin layout and numbering is as shown below.



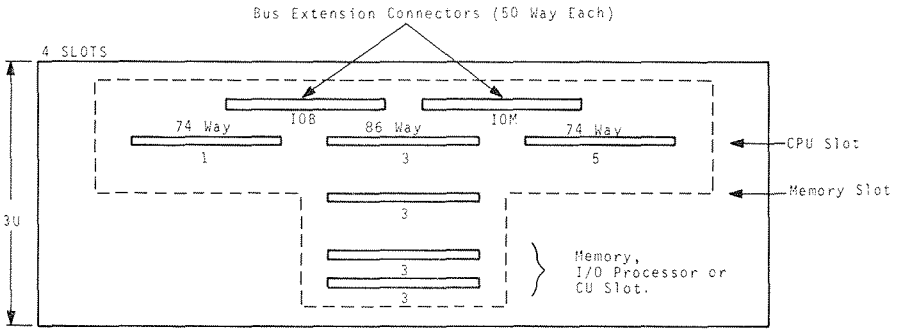


Figure 1.3. Layout of Type M1 Mounting Box Slots (P852M/P856M-100 Series)

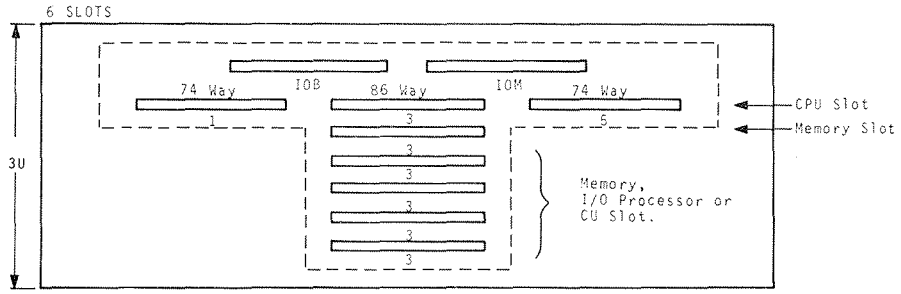


Figure 1.4. Layout of Type M2 Mounting Box Slots (P852M-200 Series)

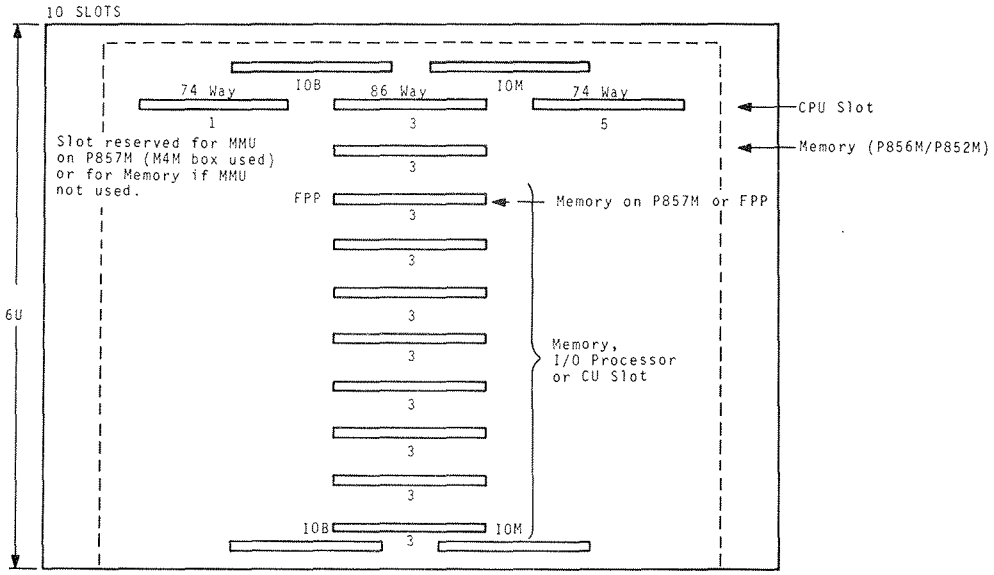


Figure 1.5. Layout of Type M4 (and M4M) Mounting Box Slots (P852M/P856M/P857M-400 Series)

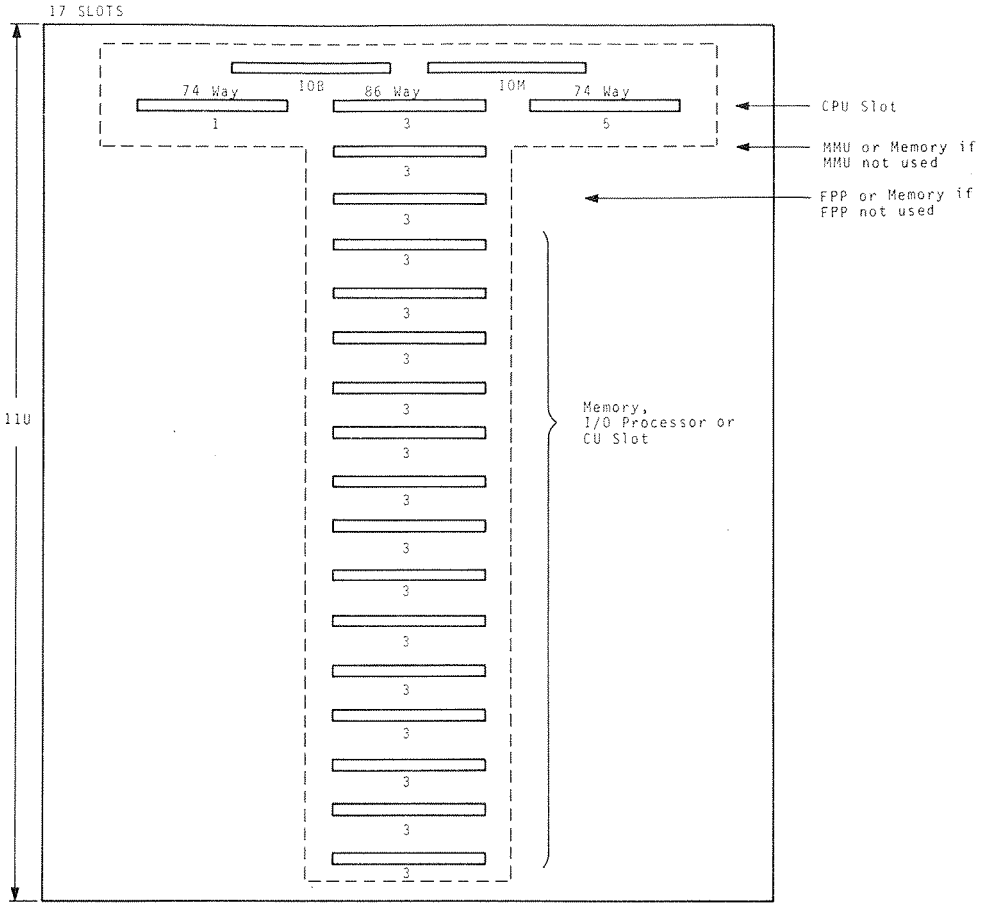


Figure 1.6. Layout of Type M5 Mounting Box Slots (P857M-500 Series)

Note: The type M5 mounting box contains two seperate power supplies- one supplying power to the first 8 slots (from the top of the box) and the other the power of the 9 remaining slots. Certain constraints apply, therefore, to the board locatins as follows:

- The CPU, MMU and FFP (floating point processor- if used) should be located in the first 3 slots.
- The memory boards should be distributed in two groups to balance the power consumption from each power supply.

  Example: 80k words.
  Place 2x16k modules in slots 4 and 5
  Place 3x16k modules in slots 9, 10 and 11

- All the other control units or I/O processors can be placed in any remaining slots.

METHOD OF CONNECTION

For sub-assemblies mounted in the basic mounting box connection to the GP bus is made directly by plugging into the common connector 3 mounted on the printed wiring board backpanel. The GP bus is taken outside the mounting box on a plug-in transmission line which extends the bus to further equipment shelves as required.

Connection to control units mounted in the basic mounting box is made via mating female connectors on the connection leads from the peripheral devices which terminate on the appropriate control unit board connector (connector 1, 2, 4, or 5). The female connector is then secured to the basic mounting box chassis; this allows the board to be extracted and re-inserted as required.

With an I/O processor board in the basic mounting box external break request connections enter the box on an 8-way cable which is terminated, using a female connector, on connector 5 of the I/O processor board. The A pins of connector 5 are connected, via printed wiring on the I/O processor board, to corresponding A pins on connector 4 and the required break request circuit is then completed to the I/O processor logic by wire wrapping connections between corresponding pins on row A and row B on connector 4. Break request lines from control units mounted in the basic mounting box are wired directly to the appropriate pin on row B of connector 4. A simplified break request connection diagram is shown in Figure 1.7 and further signal connection details are given in Chapter 2 of this Part.

More information on break request connections when using external equipment shelves can be found in Section II Part 5 Chapter 2 (Installing) of this manual. Details of the connections when using Bus Translator boards can be found in Appendix 1.

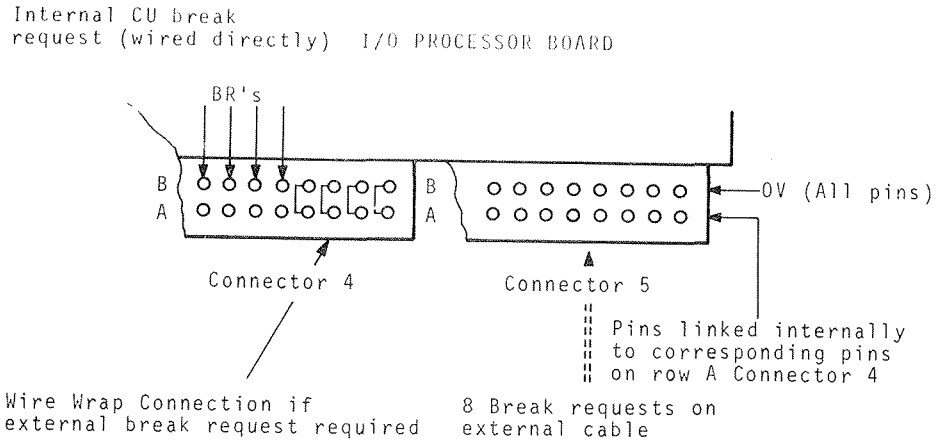


Figure 1.7 Break requests on I/O Processor Board

2 GP bus - Programmed and I/O Processor Channel

The concept of the P852M, P856M, or P857M Interface is based on the high speed transmission of data between units such as CPU, memory, and control units, which have different interface characteristics. The transmission takes place using the GP bus and the units are each connected to the bus using specific interface circuits. To ensure proper functioning of the complete system certain basic rules given in this chapter should be obeyed. The information is useful for a user who wishes either to extend the system, to improve the transmission conditions of an existing system, or to make a specific interface for a particular application.

The GP bus connections inside the basic mounting box are carried on connector 3 whose connections are commoned between each slot via the printed wiring board backpanel. This method allows any unit to be plugged into any slot to obtain access to the GP bus.

TRANSMISSION MEDIUM

The transmission medium used to extend the GP bus outside the basic mounting box consists of transmission lines made of two flat cables each of 50 conductors. Each active signal in the cable is surrounded by either 1 or 2 earthed wires according to the degree of protection required for the signal. The transmission line characteristics are as follows:

Line impedance	125 ohm ± 10%
Intrinsic propagation time	4.6 ns ± 10%
Linear resistance	⩽ 0.33 ohm per metre.

Termination Resistors

The transmission lines are required to be terminated correctly and for this purpose a resistor bridge adaption network is provided which either consists of a unit using discrete components or is in the form of integrated dual in line modules which terminate 14 signal lines each. The two possible adaptors are shown in Figure 2.1.

The transmission lines are always terminated at the basic mounting box end using the adaptor resistances, but only at the other end of the line (in an equipment shelf) if the end of the line is at a distance greater than 1 metre from the basic mounting box.

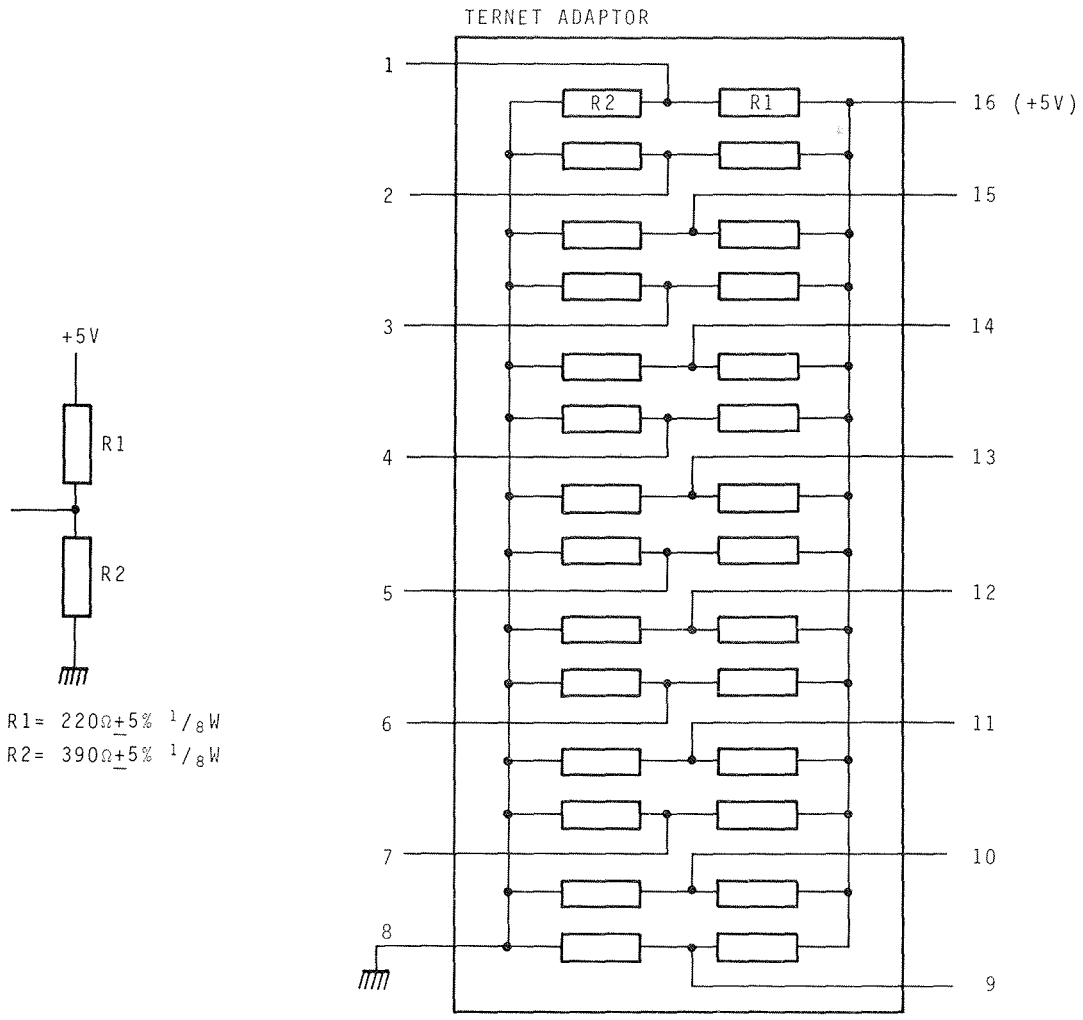


Figure 2.1. Terminator Bridge Adaption Networks

CONNECTION DETAILS

Backpanel Addresses

The standard GP bus connections, which apply to all units connected to connector 3 in any slot in the basic mounting box, are listed in Table 2.1.

Note: The earth lines MA, MB, MC shown in the table are associated with groups of signals as follows:

Earth MA	– associated with 18 lines MAD00-15, MAD64, MAD128.
Earth MB	– associated with 16 lines BIO 00N-15N.
Earth MC	– associated with all other signals comprising the control group.

Table 2.1 Backpanel Addresses Connector 3

Signal	Pin No.		Signal	Pin No.
+18V	3A01		−18V	3B01
BIEC0	3A02		EARTH	3B02
BIEC2	3A03		BIEC1	3B03
BIEC4	3A04		BIEC3	3B04
SCEIN	3A05		BIEC5	3B05
+16VM	3A06		+16VM	3B06
0VM	3A07		0VM	3B07
BIO 00N	3A08		BIO 01N	3B08
BIO 02N	3A09		BIO 03N	3B09
BIO 04N	3A10		BIO 05N	3B10
BIO 06N	3A11		BIO 07N	3B11
BIO 08N	3A12		BIO 09N	3B12
BIO 10N	3A13		BIO 11N	3B13
BIO 12N	3A14		BIO 13N	3B14
BIO 14N	3A15		BIO 15N	3B15
OKO	3A16		OKI	3B16
PWFN	3A17		RSLN	3B17
EARTH MB	3A18		−5VM	3B18
+5VL	3A19		+5VL	3B19
+5VL	3A20		+5VL	3B20
0VL	3A21		0VL	3B21
0VL	3A22		0VL	3B22
BR(CU.4)	3A23		+5VM	3B23
EARTH MA	3A24			3B24
EARTH MC	3A25		+16VM	3B25
WRITE	3A26		MAD15	3B26
CHA	3A27		MAD14	3B27
TRMN	3A28		MAD13	3B28
TMRN	3A29		MAD12	3B29
TMEN	3A30		MAD11	3B30
TMPN	3A31		MAD10	3B31
TPMN	3A32		MAD09	3B32
0V	3A33		MAD08	3B33
ACN	3A34		MAD07	3B34
SPYC	3A35		MAD06	3B35
BUSRN	3A36		MAD05	3B36
MSN	3A37		MAD04	3B37
BSYN	3A38		MAD03	3B38
CLEARN	3A39		MAD02	3B39
0V	3A40		MAD01	3B40
BR(CU.2)	3A41		MAD00	3B41
BR(CU.3)	3A42		MAD64	3B42
BR(CU.1)	3A43		MAD128	3B43

M — Memory Voltage L — Logic Voltage

GP Bus Extension Connections

The GP bus signals are extended outside the basic mounting box on connectors IOM and IOB. The pin connections for these connectors are listed in Tables 2.2 and 2.3. When the bus is extended to a Bus Translator board (used in a Type E1 equipment shelf) it is terminated on connector 6 and 5 on the board whose signal pin connections are identical to those on connectors IOM and IOB respectively (See also Section II Part 5 Chapter 2).

Table 2.2 Connector IOM Connections

Signal	Pin No.		Signal	Pin No.
Earth MA	1		Earth MC	26
MAD04	2		CLEARN	27
Earth MA	3		Earth MC	28
MAD03	4		Earth MC	29
Earth MA	5		TPMN	30
MAD08	6		Earth MC	31
Earth MA	7		Earth MC	32
MAD09	8		TMPN	33
Earth MA	9		Earth MC	34
MAD10	10		Earth MC	35
Earth MA	11		TMEN	36
MAD11	12		Earth MC	37
Earth MA	13		Earth MC	38
MAD12	14		TRMN	39
Earth MA	15		Earth MC	40
MAD13	16		Earth MC	41
Earth MA	17		Spare	42
MAD14	18		Earth MC	43
Earth MA	19		Spare	44
MAD15	20		Earth MC	45
Earth MA	21		Spare	46
ACN	22		Earth MC	47
Earth MC	23		Spare	48
	24		Earth MD	49
Earth MC	25		5V	50

Note: Each active signal is bordered by an associated earth line (MA, MB, or MC) which is connected to the appropriate earth listed in Table 2.1. Earth MD is associated with +5V for remote control purposes.

Table 2.3 Connector IOB Connections

Signal	Pin No.		Signal	Pin No.
Earth MC	1		BIO05N	26
RSLN	2		Earth MB	27
Earth MC	3		BIO04N	28
PWFN	4		Earth MB	29
Earth MB	5		BIO03N	30
BIO15N	6		Earth MB	31
Earth MB	7		BIO02N	32
BIO14N	8		Earth MB	33
Earth MB	9		BIO01N	34
BIO13N	10		Earth MB	35
Earth MB	11		BIO00N	36
BIO12N	12		Earth MB	37
Earth MB	13		BIEC5	38
BIO11N	14		Earth MC	39
Earth MB	15		SCEIN	40
BIO10N	16		Earth MC	41
Earth MB	17		BIEC3	42
BIO09N	18		Earth MC	43
Earth MB	19		BIEC4	44
BIO08N	20		Earth MC	45
Earth MB	21		BIEC1	46
BIO07N	22		Earth MC	47
Earth MB	23		BIEC2	48
BIO06N	24		Earth MC	49
Earth MB	25		BIEC0	50

Bus Translator Board Connections (See Section II Part 5 Chapter 2 and Appendix I for details of Bus Translator)

Table 2.4 I/O Bus Connections to Connector 1

Pin No.	Signal		Pin No.	Signal
1A01	+5V		1B01	0V
1A02			1B02	BIN00N
1A03	BIN00N		1B03	BIN01N
1A04	BIN01N		1B04	BIN02N
1A05	BIN02N		1B05	BIN03N
1A06	BIN03N		1B06	BIN04N
1A07	BIN04N		1B07	BIN05N
1A08	BIN05N		1B08	BIN06N
1A09	BIN06N		1B09	BIN07N
1A10	BIN07N		1B10	
1A11	BR07N		1B11	ACCN
1A12	BR06N		1B12	ACCN
1A13	ACCN		1B13	AREN
1A14	AREN		1B14	AREN
1A15			1B15	SP4
1A16			1B16	SP3
1A17	BR05N		1B17	SP2
1A18	BR04N		1B18	SP1
1A19	BR03N		1B19	
1A20	BR02N		1B20	
1A21	BIN08N		1B21	BIN08N
1A22	BIN14N		1B22	BIN14N
1A23	BR01N		1B23	
1A24	BR00N		1B24	S
1A25			1B25	
1A26	BIN13N		1B26	BIN13N
1A27	BIN15N		1B27	BIN15N
1A28	BIN09N		1B28	BIN09N
1A29	BIN10N		1B29	BIN10N
1A30	BIN11N		1B30	BIN11N
1A31	BIN12N		1B31	BIN12N

Table 2.5 I/O Bus Connections to Connector 2

Pin No.	Signal		Pin No.	Signal
2A01	BOF02N		2B01	IR04N
2A02	EOR		2B02	IR02N
2A03	BOF01N		2B03	IR06N
2A04	BOF00N		2B04	IR07N
2A05	BOU14		2B05	IR00N
2A06	BOU13		2B06	IR01N
2A07	BOU12		2B07	IR03N
2A08	BOU09		2B08	IR05N
2A09	BOU10		2B09	
2A10	BOU11		2B10	
2A11	BOU15		2B11	
2A12	MCN		2B12	
2A13	BAD03N		2B13	
2A14	DAVN		2B14	
2A15	BAD04N		2B15	
2A16	BAD05N		2B16	
2A17	BOU08		2B17	
2A18	BOU07		2B18	
2A19	BOU06		2B19	
2A20	BOU05		2B20	
2A21	BOU04		2B21	
2A22	BAD00N		2B22	
2A23	BOU03		2B23	
2A24	BAD01N		2B24	
2A25	BAD02N		2B25	
2A26	BOU02		2B26	
2A27	BOU01		2B27	
2A28	BOU00		2B28	
2A29	+5		2B29	PWFE
2A30			2B30	PWFE
2A31	+5V		2B31	0V

Table 2.6 Break Request Connections to Connector 7

Pin No.	Signal
1	BR00N(In)
2	BR00N(Out)
3	M
4	M
5	BR01N(In)
6	BR01N(Out)
7	M
8	M
9	BR02N(In)
10	BR02N(Out)
11	M
12	M
13	BR03N(In)
14	BR03N(Out)
15	M
16	M
17	BR04N(In)
18	BR04N(Out)
19	M
20	M
21	BR05N(In)
22	BR05N(Out)
23	M
24	M
25	BR06N(In)
26	BR06N(Out)
27	M
28	M
29	BR07N(In)
30	BR07N(Out)
31	M
32	M

Break Request Connections on I/O Processor Board

Table 2.7 Break Request Connections to Connector 4 and 5

Signal	Pin No.		Signal	Pin No.
	5A01			5B01
	5A02			5B02
	5A03			5B03
	5A04			5B04
	5A05			5B05
BREX07N	5A06		0V	5B06
BREX06N	5A07		0V	5B07
BREX05N	5A08		0V	5B08
BREX04N	5A09		0V	5B09
BREX03N	5A10		0V	5B10
BREX02N	5A11		0V	5B11
BREX01N	5A12		0V	5B12
BREX00N	5A13		0V	5B13

Signal	Pin No.		Signal	Pin No.
BREX07N	4A06		BR07N	4B06
BREX06N	4A07		BR06N	4B07
BREX05N	4A08		BR05N	4B08
BREX04N	4A09		BR04N	4B09
BREX03N	4A10		BR03N	4B10
BREX02N	4A11		BR02N	4B11
BREX01N	4A12		BR01N	4B12
BREX00N	4A13		BR00N	4B13

GP BUS INTERFACE CIRCUITS

All data and control signals (apart from signal RSLN) are sent or received on the bus lines using either an 1801 Emitter or a Receiver 0612 or REC 0613. Signal RSLN is generated differently to provide protection for the peripherals in case of a system break down. Each board has one emitter and/or one receiver per bus line. The characteristics of emitter 1801 and receiver 0612 or REC 0613 are given below.

Receiver 0612

Input high threshold	Vih	⩾2.1 V min
Input low threshold	Vil	⩽1.4 V max
Input current	Iih	+50 μA max at 2.4V
Input current	Iil	−100 μA max at 0.4V
Output high voltage	Voh	2.4V
Output high current	Ioh	−1 mA for REC 0612, - 4.0 mA for REC 061
Output low voltage	Vol	0.5V
Output low current	Iol	−16 mA
Propagation delay to high state	Tpdh	4 ns min to 12 ns max
Propagation delay to low state	Tpdl	8 ns typical

Note: propagation delays measured at 400 ohm 15 pF

Emitter 1801

Input high voltage	Vih	2V
Input low voltage	Vil	0.8V
Input high current	Iih	40 μA
Input low current	Iil	−1.6 mA
Output low voltage	Vol	0.4V at 55 mA
Output high leakage current	Ioh	250 μA max at 5.5V
Propagation delay to high state	Tpdh	8 ns min, 22 ns max, 14 ns typical
Propagation delay to low state	Tpdl	5 ns min, 18 ns max, 11 ns typical

Note: propagation delays measured at 133 ohm 45 pF.

The pin layout details for the emitters and receivers are shown in Figure 2.2.

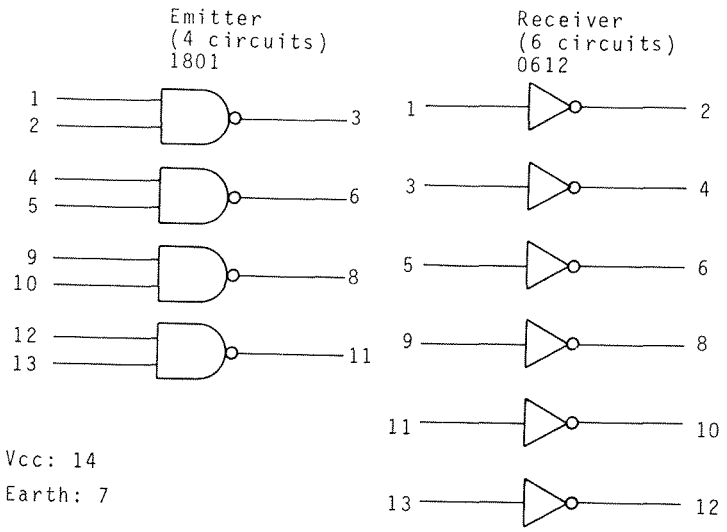


Figure 2.2 Emitter 1801 and receiver 0612/0613 pin layout

Note: Full wiring board rules for the emitter and receiver I.C's (and for all other I.C's) can be found in Chapter 4 of this Part.

EQUIPMENT SHELF DIVISION RULES

A P852M, P856M, or P857M system is composed of:

1. A basic mounting box, including power supply, (contained in a basic cabinet) which may hold:

- A central processing unit
- memory modules
- one (or more) I/O processors
- control units
- control panel.

2. One, or more, equipment shelves placed independantly of the basic cabinet and containing one or more control units whose interface may be made directly to the GP bus (or to the GP bus via a bus translator board).

It is possible to connect a total of 64 control units in the system.

Division Rules

The maximum permissable length of the GP bus is 14.50 metres and the following rules apply to a system in which the basic mounting box is situated at one end of the line, with the GP bus lines terminated using an adaption bridge, and an equipment shelf is situated at the other end of the line with the GP bus lines also terminated with an adaption bridge. Two possibilities may occur:

1. The equipment shelf may contain Type 2 control units. In this case each equipment shelf will be fitted with a Bus Translator board and one emitter and one receiver will be connected to each GP bus line via this board.

   A maximum number of 7 equipment shelves may be connected to the GP bus, and it is recommended, though not obligatory, that the shelves are connected at regular distances along the GP bus lines.

   For shelves connected at regular intervals the distance between consecutive shelves should be $\geq$ 2 metre and in this case the propagation time of the signals will be between 5 to 8 ns per metre.
   The distance between shelves should never be less than 1.5 metre except in the case where the basic mounting box and the shelf are placed in the same cabinet. In this case the GP bus will only require to be terminated at the basic mounting box end.

2. The equipment shelves contain Type 1 control units. In this case each shelf contains 6 Type 1 control units and a maximum of 6 emitters and 6 receivers per line are connected to the GP bus. Each shelf receives the GP bus

signals and the signals are terminated in the last shelf connected to the bus.
The division rules are the same as those stated for the Type 2 control unit shelves. Where a division of ≥ 2 metres between consecutive shelves is used the propagation time is between 6 to 10 ns per metre.

BREAK REQUEST CONNECTIONS

An I/O processor is always located in the basic mounting box and the user may connect 8 control units to each I/O processor; the control units may be located either in the basic mounting box or in an equipment shelf. The break request line of the CU is connected either via the back printed wiring panel of the basic mounting box or via an external cable. All break request lines are terminated at the I/O processor end and also at the equipment shelf end (in the equipment shelf which contains a control unit associated with a break line).

3 Control Units

Control units form the interface between the GP bus lines (or I/O bus lines) which carry all data and control information, and the various peripheral devices. The function of a control unit therefore is to translate I/O instructions into peripheral actions and to provide the necessary sequencing and control signals to effect data transfer.

General details are given in this chapter for assisting a user who wishes to design control units for peripherals other than those available in the standard P852M/P856M/P857M range of equipments. Also described, are two general purpose boards (for Type 1 Control Units), one partially built and one blank, for the construction of user designed control units. Full wiring rules for this purpose are given in Chapter 4 of this Part.

As a further guide to the user, in the Appendix is given the logic circuit for a specific control unit. For full information regarding all the standard control units reference should be made to the following manuals.

P800M MCU2 Service Manual - Documation M300 PER 1415	5122 991 1173X
P800M MCU3 Service Manual - Digitronics 2540 and V24	5122 991 1176X
P833-152 Cassette tape CU Service Manual - ELA	5122 991 1246X
P824-002 Disc CU Service Manual - PER 1215	5122 991 1249X
P825-040 Disc CU Service Manual - CDC 9760	5122 991
P800M DIOD Service Manual - Digital I/O	5122 991 1177X

The following explanation describes the operation of a control unit connected to the GP bus (Type 1 control unit). A control unit connected to the I/O bus (Type 2 control unit) functions in a similar manner; the names of the signals used can be found in Appendix 1 in which the facilities provided by the Bus Translator are given.

CPU COMMANDS

The following commands originate in I/O program instructions and are sent on the MAD lines (MAD 04, 08, 09) of the GP bus to the control unit addressed by MAD lines 10 to 15 of the GP bus.

CIO — To start or stop a data transfer
INR — To effect a transfer from a device to the CPU
OTR — To effect a transfer from the CPU to a device

TST — To test whether the status of a device or the CU is *busy or not busy*

SST — To send the status of a device after the transfer has been completed.

Address

The control unit address is taken from the 6-bit address field of the I/O instruction and placed on the MAD lines. Each control unit has decoding logic to recognise its own address and when this has been validated by the TMPN signal from the CPU the control unit then responds with timing signal TPMN which informs the CPU that the address has been recognised. The function decoding logic in the control unit is also primed when the address is recognised.

Address selection in the decoding logic is made by links on the control unit boards. Where more than one device or channel is controlled by a single control unit, the address logic is arranged to recognise all required addresses.

Function Codes

The commands on the MAD lines are decoded by the function code logic of the control unit as follows, provided that the control unit has recognised its address and the control unit sequencing logic is in the correct mode.

Instruction Bits	4	8	9
MAD Lines	04	08	09
CIO Start	0	1	1
CIO Stop	0	1	0
INR (Input Transfer)	1	0	0 or 1*
OTR (Output Transfer)	0	0	0 or 1*
TST (Test Status)	1	1	0
SST (Send Status)	1	1	1

* This bit can be used for specifying variations in the type of transfer.

CONTROL UNIT SEQUENCING MODES

The acceptance of a function code depends upon which mode the control unit sequencing logic is in when the command is received. As shown in Figure 3.1 the control units are sequenced through four operational states.

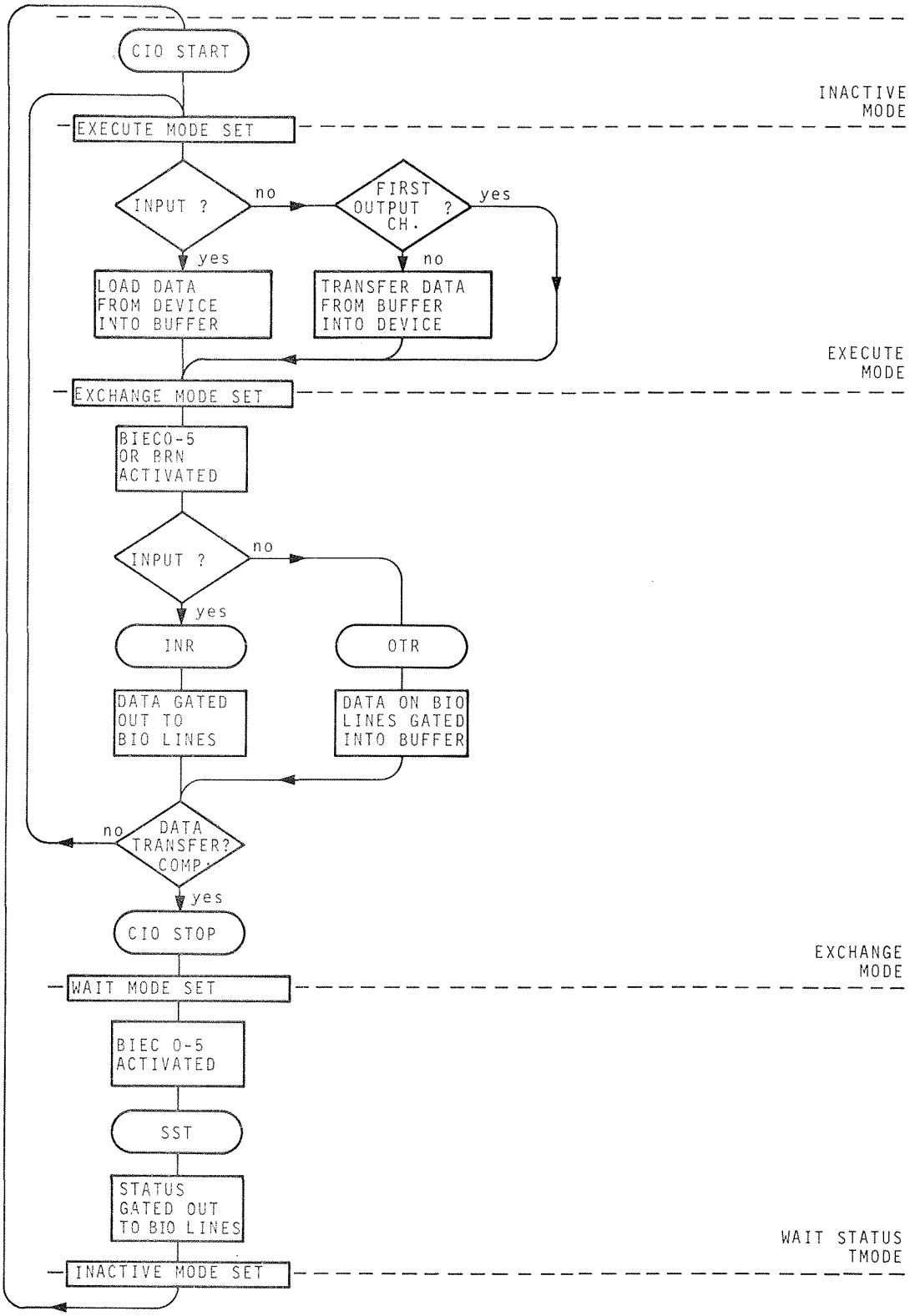


Figure 3.1. Control Unit Mode Sequence

Inactive
The only commands accepted by a control unit in this mode are CIO start and TST. The control unit responds to a TST command by placing one's on all BION lines to indicate a *not busy* condition. A CIO start command switches the control unit from the *Inactive* to the *Execute* or *Exchange* mode depending upon whether the control unit is designed for input or output.

Execute
In this mode the control unit causes the peripheral device to make a single operation — read one character, for example in the case of a tape or card reader — after which it switches to the *Exchange* mode. An Error switches the unit to the *Wait Status* mode and a TST command places a *busy* signal on the BIO lines without switching the mode. No other commands are accepted in the *Execute* mode.

Exchange
In this mode the control unit activates either the BRN break request line — if the control unit is connected for I/O processor or DMA — or the interrupt request lines BIEC 0-5 — if the control unit is connected for programmed channel operation — to indicate to the CPU that it is ready to send or receive data. The INR, OTR and TST commands are acceptable in this mode and will effect a data transfer. CIO . stop is also accepted but causes the control unit to go to the wait status state. After the transfer the control unit reverts to the *Execute* mode.

Wait Status
The control unit switches to this mode after receiving a CIO stop command whilst in the Exchange mode. Only TST and SST commands are then accepted. The TST command tests whether or not the control unit is busy and the SST command puts the control unit status word on the BIO lines. The SST command also switches the control unit into the Inactive mode in readiness for the next CIO start command.
Indication of device status in response to an SST command will vary with the kind of device connected to the control unit. The most common status bits are:

BIO line 15: Not operable/manual intervention required
BIO line 14: Throughput error
BIO line 13: Data fault
BIO line 12: Incorrect length.

TYPICAL CONTROL UNIT LOGIC CIRCUITS

Interfacing with the GP bus is standard and is fully described in Chapter 2, of this Part whereas, the device interconnections depend upon the complexity of the peripheral. Figure 3.2 is a block diagram of a typical control unit and examples of the decoding, sequencing and control logic used in standard control units are given below; these aspects of the I/O interface logic can be seen in the complete control unit logic contained in the Appendix.

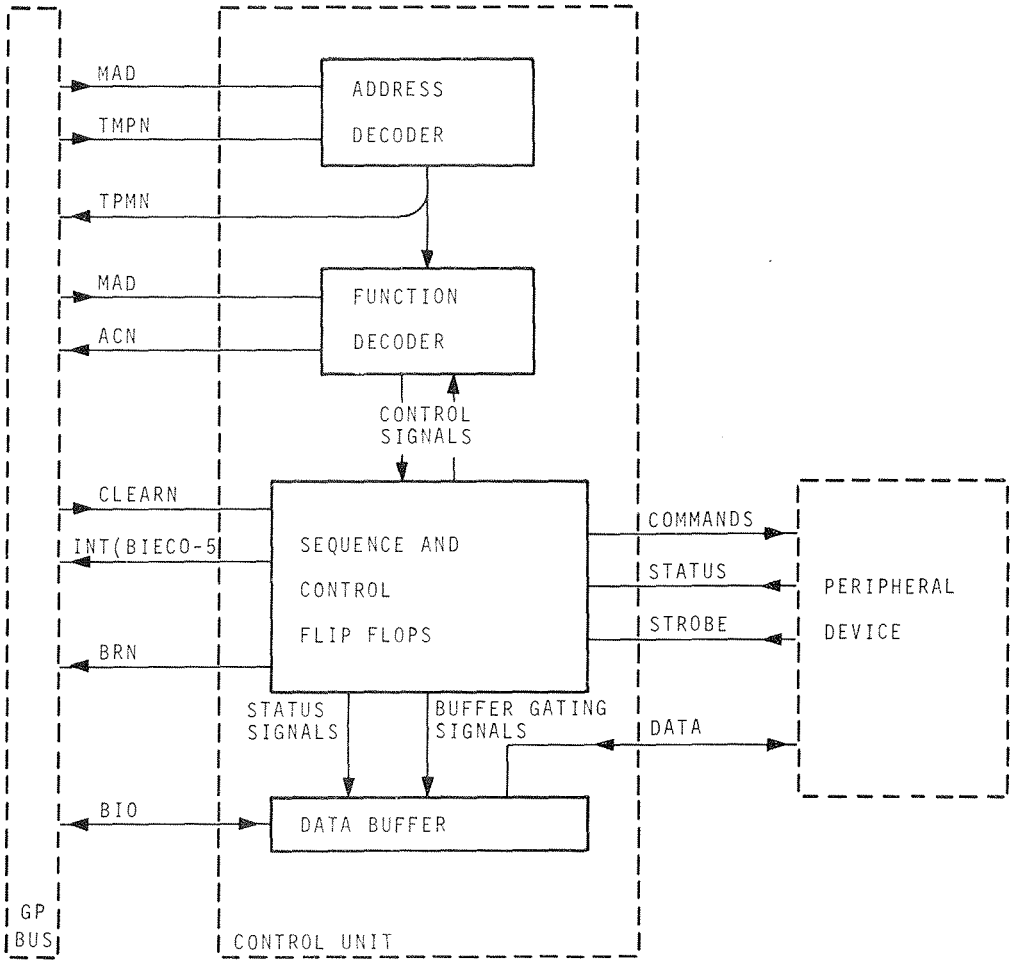


Figure 3.2. Block Diagram of Typical Control Unit

Address Decoding
Figure 3.3 shows a suitable decoding arrangement for the MAD address lines. The link connectors AD0 to AD5 allow the address of the control unit to be determined merely by positioning the links. For example device address 5 is set up in binary form by connecting the links as follows:

AD0	AD1	AD2	AD3	AD4	AD5
1 to 2	1 to 2	1 to 2	2 to 3	1 to 2	2 to 3

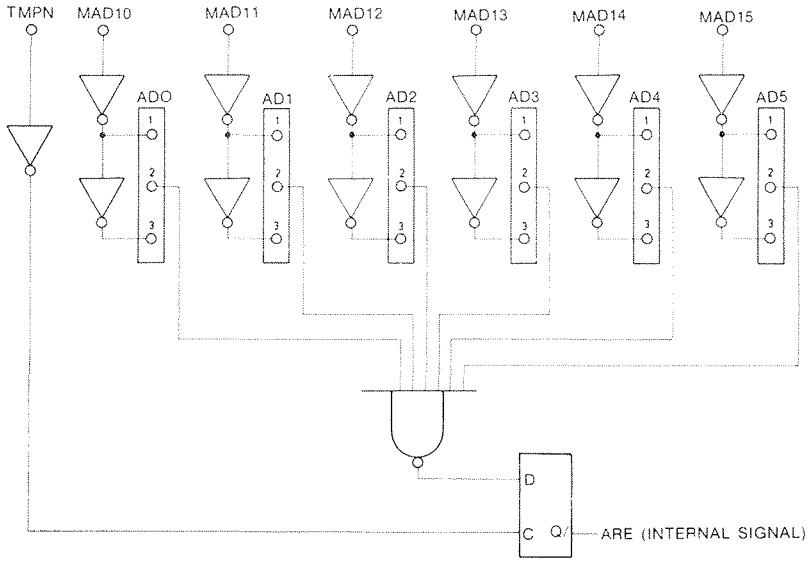


Figure 3.3 Address decode logic

Function Decoding

Once the address has been recognised the consequent ARE signal allows the function on the MAD lines to be decoded by the circuit shown in Figure 3.4.

This circuit decodes the function on the MAD lines and decides if the command can be accepted. CIO Stop and Test Status are always accepted but for all other commands the control unit must be in the appropriate state, as determined by the sequencing logic.

Sequencing
The sequencing logic controls data transfers by switching the control unit into one of four operational states to adapt the GP bus timing to the mechanical actions of the device.

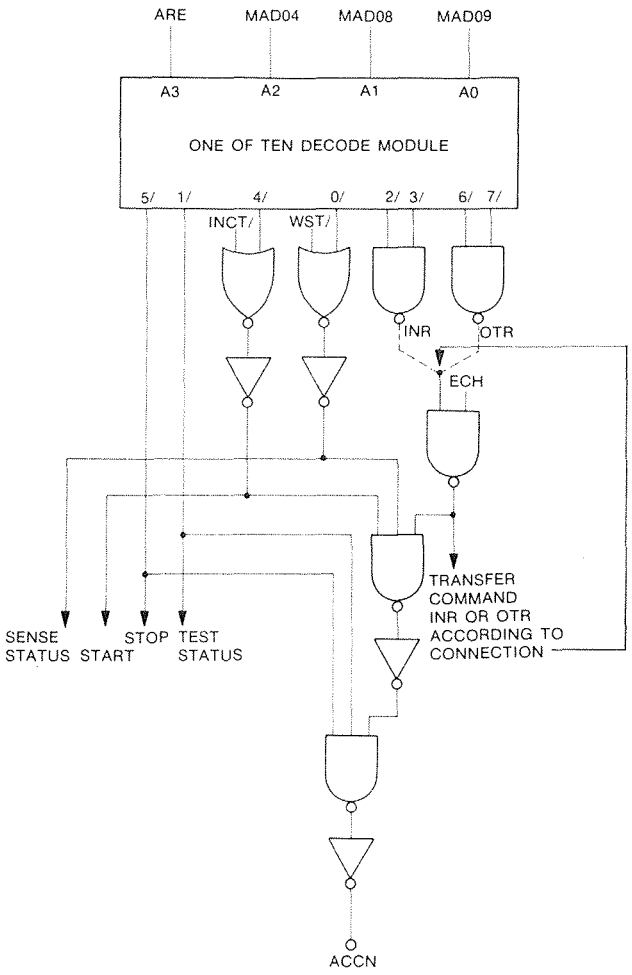


Figure 3.4 Function decode logic

In the circuit shown in Figure 3.5 the sequencing of the four states is as follows:

INCT This is the inactive state where the control unit is waiting for a CIO Start command to enable it to start a data exchange. It is switched into this state when both flip-flops A and B are reset by either a CLEARN or Send Status signal. Output from this gate allows the function decode logic to accept the CIO Start command.

EXT This is the execute state where a single mechanical action is performed by the peripheral device (i.e. a read or write operation). It is switched into this state when flip-flop A has been reset and flip-flop B has been set by the CU accepting either an INR (input) or OTR (output) command from the CPU.

ECH This state is the exchange state when the data transfer between the CPU and the control unit is actually executed. It is switched into this state when both A and B flip-flops have been set by either a CIO Start command (from the CPU) or a strobe pulse (from the device) depending on the type of exchange.

WST This is the wait status state which then enables the function decoding logic to accept a Send status command. It is switched into this state when flip-flop A is set and flip-flop B is reset by signal F HALT. F HALT will be produced by either a CIO Stop command (from the CPU) or when an error has been detected. The control until will remain in this state until it receives a send status command or CLEARN signal from the CPU. When either of these are received the CU will switch back to the *INCT* state to await the next CIO Start command.

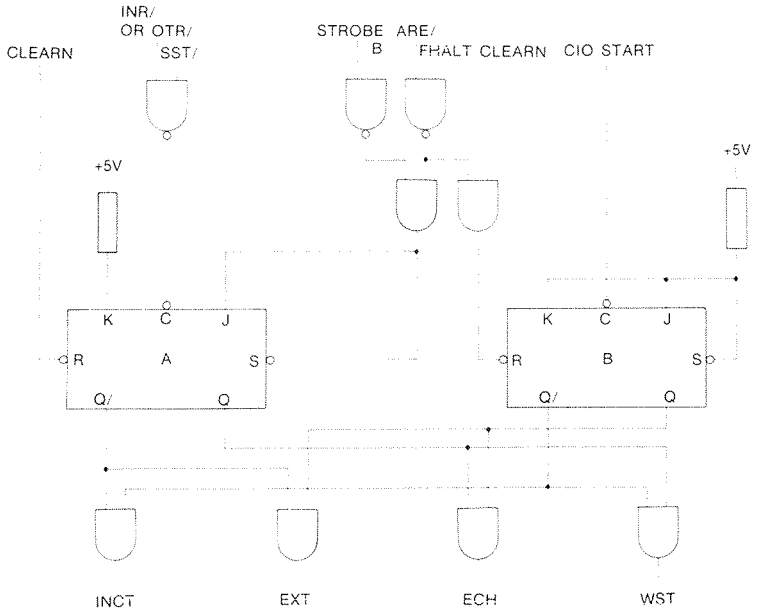


Figure 3.5 Sequencing logic

Control Flip-Flops

These flip-flops are very simple and are used to remember that certain events have occurred, such as commands from the CPU, device states or errors. The number of flip-flops needed depends upon the type of peripheral device; two examples are shown in Figure 3.6

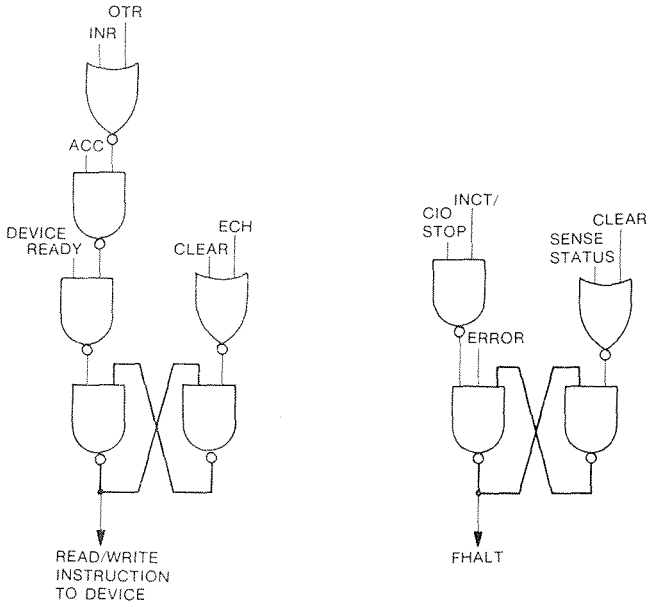


Figure 3.6 Control flip-flops

LOADING THE CU REGISTER

The diagram in Figure 3.7 shows, as an example only, the basic logic arrangement with the control unit in the exchange mode and carrying out an OTR instruction to load data sent from the CPU on the BIO lines into the CU register. The diagram illustrates that when TMPN is received, flip-flop ARE records that the control unit has recognized its address and that the function to be performed is memorized in flip-flop OUT. These actions are necessary as the contents of the MAD lines will change. The response signal TPMN from the CU occurs very rapidly and the data on the BIO lines is loaded into the register when signal ARE goes high. All response times have been made as rapid as possible to permit the use of very short guard times.

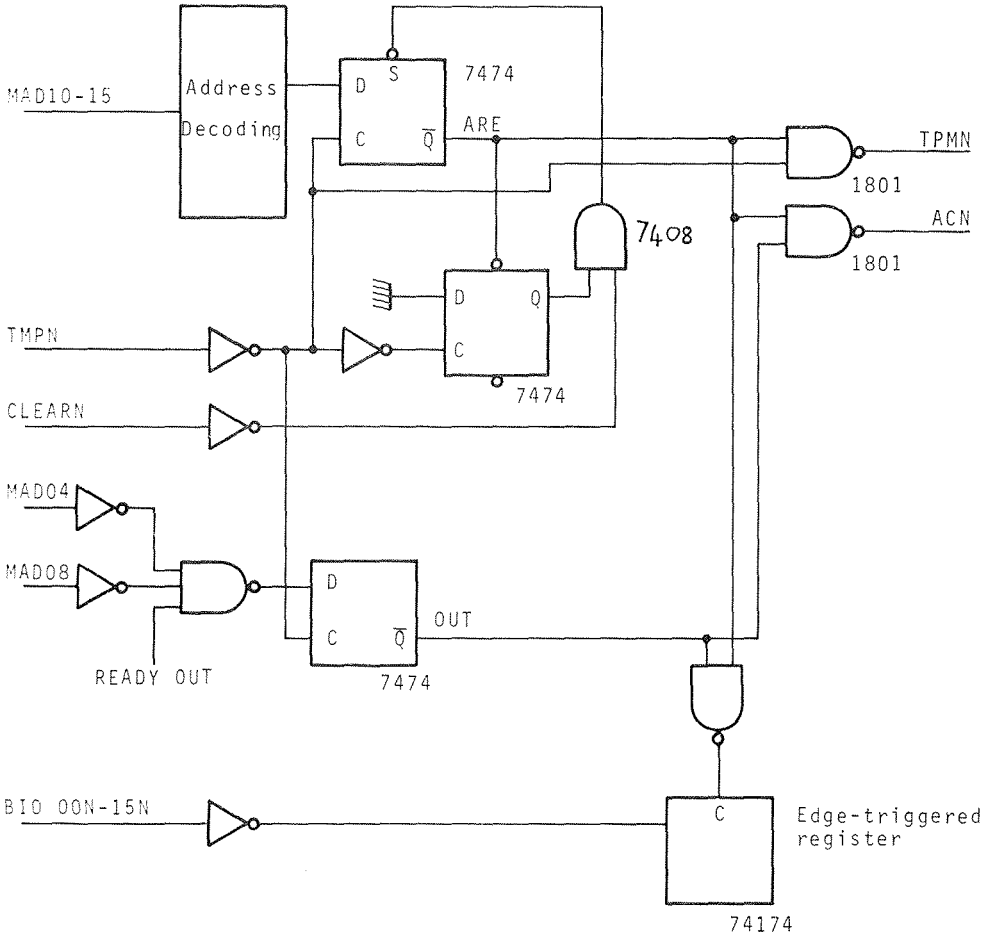


Figure 3.7 Loading the CU data register

READING FROM CU REGISTER

The diagram in Figure 3.8 shows, as an example only, the basic logic arrangement with the CU in the exchange mode and carrying out an INR instruction to send data on the BIO lines to the CPU. When the address is recognised (ARE = 1) instruction IN is used to write the contents of the register on to the BIO lines. When TMPN becomes 1 again prompt clearing of the TPMN and BIO lines is obtained by the use of a 74H11 which ensures that the BIO lines are freed within 35 ns of the resetting of TPMN.

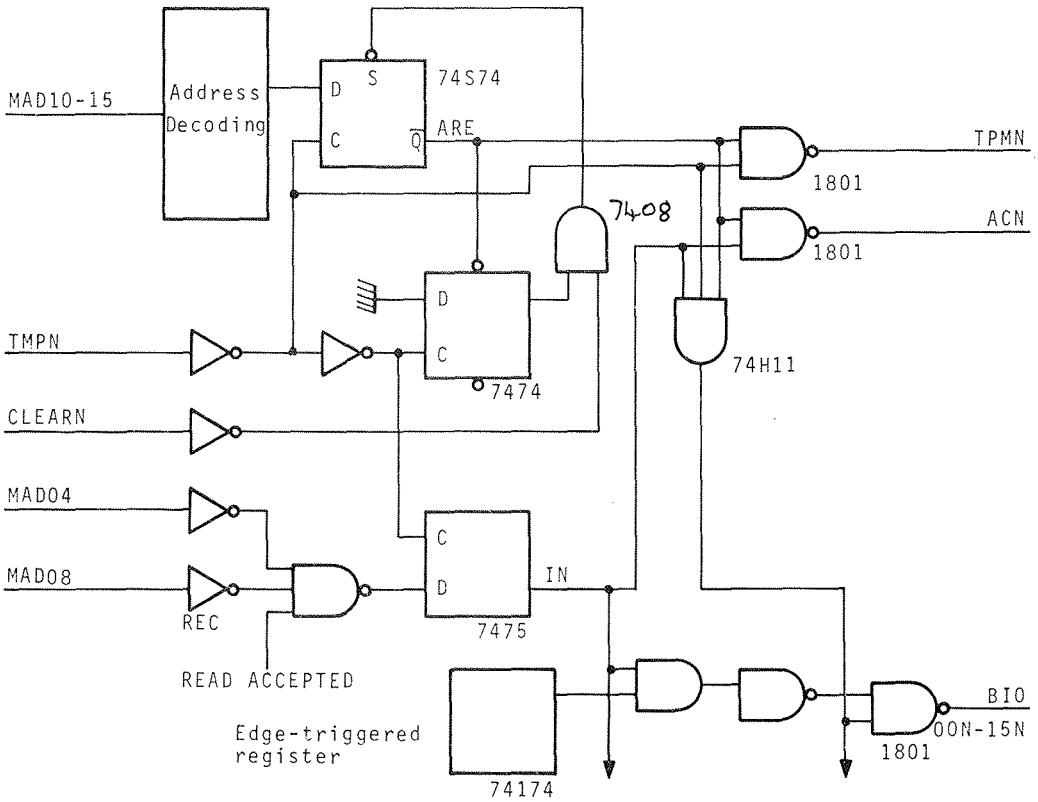
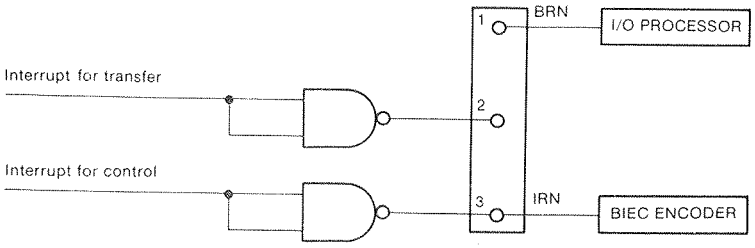


Figure 3.8 Reading the CU data register

PROGRAMMED CHANNEL OR I/O PROCESSOR SELECTION

For programmed channel operation the control unit requires only an interrupt request line (IRN) and the break request line (BRN) is not connected.



On the control unit a link is connected between 2 and 3 for programmed channel operation and between 2 and 1 for I/O processor operation.

CIRCUIT BOARDS

Physical Details

The standard dimensions applicable to control unit boards are given in Figure 3.9 (Type 1 double control unit board). The dimensions for a Type 2 control unit board can be found in Appendix 2. The spacing between the boards, whether they are mounted in the basic mounting box or in an equipment shelf, normally limits the height of the component bulk unless the neighbouring board slot is left vacant.

General Purpose Wiring Boards

The general purpose wiring board for Type 1 control shown in Figure 3.10., has hole patterns for accepting 226 dual-in-line packages with 14 or 16 pins 15 MSI, LSI packages with 24 or 40 pins and for discrete components. It is provided with printed circuit tracking for the d.c. supplies to the components and from the logic connections of each component to jumper wire connection points. A GPO bus connector for mating with the printed wiring board and header connectors for peripheral connections are mounted on the board.

Wiring Board Rules

A control unit built on the general purpose wiring board should follow the wiring rules for TTL-Circuits. A complete set of wiring rules is given in Chapter 4 of this Part.

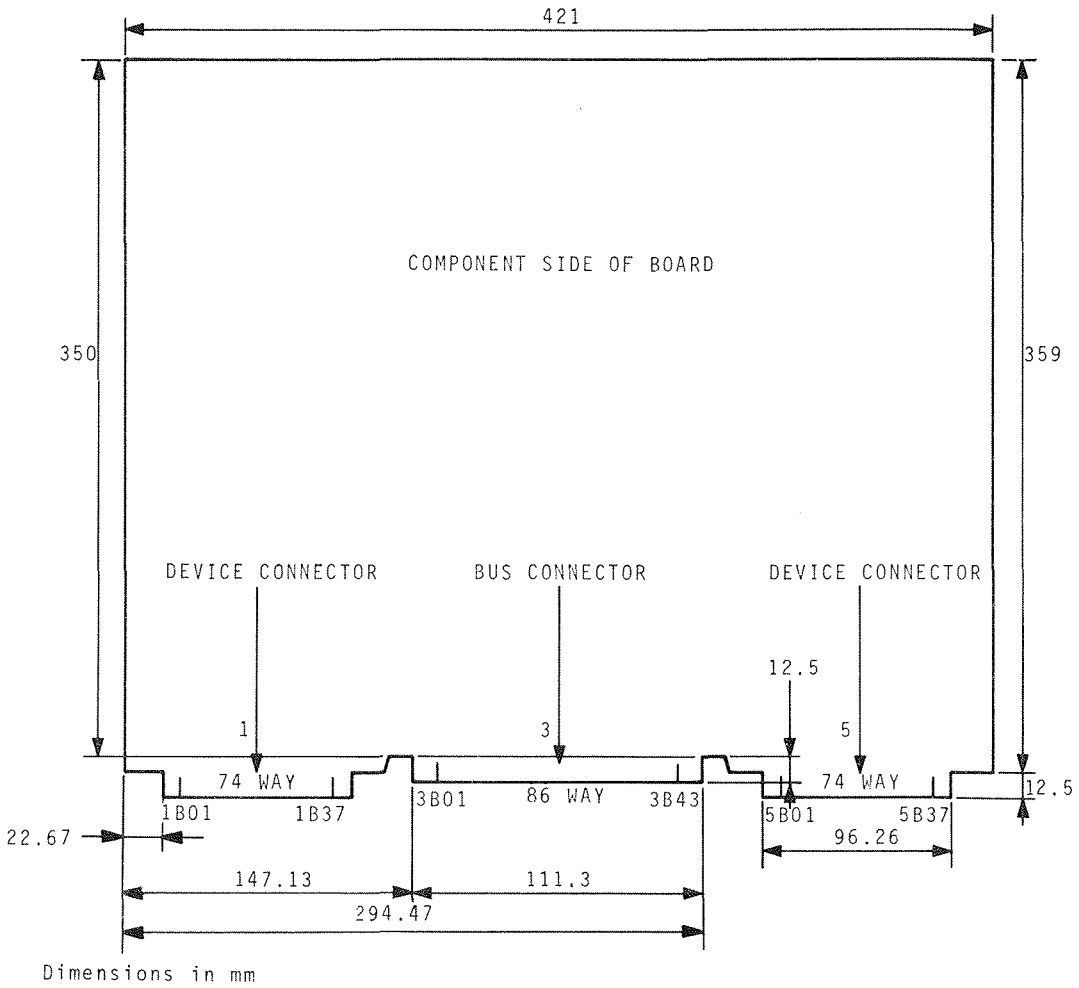


Figure 3.9. Control Unit Board Dimensions (Double CU Board)

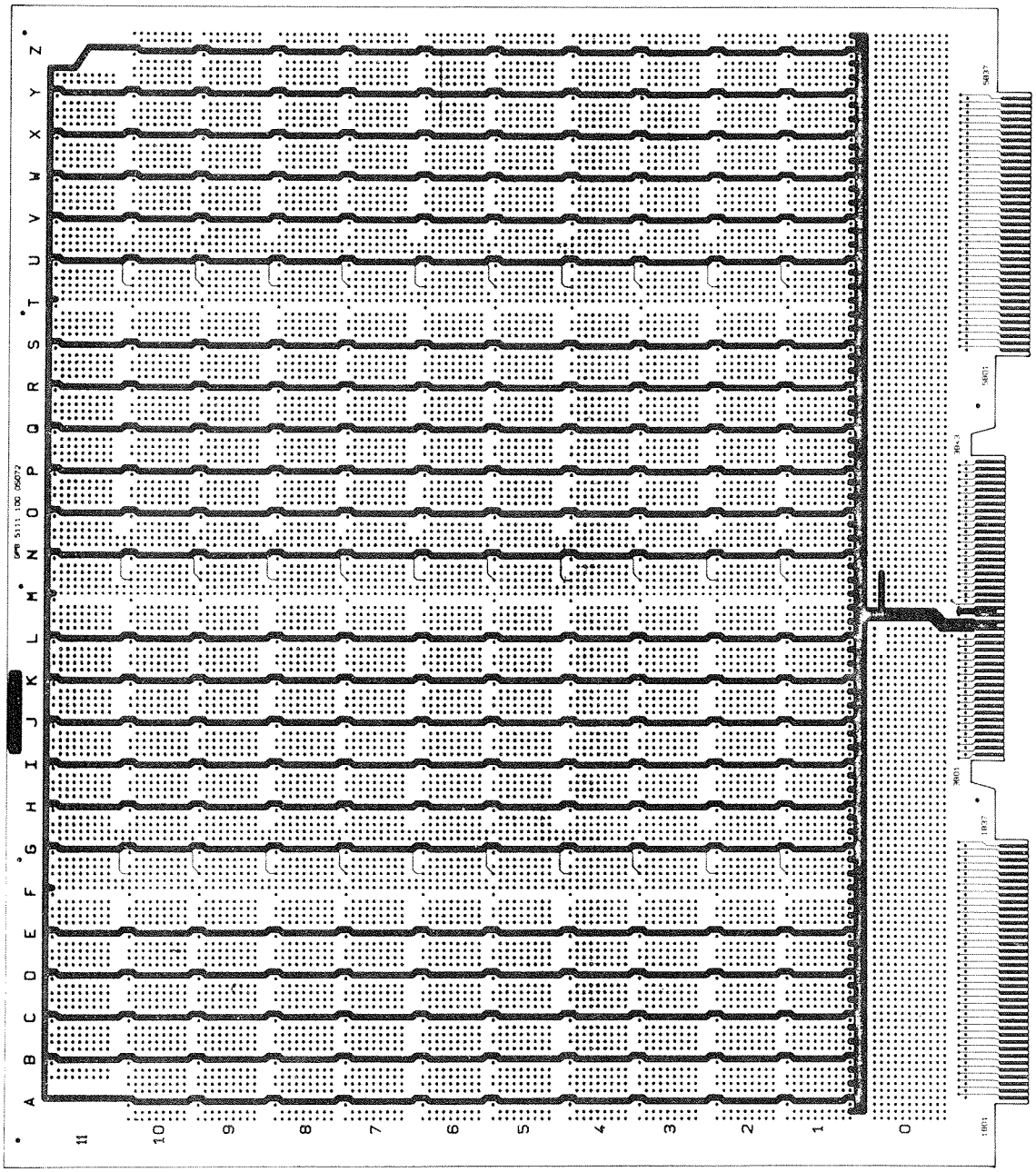


Figure 3.10 General Purpose Wiring Board

GENERAL PURPOSE INTERFACE BOARD FOR TYPE 1 CONTROL UNITS

Figure 3.11 shows the partially built general purpose interface board which is basically similar to the blank board defined earlier for Type 1 control units but contains the logic elements for handling the signal exchange across the GP bus via the programmed or I/O processor channel. Space with wiring patterns for dual-in-line packages with 14 or 16 pins and 2 MSI,LSI packages with 24 or 40 pins is provided, and two hole patterns formed by eyelets at 0.1 inch distance on which discrete components or additional dual-in-line packages can be placed.

General Purpose Interface Board — Circuit Details

The functions of the basic logic provided on the interface board, shown in Figure 3.12 are: address decoding and setting of the TPMN signal, decoding of the command signals and setting of the ACN signal. To allow for maximum flexibility for the design of the control unit no sequencing logic has been incorporated in the basic logic. An example of sequencing logic is given earlier in this chapter.

The board also provides adaption of all GP bus signals, gating of the output data on the BIO lines with signals ARE, TMPN, and encoding of the interrupt request signal onto the BIEC lines.

The hardware selection of the device address is made using 6 links and links are also used to make a selection between a common IR/BR-line (for programmed channel) or separate IR and BR lines (for I/O processor channel).

In the free area of the board, which is available for the customer to build his own control unit, the logic unit must be implemented according to the rules for TTL-circuits given in Chapter 4 of this Part.

For all 14-pins packages, and for 16 or 24-pins packages placed on locations for 36-pins packages, the appropriate connections to the power supply eyelets must be made.

The connections for the d.c. supply of +5 volts (±5%), with a maximum current of 3.0 A have been wired on the card. The maximum current needed for the basic logic is 1.0 A.

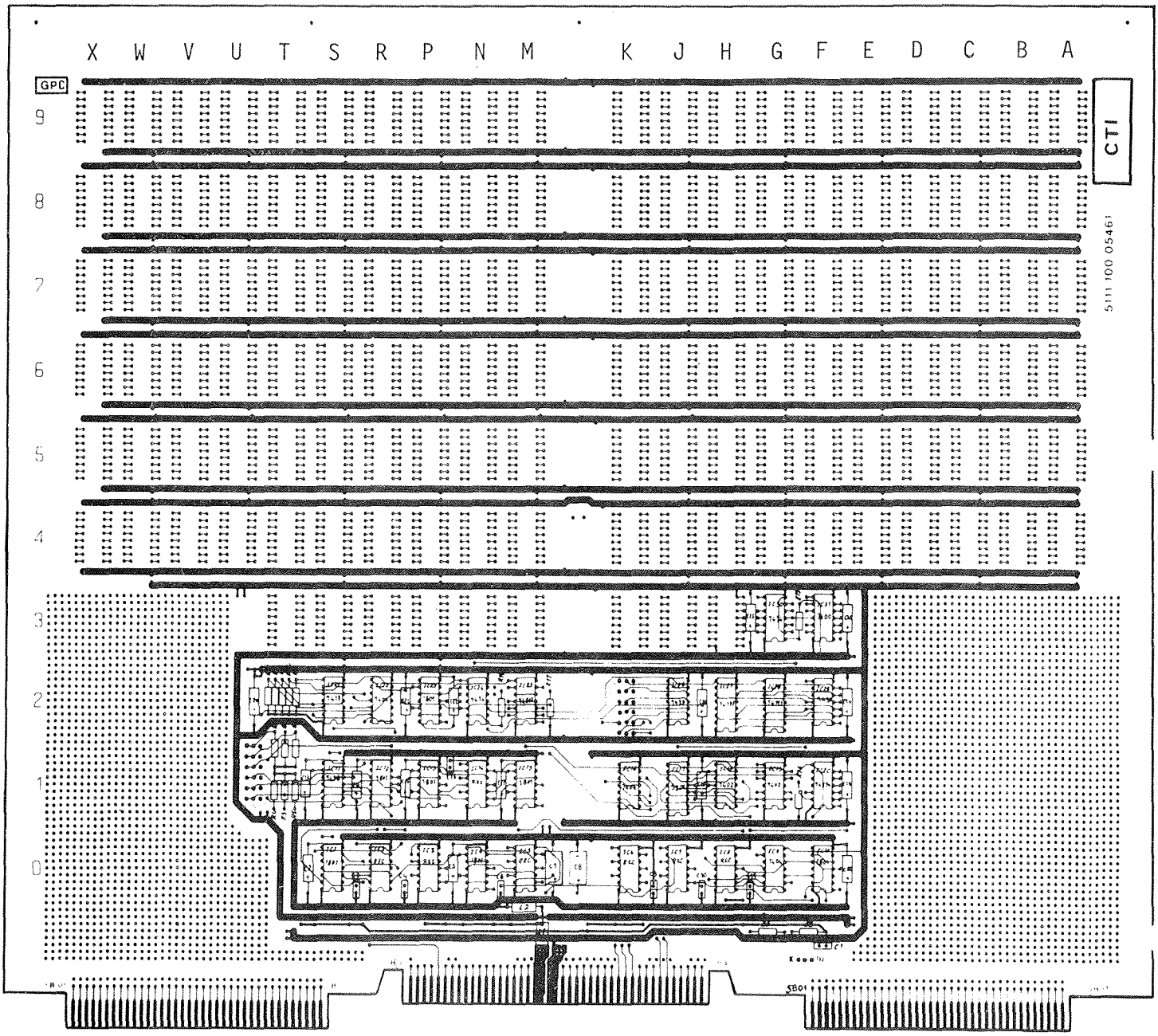


Figure 3.11 General Purpose Interface Board (P852M/P856M/P857M)

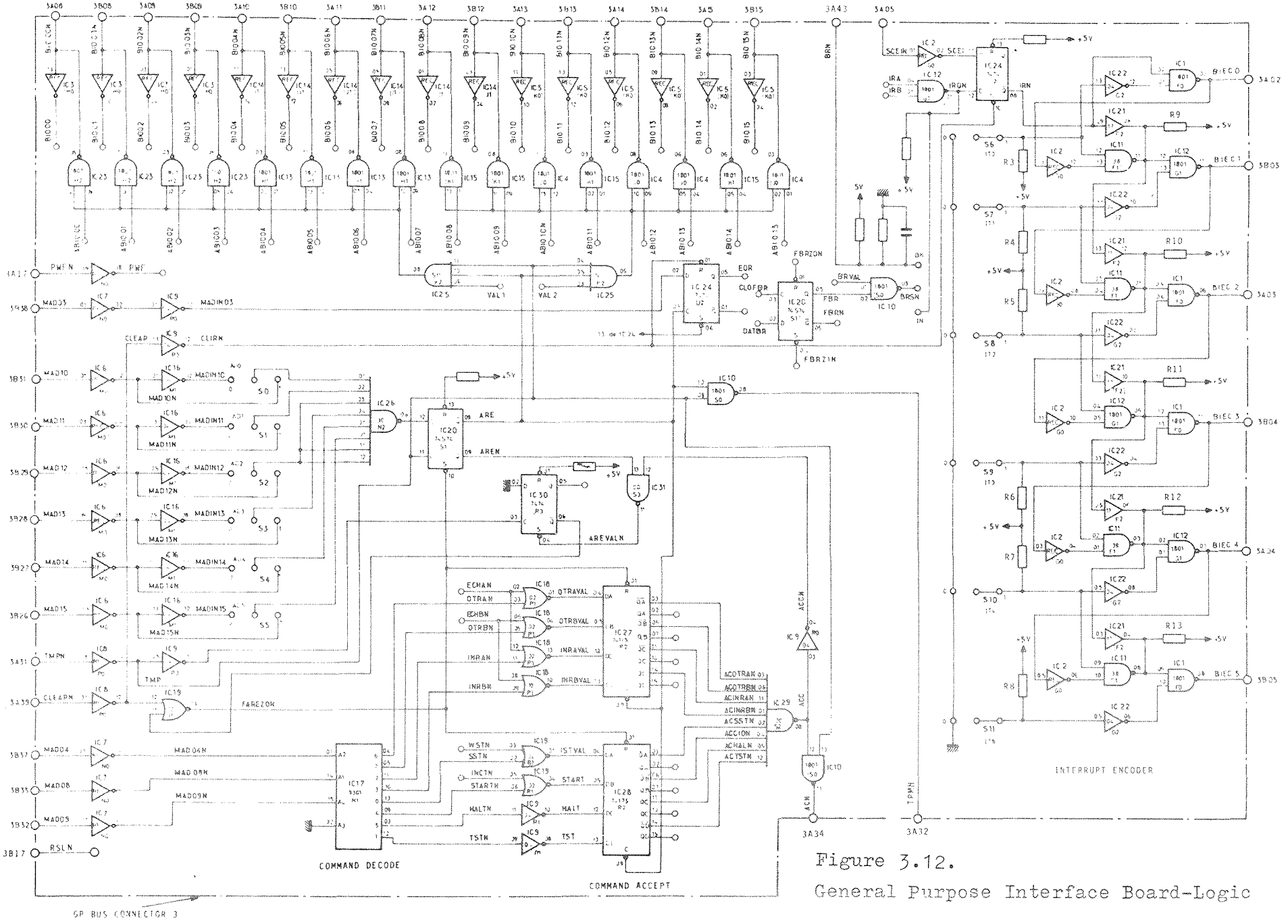


Figure 3.12.
General Purpose Interface Board-Logic

4 Wiring Board Rules

A control unit built on either of the General Purpose wiring boards should follow the wiring rules for TTL circuits. To apply the wiring rules the TTL circuits are divided into the following three groups:

74* and 93 circuits
74H circuits.
74S circuits.

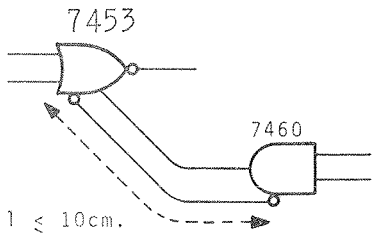
Rules for Signals Remaining Within a Board.

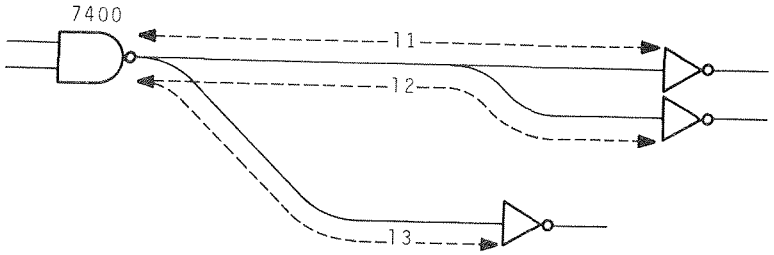
For signals whose emitter (or emitters for circuits with open collectors or 3 stage outputs) and receivers are on the same board all configurations are allowed provided that the greatest length in emitter output is not greater than the value indicated (in cm) in the table below:

Table 4.1 Maximum Emitter Lengths.

Circuit Group	Output Type		
	Gate(except with open collector or 3 states)	Flip flop and storage elements with Nand gates	Expanding gate
74 and 93	100	50	10
74H	60	30	10
74S	30	10	10

Examples





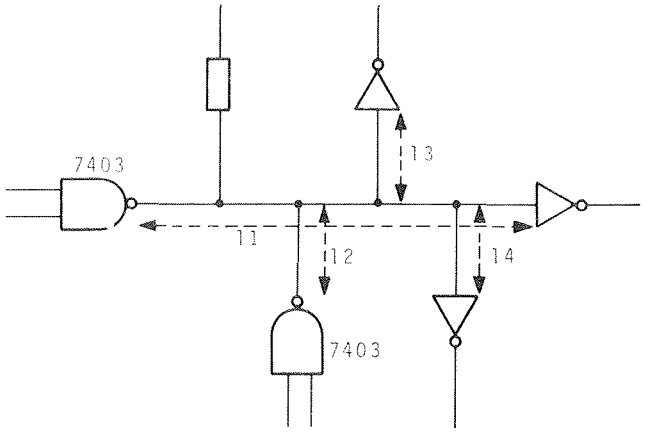
max. length for l1, l2 or l3 ≤ 100cm.

Two connections should not run in parallel for more than 20 cm (for 74 and 93 circuits), more than 15 cm (for 74H circuits), or for more than 10 cm (for 74S circuits).

Circuits With Open Collector or a 3 state Output.

A +5V load resistor is required (1 resistor is sufficient); its location is not important when the sum of the output lengths is not greater than 50 cm (for 74 circuits), or 40 cm (for 74H circuits), or 20 cm (for 74S circuits).

Example.



l1 + l2 + l3 + l4 ≤ 50cm.

Clock Circuits

In general clock sub-assemblies are constructed from a specific circuit diagram. All the components of this circuit, including supply decoupling capacitors, should be located as close as possible to each other.

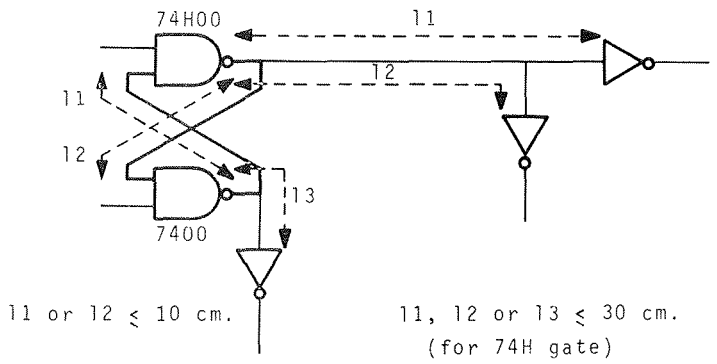
Monostable Delay Circuit

The external resistors and capacitors (which determine pulse width) and the 10 nF supply deccupling capacitor should be located as close as possible to the integrated circuit (connecting length not greater than 3 cm).

Memory Circuit

When the memory element is made from two gates, as shown in the example, the interconnection lengths shall be not greater than 10 cm. For the element output the rule given for the flip flop maximum output length earlier in this chapter shall be used.

Example

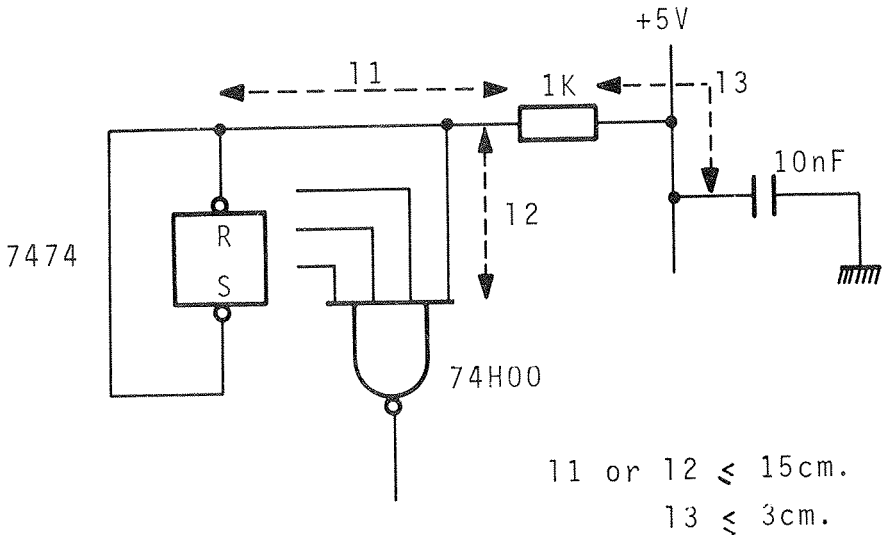


Unused Inputs

Unused inputs should be at a well defined level to avoid unpredictable switching of the associated output. When the fan-out of a driving circuit is sufficient all unused inputs of a gate shall be linked to a used input.
When the above is not possible, i.e. in the case of a NOR gate, the unused inputs should be set to level `0' by directly earthing the input. In the case of NAND gates (active low inputs) the unused inputs should be set to level `1' as follows:

> The input shall be connected to +5V via a 1k $\frac{1}{4}$W 5% resistor and the +5V supply decoupled by a 10 n.F ceramic capacitor located at less than 3 cm from the end of the resistor. Up to 10 unused inputs may be connected to one resistor provided that the inputs belong to 4 chips around the resistor and the greatest distance between the resistor end and the unused input is not greater than 15 cm.

Example



Bus Interface Signals - Wiring board Layout Rules

As described in Chapter 2 of this Part the bus signals are grouped into three classes A,B,and C (shown in Table 2.2 and 2.3 Chapter 2). Note that the signals are either:

- Emitted by an 1801
- Received by a REC 0612 or by a Ternet adaptor bridge(or adaptor of 220Ω and 390Ω connected between +5V and 0V).

Layout Rules - Double face board (interface packages and tracks)

Logic unit interface packages should be geographically distributed by groups on the board (signals of different groups should not be mixed in the same logic unit) and should be located independently of the grid usually utilized for the layout of the packages.

The printed tracks linking interface packages to board connector 3 should be arranged in groups (A,B,or C) and the tracks of signals belonging to different groups should not run in parallel. Where the tracks of different groups cross the intersection should be ideally at 90° but always not less than 60°.

Note: Where it is found impossible to prevent signals of different groups running in parallel an earth track linked to the earth grid should be arranged to pass between the two signals.

The lengths of the tracks from elements to connector 3 should be kept to a minimum and should not exceed 7 cm for all control signals (group C) or 15 cm for all other signals (groups A and B).

Note: Where it is found impossible to reduce the length of the track for a control signal to not greater than 7 cm a maximum of 10 cm may be accepted provided that an earth track, linked to the earth grid, is run along the whole length of the signal track and in close proximity to the track.

The central 0V (on pins 3A21, 3A22, 3B21, and 3B22) and the earth pins of the interface packages should be linked using tracks that are as short and as wide as possible (minimum width 2 mm).

Other earth paths should be linked together to form as small a grid as possible (max mesh area 9 cm^2 - if necessary add further earth tracks to achieve a tight mesh).

The additional earth pins on connector 3 should be linked, through the grid, to the earth pins of packages containing control signals and shall be distributed as follows:

BR No. 4 TMRN TRMN TMEN	3A 24, 25	TPMN TMPN ACN SPYC	3A 33
BUSRN MSN CLEARN BR 1-3 BSYN	3A 40	OKO OKI RSLN PWFN	3A 18
BIEC 0-5 SCEIN	3A 07		

Note: Unused connector earth pins shall be linked to the earth grid with a connection as short as possible.

Layout Rules - Multilayer Circuit (interface packages and tracks)

The earth plane shall be internal and shall cover the whole board area; all earth points of components on the board and all the OV pins of connector 3 shall be connected to the earth plane.

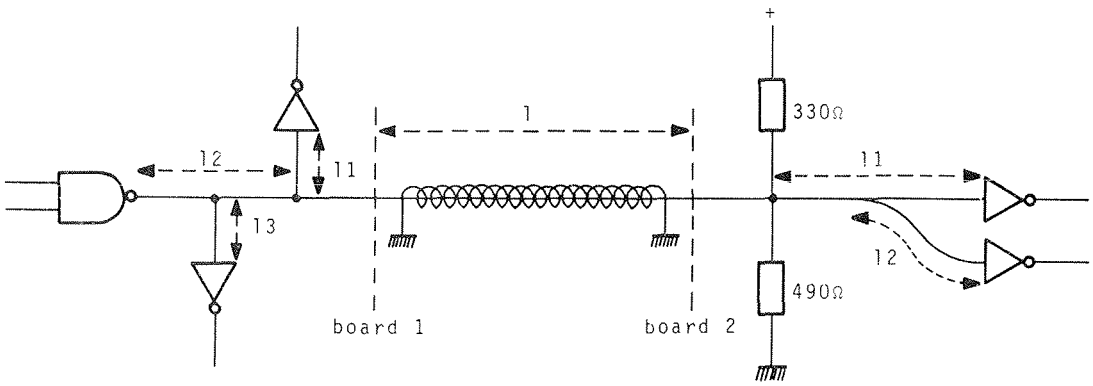
The layout of the interface packages shall be as described previously for the double face board.

The layout of tracks linking the interface packages with connector 3 shall be grouped as described for double face boards. The tracks of two signals of different groups may be parallel provided that they are separated by the earth plane. The length of the tracks should be kept to a minimum and be not greater than 7 cm for all control signals (Group C) and not greater than 15 cm for all others (Groups A, B and remaining signals).

Other Signals - Wiring Board Layout Rules

For signals one receiver of which is not on the same board as the emitter, the length of the signal path on each board and also the length of the linking path between the boards shall be taken into account. The total length should be not greater than the value indicated in Table 4.1.

If the length obtained is greater than the value given in the table then the wiring should be done as indicated below.



If length l is $\leq$ 1.5m then the active signal should be accompanied by a printed earth track.
If length l is between 1.5m to 4m then the active signal line should be made using twisted wire.
l1 + l2 + l3 should be $<$ 30 cm if using series 74 emitter
l1 + l2 + l3 should be $<$ 20 cm if using series 74H emitter
l1 + l2 + l3 should be $<$ 10 cm if using series 74S emitter
This relationship is valid at each end.

Integrated Circuits Power Supply and Power Supply Decoupling

Bus Interface IC's - Power Supply and Power Supply Decoupling

The interface packages +5V power supply shall be used as the +5V supply for all other packages on the board.

Each emitter, receiver, and Ternet package shall be decoupled by a 10 nF ceramic capacitor.

Power Supply and Decoupling of Other Circuits

Taking the connector edge of the board as a reference a 3 perpendicular bar grid shall be implemented (2 edges and the middle of the board) and as many parallel bars as there exist rows of packages. Each parallel bar shall be decoupled at both ends by a 10 µF, 25V Fitco Capacitor (2222 015 16109). The width of the perpendicular bars shall be not less than 6 mm and of the parallel bars not less than 3 mm.

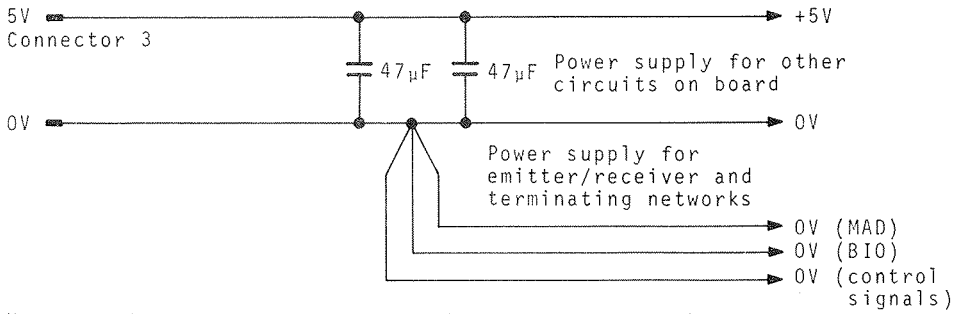
Four 47 µF, 10V Fitco Capacitors (2222 015 14479) at most (and one at least) shall be distributed among the 0V and 5V central strips.

Note: The total sum of board Capacitors must not exceed 300 µF, therefore, if necessary, the 47 µF capacitor may be replaced by 22µF 25V Fitco (2222 015 16229)

The timer circuits, monostables, 1801 emitters, 0612 receivers and Ternet networks shall be decoupled with a 10 µF, decoupling ceramic capacitor (connecting length not greater than 3 cm).

For all other packages on a board one 10 nF ceramic decoupling capacitor shall be provided for each group of 3 packages maximum.

Power Supply Distribution on Logic Boards
On logic boards the +5V input power supply must be applied via a circuit as shown in Figure 4.1 so that the supply to the emitters, receivers and terminating networks is filtered from the supply to the other circuits on the board.



Note: It is recommended to have three separate earth lines (width ≤2mm) for the BIO, MAD and Control bus signals as shown in the circuit above.
Each row of DIL is filtered by 10µF at each end.
The sum of capacitors must be ≤300µF.

Figure 4.1 Power Supply Fillter Circuit

Integrated Circuit Equivalence Table

Using Table 4.2. it is possible to substitute one type of package for another provided that length of connection and fan-out constraints are adhered to.

Example: A 7400 gate may be replaced by a 74S00 if the maximum length in output is less than 30 cm.

Back Panel (Motherboard)

A back panel circuit shall include an earth plane which shall provide the function of separating the control, BIO, and MAD signals and of connecting all connector earth points together.

Table 4.2. Integrated Circuit Substitutions

7400	with	7410,13,20,30,37,40,132,74H00,10,20,30,40,74S00, 10,20,40
7401	with	7403
7402	with	7425,27,50
7403	with	7401, 4738
7404	with	7400,02,10,13,14,20,25,27,37,40,86,132,74H00,04, 10,20,30,40,74S00,04,10,11,20,40, REC
7405	with	7401,03,06, 38
7406	with	-
7408	with	74H11
7410	with	7420,40,74H10,20,40, 74S10, 20, 40
7413	with	-
7414	with	7417,132
7416	with	7406
7417	with	-
7420	with	7440,74H20,40,7420 ,40
7425	with	-
7427	with	7425
7430	with	74H30
7432	with	-
7437	with	7440,74H40,74S40
7438	with	1801
7440	with	74H40,74S40
7450	with	-
7453	with	74H53
7460	with	74H60
7473	with	74H73
7474	with	74H74,74S74
7476	with	74H76
7486	with	74S86
74132	with	7413
74175	with	74S175
74H00	with	74H10,20,30,40,74S00,11,40,74
74H04	with	74H00,11,20,30,40,74S00,04,10,20,40, REC *

74H10	with	74H20,30,40,74S10,20,40
74H11	with	74S20
74H20	with	74H30,40,74S40,74
74H30	with	-
74H40	with	74S40
74H50	with	-
74H53	with	-
74H55	with	-
74H60	with	-
74H73	with	-
74H74	with	74S86
74H76	with	-
74S00	with	74S11,40,74
74S04	with	74S00,74S10,20,40
74S10	with	74S20,40
74S11	with	-
74S20	with	-
74S40	with	-
74S74	with	-
74S86	with	-

*Substitution possible if you admit tpd max = 12 ns (REC) instead of tpd max = 10 ns (74H04).

As well as the d.c. voltage levels needed for the memory modules and the processing and control unit logic circuits, the CPU power supply unit provides power fail (PWFN), reset (RSLN) and real time clock (RTC) logic signals. The power supply is included in the mechanical construction of the basic mounting box and all outputs from the supply are wired directly to terminations in the basic box. The mains supply to the unit is wired directly to the power supply. The block diagram in Figure 5.1 shows the interface signals and Tables 5.1 and 5.2 detail the supply characteristics.

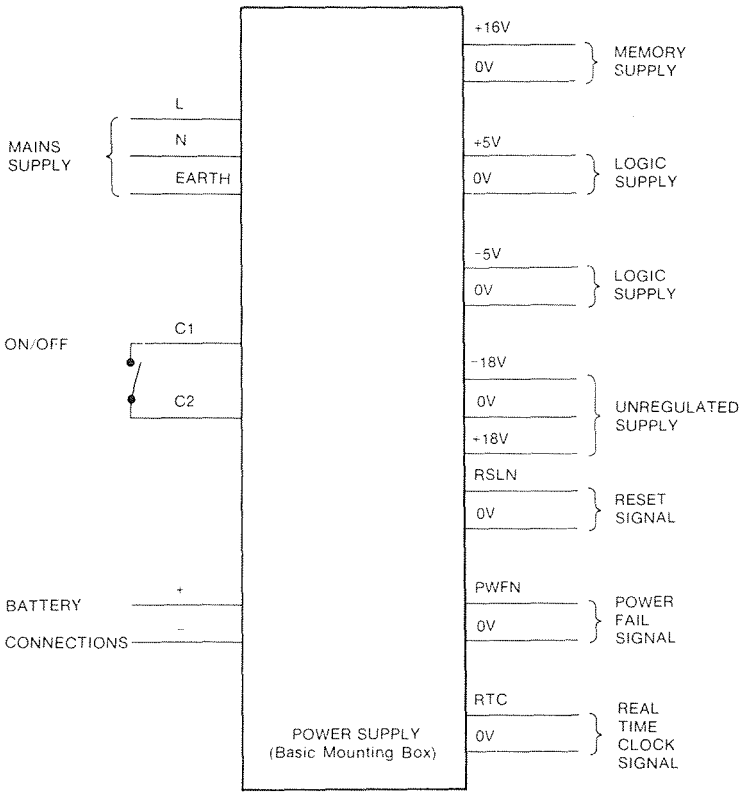


Figure 5.1 CPU power supply

Mains Input Voltage: 100V, 115V, 220 or 240V ± 10% single phase with neutral and earth or two phases with earth.

Mains Frequency: 50 Hz ± 2 Hz or 60 Hz ± 3 Hz.

Table 5.1. Output Supply Characteristics

Level	Overall Stability	Max. Current Available*			Protection to be Initiated Between	
		100/200 Series	400 Series	500 Series	Overcurrent	Overvoltage
+5V	± 5%	18A	43A	86A	+25%	6V to 7V
−5V	± 5%	0.8	2.6A	5.2A	Protected by fuse	6V to 7.5V
+16V	± 3%	4.5A	9.0A	18A	+25%	+5% to +27%
±18V	−10% +70%	1A	2.0A	4.0A	Protected by Fuse	Unprotected

Note * The maximum current available varies with the basic mounting box, more details of which can be found in Section 2 Part 2 Chapter 1.

Table 5.2. Logic Signals Characteristics

Signal	Connection	CPU Load	Logic "0" Level	Logic "1" Level
PWFN (and 0V Line)	Twisted Pair	30 TTL Nands Open collector	0 to 0.4 Volts Isink 48 mA	2.4 to 5 Volts Iload 1 mA
RSLN (and 0V line)	Twisted Pair	Open Collector Output	0 to 0.4 Volts Isink 150 mA	
RTC (and 0V line)	Twisted Pair	10 TTL Nands	0 to 0.4 Volts Isink 16 mA	2.4 to 5 Volts Iload 400 µA

LOGIC SIGNAL AND D.C. SUPPLY SEQUENCING

The sequential timing for the d.c. levels and logic signals is shown in Figure 5.2 and illustrates how automatic protection is provided for switching on and off and for power or mains failures of greater than 10ms duration. The sequencing is necessary to prevent loss of information in the memory, discs, tapes etc., and for implementing the power failure/automatic restart circuits in the CPU.

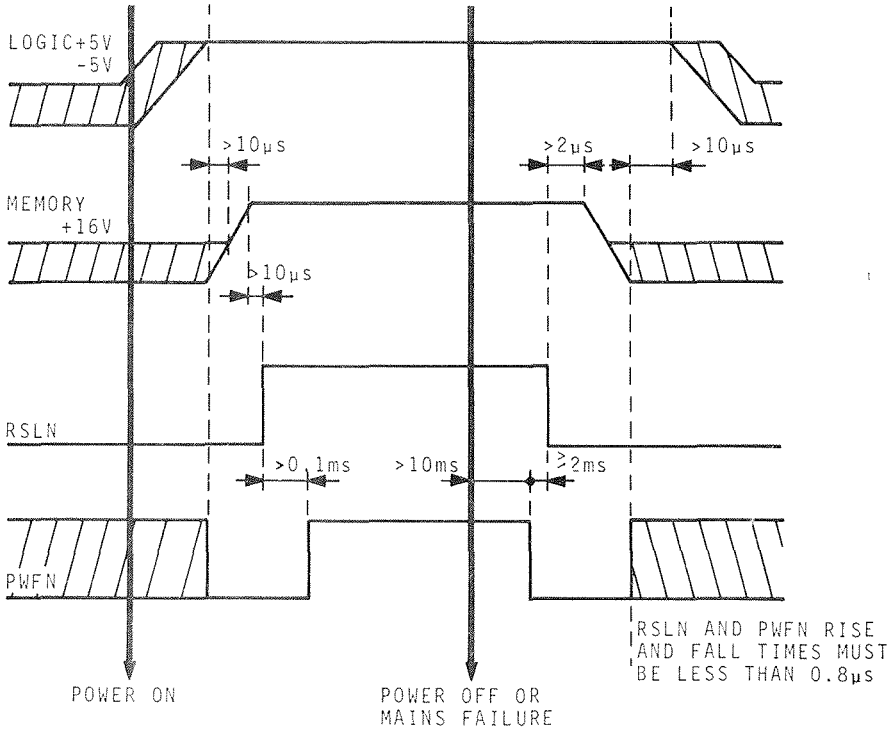


Figure 5.2 CPU power supply interface timing

Logic Signals

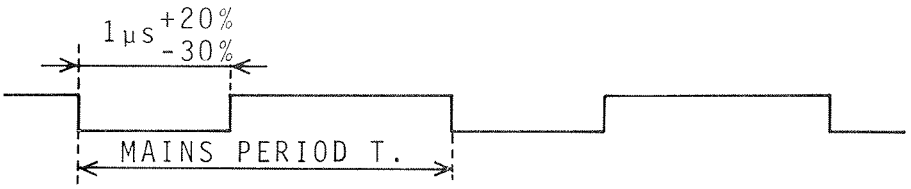
The two signals RSLN and PWFN are used by the CPU to control the power supply switching and are also extended via the GP bus to control external rack power supplies.

RSLN: This signal is used to indicate the functioning of all power supply units. If it is set to "0" the system is not operating and if it is set to "1" the ±5 V and +16 V supplies are supplied correctly.

PWFN: Transition of this signal from "0" to "1" means that, if the power is supplied correctly, the system may be set to operate. The transition of this signal from "1" to "0" means that, provided the power is supplied correctly, after a fixed delay the system will not be able to operate any longer.

Real Time Clock

The timing for the real time clock signal RTC is shown in Figure 5.3.



T= 20ms for 50Hz mains supplies
and 16.67ms for 60Hz mains supplies

Figure 5.3. Real time clock signal

POWER CONTROL WHEN USING EQUIPMENT SHELVES

The block diagram in Figure 5.4 shows the logic signal arrangement when using a system in which further power supplies are used in equipment shelves.

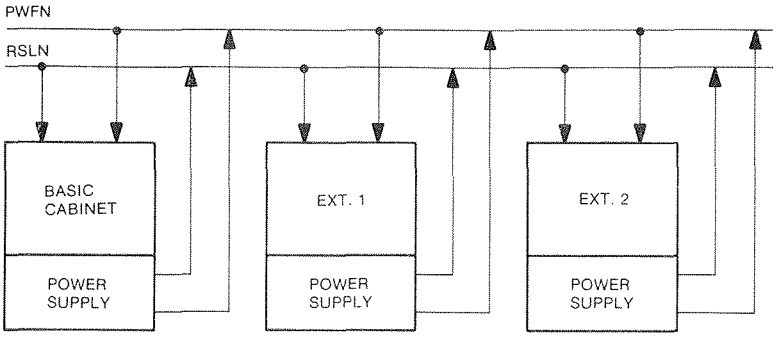


Figure 5.4 System using external power supplies

With this arrangement PWFN and RSLN are connected to each equipment shelf via the GP bus and can be activated by any one of the power supplies of the system connected on the bus. If one power supply fails the whole system becomes inoperable due to perturbation on the bus given by the failed supply. The system functions as follows:

Switching off

When a mains failure is detected ($\geq$10 ms) a PWFN signal is generated by one of the power units. This signal informs all the equipment shelves of the failure but only the CPU can take the signal into account by a priority interrupt handling. The CPU knows that the system will be not operable within 2 ms and initiates a safe guarding routine for the program running.

Signal RSLN is generated 2 ms after PWFN and provides an earth signal to protect the peripherals or memories during the decrease of the voltages. If RSLN appears during a memory cycle (due to an overtime of the power failure routine) the memory content is not guaranteed.

Switching on

Signal RSLN is used to generate a master clear signal to protect the peripherals during the increase of the voltage. Signal PWFN is used to restart the system at the point where it has been stopped.

The DIOS system is a general purpose digital input/output system which allows any external digital equipment to interface with the P852M/P856M/or P857M CPU. The system is connected to the GP bus in the same manner as a peripheral device control unit and operates on the programmed channel. Two basic DIOS units are available:

P837-001	DIOD	1 word input/1 word output.
P837-002	DIOD	2 words input/2 words output.

In these units only the output channels are buffered and TTL levels to ± 24V input and ± 30V output are provided for at the input and output.

Basic DIOS Connections

Both DIOD units are constructed on standard printed wiring boards (see Quad CU board Chapter 1 of this Part) and employ one 86-way connector(3) for connection to the GP bus and either 2x 26-way connectors(Nos. 1 and 2) for P837-001 or 4 x 26-way connectors (Nos. 1,2,4,and 5) for P837-002 to make user device connections. The boards can be located either in a basic mounting box or in an external equipment shelf P843-001; only a +5V supply is needed to drive the units .

One transmitter/receiver is employed per interface signal and for P837-002 DIOD each channel is fully independant with a separate device address and interrupt level.

As an option cables 3 metres in length with terminations can be provided but device connection between the DIOD boards and external equipment is the customers responsibility(see later information on input and output interface circuits in this chapter).

CALL Signals

For all call signals from external equipment link connections are provided which enable the user (who may be using either active high or active low signals) to select whether the call flip flop in the DIOD is clocked on the leading edge of the CALL signal or on the trailing edge. The input circuit and waveforms are shown in Figure 6.1.

Example of Input Circuit:

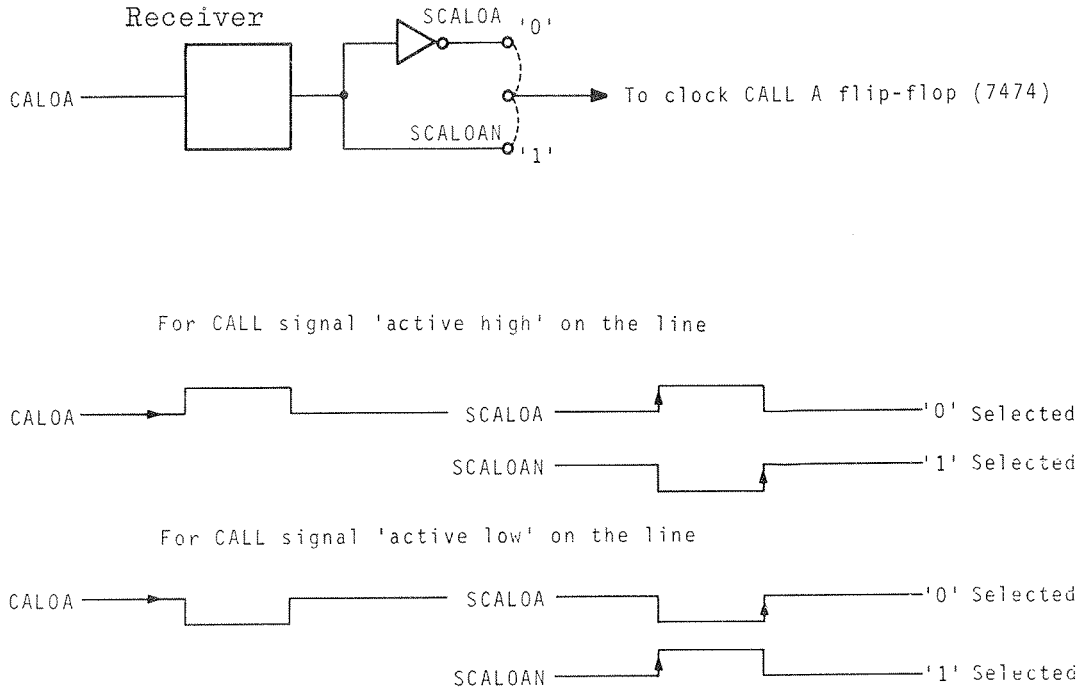


Figure 6.1 CALL Signal Input Circuit

The respective CALL flip flop is clocked only on a transition from 'Low to High' (indicated by an arrow in the diagrams) and the possible choices can be clearly seen. The links are marked on the board as follows:

`ACALO' for CALL in output mode (word A)
`ACALI' for CALL in input mode (word A)
`BCALO' for CALL in output mode (word B)
`BCALI' for CALL in input mode (word B)

The `0' and `1' positions are also marked.

Note: The minimum duration of a CALL signal must be 200 ns and, for good noise immunity, it is recommended to use CALL signals which are `active low' .

OK Signals

For the OK signals which are sent to the user device after the transfer of data (OKOA, OKIA, OKOB, or OKIB) a circuit is provided which enables the user to preselect the duration of the pulse for each OK signal. The basic circuit for each signal is shown in Figure 6.2.

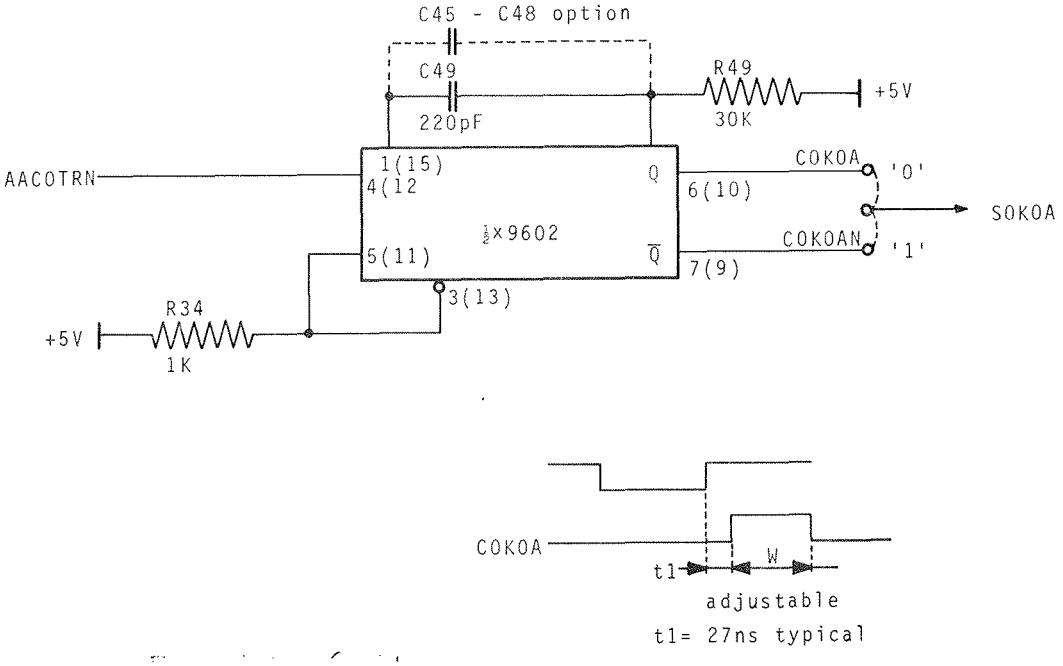


Figure 6.2. Circuit for Generation of OK Signal

The pulse width (w) is adjustable over the range 2 µs to 1 ms using capacitors C45 to C49. For the standard version C49 is always mounted which makes w= 2 µs ± 10% but if other pulse widths are required a capacitor can be mounted in parallel with C49 to change the width as follows:

C45= 1.8 nF ± 10%	w = 20 µs ± 10%
C46= 10 nF ± 10%	w = 100 µs ± 10%
C47= 47 nF ± 10%	w = 500 µs ± 10%
C48= 100 nF ± 10%	w = 1 ms ± 10%

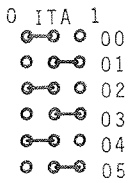
Links provided on the board allow the user to select an OK or OK/ signal. The links are identified on the board as follows:

`SOKOA ´ for OK signal in output mode (word A)
`SOKIA´ for OK signal in input mode (word A)
`SOKOB´ for OK signal in output mode (word B)
`SOKIB´ for OK signal in input mode (word B)

Each link is also identified `0´ or `1´. For `0´selection the OK signal is `active low´on the line and for `1´selection the OK signal is `active high´on the line.

Interrupt Level Selection.

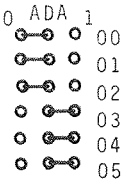
Interrupt level selection is made using links on the board. When a DIOD 2 board is used each separate channel interrupt level can be preset using two separate sets of links. The layout for one channel is as follows:



Example shows interrupt set to /15

Address Recognition

Each I/O channel has its own address (when DIOD 2 board is used) and the Address is selected by means of two separate sets of links on the board. The layout for Address A is as follows:



Example shows Address set to /07

Data Words Interface

The DIOD board has a 2 x 16-bit output buffer (DIOD 1 board has 1x16-bit buffer)connected to the BIO lines on the GP bus and the words from the CPU are buffered on the trailing edge of the respective accepted data exchange for the word A (or B). The data is available on the output lines until a new exchange is performed. The buffer details are as follows:

FOUT A(or B) 00 to 15	: Type D edge triggered flip flop (74174)
Data	: BIO 00 to BIO 15
Clock	: A (or B) ACOTR, TMPA
Reset	: CLIRN

Special Output Circuit

A total of 40 driver circuits (2x 16 data + 2x2 OK lines + 2x2 MCLN lines) are used between the DIOD 2 board and the devices. An example of one driver is given in Figure 6.3.

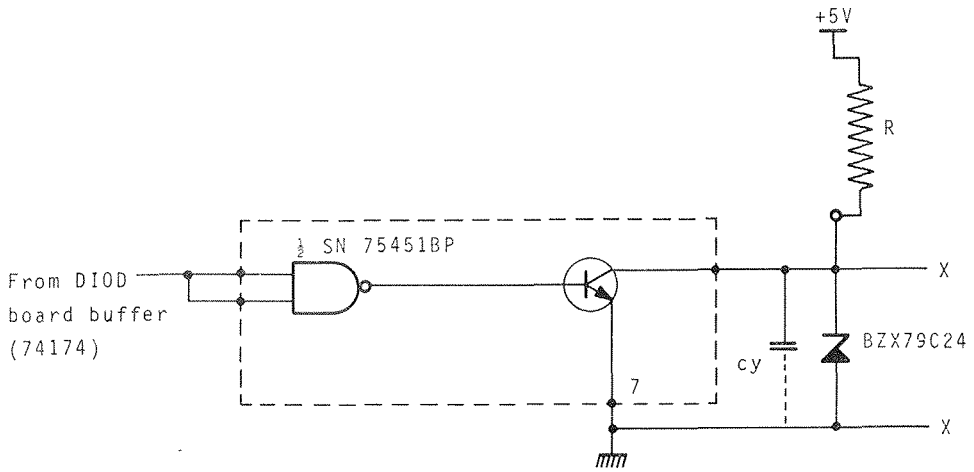


Figure 6.3. Output Driver Circuit

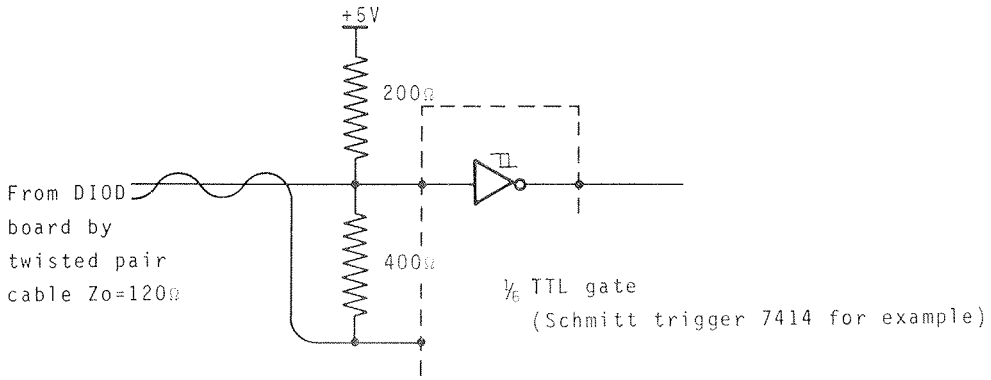
The output circuit is an open collector transistor:

- Output Voltage Vout $\leqslant$ 30 volts.
- Vol $\leqslant$ 0.4 for Iol $\leqslant$ 100 mA (TTL level)
- Vol $\leqslant$ 0.7V for Iol $\leqslant$ 300 mA
- Iol max = 300 mA
  Ioh max = 100 μA

The speed and transition times on the lines are determined by the load and length of the line (determined by customer).

Resistor R is a matching resistor which is used if the user circuitry is not adapted for the cable. The resistor is mounted on the circuit board under the customers responsibility with respect to current limitation of the output driver.

For a TTL connection it is not recommended to use the resistor R but to employ a resistor bridge at the user end. The arrangement is shown below.



Input Data Circuit

On the DIOD 2 board there are 32 receiver circuits for the BIO S00 to BIO S15 input signals from external equipment(word A and word B) plus 2x2 CALL lines. The signals are received on the circuit shown in Figure 6.4.

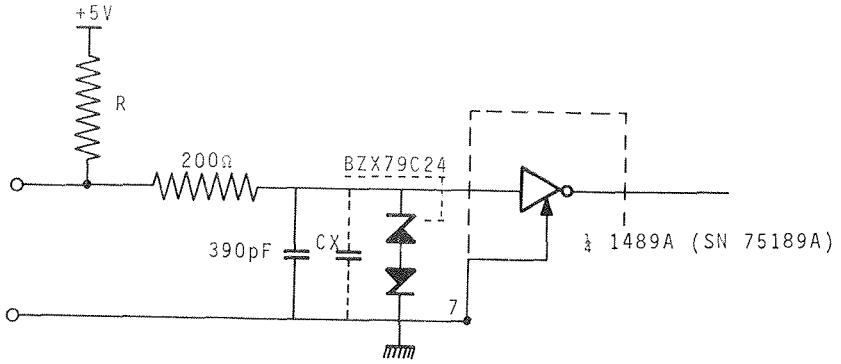


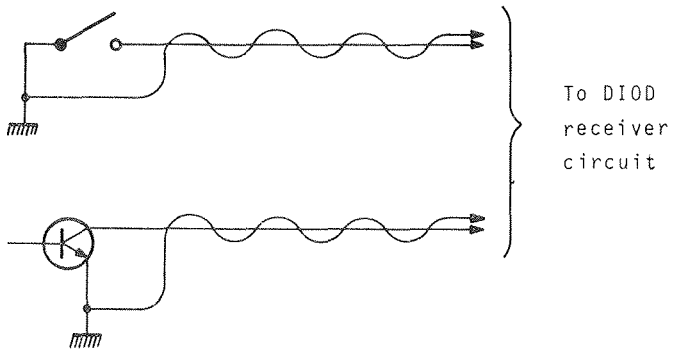
Figure 6.4. Input Receiver Circuit

Circuit performance:

- Input Voltage Vin $\pm$ 25V (Zin $\simeq$ 3k)
- Input threshold 1.75V $\leqslant$ Vih $\leqslant$ 2.25V
  0.75V $\leqslant$ Vil $\leqslant$ 1.25V

The interface complies with V28 requirements for voltage level and input impedance. Modification of the input filter is possible by the addition of capacitor Cx under customers responsibility. The speed is given for 50k words/sec.

Resistor R (120Ω $\pm$ 5% 0.125W) is mounted on the DIOD board under customers responsibility; it is necessary to mount it to allow for the possibility that lines from a switch or from an open collector may be received which have no reference voltage as shown below.



Resistor R can be used also for matched lines.

Pin Connections To DIOD Boards

The pin connections, for the connection of user devices to the DIOD boards, are listed in Tables 6.1 and 6.2.

Table 6.1 Pin Connections for DIOD 1 and DIOD 2 Boards

Signal	Pin No		Signal	Pin No
Mech Earth	1A01		Mech Earth	2A01
DINA 08	1A02		DOUTA 08	2A02
DINA 09 D3	1A03		DOUTA 09 D3	2A03
DINA 10	1A04		DOUTA 10	2A04
DINA 11	1A05		DOUTA 11	2A05
Ground D3	1A06		Ground D3	2A06
DINA 12	1A07		DOUTA 12	2A07
DINA 13 D4	1A08		DOUTA 13 D4	2A08
DINA 14	1A09		DOUTA 14	2A09
DINA 15	1A10		DOUTA 15	2A10
Ground D4	1A11		Ground D4	2A11
OKIA	1A12		OKOA	2A12
Ground CALIA	1A13		Ground CALOA	2A13
DINA 00	1B01		DOUTA 00	2B01
DINA 01 D1	1B02		DOUTA 01 D1	2B02
DINA 02	1B03		DOUTA 02	2B03
DINA 03	1B04		DOUTA 03	2B04
Ground D1	1B05		Ground D1	2B05
DINA 04	1B06		DOUTA 04	2B06
DINA 05 D2	1B07		DOUTA 05 D2	2B07
DINA 06	1B08		DOUTA 06	2B08
DINA 07	1B09		DOUTA 07	2B09
Ground D2	1B10		Ground D2	2B10
MCLA1N	1B11		MCLA2N	2B11
Ground(MCL,OK)	1B12		Ground(MCL,OK)	2B12
CALIA	1B13		CALOA	2B13

Connector P843-022 (26 Pins)

Cable 20 twisted pairs cable minimum (Zo = 120 ohm)

Table 6.2 Additional Pin Connections for DIOD 2 Board (P837-002)

Signal		Pin No.		Signal		Pin No.
Mech Earth		5A01		Mech Earth		4A01
DINB 08		5A02		DOUTB 08		4A02
DINB 09	D3	5A03		DOUTB 09	D3	4A03
DINB 10		5A04		DOUTB 10		4A04
DINB 11		5A05		DOUTB 11		4A05
Ground D3		5A06		Ground D3		4A06
DINB 12		5A07		DOUTB 12		4A07
DINB 13	D4	5A08		DOUTB 13	D4	4A08
DINB 14		5A09		DOUTB 14		4A09
DINB 15		5A10		DOUTB 15		4A10
Ground D4		5A11		Ground D4		4A11
OKIB		5A12		OKOB		4A12
Ground CALIB		5A13		Ground CALOB		4A13
DINB 00		5B01		DOUTB 00		4B01
DINB 01	D1	5B02		DOUTB 01	D1	4B02
DINB 02		5B03		DOUTB 02		4B03
DINB 03		5B04		DOUTB 03		4B04
Ground D1		5B05		Ground D1		4B05
DINB 04		5B06		DOUTB 04		4B06
DINB 05	D2	5B07		DOUTB 05	D2	4B07
DINB 06		5B08		DOUTB 06		4B08
DINB 07		5B09		DOUTB 07		4B09
Ground D2		5B10		Ground D2		4B10
MCLB1N		5B11		MCLB2N		4B11
Ground(MCL,OK)		5B12		Ground(MCL,OK)		4B12
CALIB		5B13		CALOB		4B13

SECTION 1 – INTERFACING

PART 3 INPUT/OUTPUT PROGRAMMING

Data transfer between input/output devices and the central processor is controlled by device control units each of which may have one or several devices attached to it, depending on the type of device.

Table 1.1 shows the number of devices which can be attached to a control unit; the last two columns indicate with an X the channels to which the control units can be connected.

Theoretically, any device, apart from very fast devices such as discs and magnetic tapes, can be connected to any channel. The standard *system software* is the limiting factor.

Control Unit for:	devices per c.u.	Prog. channel	I/O Processor channel
operator's typewriter	1	X	
punched tape reader	1	X	
tape punch	1	X	
line printer	1		X
magnetic tape	4		X
cassette tape	3		X
moving head disc	2		X
card reader	1		X

Table 1.1 Connection of Peripheral Devices

Each control unit is attached to the central processor by an interrupt line, address lines, and other signal lines which are used by the computer to determine whether a data transfer can be performed.

Data are transferred between the devices and the central processor via Programmed Channel – where each word or character must be programmed or I/O Processor Channel – where only blocks of data to be transferred are programmed.

PERIPHERAL DEVICES

The *standard software* of the P852M, P856M, and P857M handles the following devices:

- I/O typewriter
- Punched tape equipment
- Punched card reader
- Magnetic tape equipment (including cassette tape)
- Disc equipment
- Line printing equipment
- Display equipment

BASIC INSTRUCTION SEQUENCE

The various commands are used as follows:

- *TST (Test Status):* can be used before starting a data transfer (i.e. before CIO) to check whether the device control unit is in the busy state;

- *CIO (Control I/O):* used to start or stop a device and to perform control operations on peripherals.

- *INR (INput to Register):* used to transfer a character or word from a device control unit connected to the programmed channel to a specified register of the central processor;

- *OTR (OuTput from Register):* used to transfer a character or word from a specified register to the buffer of a device control unit, connected to the programmed channel.

- *SST (Send Status):* used to get a status word from the device control unit.

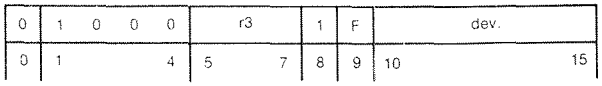
A branch instruction should be used after each I/O command to check whether it has been accepted. The following conditions are indicated by the condition register:

CR = 0: command accepted (control unit ready);
1: command refused (control unit busy or command not compatible with state of control unit);
3: device address unknown.

Notes

Notes

CIO **Control Input/Output** CIO



Syntax
[< ident >]␣CIO␣ < r3 > ,[0|1], < dev >

The I/O instruction has the following functions:

– Start an I/O operation on a peripheral device (F = 1).
– Stop a data transfer or reset the state of a peripheral device control unit (F = 0).

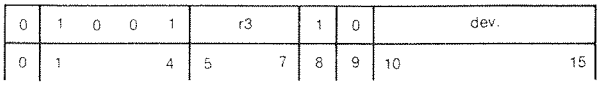
During execution < dev > field is sent, via the GP bus, to the peripheral device control unit. The contents of the 16-bit register specified by r3 are also sent on the GP bus to provide further information for some sophisticated control units (see chapter 3, under "Special device handling"). A start I/O command is not accepted if the device address is unknown or if the corresponding device is busy. Halt I/O is always accepted. The control unit will switch to the wait state after termination of data transfer with the peripheral device.

Execution Time = M + 1,4 μs + I/O
where M = Memory cycle time
and I/O = 0.6 to 6.4 μs (Asynchronous response time of the device control unit).

Condition Register
CR = 0 command accepted
1 command not accepted
3 device address unknown

Remark
r3 ≠ 0 – from A1 to A7

TST **Test STatus** TST



Syntax
[< ident >]␣TST␣ < r3 > , < dev >

This instruction may be used before starting any I/O operation to test if the device control unit is in the *inactive* state. It is always accepted by the control unit.

During the execution of the TST instruction a status word is sent from the control unit to the register specified by r3.
A one in position 15 of r3 indicates that the control unit is busy. Other bits are not significant.

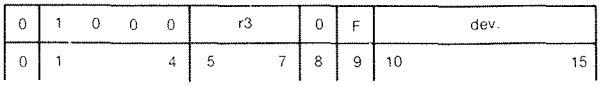
Execution Time = M + 1,3 μs + I/O

Condition Register
CR = 0 command accepted
3 device address unknown

Remark
r3 ≠ 0 — from A1 to A7

Note: In the *inactive* state the device control unit can be used by other programs for I/O exchange (CIO start will be accepted).

OTR **OuTput from Register** **OTR**



Syntax
[< ident >]␣OTR␣ < r3 > ,[0|1], < dev >

If OTR is accepted a data character or word from the register specified by r3 is transferred to the device, connected to the programmed channel. According to the type of device the lower bits (right place) of r3 or the whole 16 bits of r3 are taken into account by the control unit.

During execution, "F" and < dev > field are sent to the device via the GP bus. "F" may be used to specify a particular output function (e.g. binary or ASCII).

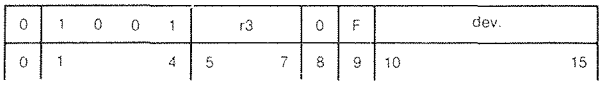
OTR is not accepted if the device control unit is not in the *exchange* state.

Execution Time = M + 1,4 μs + I/O

Condition Register
CR = 0 command accepted
1 command not accepted
3 device address unknown

Remark
r3 ≠ 0 — from A1 to A7

INR **INput to Register** INR



Syntax
[< ident >]␣INR␣ < r3 > ,[0|1], < dev >

If the I/O instruction is accepted a data word or character is transferred from the device connected to the programmed channel to the register specified by r3. Acoording to the type of the device the lower bits (right placed) or the 16 bits are significant. In the first case the higher bits of r3 are reset to zero. During execution "F" and < dev > fields are sent to the device via the GP bus. F bit may be used to specify a particular input function. This I/O instruction is not accepted if the device control unit is not in *exchange* state. After a not accepted INR instruction the contents of the r3-register are not significant.

Execution Time = M + 1,3 μs + I/O

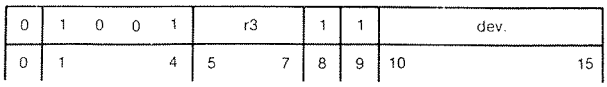
Condition Register
CR = 0 command accepted
1 command not accepted
3 device address unknown

Remark
r3 ≠ 0 — from A1 to A7

SST

Send Status

SST



Syntax
[< ident >]␣SST␣ < r3 > , < dev >

If SST is accepted a status word is transferred from the device to the register specified by r3. The following bits, if significant for the device concerned, have fixed positions:

15: not operable
14: throughput error
13: data fault
12: incorrect length
11: program error

A device does not accept an SST instruction if it is not in *exchange* status state.

Echange status state occurs when:

— an error is detected by the device (not operable, throughput error, data fault, incorrect length etc.).

— transfers are terminated (when the device is connected to the I/O processor).

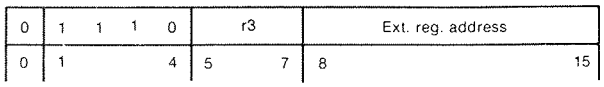
— A CIO halt is executed (when the device is connected to the programmed channel).

Execution Time = M + 1,3 μs + I/O

Condition Register
CR = 0 command accepted
1 command not accepted
3 device address unknown

Remarks
r3 ≠ 0 — from A1 to A7.

WER	Write External Register	WER



Syntax
[< ident >]␣WER␣ < r3 > , < address >

This I/O instruction is used to transfer control words to any external working register. In this application it is used to transfer the control words to the working registers of an I/O processor subchannel.

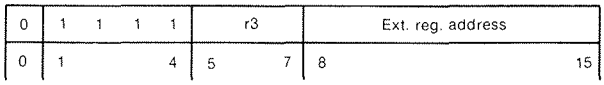
During execution the contents of the 16-bit register specified by r3, previously loaded with a control word are sent on the GP bus to the external working register of the I/O processor specified by the 8 right bits of the instruction.

Execution Time = M + 0,9 μs + I/O

Condition Register
Unchanged

Remark
< r3 > must be ≠ 0.

RER **Read External Register** **RER**



Syntax
[< ident >]␣RER␣ < r3 > , < address >

This I/O instruction is used to transfer the contents of an external register whose address is specified by the 8 right bits of the instruction to the 16-bit register specified by r3. The contents of the external register remain unchanged.

Execution Time = M + 0,9 µs + I/O

Condition Register
Contents of data bus 6 and 7 copied into CR.

Remark
< r3 > must be ≠ 0.

Input/output routines can be written either as a part of the main program, in which case processing speed is limited by the speed of the device (wait mode) or in separate I/O routines which will, on request, interrupt the main program.

In the latter case the main program, having activated the device by CIO start, can continue with other processing while mechanical actions are being performed by the device, and only when a character or word is ready for transfer will there be an interrupt from the device control unit.

INITIATION OF A DATA EXCHANGE

The Test Status instruction (TST) may be used before starting an I/O operation, to test whether the device control unit is in the inactive state. The TST instruction causes a status word to be transferred to a specified register; if bit 15 of this status word is 0 the control unit is in the inactive state; if 1, it is busy. The sequence is shown in Figure 3.1.

It is also possible when using the programmed channel to make a direct attempt to start, without using TST. In this case the readiness of the device control unit is tested by using a conditional branch after the CIO instruction: the following states of the condition register are relevant:

- CR = 0: control unit ready: command accepted.
- CR = 1: control unit busy: command not accepted.
- CR = 3: device address unknown.

The sequence is shown in Figure 3.2.

DATA TRANSFER IN WAIT MODE (PROGRAMMED CHANNEL)

Once it has been ascertained that the device is ready, the data transfer can be started. A test must be performed after each character or word transfer, to see whether the command is accepted. If the device is still in the *ready* state, the number of characters or words transferred is counted and, depending on the result of the count, either the transfer is continued or a CIO instruction in *halt* mode is issued. If the command is not accepted the transfer may be complete (after CIO-Halt) or the device may still be transferring a character or word. In either case, an SST instruction is given the result of which is followed by an exit routine or a loop through the exchange procedure, depending whether SST is accepted or not. The sequence is shown in Figure 3.3.

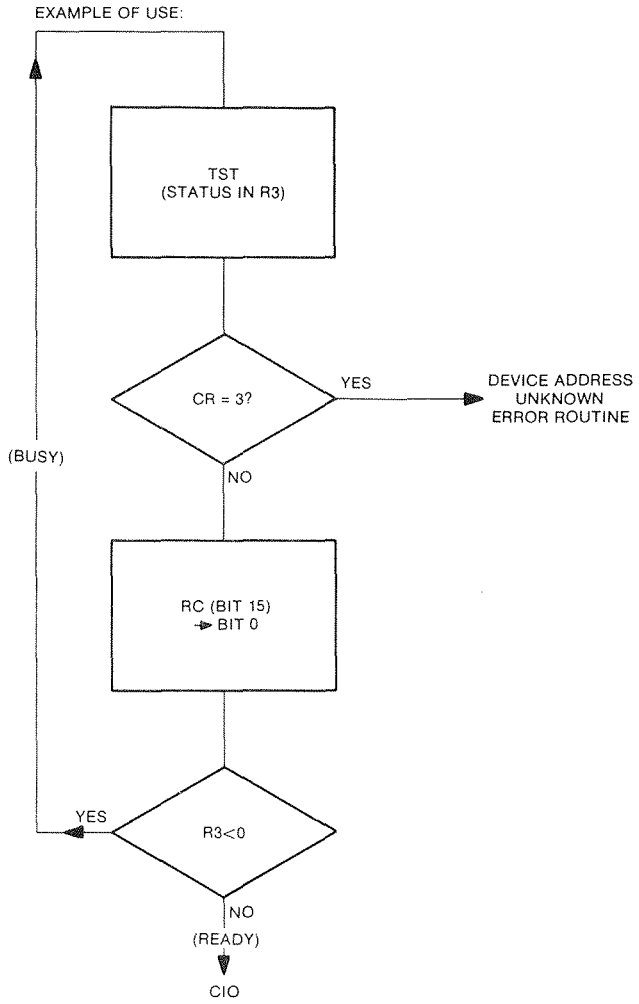


Figure 3.1 Data exchange initiation — with TST

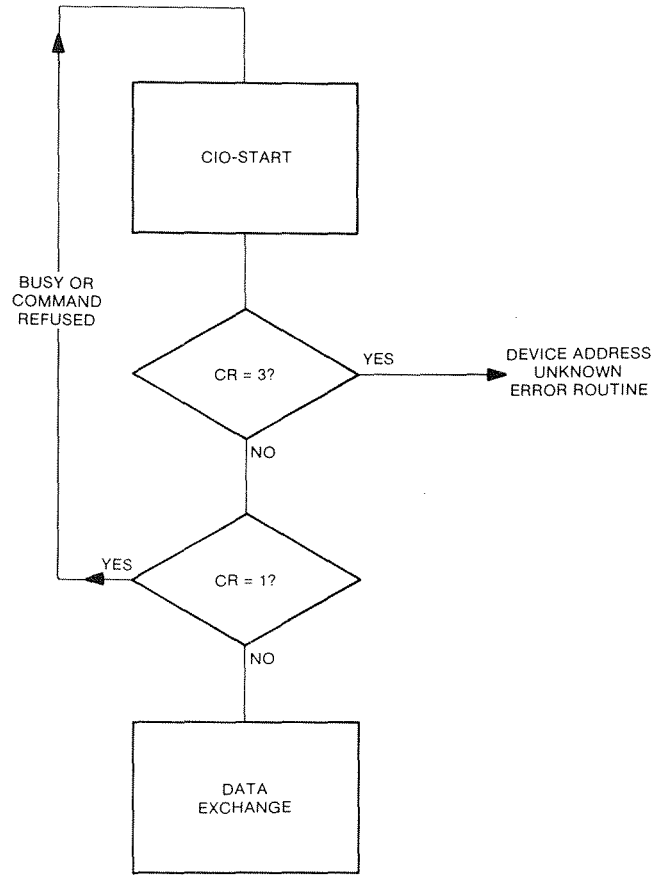


Figure 3.2 Data exchange initiation — without TST

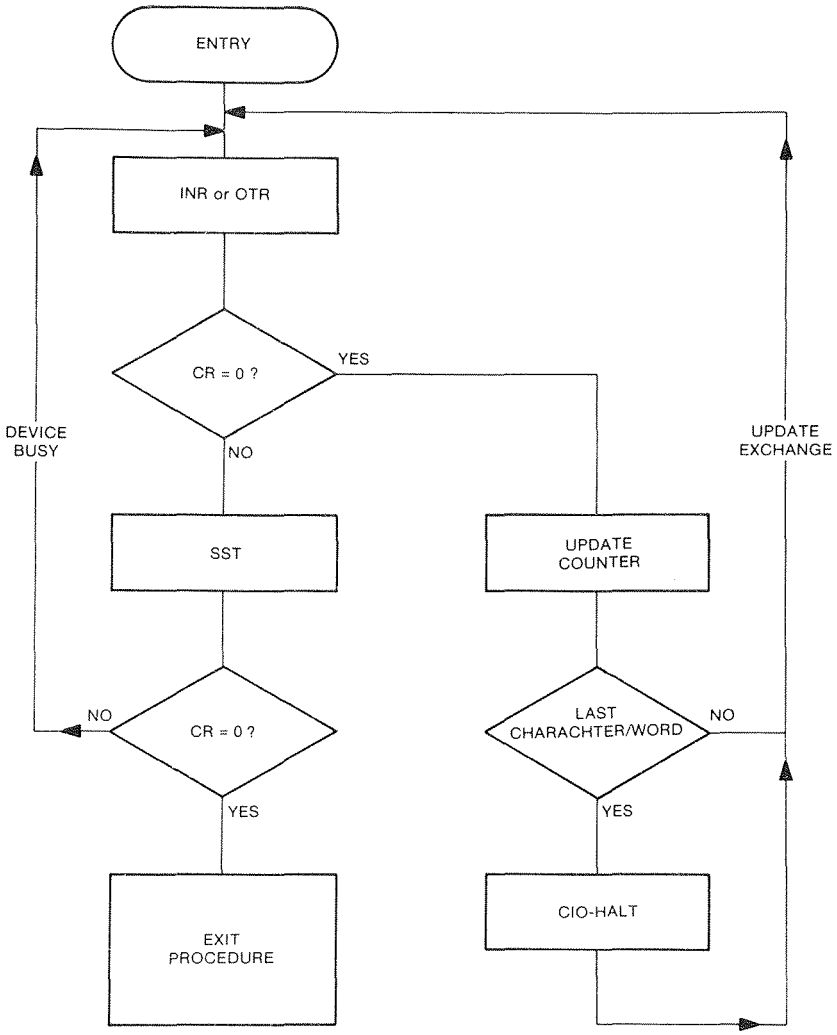


Figure 3.3 Data transfer in wait mode (programmed channel)

DATA TRANSFER IN INTERRUPT MODE (PROGRAMMED CHANNEL)

If the I/O routine is written as an interrupt routine, it can be initialized in either of the two ways described earlier in this chapter — i.e. with or without the TST instruction. The sequence is shown in Figure 3.4.

After this procedure the interrupted program can be processed: it will be interrupted, each time a data item has to be transferred, by the I/O exchange procedure, which is of higher priority.

DATA TRANSFER WAIT AND INTERRUPT MODES (I/O PROCESSOR CHANNEL)

Wait Mode

Data transfers in this mode, when using the I/O Processor, follow the sequence shown in Figure 3.5.

Interrupt Mode

In this mode the interrupt occurs only when the devices are in the exchange status state i.e. when the transfer is completed or when an error occurs. The sequence is shown in Figure 3.6.

SPECIAL DEVICE HANDLING

The instruction sequence outlined in the foregoing, varies slightly according to the *standard device.*

I/O Typewriter

The I/O typewriter must be enabled in input or output mode. For this purpose the CIO-start instruction must specify one of the registers 1 to 7, bit 15 of which must contain:

— 0: for output mode:
— 1: for input mode.

For the punched tape reader of the I/O typewriter the data stream must finish with X-off.

For the tape punch of the I/O typewriter, the data stream must start with *tape-on* and end with *tape-off.*

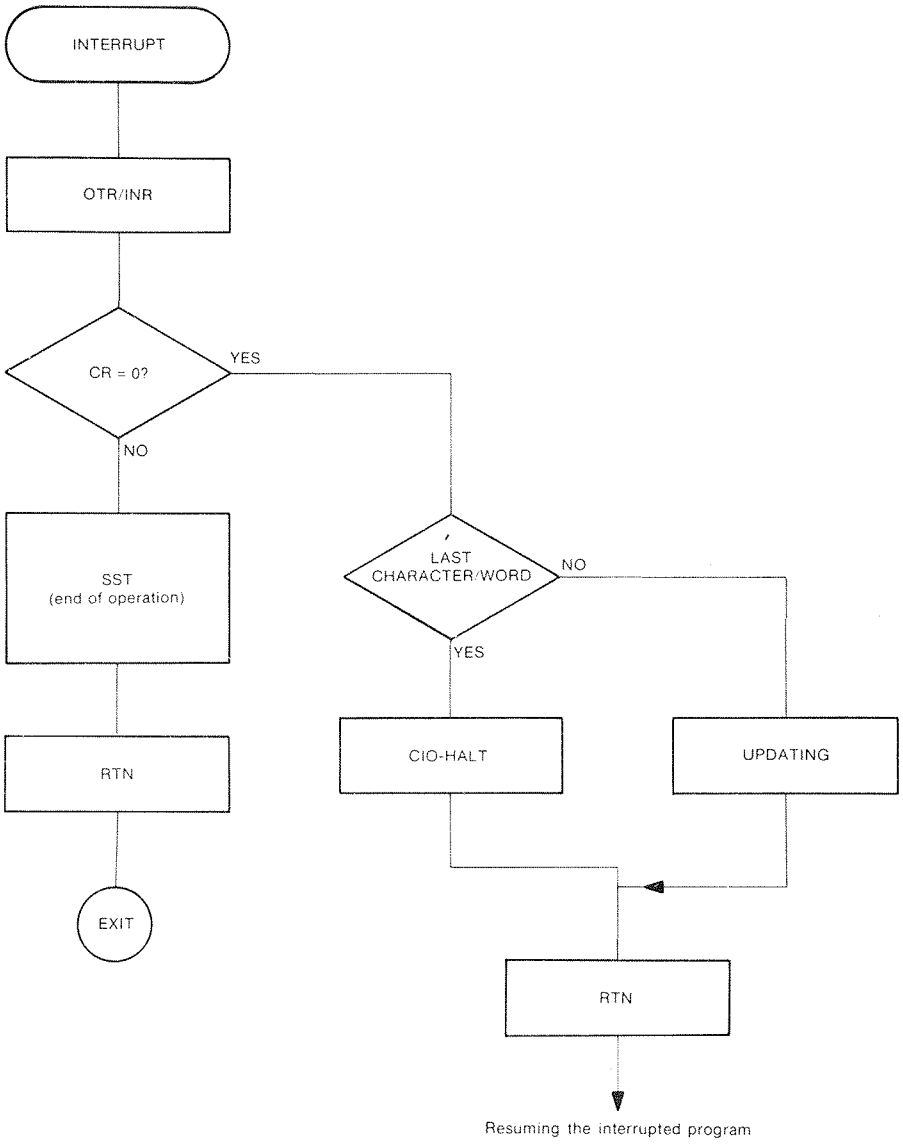


Figure 3.4 Data transfer in interrupt mode (programmed channel)

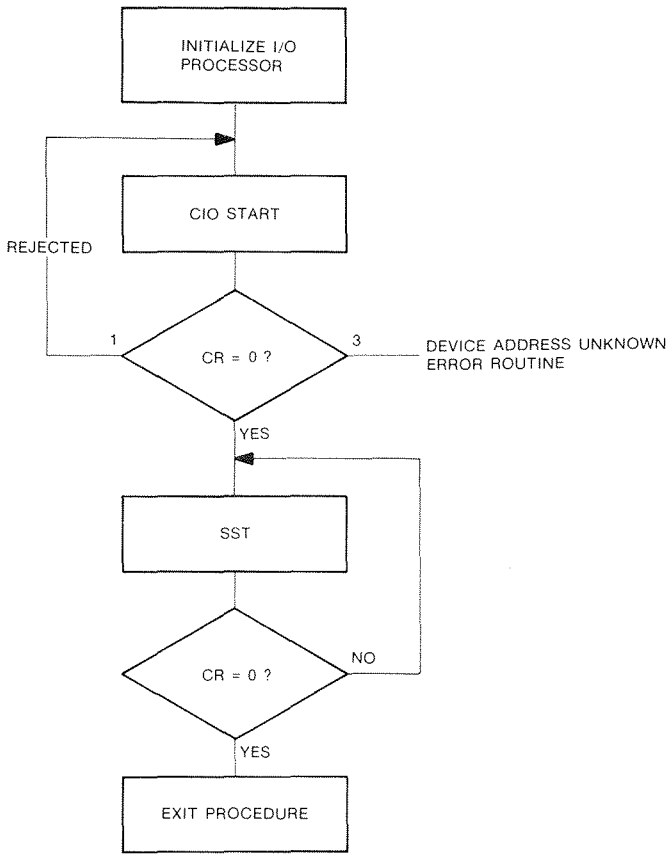


Figure 3.5 Data transfer in wait mode (I/O processor)

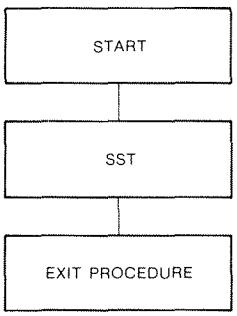


Figure 3.6 Data transfer in interrupt mode (I/O processor)

High-Speed Punched Tape Reader
If it is desired to process each character as it is input, the high-speed tape reader must be stopped between characters; i.e. a CIO-Start-INR-CIO Stop-SST loop may be performed for each character when the processing takes such an amount of time that throughput errors might occur.

Magnetic Cassette Tape
The TST command indicates not only whether the control unit is ready, but also if the cassette is rewinding.

– bit 15 of r3 register = 1: rewinding or CU busy
– bit 15 of r3 register = 0: CU and cassette inactive.

The contents of the register specified in the CIO command indicate the type of operation:

Setting of r3-register bits 12-15	Operation required
0000	Lock/unlock
0001	Erase forward
0010	Backwards space block
0101	Write a block forward
0111	Read a block forward
1000	Rewind at fast speed
1001	Write tape mark
1010	Search tape mark backward
1011	Search tape mark forward

Table 3.1 Cassette Tape Operation

The CIO extensions are used to read and write records when the cassette unit is connected via the I/O processor (as will be the case when standard system software is used). If the Programmed Channel is used (with special system software) characters are transferred singly by INR and OTR commands.

Moving Head Disc Unit
The contents of the register in the CIO command indicate the type of operation.

Setting of r3 register bits	Operation required
bits 4-12 Define cylinder number bit 13: Dont care bit 14 = 1, bit 15 = 0	seek
bits 14-15: 11	seek to zero
bits 9-13: sector address from 0-31. bits 14-15: 01	write a sector
bits 9-13: sector address from 0-31. bits 14 = 0, bit 15 = 0.	read a sector

Table 3.2 Moving Head Disc Operation

CONTROL UNIT/DEVICE ADDRESSING

All peripheral units and other external devices for the input or output of data are connected to the GP bus, through a control unit.
Each control unit must have a unique address decoded from the address lines of the GP bus and corresponding to the I/O command.

If more than one device can be connected to a control unit, which is the case for magnetic tape and disc, separate bits are used to address the device. As an example, the following table shows addresses for control units and devices.

Table 3.3 *Example* of C.U./Device Addresses

I/O command	device address						PROG. CH.	I/O PROCESSOR	device
9	10					15			
	0	1	0	0	0	0	x		I/O typewriter
	1	0	0	0	0	0	x		punched tape reader
	1	1	0	0	0	0	x		tape punch
	0	0	0	0	0	1		x	M.H. disc no. 1
	0	1	0	0	0	1		x	M.H. disc no. 2
	1	0	0	0	0	1		x	M.H. disc no. 3
	1	1	0	0	0	1		x	M.H. disc no. 4
	0	0	0	0	1	0		x	M.H. disc no. 1
	0	1	0	0	1	0		x	M.H. disc no. 2
	1	0	0	0	1	0		x	M.H. disc no. 3
	1	1	0	0	1	0		x	M.H. disc no. 4
	0	0	0	0	1	1		x	M.H. disc no. 1
	0	1	0	0	1	1		x	M.H. disc no. 2
	1	0	0	0	1	1		x	M.H. disc no. 3
	1	1	0	0	1	1		x	M.H. disc no. 4
	0	0	0	1	0	0		x	Mag tape no. 1
	0	1	0	1	0	0		x	Mag tape no. 2
	1	0	0	1	0	0		x	Mag tape no. 3
	1	1	0	1	0	0		x	Mag tape no. 4
	0	0	0	1	0	1	x	x	Cass tape no. 1
	0	1	0	1	0	1	x	x	Cass tape no. 2
	1	0	0	1	0	1	x	x	Cass tape no. 3
	1	1	0	1	0	1	x		any control unit
	0	0	0	1	1	0		x	Card reader
	0	1	0	1	1	0	x		AMA
	1	0	0	1	1	0	x		AMA
	1	1	0	1	1	0	x		AMA

Table 3.3 continued

I/O command	device address	PROG. CH.	I/O PROCESSOR	device
9	10 15			
	0 0 0 1 1 1		x	Line printer
	0 1 0 1 1 1			
	1 0 0 1 1 1			
	1 1 0 1 1 1			
	0 0 1 0 0 0			SLCU 2/4
	0 1 1 0 0 0			
	1 0 1 0 0 0			
	1 1 1 0 0 0			
	0 0 1 0 0 1			SLCU 2/4
	0 1 1 0 0 1			
	1 0 1 0 0 1			
	1 1 1 0 0 1			
	0 0 1 0 1 0			SLCU 2/4
	0 1 1 0 1 0			
	1 0 1 0 1 0			
	1 1 1 0 1 0			
	0 0 1 0 1 1			SLCU 2/4
	0 1 1 0 1 1			
	1 0 1 0 1 1			
	1 1 1 0 1 1			
	0 0 1 1 0 0			SLCU 2/4
	0 1 1 1 0 0			
	1 0 1 1 0 0			
	1 1 1 1 0 0			
	0 0 1 1 0 1			
	0 1 1 1 0 1			
	1 0 1 1 0 1			
	1 1 1 1 0 1			
	0 0 1 1 1 0			adaptor
	0 1 1 1 1 0			
	1 0 1 1 1 0			
	1 1 1 1 1 0			
	0 0 1 1 1 1			adaptor
	0 1 1 1 1 1			
	1 0 1 1 1 1			
	1 1 1 1 1 1			

4 Coding of Input/Output Drivers

Input/output operations involve a number of monitor modules, aside from the Device Work Table (DWT) and File Code Table (FCT) explained in Vol I of Programmer's Guides 1 and 2 (publication numbers 5122 991 2732X and 5122 991 2737X

The most important of these are the I/O drivers, one for each type of peripheral.
Every I/O monitor request is handled by a driver. The user can, if he wishes, write his own drivers and insert them into the system. Such user-written drivers must interface with certain tables and modules which make up the I/O system of the monitor. The necessary information which must be taken into account is described in the following paragraphs.

Three tables are used by the I/O system to know the necessary details about the peripheral devices used:

- *File Code Table:* which enables the user to assign a file code (a logical number) to a device and use this file code in programming.

- *Device Work Table:* contains all parameters about the peripherals which are necessary for the I/O system to know.

- *Control Unit Status Table:* one for each device control unit. This is a free format table of unspecified length in which the user can put information for use by his I/O driver. The I/O system knows one word of this table (referred to by word DWT + 32) which contains the status of the control unit. The first bit of this status word is reset to 0 (= busy) as soon as an I/O operation for a device is started and set to 1 when it is terminated (= free).

INPUT/OUTPUT SYSTEM

The I/O system can be divided into four main parts:

- The I/O request module (M : IORM)
- Driver
- End of I/O module (ENDIO)
- Service routines (COMIO and M:RETR).

M:IORM
When the user has given an I/O monitor request (LKM 1), this module will receive control, via the I:LKM module and T:LKM table. First this module

- checks the validity of the request
- increments the event count (PCT)
- computes the Device Work Table address
- checks whether the device control unit is busy or free (via DWT + 32). If the unit is busy, the user is put in wait.
- checks whether the device is busy or not. If it is busy, the user is put in wait.

The M:IORM sets some parameters in the DWT and in the control unit status table:

- control unit is set to busy (bit 0 = 1)
- any scheduled label parameters are set in DWT + 28, DWT + 30. DWT + 24 and DWT + 26 are set to zero
- the user Event Control Block is initialized, i.e. the left character of ECB 0 and ECB + 8 (status) are set to zero
- the I/O order (A7) is analyzed which may result in initialization of some DWT location.

At the end of this process a branch is made to the I/O driver concerned, the address of which is found in DWT + 6.

Driver
On entry into a driver, these registers must contain the following parameters:

A7: User I/O order
A8: User ECB address
A6: DWT address.

The DWT must contain:

Words 0 to 8: not modified
Word 10: User ECB address
Word 12: User buffer address
Word 14: Requested length (may be non-significant)
Word 16: Zero
Word 18: I/O order (without wait or entry bit; may be non-significant, e.g. skip orders)
Word 20: Retry bit
Word 22: Not significant.

The first part of the I/O driver performs the I/O initialization. An Exit to the C:WAIT module is made with the user order (with wait bit!) in A7 and the user ECB address in A8.

The second part of the driver is made up by the interrupt routine:

- interrupt sequence generated by SYSGEN:

  for single-card control units:

```
STR   A1,   A15
STR   A2,   A15
 |     |     |
 |     |     |
STR   A8,   A15
LDKL  A6,   DWT address
```

  for multi-card control units: e.g. cassette and magnetic tape

```
STR   A1,   A15
STR   A2,   A15
 |     |     |
 |     |     |
STR   A8,   A15
LDKL  A6,   controller status
            address
```

- the exit from the interrupt routine is made to R:TURN (in ENDIO module) if the I/O is not yet finished or to R:TUR4 (in ENDIO) if it is finished. The exit to R:TUR4 must be made with A2 containing the I/O status and A6 containing the DWT address.

- for Retry procedures a branch must be made:
  ABL M:RETR
  In this case, the calling sequence is:
  A6: DWT address
  A2: Status to be printed
  A1: Retry flag (0 if retry)[1])
  A3: Hardware order (register contents with which CIO start will be given.)

Note: The I/O processor control words must have been reinitialized.

ENDIO
The calling sequence to be used by user I/O drivers for this module is:

A6: DWT address
A2: I/O software status
ABL R:TUR4

---

[1]) For BRTM, the RETRY facility is not available, it is only used to obtain the status, i.e. A1 and A3 do not apply.

Moreover, ENDIO must find the following parameters set in the DWT:

- effective length (word 16)
- ECB address (word 10)
- scheduled label parameters (words 28, 30).

Then the following functions are performed by the ENDIO module:

- control unit status is updated (bit 0 is set to 0: free)
- the event count is decremented
- the event character in the user ECB is updated (0 = 0)
- the effective length is set in the user ECB (word 3)
- the software status is set in the user ECB (word 4)
- a branch is made to the dispatcher (M:DISP).

COMIO
This routine has two functions: executing I/O hardware instructions and building the I/O processor control words and putting a requesting program in *wait* state. Below the calling sequences for the functions are listed. (Note, that in these sequences the instructions INH-STR-ABL must be given in the order specified).

- CIO Start:

```
A6: DWT address
A2: hardware address
A3: return address
INH
STR     A1,     A15
STR     A2,     A15
 |       |       |
 |       |       |
STR     A8,     A15
ABL S:TIO
Return:         - condition register set
                - inhibit mode.
```

- CIO Stop:

```
A6: DWT address
A3: return address
INH
STR     A1,     A15
STR     A2,     A15
 |       |       |
 |       |       |
STR     A8,     A15
ABL H:LTIO
Return:         - condition register set
                - inhibit mode.
```

— OTR:

A6: DWT address
A3: return address
A1: word to be output
INH
STR A1, A15
STR A2, A15
| | |
| | |
STR A8, A15
ABL O:TRIO
Return: — condition register set
— inhibit mode.

— INR:

A6: DWT address
A3: return address
INH
STR A1, A15
STR A2, A15
| | |
| | |
STR A8, A15
ABL I:NRIO
Return: — condition register set
— inhibit mode
— word or character to be input: in A1.

— SST:

A6: DWT address
A3: return address
INH
STR A1, A15
STR A2, A15
| | |
| | |
STR A8, A15
ABL S:SST
Return: — condition register set
— inhibit mode
— status in A2.

— the control words loaded into the I/O processor are built as follows:

```
A1, A2: two-word I/O processor contents
A6: DWT address
A3: return address
INH
STR     A1,     A15
STR     A2,     A15
 |       |       |
 |       |       |
STR     A8,     A15
ABL M:TEX
```

The requesting program is put in *wait* state, if this is necessary, as follows:

```
A7: User I/O order
A8: User ECB address
ABL C:WAIT
```

Note: Any system programs, which use an I/O monitor request must do so without using the wait bit in the I/O order.

M:RETR

This module is involved when a retry has to be made for an I/O operation. The calling sequence is:

A6: DWT address
A2: status to be printed
A1: retry flag (= 0, if the operation must be retried)
A3: I/O request (hardware order for CIO start), only for BOM and DOM
ABL M:RETR
(The following applies to the BOM and DOM only, except that for BRTM/DRTM, status is set to /8000 and a branch is made to ENDIO).

Note: The control words must already have been initialized by the calling I/O driver.

The M:RETR module now puts the request in the DWT and sends the message PU, DNXX, STAT [,RY] to the typewriter.
DN is the device name
XX is the device address
STAT is the status
RY, if specified, means that the operator may give a retry message on the typewriter.
The operator may then type in RY or RD (release device):

— If he types RD, the status is set to /8000 and a branch is made to ENDIO.
— If he types RY, the following actions must be taken:

Compute and execute CIO instruction.
Reset buffer address in DWT + 12.
Reset effective length to zero in DWT + 16.
Reset DWT + 24 and DWT + 26 to zero.
Reset effective length and status in user ECB to zero.
Branch to return address of control panel routine.

EXAMPLE OF AN I/O PROGRAM IN INTERRUPT MODE

First, words are reserved to store parameters.
The "PHILIPS-ELECTROLOGICA" "APELDOORN" is written into an area labeled BUFFER, 36 characters long. (CR-LF are given after "PHIL-EL." and after "APELDOORN").
Then, the main program:

- Stack pointer is set in A15.
- The address (INTY) of the interrupt routine is loaded into A1 and stored in interrupt location /0C
- The buffer length is loaded into A1 (36).
- The buffer address is loaded into A2 (BUFFER).
- Zeroes are loaded into A3 and A4.
- These 4 parameters are stored respectively in the reserved word BULEN, BUAD, ACT and READY.
- The ASR is started for output.

Now, the interrupt routine:
Contents of A1 to A4 are stored in stack (A15).
The parameters 36, BUFFER 0, are loaded into A1, A2 and A3. One character of the buffer is loaded and output on the Teletype. If it is not accepted: SST to test the status of the typewriter. If accepted the ACTual length (A3) is incremented by 1 and compared with A1, until it is 36. Then all characters have been output. If not, return is made to main program. If output is ready: stop ASR.

```
00000                          IDENT    OUTINT
00001                  *
00002                  *
00003                  * OUTPUT PROGRAM IN INTERRUPT MODE
00004                  * INTERRUPT ADDRESS IS /0C
00005                  *
00006                  TY        EQU      /10
00007  0000                      RES      /100
00008  0200            BULEN     RES      1                    BUFFER LENGTH
00009  0202            BUAD      RES      1                    BUFFER ADDRESS
00010  0204            ACT       RES      1                    ACTUAL LENGTH
00011  0206            READY     RES      1                    READY AND ERROR BIT
00012  0208  5048      BUFFER    DATA     'PHILIPS-ELECTROLOGICA'
       020A  494C
       020C  4950
       020E  5320
       0210  454C
       0212  4543
       0214  5452
       0216  4F4C
       0218  4F47
       021A  4943
       021C  4120
00013  021E  0D0A                DATA     /0D0A
00014  0220  4150                DATA     'APELDOORN'
       0222  454C
       0224  444F
       0226  4F52
       0228  4E20
00015  022A  0D0A                DATA     /0D0A
```

```
00016                          *
00017                          * MAIN PROGRAM
00018                          *
00019  022C  87A0       STOUT  LDKL   A15,BULEN-2          STACKPOINTER
       022E  01FE  R
00020  0230  8120              LDKL   A1,INTY
       0232  0000  F
00021  0234  8141              ST     A1,/0C               FILL IN INTERRUPT ADDRESS
       0236  000C
00022  0238  0124              LDK    A1,36
00023  023A  8220              LDKL   A2,BUFFER
       023C  0208  R
00024  023E  0300              LDK    A3,0
00025  0240  0400              LDK    A4,0
00026  0242  8141              ST     A1,BULEN
       0244  0200  R
00027  0246  8241              ST     A2,BUAD
       0248  0202  R
00028  024A  8341              ST     A3,ACT               FILL IN PARAMETERS
       024C  0204  R
00029  024E  8441              ST     A4,READY
       0250  0206  R
00030  0252  0500              LDK    A5,0
00031  0254  45D0              CIO    A5,1,TY              START ASR
00032  0256  8140       WAIT   LD     A1,READY             TEST READY BIT
       0258  0206  R
00033  025A  5806              RB(0)  WAIT
00034  025C  207F              HLT
00035                          *
00036                          * TYPEWRITER INTERRUPT ROUTINE
00037                          *
00038  025E  813F       INTY   STR    A1,A15
00039  0260  823F              STR    A2,A15
00040  0262  833F              STR    A3,A15
00041  0264  843F              STR    A4,A15               A1,A2,A3,A4 TO STACK
00042  0266  8140              LD     A1,BULEN
       0268  0200  R
00043  026A  8240              LD     A2,BUAD              3 PARAMETERS
       026C  0202  R
00044  026E  8340              LD     A3,ACT
       0270  0204  R
00045  0272  920C              ADR    A2,A3
00046  0274  E428              LCR    A4,A2                LOAD CHAR.
00047  0276  4410              OTR    A4,0,TY
00048  0278  5100  F           RF(1)  SST                  NOT ACCEPTED
00049  027A  9041              IM     ACT                  UPDATE ACTLEN
       027C  0204  R
00050  027E  E90C              CWR    A1,A3                TEST IF READY
00051  0280  5100  F           RF(1)  RETURN               NOT READY
00052  0282  5700  F           RF     RETURN
00053  0284  4AD0       SST    SST    A2,TY
00054  0286  5100  F           RF(1)  ERROR
00055  0288  8120              LDKL   A1,/8000
       028A  8000
00056                          *
00057                          * SET CONTROL BITS
00058                          * BIT 0 IS READY BIT
00059                          * BIT 1 IS ERROR BIT
00060  028C  8141       SETBIT ST     A1,READY             STORE CONTROL BITS
       028E  0206  R
00061  0290  811E       RETURN LDR    A1,A15
00062  0292  821E              LDR    A2,A15
00063  0294  831E              LDR    A3,A15
00064  0296  841E              LDR    A4,A15
00065  0298  F03E              RTN    A15
00066  029A  8120       ERROR  LDKL   A1,/C000
       029C  C000
00067  029E  5F14              RB     SETBIT
00068                          END    STOUT
```

SYMBOL TABLE

TY	0010	A	BULEN	0200	R	BUAD	0202	R	ACT	0204	R
READY	0206	R	BUFFER	0208	R	STOUT	022C	R	INTY	025E	R
WAIT	0256	R	SST	0284	R	RETURN	0290	R	ERROR	029A	R
SETBIT	028C	R									

ASS.ERR. 00000

```
:EOF
PROG ELAPSED TIME:  00H-00M-00S-000MS-
```

EXAMPLE OF A PROGRAM TO READ 20 CHARACTERS VIA I/O PROCESSOR

```
00000                          IDENT    IOPROC
00001                  *
00002                  * PROGRAM TO READ 20 CHAR
00003                  * VIA I/O PROCESSOR
00004                  * DEVICE ADDR. = /9
00005                  * INTERR LOC. =  /008
00006                  * MEMORY = 16K
00007                  * I/O PROC.NUM = 001
00008                  * SUBCHAN. NUM = 001
00009  0000                    RES      3
00010                  EOS     EQU      *-2
00011  0006            READY   RES      1
00012  0008            BUF     RES      10
00013                  *
00014                  * MAIN PROGRAM
00015                  *
00016  001C  87A0      START   LDKL     A15,EOS          STACK POINTER
       001E  0004  R
00017  0020  8120              LDKL     A1,INT
       0022  0000  F
00018  0024  8141              ST       A1,/08           LOCATION /08
       0026  0008
00019  0028  0114              LDK      A1,[illegible]   CHAR.INP. 20
00020  002A  8220              LDKL     A2,B[illegible]  BEGIN OF BUFFER
       002C  0008  R
00021  002E  7112              WER      A1,[illegible]
00022  0030  7213              WER      A2,[illegible]
00023  0032  0300              LDK      [illegible]
00024  0034  8341              ST       [illegible]DY    CLEAR READY
       0036  0006  R
00025  0038  44C9              CIO      A4,9
00026  003A  8140      WAIT    LD       A1,READY
       003C  0006  R
00027  003E  5806              RB(0)    WAIT
00028  0040  207F              HLT
00029                  *
00030                  * INTERRUPT PROGRAM
00031                  *
00032  0042  813F      INT     STR      A1,A15           A1 TO STACK
00033  0044  49C9              SST      A1,/9
00034  0046  9041              IM       READY            SET READY
       0048  0006  R
00035  004A  813E              LDR*     A1,A15
00036  004C  F03E              RTN      A15
00037                          END      START
```

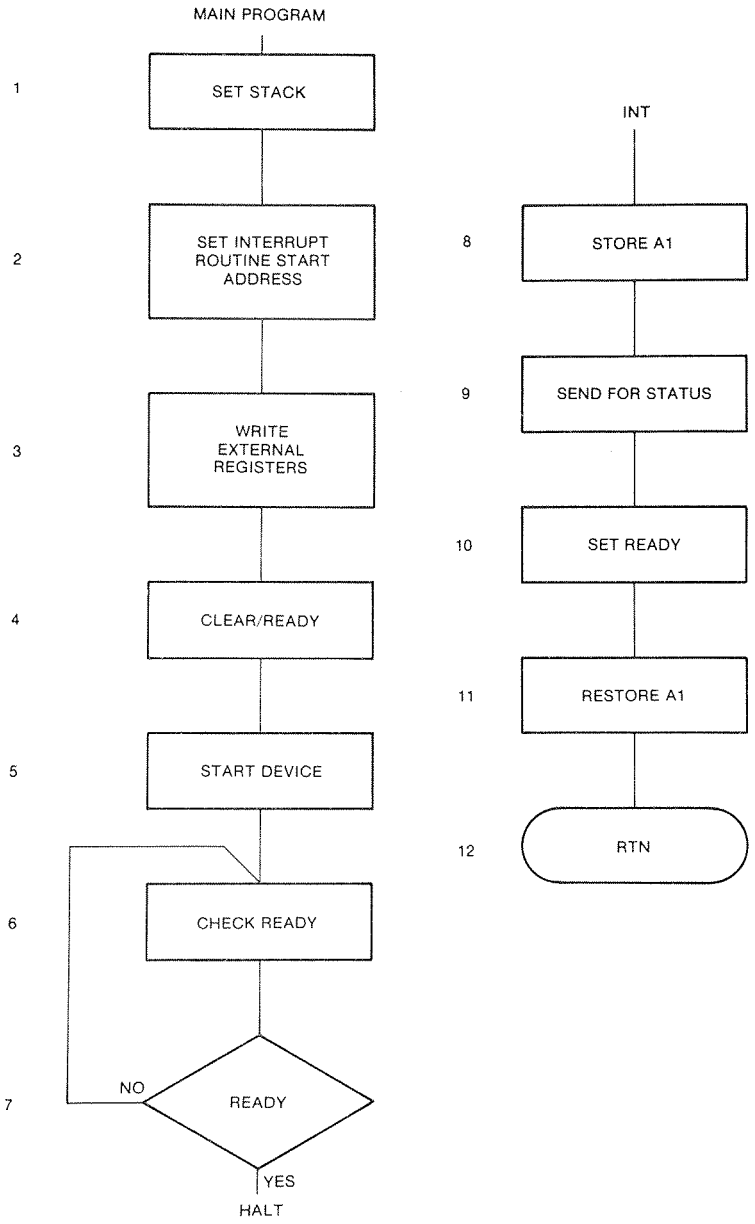
SYMBOL TABLE

EOS	0004	R	READY	0006	R	BUF	0008	R	START	001C	R
INT	0042	R	WAIT	003A	R						

ASS.ERR. 00000

```
:EOF
PROG ELAPSED TIME:  00H-00M-00S-000MS-
LSD
LABEL = OEMSEGR          DATE =  28-05-75        PACK NBR =  000     SAG
DEL IOPROC,/S
LABEL = OEMSEGR          DATE =  28-05-75        PACK NBR =  000     SAG
DEL IOPROC,/L
LABEL = OEMSEGR          DATE =  28-05-75        PACK NBR =  000     SAG
BYE
DATE     /   /     TIME 24H-60M-60S-
```

The following flowcharts and brief descriptions of each stage explain the example shown.



The initial assembly directives define the end of interrupt stack address as EOS and reserve a single location to indicate a ready condition, designated READY, and a 10 word buffer area, the start address of which is BUFF.

1. Register A15 is loaded with the end of stack address EOS.

2. The hardware interrupt location for level 4, location /0003, is loaded with the start address of the interrupt routine, designated INT.

3. The 2 I/O processor control words, are loaded with the required parameters:

   word 0 bit 0 = 0 character transfer
   1 = 0 input
   2 = 0 not used
   3 = 0 not used
   bits 4-15 = Transfer Length.
   The hexadecimal value to be loaded is therefore /0014.

   Word 1 = First Character Address.
   That is BUF.

4. The location READY is cleared to indicate a busy condition.

5. The device is started using a CIO start command.

At this point the transfer is commenced and each character transfer is initiated by a break signal from the device control unit concerned. The main program continues.

6. The state of the busy indication is checked by loading the contents of READY into register A1.

7. A conditional branch is then carried out with respect to the value in register A1, if the value is zero the program branches back in a loop to wait for a non-busy indication. The non-busy indication will occur when the value loaded into register A1, from location READY, is no longer zero, and in this case no branch occurs and the program continues sequentially with a HALT instruction.

   The interrupt routine is responsible for setting the busy indication to non zero, this should occur when the transfer is complete and the device control unit concerned raises an interrupt.

8. The interrupt routine commences by storing register A1 in the stack.

9. The device status is requested and placed into register A1.

10. The indication of busy is cleared by setting location READY equal to non zero, carried out in the main program by an Increment Memory instruction.

11. Register A1 is then restored and a return made to the main program.

Note: An I/O Processor transfer may also terminate on an error which will cause the device control unit to raise an interrupt. It is the program responsibility to decide whether or not any transfer has terminated correctly and continue accordingly. In the program example shown no errors are assumed to occur and thus no routines are shown to cope with them.