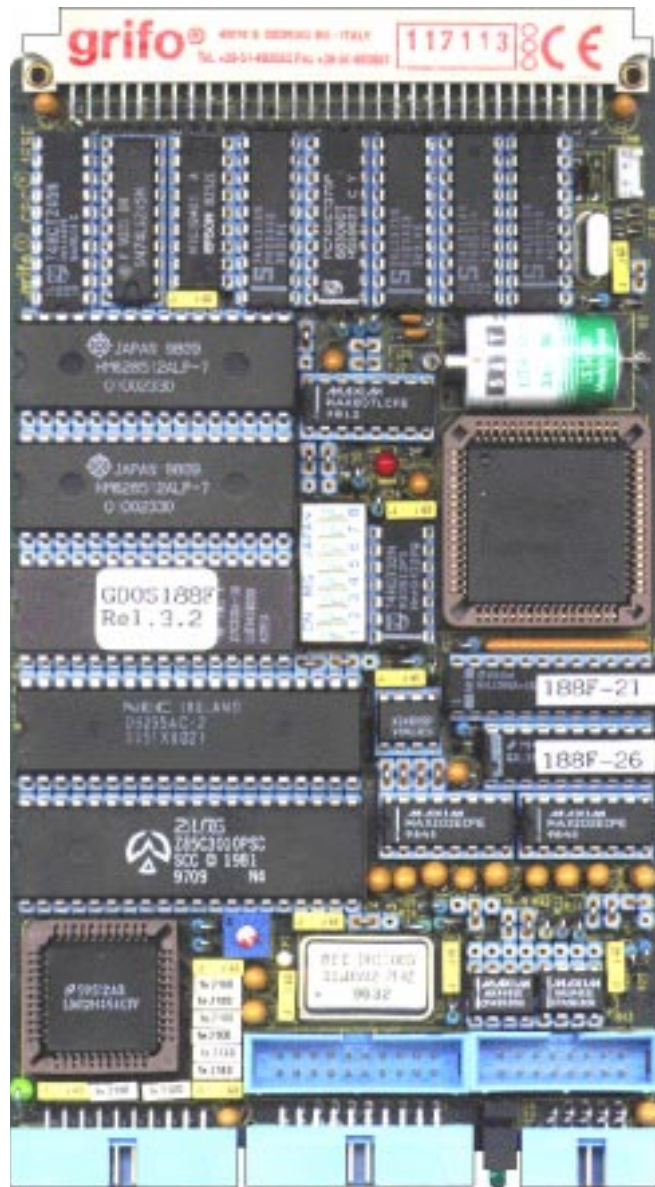


GPC[®] 188F

General Purpose Controller 80C188

TECHNICAL MANUAL



grifo[®]

ITALIAN TECHNOLOGY

Via dell' Artigiano, 8/6
40016 San Giorgio di Piano
(Bologna) ITALY

E-mail: grifo@grifo.it

<http://www.grifo.it>

<http://www.grifo.com>

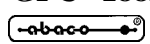
Tel. +39 051 892.052 (a.r.) FAX: +39 051 893.661



GPC[®] 188F

Edition 5.10

Rel. 12 April 2000

, GPC[®], grifo[®], are trade marks of grifo[®]

GPC[®] 188F

General Purpose Controller 80C188

TECHNICAL MANUAL

Single Euro size 100x160mm with interface to **Abaco[®] industrial BUS**; **80C188 CPU** with **20 MHz** crystal; up to **1M EPROM** or **512K FLASH EPROM** and up to **1M SRAM** with 512K write-protectable and so managed ad **RAM disk**; serial **EEPROM** up to 8K; 8 way **Dip Switch** and configuration jumpers readable by software; activity **LEDs**, placed on the front side, driven through software; **2 RS 232** serial lines, one configurable in **RS 422**, **RS 485** or **Current Loop** managed by the powerful **Z85C30** supporting **HDLC**, **SDLC** etc. protocols and software selectable baud rate, up to **115 KBaud**; **24 I/O TTL** lines, set via software: 24 managed by **PPI 82C55**; **one** 16 bits **timer** and two 16 bits **timer counters** capable to generate **PWM** signals; **8 A/D** converter lines with **Sample & Hold**, 5.5 μ s, range 0÷2.048 V and possibility to work on differential inputs (± 2.0485), 12 bits+sign, managed by **LM 12H458**. It develops more than **140.000** conversions per second, it is provided with an internal sequencer, **Self Calibration** function, **Conversion Rates** programming and it is able to check an analog input generating an **INT** when defined limits are exceeded; **power failure** circuit capable to generate interrupt; **Real Time Clock** able to up date day, month, year, week day, seconds, minutes and hours. It can be programmed to issue an **INT** at intervals defined by software; **Watch Dog** resettable by software and display through **LED**; **back up** circuit for SRAM and RTC provided of **lithium battery** and connector for external battery. The battery charge status can be acquired by software; single power supply **+5Vdc**, **386 mA**; wide range of base software and **development tools** that allow card use with only a standard PC, connected through serial line. Among these: **GDOS 188**; **PASCAL 188**; **Monitor Debugger**; **GCTR 188**; **ROM-DOS**; **HI TECH C 86**; **GET 188**; **DDS MICROC 86**; **FLASH WRITER**; etc.

grifo[®]

ITALIAN TECHNOLOGY

Via dell' Artigiano, 8/6
40016 San Giorgio di Piano
(Bologna) ITALY

E-mail: grifo@grifo.it

<http://www.grifo.it>

<http://www.grifo.com>

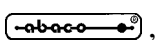
Tel. +39 051 892.052 (a.r.) FAX: +39 051 893.661



GPC[®] 188F

Edition 5.10

Rel. 12 April 2000

, GPC[®], grifo[®], are trade marks of grifo[®]

DOCUMENTATION COPYRIGHT BY grifo[®], ALL RIGHTS RESERVED

No part of this document may be reproduced, transmitted, transcribed, stored in a retrieval system, or translated into any language or computer language, in any form or by any means, either electronic, mechanical, magnetic, optical, chemical, manual, or otherwise, without the prior written consent of **grifo[®]**.

IMPORTANT

Although all the information contained herein have been carefully verified, **grifo[®]** assumes no responsibility for errors that might appear in this document, or for damage to things or persons resulting from technical errors, omission and improper use of this manual and of the related software and hardware.

grifo[®] reserves the right to change the contents and form of this document, as well as the features and specification of its products at any time, without prior notice, to obtain always the best product.

For specific informations on the components mounted on the card, please refer to the Data Book of the builder or second sources.

SYMBOLS DESCRIPTION

In the manual could appear the following symbols:




Attention: Generic danger



Attention: High voltage

Trade Marks

 , GPC[®], grifo[®] : are trade marks of grifo[®].

Other Product and Company names listed, are trade marks of their respective companies.

GENERAL INDEX

INTRODUCTION	1
CARD VERSION	1
GENERAL INFORMATIONS	2
CPU	3
REAL TIME CLOCK	3
SERIAL COMMUNICATION	3
ABACO® BUS	4
CLOCK DEVICES	4
A/D CONVERTER	4
WATCH DOG SECTION	6
MEMORY DEVICES	6
MEMORY MANAGEMENT UNIT	6
DIGITAL I/O LINES	6
TECHNICAL FEATURES	8
GENERAL FEATURES	8
PHYSICAL FEATURES	8
ELECTRIC FEATURES	9
INSTALLATION	10
CONNECTIONS	10
CN6 - EXTERNAL BACK UP BATTERY CONNECTOR	10
CN5 - CONNECTOR FOR PPI 82C55 PORT B	11
CN2 - CONNECTOR FOR PPI 82C55 PORT A AND C	12
CN3 - A/D CONVERTER INPUTS CONNECTOR	14
CN1 - RS 232 SERIAL LINES AND TIMER COUNTER CONNECTOR	16
CN4 - CONNECTOR FOR RS 422, RS 485 AND CURRENT LOOP	18
K1 - CONNECTOR FOR ABACO® BUS	24
TRIMMER AND CALIBRATION	26
TEST POINT	26
I/O CONNECTION	28
VISUAL FEEDBACK	28
DIGITAL I/O INTERFACES	29
JUMPERS	30
2 PINS JUMPERS	32
4 PINS JUMPER	32
3 PINS JUMPERS	33
5 PINS JUMPER	33
RESET AND WATCH DOG	34
BACK UP	34
POWER FAILURE	34
INTERRUPTS	35
CONFIGURATION INPUTS	35

SERIAL COMMUNICATION SELECTION	36
DMA TRANSFERS	38
MEMORY SELECTION	39
SOFTWARE DESCRIPTION	40
ADDRESSES AND MAPS	42
ON BOARD RESOURCES ALLOCATION	42
I/O ADDRESSES	43
ABACO® BUS ADDRESSES	45
MEMORY ADDRESSES	45
PERIPHERAL DEVICES SOFTWARE DESCRIPTION	48
MEMORY MANAGEMENT UNIT	48
A/D CONVERTER	49
WATCH DOG	49
SERIAL EEPROM	49
BATTERY STATUS	50
CONFIGURATION INPUTS	50
ACTIVITY LEDS	51
WRITE PROTECTION	51
DISABLE DMA REQUEST	51
REAL TIME CLOCK	52
SCC 85C30	53
PPI 82C55	61
CPU INTERNAL DEVICES	62
EXTERNAL CARDS	63
BIBLIOGRAPHY	67
APPENDIX A: ELECTRIC DIAGRAMS	A-1
APPENDIX B: ON BOARD COMPONENTS DESCRIPTION	B-1
CPU 80C188	B-1
A/D CONVERTER LM12H458	B-19
APPENDIX C: ALPHABETICAL INDEX	C-1

FIGURES INDEX

FIGURE 1: BLOCK DIAGRAM	5
FIGURE 2: CARD PHOTO	7
FIGURE 3: CN6 - EXTERNAL BACK UP BATTERY CONNECTOR	10
FIGURE 4: CN5 - CONNECTOR FOR PPI 82C55 PORT B	11
FIGURE 5: CN2 - CONNECTOR FOR PPI 82C55 PORT A AND C	12
FIGURE 6: PPI I/O LINES CONNECTION DIAGRAM	13
FIGURE 7: CN3 - A/D CONVERTER INPUTS CONNECTOR	14
FIGURE 8: A/D CONVERTER INPUT DIAGRAM	15
FIGURE 9: CN1 - RS 232 SERIAL LINES AND TIMER COUNTER CONNECTOR	16
FIGURE 10: TIMER COUNTER CONNECTION DIAGRAM	17
FIGURE 11: SERIAL COMMUNICATION CONNECTION DIAGRAM	17
FIGURE 12: CN2 - RS 422, RS 485 AND CURRENT LOOP SERIAL LINE B CONNECTOR	18
FIGURE 13: RS 232 POINT-TO-POINT CONNECTION EXAMPLE	19
FIGURE 14: RS 422 POINT-TO-POINT CONNECTION EXAMPLE	19
FIGURE 15: RS 485 POINT-TO-POINT CONNECTION EXAMPLE	19
FIGURE 16: RS 485 NETWORK CONNECTION EXAMPLE	20
FIGURE 17: 4 WIRES CURRENT LOOP POINT-TO-POINT CONNECTION EXAMPLE	21
FIGURE 18: 2 WIRES CURRENT LOOP POINT-TO-POINT CONNECTION EXAMPLE	21
FIGURE 19: PASSIVE CURRENT LOOP NETWORK CONNECTION EXAMPLE	22
FIGURE 20: COMPONENTS MAP	23
FIGURE 21: K1 - ABACO® BUS CONNECTOR	24
FIGURE 22: CONNECTORS, TRIMMER, MEMORIES, ETC. LOCATION	27
FIGURE 23: VISUAL FEEDBACK TABLE	28
FIGURE 24: JUMPERS SUMMARIZING TABLE	30
FIGURE 25: JUMPERS LOCATION	31
FIGURE 26: 2 PINS JUMPERS TABLE	32
FIGURE 27: 4 PINS JUMPER TABLE	32
FIGURE 28: 3 PINS JUMPERS TABLE	33
FIGURA 29: 5 PINS JUMPER TABLE	33
FIGURE 31: SERIAL COMMUNICATION DRIVERS LOCATION	37
FIGURE 32: MEMORY SELECTION TABLE	39
FIGURE 33: PERIPHERAL DEVICES HARDWARE CONNECTIONS	42
FIGURE 34: I/O ADDRESSES TABLE - PART 1	43
FIGURE 35: I/O ADDRESSES TABLE - PART 2	44
FIGURE 36: MEMORY ADDRESSES - PART 1	46
FIGURE 37: MEMORY ADDRESSES - PART 2	47
FIGURE 38: POSSIBLE CONECTIONS DIAGRAM	65
FIGURE A1: IAC 01 ELECTRIC DIAGRAM	A-1
FIGURE A2: KDX x24 ELECTRIC DIAGRAM	A-2
FIGURE A3: QTP 16P ELECTRIC DIAGRAM	A-3
FIGURE A4: QTP 24P ELECTRIC DIAGRAM - PART 1	A-4
FIGURE A5: QTP 24P ELECTRIC DIAGRAM - PART 2	A-5
FIGURE A6: SPA 01 ELECTRIC DIAGRAM	A-6



INTRODUCTION

The use of these devices has turned - IN EXCLUSIVE WAY - to specialized personnel.

The purpose of this handbook is to give the necessary information to the cognizant and sure use of the products. They are the result of a continual and systematic elaboration of data and technical tests saved and validated from the manufacturer, related to the inside modes of certainty and quality of the information.

The reported data are destined- IN EXCLUSIVE WAY- to specialized users, that can interact with the devices in safety conditions for the persons, for the machine and for the environment, impersonating an elementary diagnostic of breakdowns and of malfunction conditions by performing simple functional verify operations , in the height respect of the actual safety and health norms.

The informations for the installation, the assemblage, the dismantlement, the handling, the adjustment, the reparation and the contingent accessories, devices etc. installation are destined - and then executable - always and in exclusive way from specialized warned and educated personnel, or directly from the TECHNICAL AUTHORIZED ASSISTANCE, in the height respect of the manufacturer recommendations and the actual safety and health norms.

The devices can't be used outside a box. The user must always insert the cards in a container that respect the actual safety normative. The protection of this container is not threshold to the only atmospheric agents, but specially to mechanic, electric, magnetic, etc. ones.

To be on good terms with the products, is necessary guarantee legibility and conservation of the manual, also for future references. In case of deterioration or more easily for technical updates, consult the AUTHORIZED TECHNICAL ASSISTANCE directly.

To prevent problems during card utilization, it is a good practice to read carefully all the informations of this manual. After this reading, the user can use the general index and the alphabetical index, respectly at the begining and at the end of the manual, to find information in a faster and more easy way.

CARD VERSION

The present handbook is reported to the **GPC® 188F** card release **121198** and later. The validity of the bring informations is subordinate to the number of the card release. The user must always verify the correct correspondence among the two denotations. On the card the release number is present in more points both board printed diagram (serigraph) and printed circuit (for example near socket IC18 both on the component side and on the solder side).

GENERAL INFORMATION

The **GPC® 188F** card is a powerful control and managing card in the 100x160mm standard single Europe size. It relies on the powerful **Industrial ABACO® BUS** and exploits the numerous intelligent and non intelligent peripherals, available on this BUS.

The **GPC® 188F** is based on the powerful and notorious **CPU 80C188Intel**, therefore being code compatible with any **PC**, and it has considerable hardware resources available on board. The 8 lines of 13 bits high performances A/D converter are particularly interesting. Its modularity and the remarkable hardware resources allow this card to be easily used even in complex applications. Moreover this card is the right component for all the applications that require a lot of memory, in fact up to **2 M Bytes** can be mounted on board.

The **GPC® 188F** use is simplified by a wide range of software development tools based on high level languages which, in an efficient and friendly environment, allow to work at the best using a standard PC. Noteworthy among these tools the **operating systems**, an efficient **PASCAL romated compiler** or the powerful C compilers. Special care has been devoted to the difficult operation of **debug**, by programs which allow the **remote symbolic debugger** directly on the card.

- Single **Euro** size 100x160mm with interface to **Abaco® Industrial BUS**
- **20 MHz** cmos **80C188 CPU**
- Up to **1M EPROM** or up to **512K FLASH EPROM**
- Up to **1M SRAM**. A write protect circuit can be enabled for 512K SRAM obtaining a reliable **RAM disk**
- Serial **EEPROM** up to 8K.
- 8 way **Dip Switch** readable by software
- **2** activity **LEDs**, placed on the card front, driven through software
- **2 RS 232** serial lines, one configurable in **RS 422**, **RS 485** or **Current Loop** managed by the powerful **Z85C30** supporting **HDLC**, **SDLC** etc. protocols and software selectable baud rate, up to **115 KBaud**
- **24 I/O TTL** lines, set via software, managed by **PPI 82C55**
- **One** 16 bits **Timer** and **two** 16 bits **Timer Counter** capable to generate **PWM** signals
- **8 A/D** Converter lines with **Sample & Hold**, 5,5 μ s, range 0÷2.048V and possibility to work on differential inputs (\pm 2.048V), 12 bits+sign, managed by **LM 12H458**. It develops more than **140.000** conversions per second, it is provided with an internal sequencer, **DMA** data transfer, **Self Calibration** function, **Conversion Rates** programming and it is able to check an analog input generating an **INT** when defined limits are exceeded.
- **Power Failure** circuit capable to generate interrupt
- **Real Time Clock** able to up date day, month, year, week day, seconds, minutes and hours. It can be programmed to issue an **INT** at intervals defined by software
- **Watch Dog** resettable by software and display through **LED**
- **Back up** circuit for SRAM and RTC provided of **lithium battery** and connector for external battery. The battery charge status can be acquired by software.
- Single power supply **5Vdc, 386mA**
- Wide range of base software and **development tools** that allow card use with only a standard PC, connected through serial line. Among these: **GDOS 188**; **PASCAL 188**; **Monitor Debugger**; **GCTR 188**; **ROM-DOS**; **HI TECH C 86**; **GET 188**; **FLASH WRITER**; **DDS MICROC 86**; etc.

CPU

GPC® 188F board is designed to employ the **80C188** microcontroller manufactured by **INTEL**. This 16 bits CPU is code compatible with Intel 86 family so it features: an extended instructions set, high speed of execution, efficient vectored interrupts management, wide range of addressing modes and and interesting data manipulation techniques. Remarkable is the presence of these peripherals inside the CPU:

- Two 16 bits timers counter and one 16 bits timer, capable to generate pulse width modulation signals (PWM);
- Two DMA channels for high speed data transfers (DMA);
- One programmable interrupt controller (PIC) ;
- One power management section;
- One section that generates control signal for peripheral devices management,with programmable wait states generator;

For further informations about this component please refer to the manufacturer documentation, or see appendix B of this manual.

REAL TIME CLOCK

GPC® 188F has installed on-board a Real Time Clock capable of a completely autonomous management of hours, minutes, seconds, day of month, month, year and day of week. The device is completely software programmable by 16 registers mapped in the CPU I/O space and is supplied by a back-up circuitry which warrants the validity of its data in any operating condition. In addition RTC section is capable to generate periodic interrupts, for example to wake the CPU away from halt, idle or stop conditions.

SERIAL COMMUNICATION

The serial communication lines are completely software configurable for protocol and speed (from 50 to 115200 Baud) independently for both the communication lines. It is remarkable the possibility to employ SDLC and HDLC logic protocols, having so the chance to insert the **GPC® 188F** board in a network with more than one intelligent card. The settings are performed through the software programming of 85C30 registers, for further informations please refer to the paragraph "PERIPHERAL DEVICES SOFTWARE DESCRIPTION".

One of the two serial lines is always buffered with RS 232 electric protocol, while the second one is hardware configurable in fact connecting some jumpers, the User can select the electric standard interface between RS 232, RS 422, RS 485 and Current loop; for RS 422 it is also possible to specify Half Duplex or Full Duplex communication. The chapter "SERIAL COMMUNICATION SELECTION" contains a detailed description of available hardware configurations.

Normally the card is provided with two RS 232 interfaces and a different configuration must be specified when ordering.

ABACO® BUS

One of the most important features of **GPC® 188F** is its possibility to be interfaced to industrial **ABACO® BUS**. Thanks to its standard **ABACO® BUS** connector, the card can be connected to some of the numerous **grifo®** boards, both intelligent and not. For example the User can directly use cards for analog signals acquisition (A/D), cards for analog signals generation (D/A), cards for digital I/O signals management, cards with timers and counters, cards for temperature controls, etc. Through mother boards like **ABB 03** and **ABB 05** it is also possible to manage serie 3 and 4 boards, which are provided with **ABACO® I/O BUS**. So, **GPC® 188F** becomes the right component for each industrial automation systems, in fact **ABACO® BUS** makes the card easily expandable with the best price/performance ratio.

CLOCK DEVICES

On **GPC® 188F** there are three separate circuits with crystal to generate the clock signal for the microprocessor (**20 MHz**), A/D Converter section (**8 MHz**) and the timing signal for **SCC 85C30** (**11.0592 MHz**). Please remark that CPU generates its own clock signal through the external quartz dividing its frequency by two and eventually reducing it when the power management section is programmed properly. The choice of using three circuits and as many separated crystals, has the advantage to change the microprocessor working speed (when best performances are required) without additional changes in communication software or firmware. The best time performances are always obtained both for execution time and serial communication, fulfilling the User necessity.

A/D CONVERTER

The optional A/D Converter section on the **GPC® 188F** board is based on the powerful **LM12H458** peripheral can acquire 8 channels with a maximum resolution of 12 bits plus sign bit in the range $0 \div 2.048V$ or 4 channels connected to differential signals in the range $\pm 2.048 V$ in "fully differential mode".

This section is provided with Sample & Hold circuit, a fully-differential self calibrating Analog-to-Digital converter capable to perform $5.5 \mu s$ conversion time and over **140,000** conversions per second. Some other features like: internal **sequencer**, **DMA** data transfer, two different **self-calibration** modes, programmable **conversion rates**, **resolution** setting, **FIFO** to store conversion data and autonomous limits control (this means monitoring an analog input and generating and interrupt when it is above or below either of the two limits), easy remarkably its management without requiring CPU intervenes.

By software, all the section functions are programmable through 27 registers allocated in the I/O addressing space.

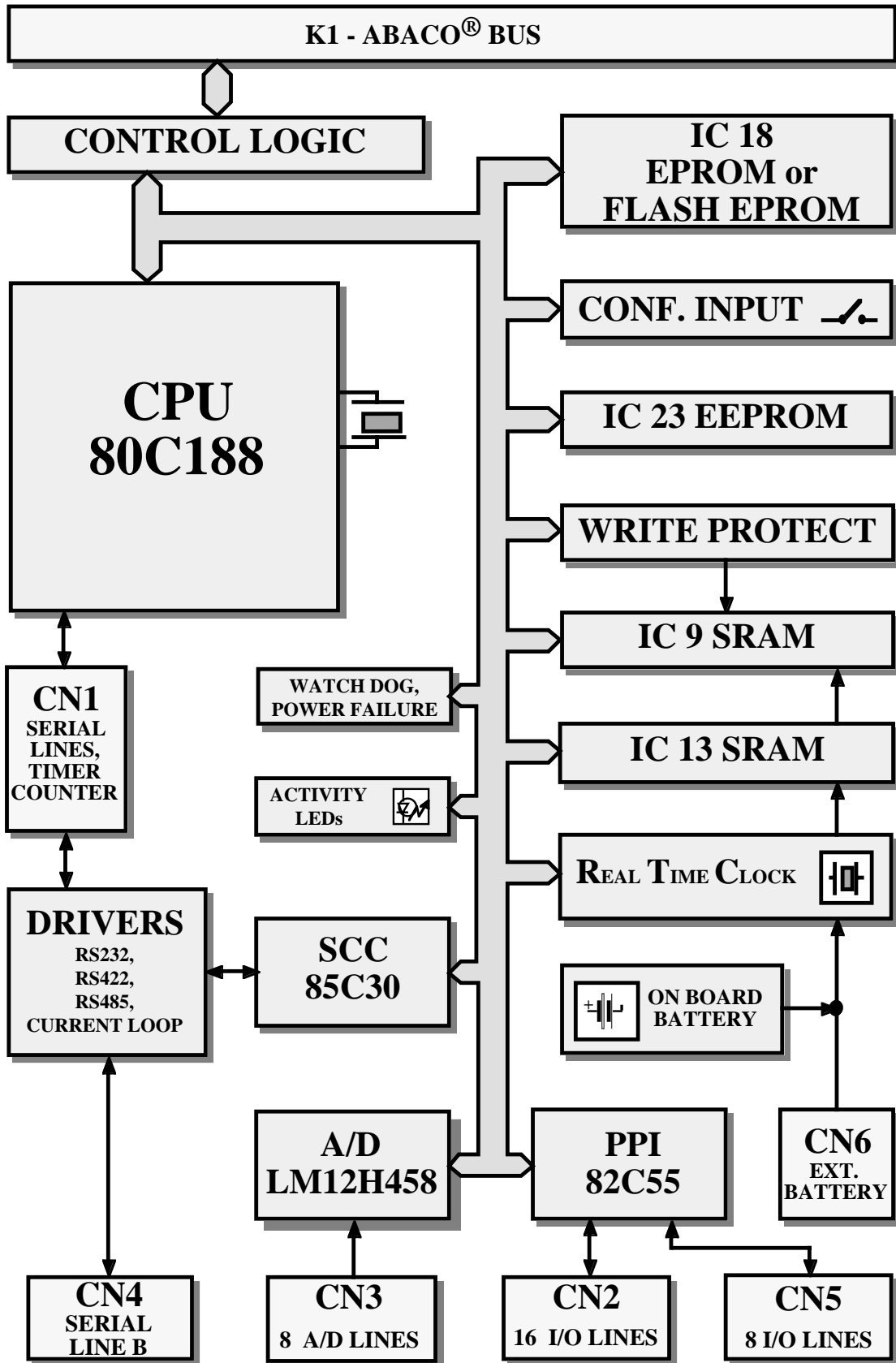


FIGURE 1: BLOCK DIAGRAM

WATCH DOG SECTION

GPC® 188F is provided with one Watch Dog circuit that can reset the card at programmable time intervals, if not retriggered. Watch dog circuit is used when the User want to exit from endless loops or to reset anomalous conditions not estimated by application program. There is an astable section with 1480 ms fixed intervention time. By software the User can perform a complete management of the devoces, using specific registers allocated in microprocessor I/O addressing space.

MEMORY DEVICES

On the card can be monted 2056K Bytes of memory divided with a maximum of 1024K Bytes EPROM or 512K Bytes FLASH EPROM, 1024K Bytes SRAM and 8K Bytes serial EEPROM. Please remark that up to 512K Bytes of SRAM can be write-protected through a specific jumper. The **GPC® 188F** memory configuration must be chosen considering the application to realize or the specific requirements of the User. Normally the card is equipped with 128K Bytes of SRAM plus 512 Bytes of serial EEPROM and all different configurations must be specified from the User, at the moment of the order.

With the on board back up circuit there is the possibility to keep data , also when power supply is failed; in this way the card is always able to maintain parameters, logged data, system status and configuration, etc. without using expensive external UPS. The back up circuit is supplied by a on board lithium battery or an external battery to be connected to a specific connector.

The addressing of memory devices is controlled by a specific circuitry, that provides to allocate the devices in the microprocessor address space.

For further information about memory configuration, sockets description and jumpers connection, please refer to "ADDRESSES AND MAPS" and "PERIPHERAL DEVICES SOFTWARE DESCRIPTION" chapters. For informations about memory devices please refer to "MEMORY SELECTION" paragraph.

MEMORY MANAGEMENT UNIT

A specific MMU section has been designed to manage in a practical and effcient way the memory configurations that the **GPC® 150** board can assume. The use is provided with a 64K work area, which can be easily allocated anywhere in the 5128K maximun memory space.

DIGITAL I/O LINES

It manages 24 TTL I/O lines divided in three 8 bit parallel ports. The lines direction is software settable at byte level. These I/O lines allow the possibility to connect several devices (for example: User interfaces) even when the handshake is completely software driven The PPI 82C55 is completely driven by software programming four registers allocated in microprocessor I/O addressing space, by a specific control logic.

TECHNICAL FEATURES

GENERAL FEATURES

On board resources:	24 TTL programmable Input/Output TTL (PPI 82C55) 2 sixteen bits TTL Timer Counter (CTC) 1 sixteen bits Timer 2 DMA channels for data transfer 1 RS 232 bidirectional line (SCC A) 1 RS 232, RS 422, RS 485 or Current Loop (SCC B) 1 astable Watch Dog 1 Real Time Clock (RTC) 1 eight dips Dip Switch 8 A/D Converter signals Industrial ABACO® BUS
Addressable memory:	IC 18: EPROM from 128K x 8 to 1024K x 8 FLASH EPROM from 128K x 8 to 512K x 8 IC 9: SRAM 128K x 8 or 512Kx8 IC 13 SRAM 128K x 8 or 512Kx8 IC 23: serial EEPROM from 256 byte to 8192 byte
CPU:	Intel 80C188
CPU clock frequency:	20 MHz
A/D frequency:	8 MHz
SCC clock frequency:	11.0592 MHz
A/D Converter resolution:	12 bit + sign
A/D conversion time:	5,5 µsec
Watch Dog intervent time:	from 940 msec to 2060 msec (typical 1420 msec)

PHYSICAL FEATURES

Size (W x H x D):	EUROPE format : 100 x 160 x 15 mm
Weight:	220 g (basic configuration)
Connectors:	K1: 64 pins DIN 41612 M 90 degreeses A+C type C CN1: 16 pins low profile vertical M CN2: 20 pins low profile 90 degreeses M

CN3: 20 pins low profile 90 degrees M
 CN4: 10 pins low profile 90 degrees M
 CN5: 20 pins low profile vertical M
 CN6: 2 pins low profile vertical M

Temperature range: from 0 to 70 Centigrad degrees

Relative humidity: 20% up to 90% (without condense)

ELECTRIC FEATURES

Power supply: +5 Vcc

Consumption on 5 Vdc: 390 mA in basic configuration
 440 mA in full configuration

On board back up battery: 3.0 Vdc; 1/2 AA

External back up battery: 3.6÷5 Vdc

Back up current: 3.8 μ A (on board battery)
 5.5 μ A (3.6 V external battery)

Analog inputs: 0÷2.048 V; \pm 2.048

Analog inputs impedance: < 4K Ω

RS422, 485 termination network: line temination= 120 Ω
 Pull-up on positive= 3.3K Ω
 Pull-down on negative= 3.3K Ω

Power Failure threshold: 52 mV before the reset occurs

INSTALLATION

In this chapter there are the information for a right installation and correct use of the card. The User can find the location and functions of each connectors, jumpers and some explanatory diagrams.

CONNECTIONS

The **GPC®188F** board has 7 connectors that can be linkeded to other devices or directly to the field, according to system requirements. In this paragraph there are connectors pin out, a short signals description (including the signals direction), connectors location (see figure 22) and some electrical diagrams that show the on board circuit of each connector.

CN6 - EXTERNAL BACK UP BATTERY CONNECTOR

CN6 is a 2 pins, vertical, male connector with 2.54mm pitch.

Through CN6 the User can connect an external battery for SRAM and RTC back up when the power supply is switched off (for further information please refer to chapter "BACK UP").

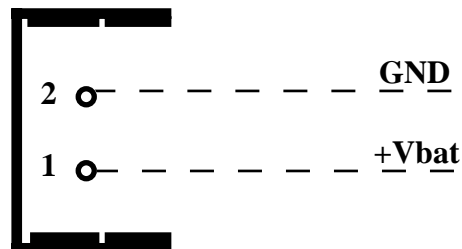


FIGURE 3: CN6 - EXTERNAL BACK UP BATTERY CONNECTOR

Signals description:

+Vbat	=	I	- External back up battery positive pin
GND	=		- External back up battery negative pin

CN5 - CONNECTOR FOR PPI 82C55 PORT B

CN5 is a 20 pins, male, vertical, low profile connector with 2.54 mm pitch.

On CN5 connector are available PPI 82C55 port B signals that equal to 8 I/O digital lines. Any parameter of this device (like signals direction, data management mode, etc.) is completely software definible by programming the device itself. All signals are at TTL level and follow the standard pinout I/O **ABACO®** pin-out.

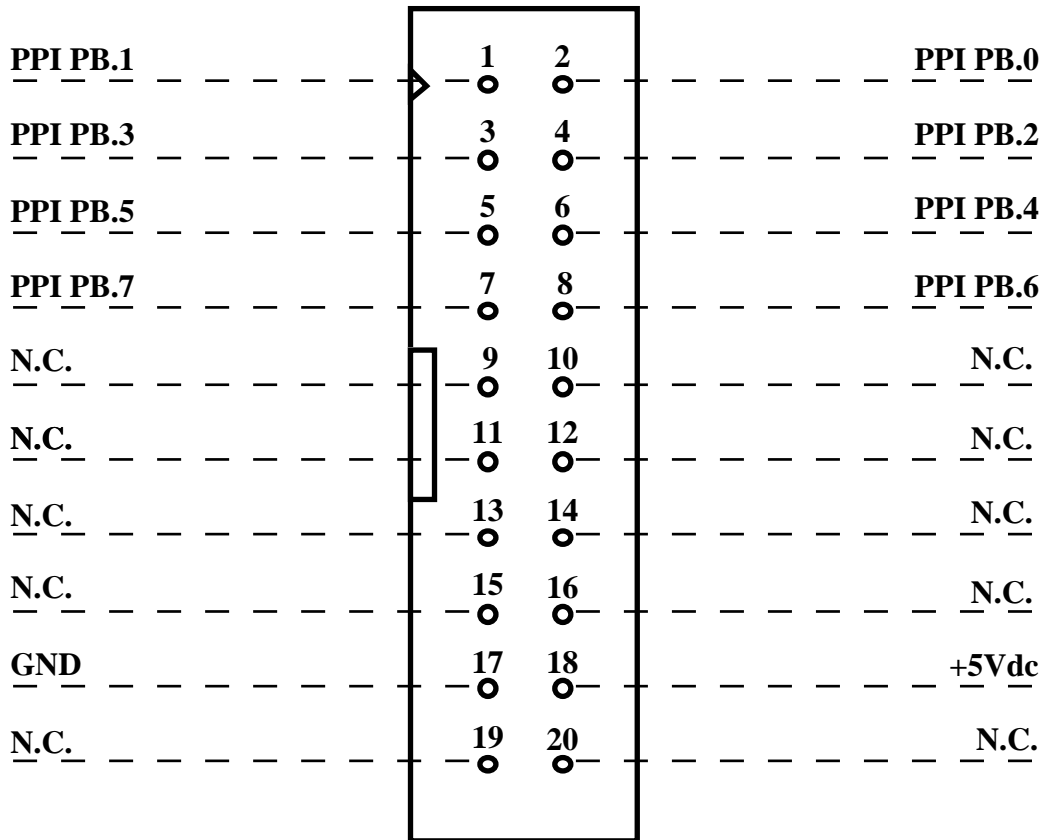


FIGURE 4: CN5 - CONNECTOR FOR PPI 82C55 PORT B

Signals description:

- PPI PB.n** = I/O - PPI 82C55 port B digital line n
- +5 Vdc** = O - Line connected to +5 Vdc power supply
- GND** = - Ground signal
- N.C.** = - Not connected

CN2 - CONNECTOR FOR PPI 82C55 PORT A AND C

CN2 is a 20 pins, male, 90 degrees, low profile connector with 2.54 mm pitch.

On CN2 connector are available PPI 82C55 port A and C signals that equal to 16 I/O digital lines. Any parameter of this device (like signals direction, data management mode, etc.) is completely software definible by programming the device itself. All signals are at TTL level and follow the standard pinout I/O **ABACO**® pin-out.

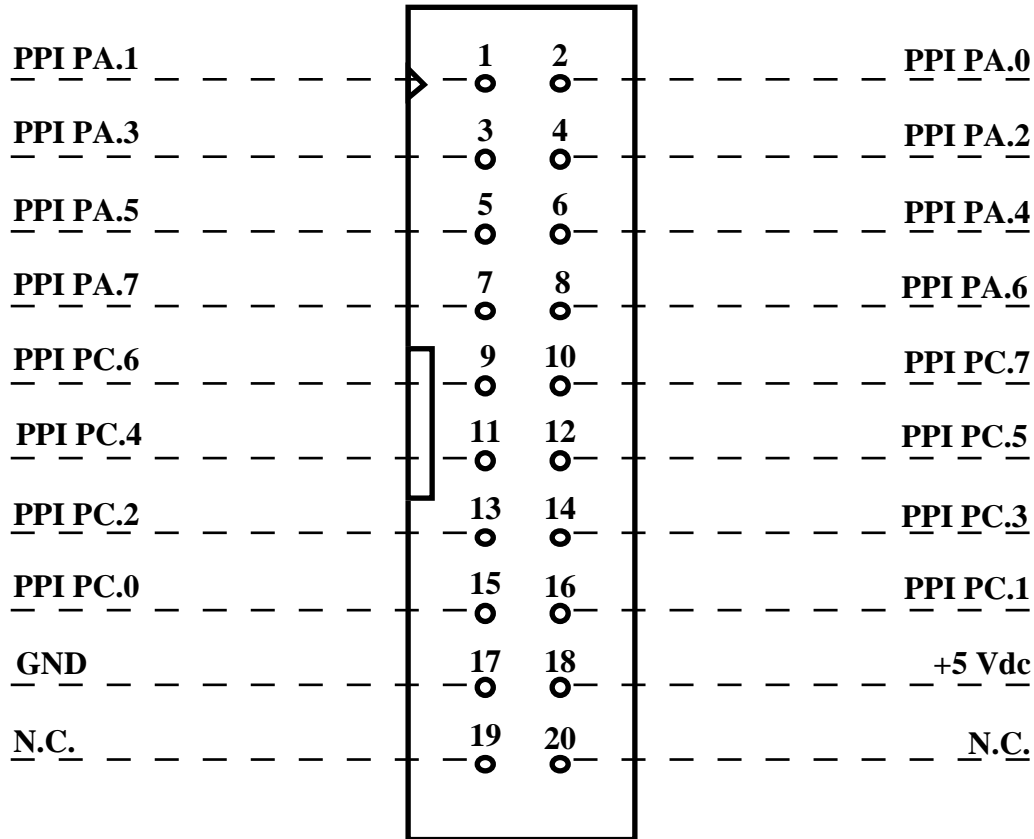


FIGURE 5: CN2 - CONNECTOR FOR PPI 82C55 PORT A AND C

Signals description:

PPI PA.n	=	I/O	- PPI 82C55 port A digital line n
PPI PC.n	=	I/O	- PPI 82C55 port C digital line n
+5 Vdc	=	O	- Line connected to +5 Vdc power supply
GND	=		- Ground signal
N.C.	=		- Not connected

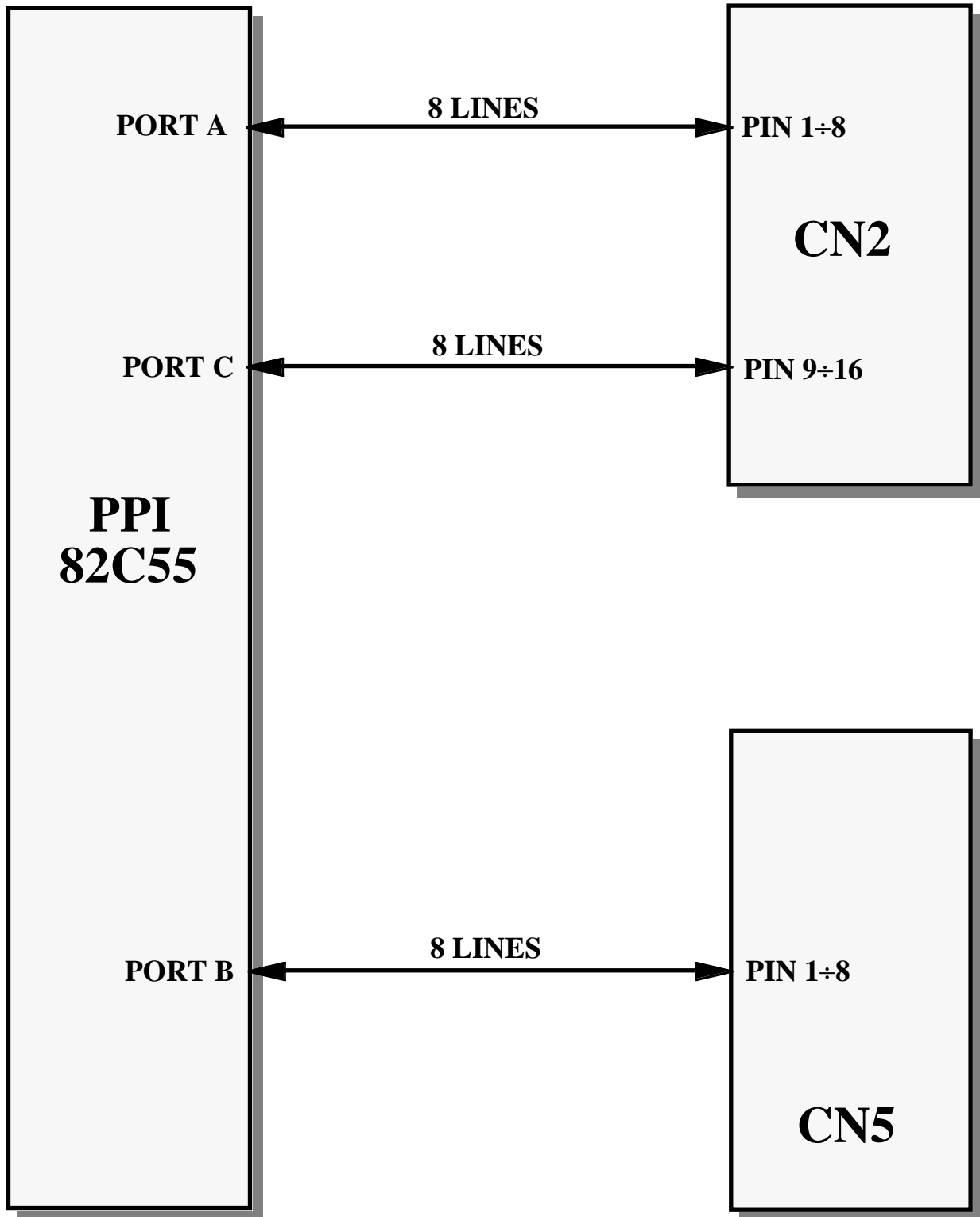


FIGURE 6: PPI I/O LINES CONNECTION DIAGRAM

CN3 - A/D CONVERTER INPUTS CONNECTOR

CN3 is a 20 pins, male, 90 degrees, low profile connector with 2.54 mm pitch.

Through CN3 an external device can connect up to 8 analog inputs. These are low impedance lines directly connected to the on board A/D Converter and are provided with a capacitive filter. The inputs may vary in the range $0 \div 2.048$ V single ended or in the range ± 2.048 V fully differential. Conversion management is completely software driven through programming LM 12H458; signals location on this connector has been designed to reduce problems due to interference.

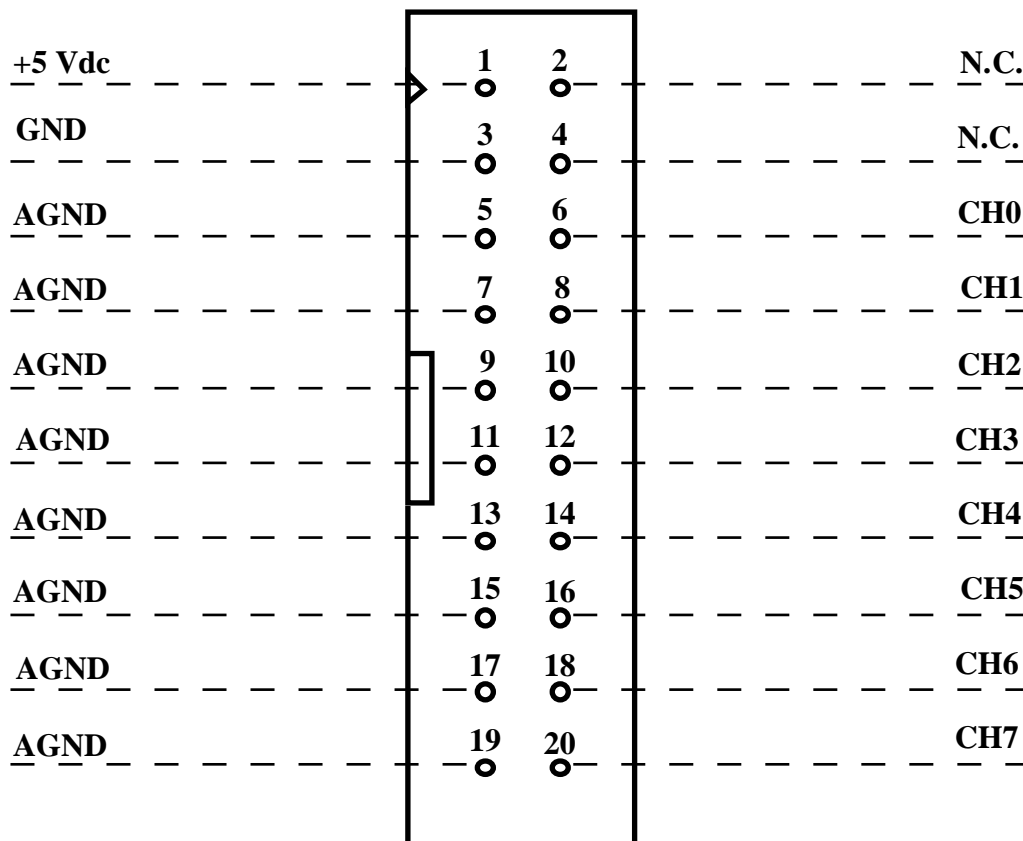


FIGURE 7: CN3 - A/D CONVERTER INPUTS CONNECTOR

Signals description:

CHn	= I	- Analog input signal connected to A/D Converter n-th channell
AGND	= -	- Analog ground signal
+5 Vdc	= O	- Power supply signal +5 Vcc
GND	= -	- Digital ground signal
N.C.	= -	- Not Connected

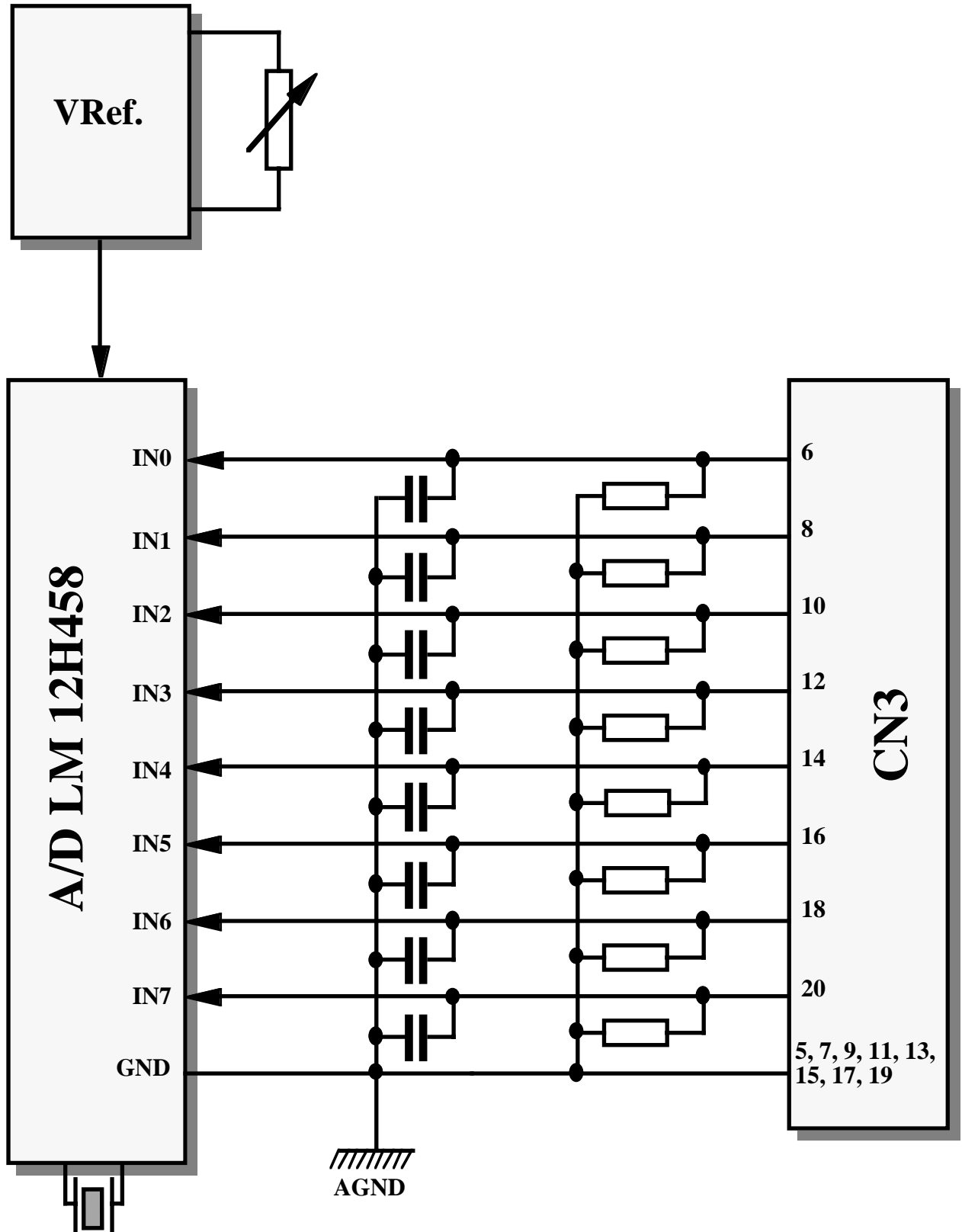


FIGURE 8: A/D CONVERTER INPUT DIAGRAM

CN1 -RS 232 SERIAL LINES AND TIMER COUNTER CONNECTOR

CN1 is a 16 pins, male, vertical, low profile connector with 2.54 mm pitch.

Through CN1 it is possible to connect the RS 232 serial lines and the two timer counters to the external world. To manage the CTC signals the User needs to connect specific jumpers in the proper positions and program specific CPU internal registers, to manage the serial lines the User needs to connect specific jumpers in the proper positions and program specific SCC 85C30 registers.

The signals on this connector are at TTL level and RS 232 signals follow the CCITT specifications; signals location on this connector has been designed to reduce problems due to interference.

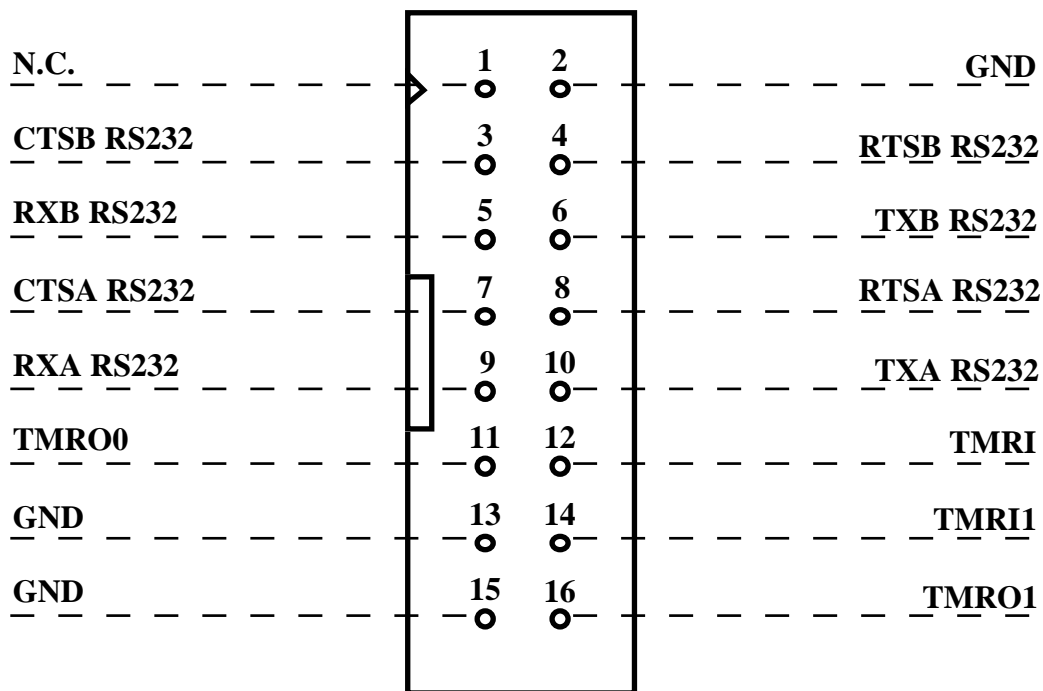


FIGURE 9: CN1 - RS 232 SERIAL LINES AND TIMER COUNTER CONNECTOR

Signals description:

- RXA, B RS232** = I - RS 232 serial line A, B Receive Data signal
- TXA, B RS232** = O - RS 232 serial line A, B Trasmit Data signal
- CTS A, B RS232**= I - RS 232 serial line A, B Clear To Send signal
- RTS A, B RS232**= O - RS 232 serial line A, B Request To Send signal
- TMRI0, 1** = I - TTL level CTC n-th input
- TMRO0, 1** = O - TTL level CTC n-th output
- GND** = - Ground signal

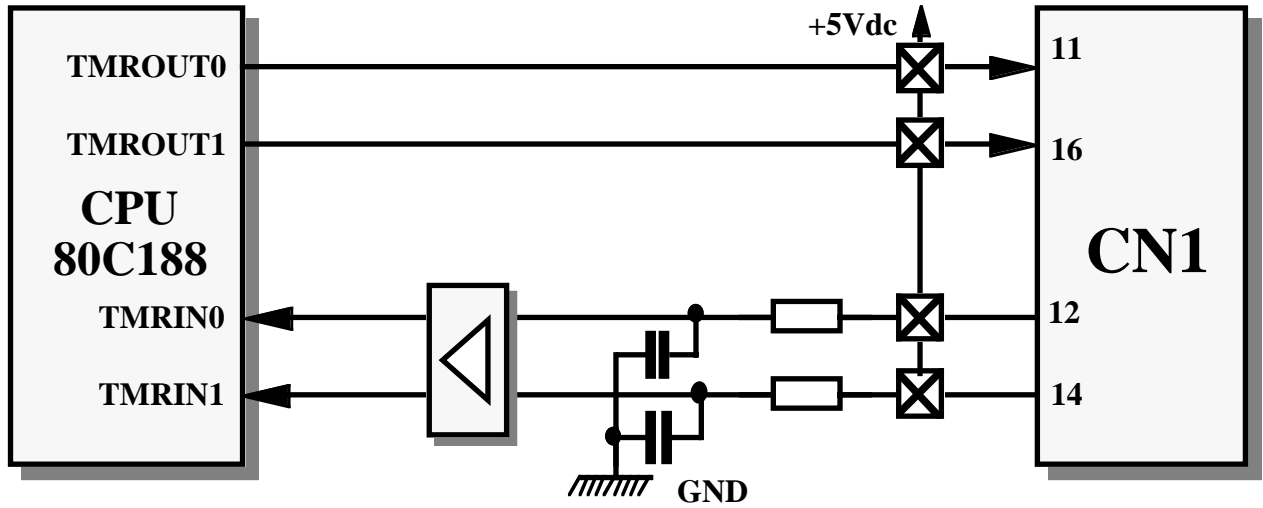


FIGURE 10: TIMER COUNTER CONNECTION DIAGRAM

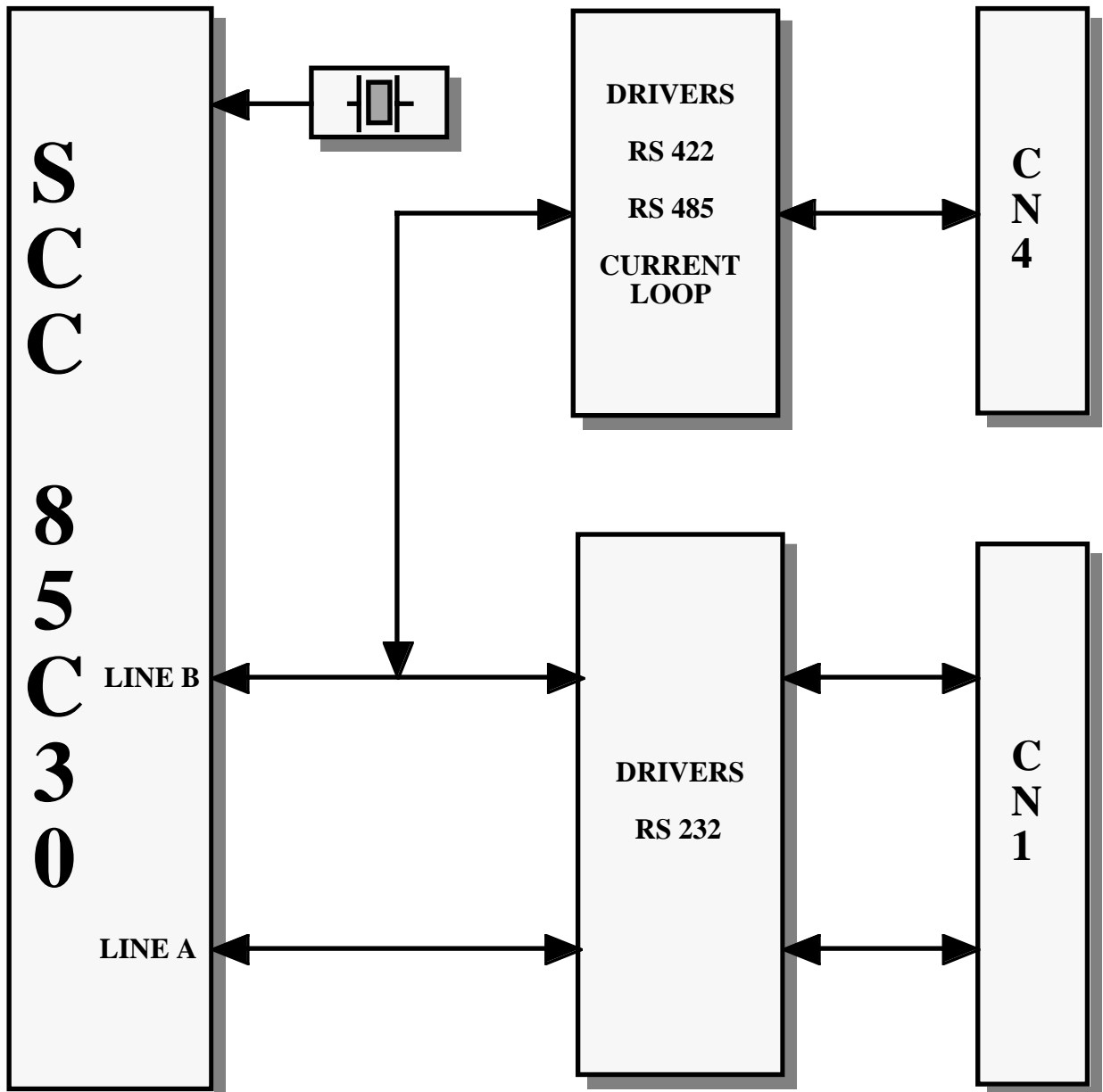


FIGURE 11: SERIAL COMMUNICATION CONNECTION DIAGRAM

CN4 - CONNECTOR FOR RS 422, RS 485 AND CURRENT LOOP

CN2 is a 10 pins, 90 degrees, male connector with 2,54mm pitch.

This connector carries all the signals of RS 422, RS 485 and Current Loop serial line B.

signals location on this connector has been designed to reduce problems due to interference; all the signals follow the CCITT normatives for each of the communication standards used. Please remark that the Current Loop serial line is passive.

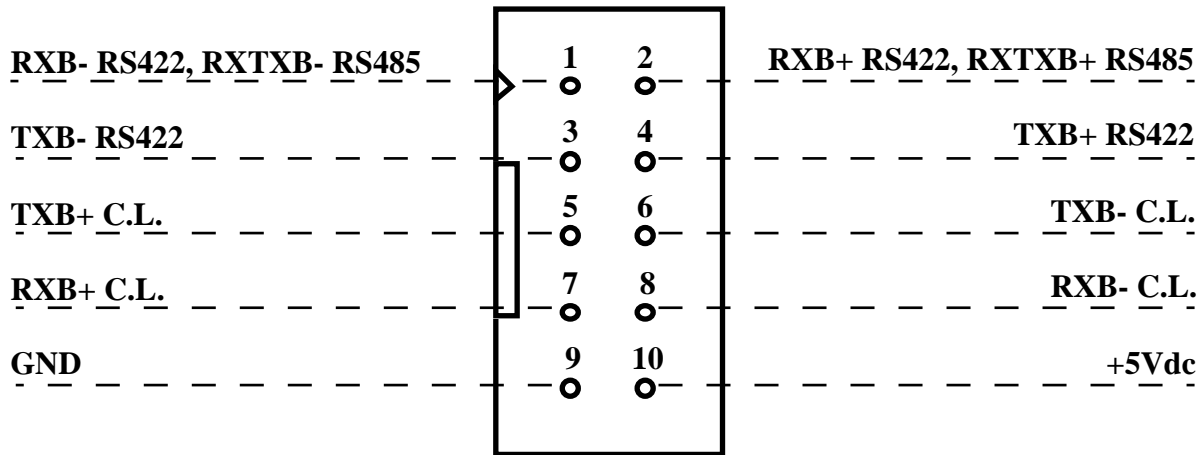


FIGURE 12: CN2 - RS 422, RS 485 AND CURRENT LOOP SERIAL LINE B CONNECTOR

Signals description:

RXB- RS422	= I - RS 422 serial line B Receive Data Negative signal
RXB+RS422	= I - RS 422 serial line B Receive Data Positive signal
TXB- RS422	= O - RS 422 serial line B Transmit Data Negative signal
TXB+ RS422	= O - RS 422 serial line B Transmit Data Positive signal
RXTXB- RS485	= I/O - RS 485 serial line B Receive Transmit Data Negative signal
RXTXB+ RS485	= I/O - RS 485 serial line B Receive Transmit Data Positive signal
RXB- C.L.	= I - Current Loop serial line B Receive Data Negative signal
RXB+ C.L.	= I - Current Loop serial line B Receive Data Positive signal
TXB- C.L.	= O - Current Loop serial line B Transmit Data Negative signal
TXB+ C.L.	= O - Current Loop serial line B Transmit Data Positive signal
+5 Vdc	= I - Power supply signal+5 Vcc
GND	= - Ground signal

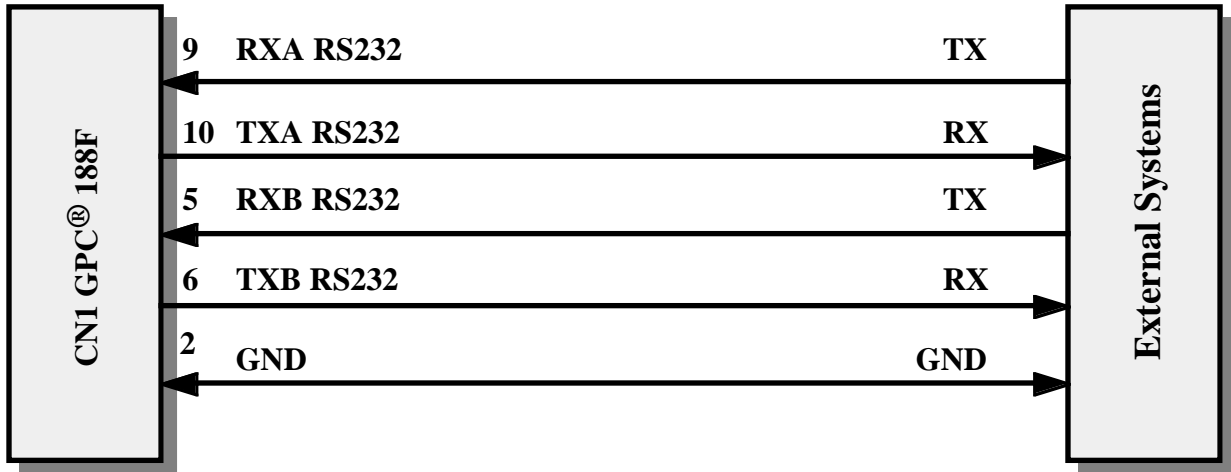


FIGURE 13: RS 232 POINT-TO-POINT CONNECTION EXAMPLE

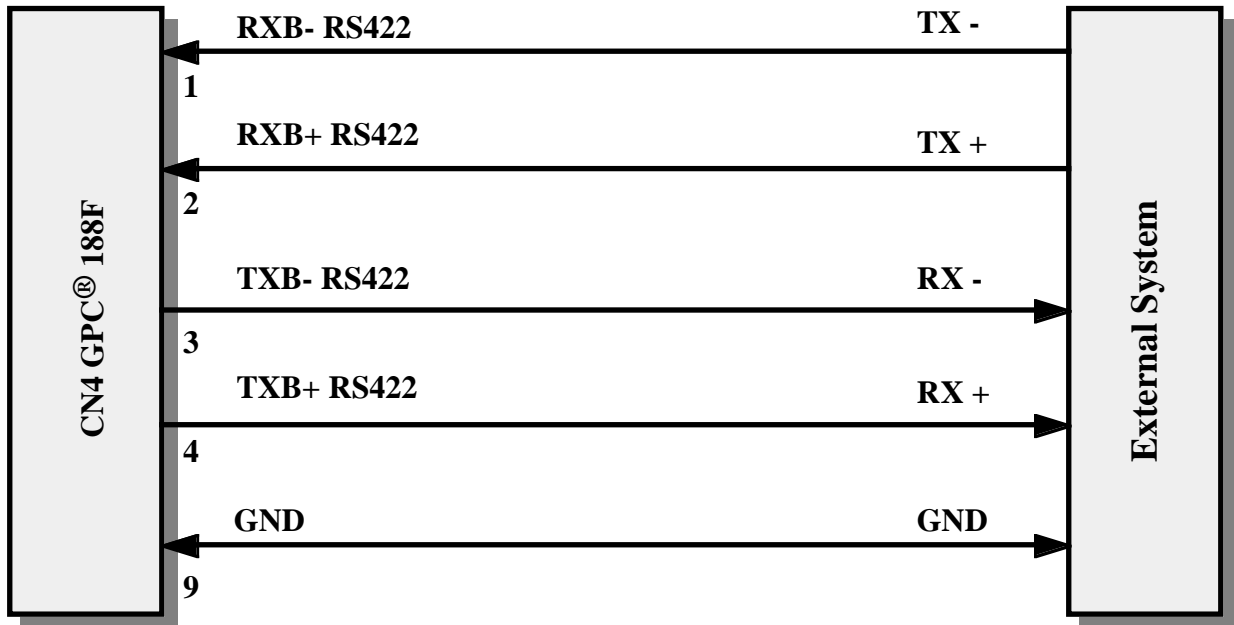


FIGURE 14: RS 422 POINT-TO-POINT CONNECTION EXAMPLE

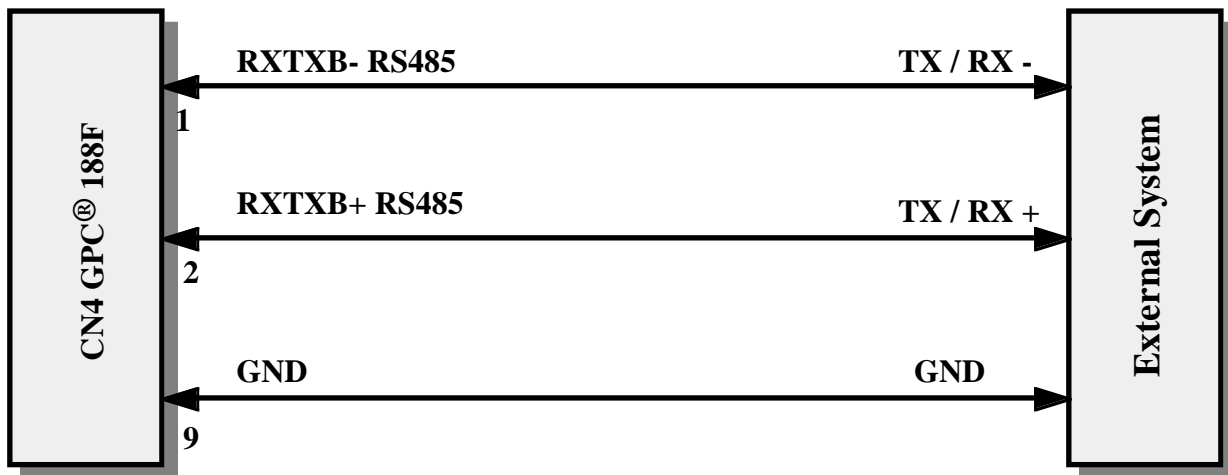


FIGURE 15: RS 485 POINT-TO-POINT CONNECTION EXAMPLE

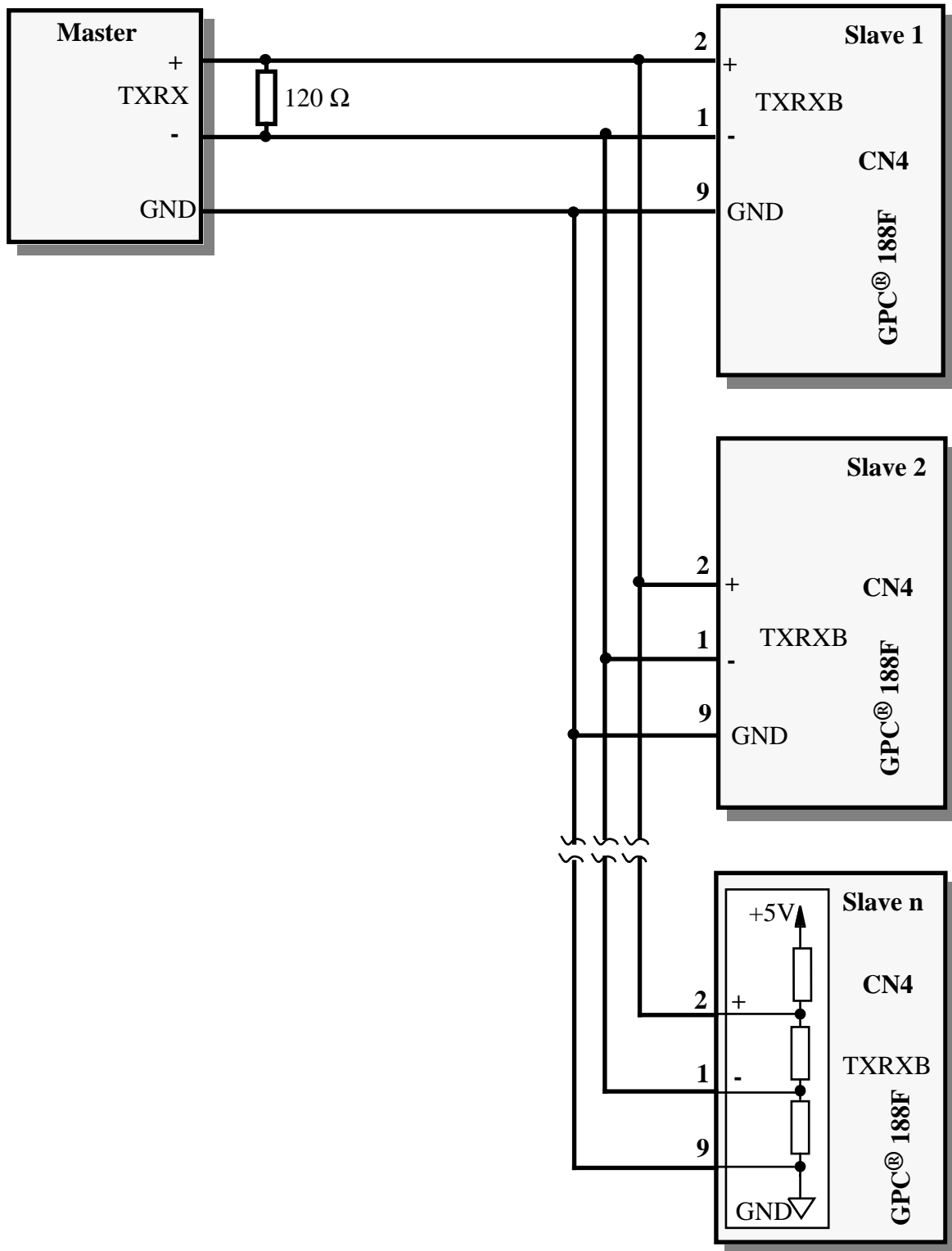


FIGURE 16: RS 485 NETWORK CONNECTION EXAMPLE

Please remark that in a RS 485 network two forcing resistors must be connected across the net and two termination resistors (120 Ω) must be placed at its extremities, respectively near the Master unit and the Slave unit at the greatest distance from the Master.

Forcing and terminating circuitry is installed on GPC® 188F board. It can be enabled or disabled through specific jumpers, as explained later.

For further information please refer to TEXAS INSTRUMENTS Data-Book, "RS 422 and RS 485 Interface Circuits", the introduction about RS 422-485.

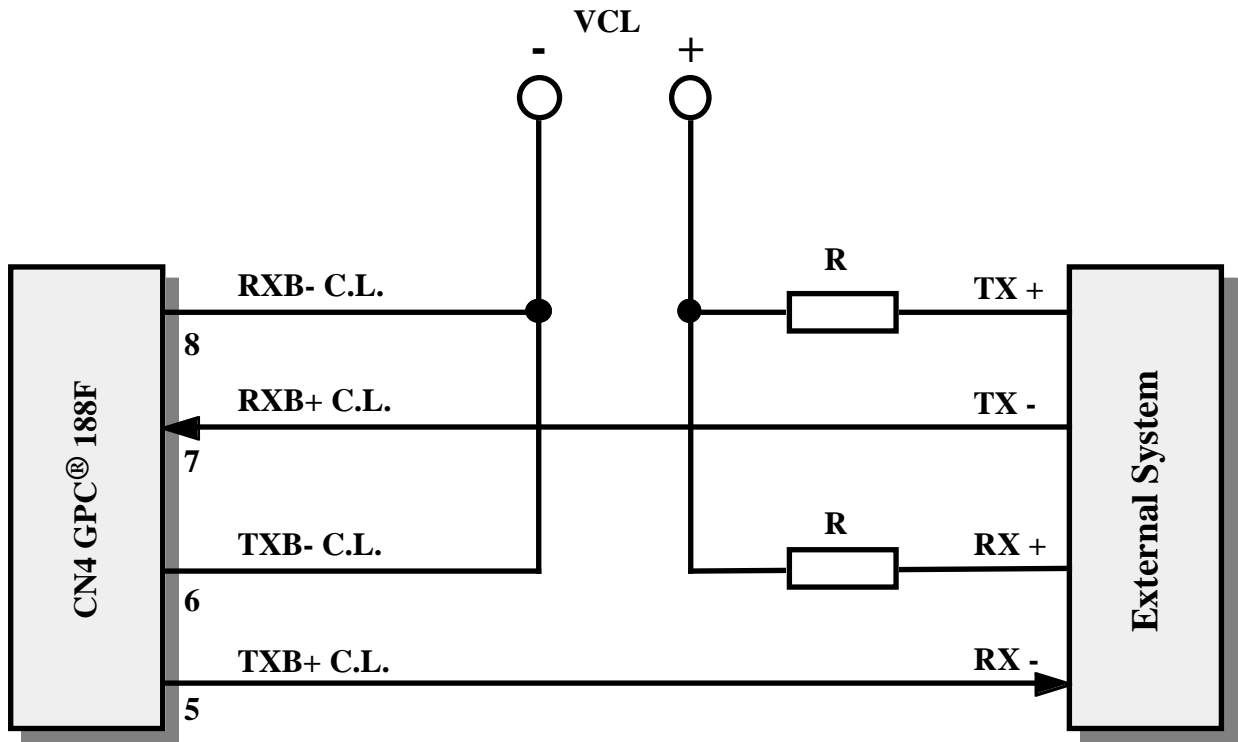


FIGURE 17: 4 WIRES CURRENT LOOP POINT-TO-POINT CONNECTION EXAMPLE

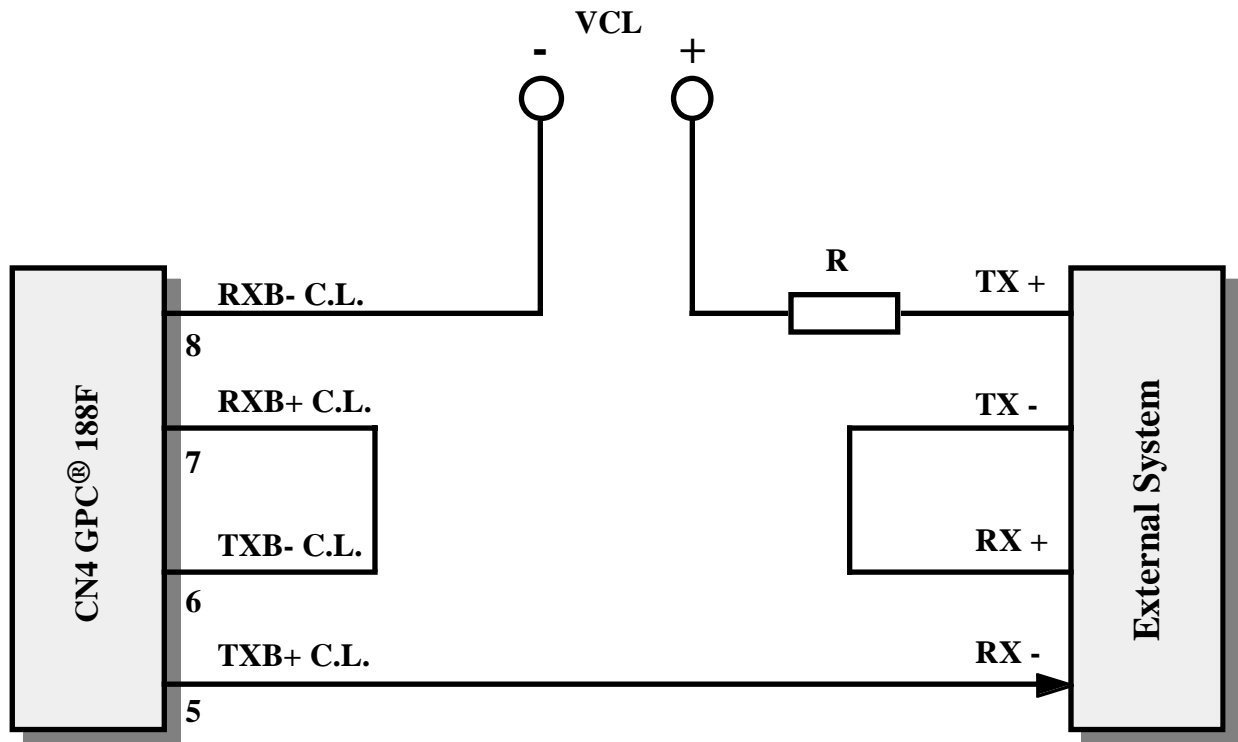


FIGURE 18: 2 WIRES CURRENT LOOP POINT-TO-POINT CONNECTION EXAMPLE

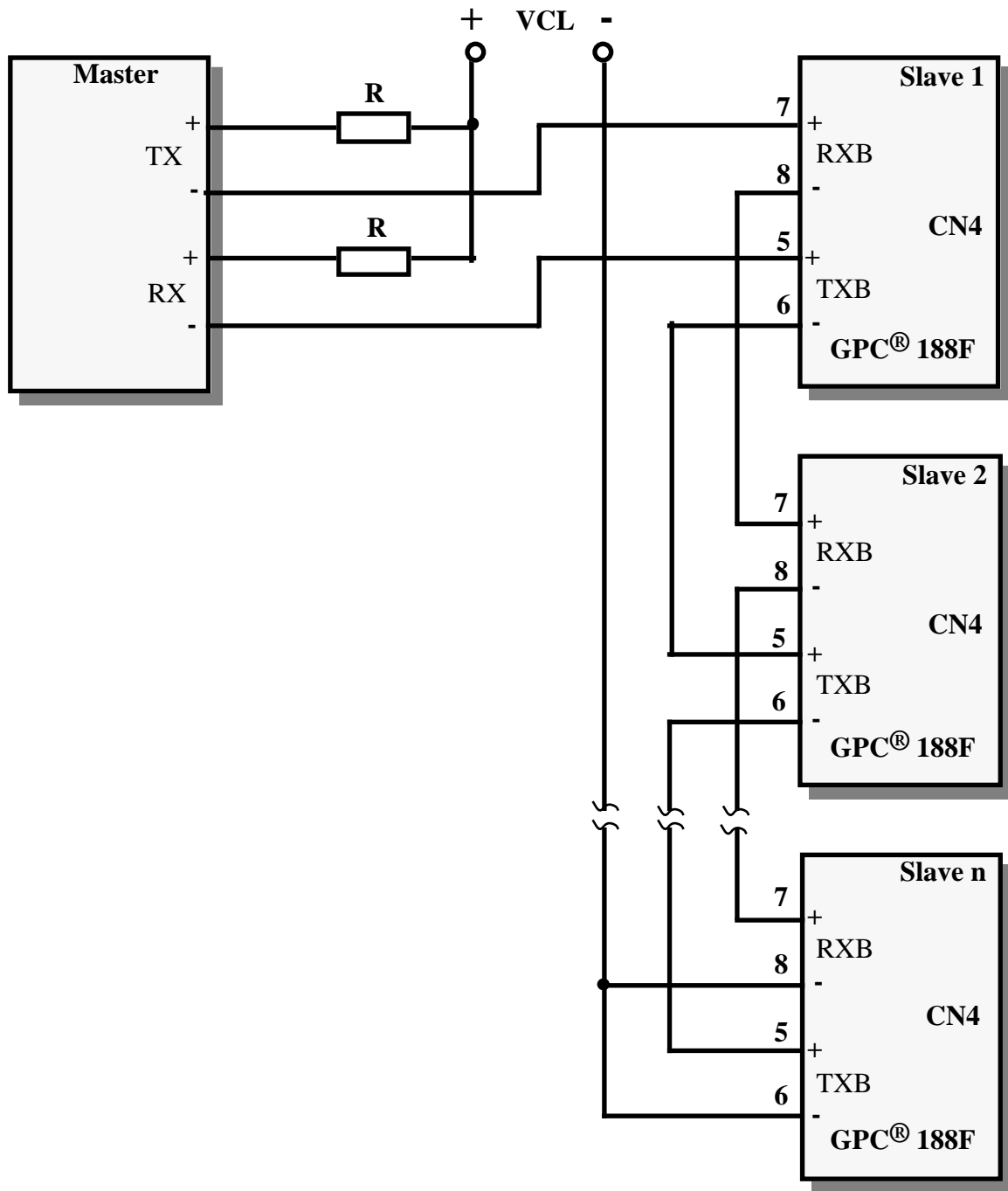


FIGURE 19: PASSIVE CURRENT LOOP NETWORK CONNECTION EXAMPLE

Possible Current Loop connections are two: 2 wires and 4 wires. These connections are shown in figures 19 and 20 where it is possible to see the voltage for **VCL** and the resistances for current limitation (**R**). The supply voltage varies in compliance with the number of connected devices and voltage drop on the connection cable.

The choice of the values for these components must be done considering that:

- circulation of a **20 mA** current must be guaranteed;
- potential drop on each transmitter is about **2,35 V** with a 20 mA current;
- potential drop on each receiver is about **2,52 V** with a 20 mA current;
- in case of shortcircuit each transmitter must dissipate at most **125 mW**;
- in case of shortcircuit each receiver must dissipate at most **90 mW**.

For further info please refer to HEWLETT-PACKARD Data Book, (**HCPL 4100** and **4200** devices).

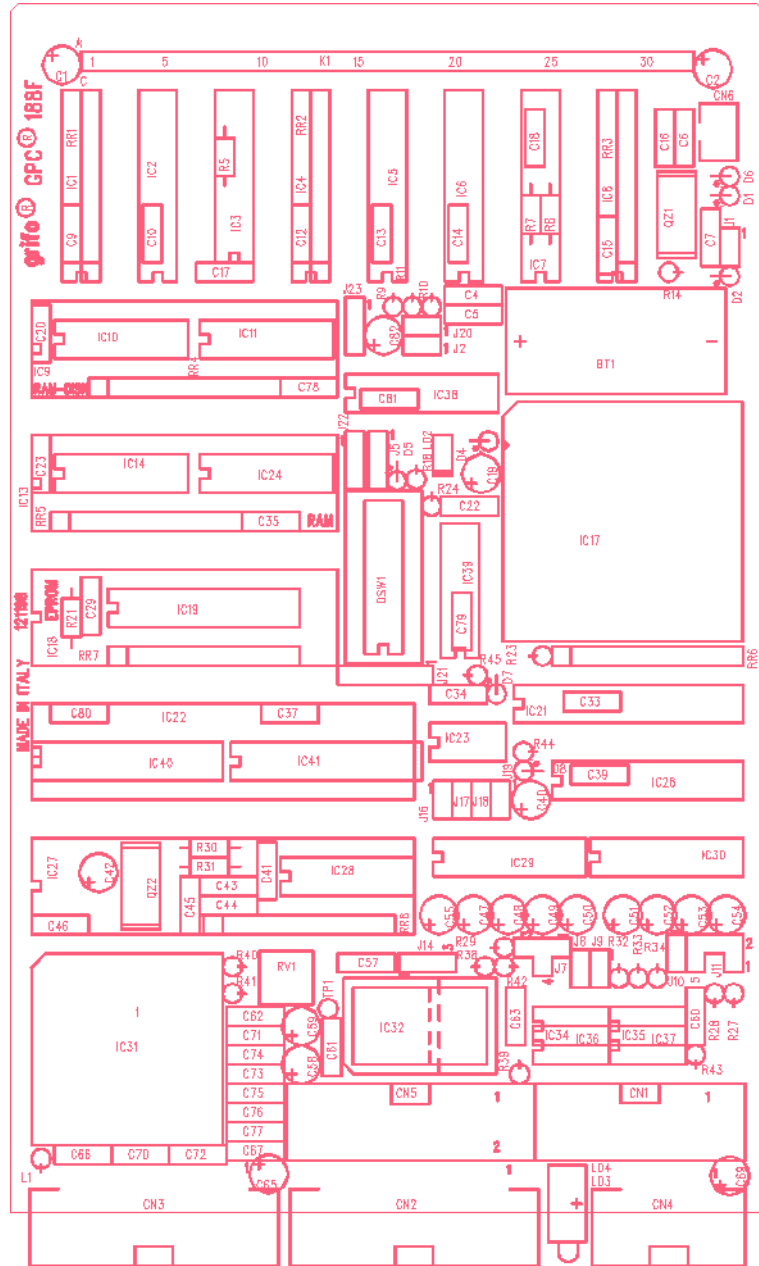


FIGURE 20: COMPONENTS MAP

K1 - CONNECTOR FOR ABACO® BUS

K1 is a 64 pins, male, 90°, DIN 41612 connector with 2.54 pitch.

On K1 are available all the industrial **ABACO®** BUS signals and it can be used for connections to many other peripheral cards. In the table below there are the standard pin outs both for 8 bits and 16 bits CPU and the signal connected on **GPC® 188F**. All signals follow TTL standard.

A 16 bit BUS	A 8 bit BUS	A GPC® 188F	PIN	C GPC® 188F	C 8 bit BUS	C 16 bit BUS
GND	GND	GND	1	GND	GND	GND
+5 Vdc	+5 Vdc	+5 Vdc	2	+5 Vdc	+5 Vdc	+5 Vdc
D0	D0	D0	3	N.C.		D8
D1	D1	D1	4	N.C.		D9
D2	D2	D2	5	N.C.		D10
D3	D3	D3	6	/INT	/INT	/INT
D4	D4	D4	7	/NMI	/NMI	/NMI
D5	D5	D5	8	N.C.	/HALT	D11
D6	D6	D6	9	N.C.	/MREQ	/MREQ
D7	D7	D7	10	/IORQ	/IORQ	/IORQ
A0	A0	A0	11	/RD	/RD	/RD LDS
A1	A1	A1	12	/WR	/WR	/WR LDS
A2	A2	A2	13	N.C.	/BUSAK	D12
A3	A3	A3	14	N.C.	/WAIT	/WAIT
A4	A4	A4	15	N.C.	/BUSRQ	D13
A5	A5	A5	16	/RESET	/RESET	/RESET
A6	A6	A6	17	N.C.	/M1	/IACK
A7	A7	A7	18	N.C.	/RFSH	D14
A8	A8	A8	19	N.C.	/MEMDIS	/MEMDIS
A9	A9	A9	20	N.C.	VDUSEL	A22
A10	A10	A10	21	N.C.	/IEI	D15
A11	A11	A11	22	N.C.		
A12	A12	A12	23	N.C.	CLK	CLK
A13	A13	A13	24	N.C.		/RD UDS
A14	A14	A14	25	N.C.		/WR UDS
A15	A15	A15	26	N.C.		A21
A16		N.C.	27	N.C.		A20
A17		N.C.	28	N.C.		A19
A18		N.C.	29	/R.T.	/R.T.	/R.T.
+12 Vdc	+12 Vdc	N.C.	30	N.C.	-12 Vdc	-12 Vdc
+5 Vdc	+5 Vdc	+5 Vdc	31	+5 Vdc	+5 Vdc	+5 Vdc
GND	GND	GND	32	GND	GND	GND

FIGURE 21: K1 - ABACO® BUS CONNECTOR

Signals description:

8 bits CPU

A0-A15	=	O	- Address BUS
D0-D7	=	I/O	- Data BUS
INT	=	I	- Interrupt request
NMI	=	I	- Non Maskable Interrupt
HALT	=	O	- Halt state
MREQ	=	O	- Memory Request
IORQ	=	O	- Input Output Request
RD	=	O	- Read cycle status
WR	=	O	- Write cycle status
BUSAK	=	O	- BUS Acknowledge
WAIT	=	I	- Wait
BUSRQ	=	I	- BUS Request
RESET	=	O	- Reset
M1	=	O	- Machine cycle one
RFSH	=	O	- Refresh for dynamic RAM
MEMDIS	=	I	- Memory Display
VDUSEL	=	O	- VDU Selection
IEI	=	I	- Interrupt Enable Input
CLK	=	O	- System clock
R.B.	=	I	- Reset button
+5 Vdc	=	I	- Power supply at +5 Vdc
+12 Vdc	=	I	- Power supply at +12 Vdc
-12 Vdc	=	I	- Power supply at -12 Vdc
GND	=		- Ground signal

16 bits CPU

A16-A22	=	O	- Address BUS
D8-D15	=	I/O	- Data BUS
RD UDS	=	O	- Read Upper Data Strobe
WR UDS	=	O	- Write Upper Data Strobe
IACK	=	O	- Interrupt Acknowledge
RD LDS	=	O	- Read Lower Data Strobe
WR LDS	=	O	- Write Lower Data Strobe

N.B.

Directionality indications as above stated are referred to a master (CPU o GPC®) board and have been kept untouched to avoid ambiguity in case of multi-boards systems.

TRIMMER AND CALIBRATION

On **GPC® 188F** is available a trimmer, named **RV1**, that calibrates the V_{ref} voltage of the A/D Converter section. The **GPC® 188F** is subjected to a careful test that verifies and calibrates all the card sections. To easily locate the trimmer, please refer to figure 20. The calibration is executed in laboratory, with a controlled $+20\text{ C}^\circ$ room temperature, following these steps:

- The A/D voltage reference (V_{ref}) is calibrated through RV1 trimmer, by using a 5 digits precision multimeter, to a value of $+2.048\text{ Vdc}$, on test point TP1.
- The corrispondance between the analog input signal and the combination read from A/D is verified. This check is performed with a reference signal connected to A/D inputs and testing that the A/D combination and the theoretic combination differ at maximum of the A/D section errors sum.
- The trimmer is blocked with paint.

The analog interfaces use high precision components that are selected during mounting phase to avoid complicate and long calibration procedures. After the calibration, the on board trimmer is blocked with paint to mantain calibration also in presence of mechanic stresses (vibrations, movings, delivery, etc.).

The User must not modify the card calibration, but if thermic drifts,time drifts and so on, make necessary a new calibration, the User must strictly follow the previous described procedure.

The circuitry that generates the reference voltage defines also the full range for all the 8 analog input channels; by software it is possible to define the signals acquisition modality between "single ended" (8 inputs grounded to AGND in the range $0\div 2.048\text{ V}$) and "fully differential" (4 differential inputs in the range $\pm 2.048\text{ V}$), as described in the appendix B of this manual.

The User most NOT intervernt on the circuit that generates the reference voltage, however if this should be necessary (exampe: for time derives) then he/she must follow the above mentioned procedure.

To easily locate RV1 and TP1 please refer to figure 22.

TEST POINT

The board is provided with a test point called TP1, that allows to read, through a galvanically isolated multimeter, the reference voltage which is calibrated in laboratory and whose value is $V_{ref}=2.048\text{ V}$. TP1 is made of one contact connected directly to V_{ref} signal.

To easily locate the test point please refer to figure 22, while for further informations about V_{ref} signal please refer to the paragraph "TRIMMER AND CALIBRATION".

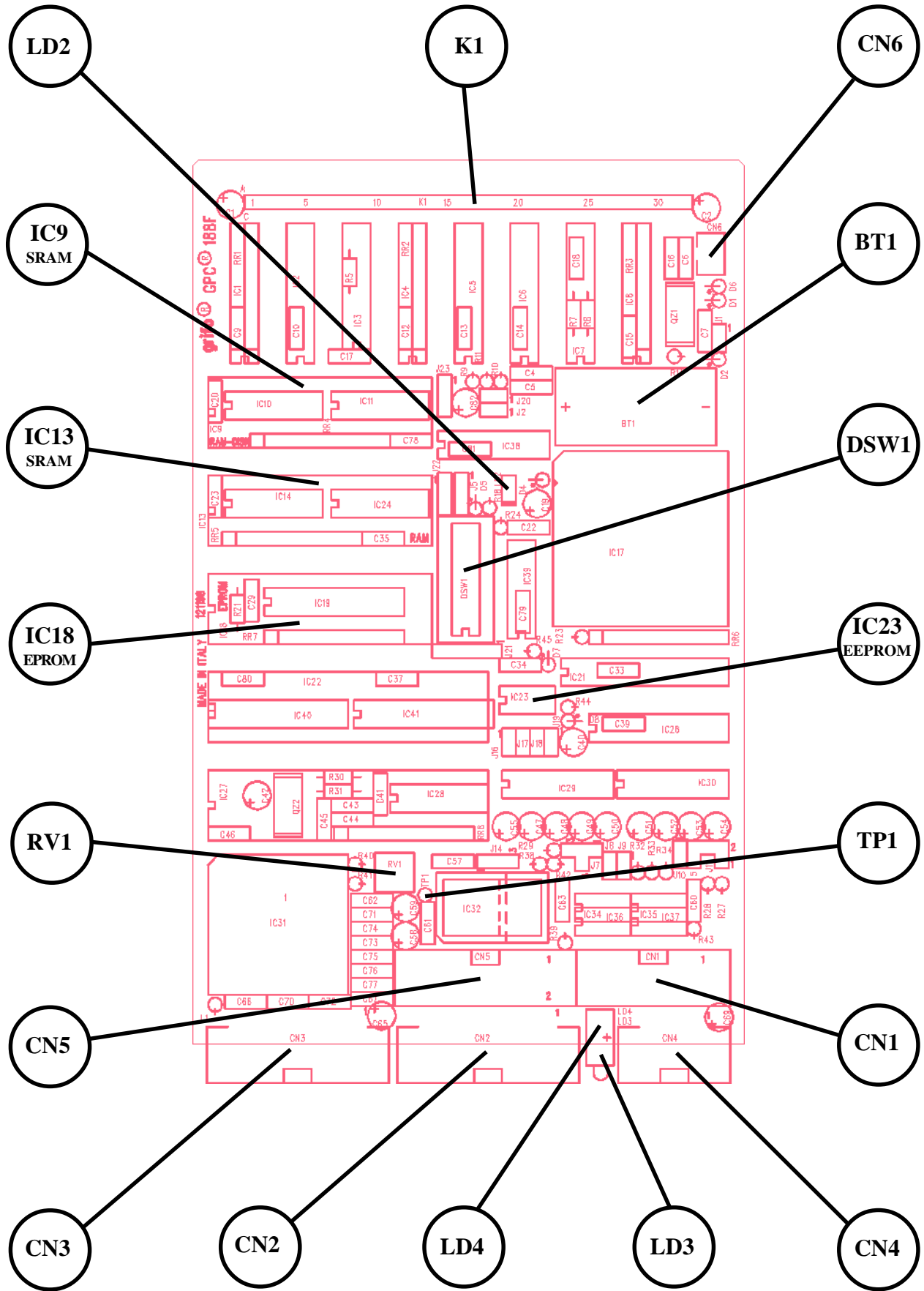


FIGURE 22: CONNECTORS, TRIMMER, MEMORIES, ETC. LOCATION

I/O CONNECTION

To prevent possible connecting problems between **GPC® 188F** and the external systems, the user has to read carefully the information of the previous paragraphs and he must follow these instructions:

- For RS 232, RS 422, Current Loop or RS 485 communication signals the User must follow the standard rules of these protocols.
- For all TTL signals the user must follow the rules of this electric standard. The connected digital signal must be always referred to card digital ground (GND). For TTL signals, the 0 Vdc level corresponds to logic state "0", while 5Vdc level corresponds to logic state "1".
- The analog inputs (A/D section) must be connected to low impedance signals in the following ranges: $0 \div 2.048$ V or ± 2.048 V according to selected signals acquisition modality. Remember that the eight analog inputs available on CN3 are provided of filter capacitors that ensure an higher stability of the acquired signals, but reduce at the same time the bandwidth frequency. For further informations please refer to the paragraph "TYPE OF ANALOG INPUT SELECTION".

VISUAL FEEDBACK

GPC® 188F board is provided with three LEDs to signal status conditions, as described in the following table:

LEDs	COLORE	FUNZIONE
LD1	Rosso	Segnala l'attivazione della circuiteria di watch dog.
LD2	Verde	LED di attività gestito via software.
LD3	Verde	LED di attività gestito via software.

FIGURE 23: VISUAL FEEDBACK TABLE

The main purpose of these LEDs is to give a visual indication of the board status, making easier the operations of system working verify. To easily locate these LEDs on the board, please refer to figure 22.

DIGITAL I/O INTERFACES

Through CN2 and CN5 (I/O Abaco® standard connector) the GPC® 188F card can be connected to all of the numerous grifo® boards featuring the same standard pin out. Installation of these interface is very easy; in fact only a 20 pins flat cable (code FLT.20+20) or a 26 pins GPC®-side and 20 pins interface-side flat cable (code FLT.26+20) is required, while the software management of these interfaces is as easy; in fact most of the software packages available for GPC® 188F card are provided with the necessary procedures. Remarkable modules are:

- **QTP 16P, QTP 24P, KDL x24, KDF 224, DEB 01**, etc. that solve all the local operator interfacing problems. These modules are provided with all the resources needed to obtain a high-level human-machine interface (they feature alphanumeric displays, matrix keyboard and visualization LEDs) at a short distance from GPC® 188F card. The available software drivers allow to manage the operator interface resources directly through the high-level instructions for console management.
- **IAC 01, DEB 01**: it is an interface for CENTRONICS parallel printer that can be connected with a standard printer cable. The printer is managed by software through the high level instructions of the selected programming language (LPRINT for BASIC, PRINTF for C, etc.).
- **MCI 64**: it a large mass memory support that can directly manage the PCMCIA memory cards (RAM, FLASH, ROM, etc.) in their available sizes. About software the developed drivers provide a complete file system interfacing allowing to access the informations stored in the memory card directly through the high-level file management instructions.
- **RBO xx, TBO xx, XBI xx, OBI xx**: these are buffer interfaces for I/O TTL signals. With these modules the the TTL input signals are converted in NPN or PNP optoisolated inputs and the TTL output signals are converted in relays or transistor optoisolated outputs. Some of the above mentioned interfaces can be connected directly to CN4.

For more informations refer to "EXTERNAL CARDS" chapter and the software tools documentation.

JUMPERS

On **GPC® 188F** there are 17 jumpers for card configuration. Connecting these jumpers, the User can define for example the memory type and size, the peripheral devices functionality and so on. Below there is the jumpers list, location and function:

JUMPERS	N. PINS	USE
J1	2	It connects the on board Lithium battery to the Back Up circuitry.
J2	2	It enables the Write Protect circuitry of SRAM on IC9.
J5	3	It connects the Watch Dog circuitry.
J7	4	It selects the serial communication type for serial line B (RS 232, RS 422, RS 485, Current Loop).
J8, J9	2	It connects the termination and forcing circuitry for RS 422, RS 485.
J10	2	It forces the status of CTSB handshake signal.
J11	5	It selects directionality and activation modality for serial line B in RS 422, RS 485.
J14	3	It selects the source for DMA channel 0.
J16	2	It forces the status of DCDA handshake signal.
J17	2	It forces the status of DCDB handshake signal.
J18	2	It forces the status of SYNCA handshake signal.
J19	2	It forces the status of SYNCB handshake signal.
J20	2	It connects the Power Failure circuitry.
J21	5	It selects memory type and size for IC18.
J22	3	It selects SRAM size for IC13.
J23	3	It selects SRAM size for IC9.

FIGURE 24: JUMPERS SUMMARIZING TABLE

The following tables describe all the right connections of **GPC® 188F** jumpers with their relative functions. To recognize these valid connections, please refer to the board printed diagram (serigraph) or to figure 20 of this manual, where the pins numeration is listed; for recognizing jumpers location, please refer to figure 25.

The "*" used in the following tables, denotes the default connection, or on the other hand the connection set up at the end of testing phase, that is the configuration the User receives.

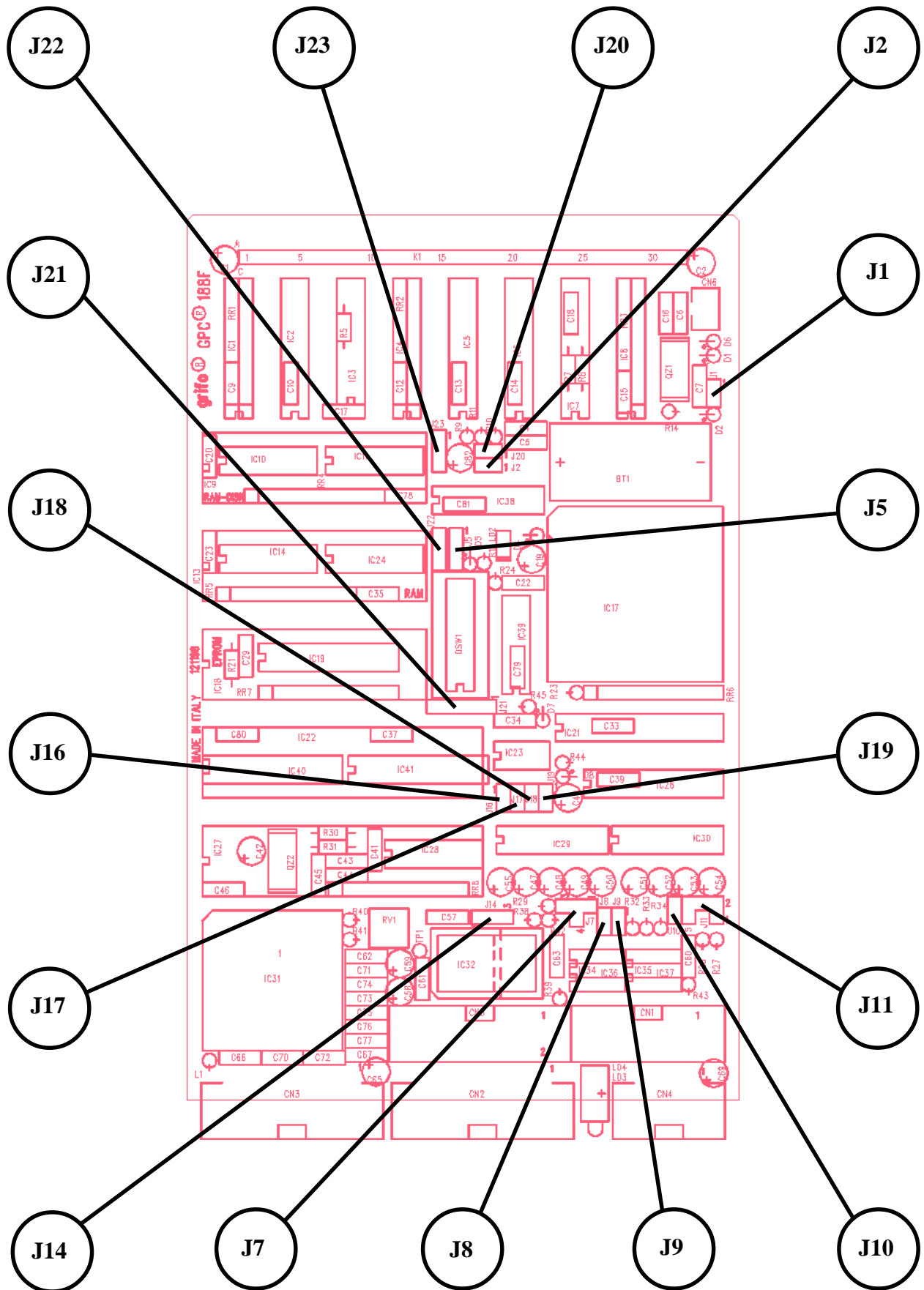


FIGURE 25: JUMPERS LOCATION

2 PINS JUMPERS

JUMPERS	CONNECTION	USE	DEF.
J1	not connected	It does not connect battery BT1 to Back Up circuit.	*
	connected	It connects battery BT1 to Back Up circuit.	
J2	not connected	Disables Write Protect circuitry.	*
	connected	Enables Write Protect circuitry.	
J8, J9	not connected	It does not connect the termination and forcing circuitry to RS 485 serial line or to RS 422 reception signal.	*
	connected	It connects the termination and forcing circuitry to RS 485 serial line or to RS 422 reception signal.	
J10	not connected	It does not force the handshake CTSB status.	*
	connected	It forces active (GND) the handshake CTSB status.	
J16	not connected	It sets to logic status 1 the DCDA signal.	*
	connected	It sets to logic status 0 the DCDA signal.	
J17	not connected	It sets to logic status 1 the DCDB signal.	*
	connected	It sets to logic status 0 the DCDB signal.	
J18	not connected	It sets to logic status 1 the SYNCA signal.	*
	connected	It sets to logic status 0 the SYNCA signal.	
J19	not connected	It sets to logic status 1 the SYNCB signal.	*
	connected	It sets to logic status 0 the DCSYNCBDA signal.	
J20	not connected	It does not connect Power Failure circuitry to NMI signal.	*
	connected	It connects Power Failure circuitry to NMI signal.	

FIGURE 26: 2 PINS JUMPERS TABLE
4 PINS JUMPER

JUMPER	CONNECTION	USE	DEF.
J7	position 1-2	It sets serial line B in Current Loop.	*
	position 2-3	It sets serial line B in RS 232.	
	position 2-4	It sets serial line B in RS 422 or RS 485.	

FIGURE 27: 4 PINS JUMPER TABLE

3 PINS JUMPERS

JUMPERS	CONNECTION	USE	DEF.
J5	position 1-2	It does not connect Watch Dog circuitry to reset circuitry.	*
	position 2-3	It connects Watch Dog circuitry to reset circuitry.	
J14	not connected	No source is connect to microprocessor DMA channel 0.	*
	position 1-2	It connects SCC 8530 to microprocessor DMA channel 0.	
	position 2-3	It connects A/D LM 12458 to microprocessor DMA channel 0.	
J22	position 1-2	It configures socket IC13 for 128K Bytes SRAM.	*
	position 2-3	It configures socket IC13 for 512K Bytes SRAM.	
J23	position 1-2	It configures socket IC9 for 128K Bytes SRAM.	*
	position 2-3	It configures socket IC9 for 512K Bytes SRAM.	

FIGURE 28: 3 PINS JUMPERS TABLE

5 PINS JUMPER

JUMPERS	CONNECTION	USE	DEF.
J11	position 1-2, 3-4	It selects communication on serial line B in RS 485 (2 wires half duplex).	*
	position 2-3, 4-5	It selects communication on serial line B in RS 485 (4 wires full duplex or half duplex).	
J21	position 3-4	It configures socket IC18 for EPROM 1024K Bytes.	*
	position 1-2, 3-4	It configures socket IC18 for EPROM up to 512K Bytes.	
	position 2-3, 4-5	It configures socket IC18 for FLASH EPROM up to 512K Bytes.	

FIGURA 29: 5 PINS JUMPER TABLE

The "*" used in the following tables, denotes the default connection, or on the other hand the connection set up at the end of testing phase, that is the configuration the User receives.

RESET AND WATCH DOG

GPC® 188F is provided with a very efficient Watch Dog circuitry that are really efficient and easy to use. Its most important features are:

- astable mode;
- intervention time fixed at 1420 ms;
- enable function by hardware;
- retrigger by software;

In astable mode when intervention time is elapsed the circuit becomes active, it stays active till the end of reset time (185 msec) and after it is again deactivated.

Through jumper J5 the User can select whether to connect the Watch Dog circuitry to the reset network.

In response to a /RESET signal activation and successive deactivation the board restarts the execution of the program stored at address FFFF0H on IC13.

Please remark that /RESET signal generated by **GPC® 188F** the board is connected also to pin 16C of K1 connector. Other reset sources, in addition to the Watch Dog circuit, are: reset contact (R.T., pin 29C of connector K1) and power good circuitry.

About retrigger operation of internal and external watch dog circuits, please refer to paragraph "WATCH DOG" in chapter "PERIPHERAL DEVICES SOFTWARE DESCRIPTION" and to appendix B of this manual.

BACK UP

GPC® 188F has an on-board lithium battery BT1 for the back up of SRAM and RTC content when power supply is switched off. Jumper J1 connects physically the battery so it can be disconnected to save its duration whenever back-up is not needed. Through CN6 connector it is possible to connect an external battery: configuration of jumper J1 does not affect the working of this battery and it can replace BT1 completely.

Please refer to the paragraph "ELECTRIC FEATURES" to choose the type of the external back-up battery, to easily locate see see figure 22.

POWER FAILURE

In addition to the CPU controlled power management circuitry, **GPC® 188F** card also features an efficient Power Failure circuitry. Through jumper J20 this latter can be connected to the microprocessor /NMI interrupt signal.

The task of this circuitry is to keep under control power supply voltage and activate on output to request a CPU action when this voltage reaches a value lower than a threshold (52 mV above the reset interval) and jumper J20 is connected.

Please remark that the time interval between Power Failure activation and reset activation changes according to the type of supply being used; it is however about 100 µsec, long enough only to execute a fast response routine (for example to save a flag in the backed SRAM).

Typical use of power failure is to inform the board about the imminent power supply black out, so the CPU can save appropriate informations.

INTERRUPTS

A remarkable feature of **GPC® 188F** card is the powerful interrupt management. Here follows a short description of wich devices can geneate interrupts and their modalities; for further informations about interrputs management please refer to the microprocessor data sheet or to the appendix B of this manual.

- **ABACO® BUS** -> Generates a CPU /NMI through K1 connector /NMI signal.
 Generates a CPU INT0 through K1 connector /INT signal.
- Power Failure -> Generates a CPU /NMI, according to the connection o jumper J20.
- Real Time Clock -> Generates a CPU INT2.
- A/D Converter -> Generates a CPU INT3.
- SCC serial lines -> Generate a CPU INT1.
- CPU periherals -> Generate an internal interrupt. CPU internal interrupt sources are: TC 0, TC 1, TC 2, DMA 0, DMA 1, software interrupts.

On the **GPC® 188F** board there is an interrupt manager capable to enable, disable and mask the interrupt sources and to manage more interrupts simultaneously. This way the User can always respond promptly and efficently to any external event, also deciding the priority to assign to the several event sources.

For further informations please refer to appendix B of this manual.

CONFIGURATION INPUTS

GPC® 188F is provided with an on board 8 pins Dip Switch (DSW1) and four jumpers (J16÷J19), typically used for systm configuration and software readable. Most commin applications for these devices are working conditions settings or firmware parameters input, etc.

Configuration inputs read modalities can be found in the chapter "PERIPHERAL DEVICES SOFTWARE DESCRIPTION", while to easily locate them on the board please refer to figures 22 and 25.

SERIAL COMMUNICATION SELECTION

Serial line A can be buffered only as RS 232 while serial line B can be buffered in RS 232, RS 422, RS 485 or Current Loop. By hardware can be selected which one of these electric standards is used, through jumpers connection (as described in the previous tables) and drivers installation. By software the serial line can be programmed to operate with all the standard physical protocols, in fact the bits per character, parity, stop bits and baud rates can be decided by setting opportunes SCC 85C30 internal registers. In the following paragraphs there are all the informations on serial communication configurations.

Some devices needed for RS 422, RS 485 and Current Loop configurations are not mounted on the board in standard configuration; this is why each fist non-standard (non-RS 232) serial configuration for line B must be always performed by **grifo**® technicians. This far the User can change in autonomy the configuration following the informations below:

- SERIAL LINE B IN RS 232 (default configuration)

J7	=	position 2-3	IC29	= driver MAX 202
J8, J9	=	don't care	IC34	= don't care
J10	=	not connected	IC35	= don't care
J11	=	don't care	IC36	= don't care
			IC37	= don't care

- SERIAL LINE B IN CURRENT LOOP (option .CLOOP)

J7	=	position 1-2	IC29	= don't care
J8, J9	=	don't care	IC34	= driver HP 4200
J10	=	(*2)	IC35	= driver HP 4200
J11	=	don't care	IC36	= no device
			IC37	= no device

Please remark that Current Loop serial interface is passive, so it must be connected an active Current Loop serial line, that is a line provided with its own power supply. Current Loop interface can be employed to make both point-to-point and multi-point connections through a 2-wires or a 4-wires connection.

- SERIAL LINE B IN RS 422 (option .RS 422)

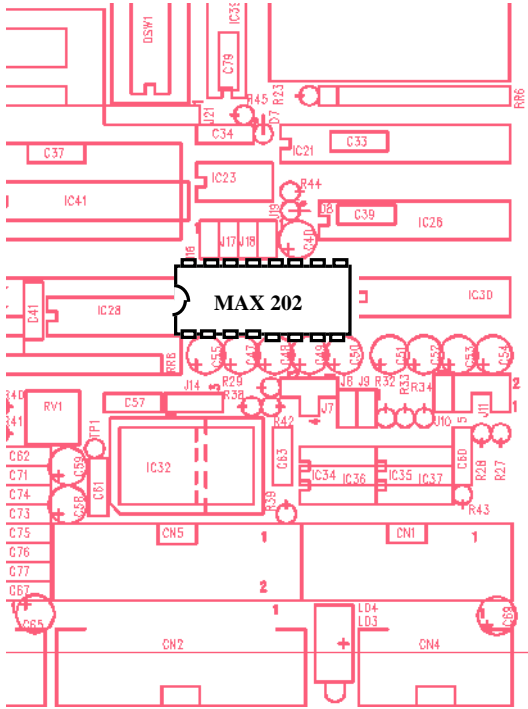
J7	=	position 2-4	IC29	= don't care
J8, J9	=	(*1)	IC34	= no device
J10	=	(*2)	IC35	= no device
J11	=	position 2-3, 4-5	IC36	= driver SN 75175 or MAX 483
			IC37	= driver SN 75175 or MAX 483

Status of signal /RTSB, which is software managed, allows to enable or disable the transmitter as follows:

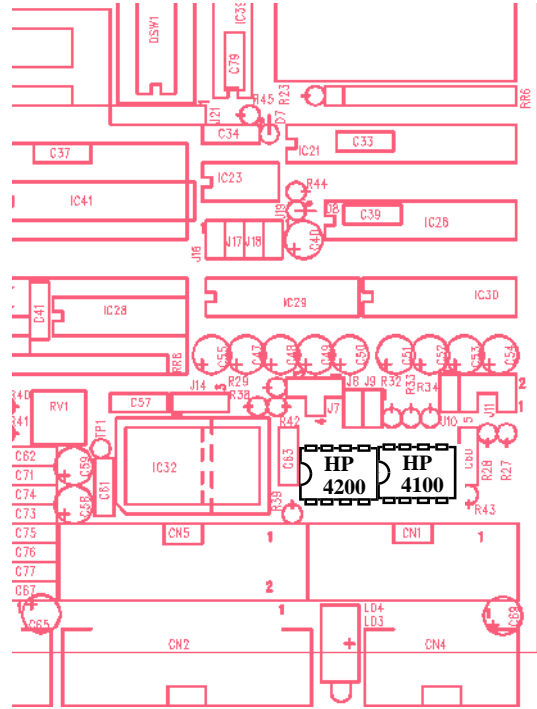
/RTSB = low level = logic state 0 -> transmitter enabled

/RTSB = high level = logic state 1 -> transmitter disabled

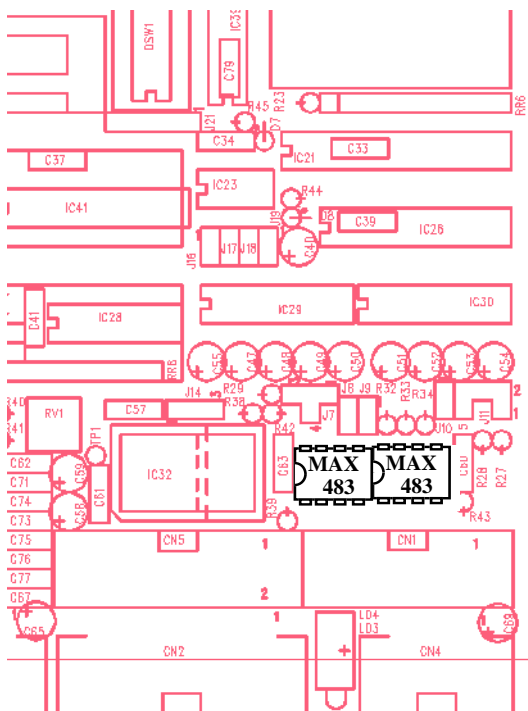
In point-to-point connections, signal /RTSB can be always kept low (trasnmitter always enabled), while in multi-point connections transmitter must be enabled only when a transmission is requested.



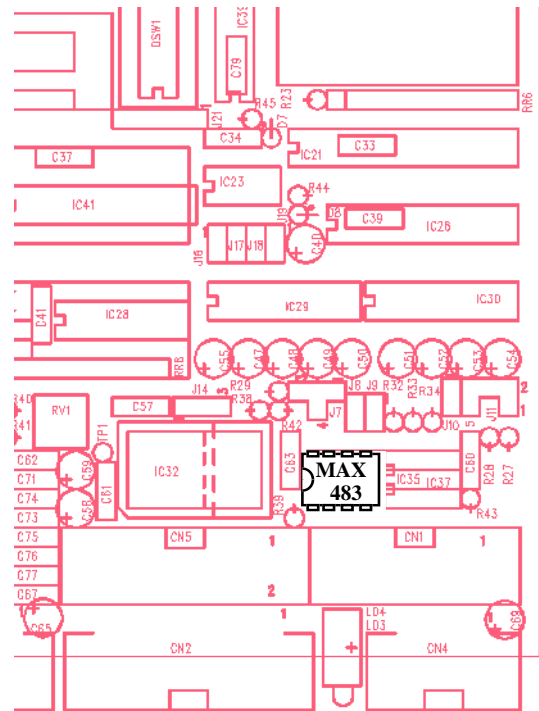
Serial B in RS 232



Serial B in Current Loop



Serial B in RS 422



Serial B in RS 485

FIGURE 31: SERIAL COMMUNICATION DRIVERS LOCATION

- SERIAL LINE B IN RS 485 (option .RS 485)

J7	=	position 2-4	IC29	=	don't care
J8, J9	=	(*1)	IC34	=	no device
J10	=	(*2)	IC35	=	no device
J11	=	position 1-2, 3-4	IC36	=	driver SN 75175 or MAX 483
			IC37	=	no device

In this modality the signals to use are pins 4 and 5 of connector CN2, that become transmission or reception lines according to the status of signal /RTSB, managed by software, as follows:

/RTSB = low level = logic state 0 -> transmitter enabled

/RTSB = high level = logic state 1 -> transmitter disabled

This kind of serial communication can be used for multi-point connections, in addition it is possible to listen to own transmission, so the User is allowed to verify the succes of transmission. In fact, any conflict on the line can be recognized by testing the received character after each transmission.

(*1) If using the RS 422 or RS 485 serial line, it is possible to connect the terminating and forcing circuit on the line by using J11 and J12. This circuit must be always connected in case of point-to-point connections, while in case of multi-point connections it must be connected only in the farrest boards, that is on the edges of the communication line.

(*2) The purpose of jumper J10 is to keep active the input handshake /CTS_B when serial line B is not configured in RS 232. In detail, if the User sets the autoenable modality (where reception and transmission are automatically decided by the status of the hardware handshakes) this jumper must be connected in order to keep the serial transmission enabled.

When a reset or a power on occur, signal /RTSB is kept to a logic level high, so in one of these two cases driver RS 485 is receiving or RS 422 driver is disabled, avoiding eventual conflicts in communication.

For further informations about serial communication please refer to the examples of figures 13÷19 and to appendix B of this manual.

DMA TRANSFERS

Some devices on **GPC® 188F** board support high speed (DMA) transfers to the work memory:

- SCC 85C30 -> Data to transmit through serial line B can be transferred through CPU DMA channel 1.
Data received from serial line B can be transferred through CPU DMA channel 0, when J14 is connected in position 1-2.
- A/D LM 12H458 -> Data sampled by A/D converter can be transferred through CPU DMA channel 0, when J14 is connected in position 2-3.

For further informations please refer to Appendix B to the paragraph "DISABLE DMA REQUEST".

MEMORY SELECTION

On **GPC® 188F** can be mounted up to 2056K bytes of memory divided in several configurations, as described in the following table:

IC	DEVICE	DIMENSIO SIZE	JUMPERS CONNECTION
18	EPROM	128K Byte	J21 in position 3-4
	EPROM	256K Byte	J21 in position 3-4
	EPROM	512K Byte	J21 in position 3-4
	EPROM	1024K Byte	J21 in position 1-2 and 3-4
	FLASH EPROM	128K Byte	J21 in position 2-3 and 4-5
	FLASH EPROM	512K Byte	J21 in position 2-3 and 4-5
9	SRAM	128K Byte	J23 in position 1-2
	SRAM	512K Byte	J23 in position 2-3
13	SRAM	128K Byte	J22 in position 1-2
	SRAM	512K Byte	J22 in position 2-3
23	EEPROM	256÷8K Byte	-

FIGURE 32: MEMORY SELECTION TABLE

All the above described devices must feature a JEDEC compliant pin out except for the serial EEPROM installed on IC23 that must be requested to **grifo®** in the ordering phase. To determine the name of the memory devices that can be mounted, please refer to the manufacturer documentation.

GPC® 188F is delivered in its default configuration, this means 128K SRAM on IC13 and 512 bytes serial EEPROM on IC23; any different memory configuration can be mounted by the User in autonomy or requested to **grifo®** in the order. Below are reported the order codes for the several optional memory configurations:

.512K	->	512K SRAM
.640K	->	128K + 512K SRAM
.1M	->	512K + 512K SRAM
.EE08	->	1K serial EEPROM
.EE16	->	2K serial EEPROM
.EE64	->	8K serial EEPROM

For further informations about memory options and their cost please contact **grifo®**, while to easily locate the memory devices on the board please refer to figure 22.

SOFTWARE DESCRIPTION

A wide selection of software development tools can be obtained, allowing use of the **GPC® 188F** as a system for its own development, both in assembler and in other high level languages; in this way the user can easily develop all the requested application programs in a very short time. Generally all software packages available for the mounted microprocessor, or for the 8086 and derived ones, can be used. Even many of the P.C. softwares could be used on **GPC® 188F** but the numerous hardware difference make it really difficult. Below is described a brief list of the available software tools:

GET 188

It is a complete program with Editor, Communication driver, and Mass Memory management for all 80188 family cards. This program, developed by **grifo®**, allows to operate in the best conditions when **GDOS**, **FGDOS** or **FWR** software tools are used; **GET 188** is supplied when one of these tools is ordered and it is personalized with name and general data of the customer. A useful list of pull down menus and the possibility of using mouse, make program use very comfortable. **GET 188** program can be executed both on MS-DOS system and on **MACINTOSH** computers too, through SOFT-PC program. It is supplied on MS-DOS 3 1/2 floppy disk with the documentation on **GDOS 188** manual.

GDOS 188

It is a complete development tool for **GPC® 188F** card. It is supplied together with **GET 188** program to allow an easy and immediate use of this powerful development system. **GDOS** is divided in two different structures : the first one works on PC maintaining serial communication with the second one. The second structure is on EPROM, it works on board of the card and it is an efficient operating system that executes many low level functions and, at the same time, it allows high level language use. The combination of the said structures results in a complete machine, in fact the card uses PC resources (like floppy disk, hard disk, printer, keyboard, monitor, etc.) as its own peripheral devices. This resulting “virtual machine” performs operation in a transparent way for the user, so this latter can operate with the same modality of standard PC languages.

Moreover, **GDOS** can manage a portion of the on board memory as RAM disk. The on board RAM devices can directly be used performing data read and write operations with the comfortable high level file system mode.

This software tools is supplied on EPROM with MS-DOS **GET 188** floppy disk, some examples, utilities and the operating system documentation.

FGDOS 188

It is really similar to **GDOS**, but it can program and erase the on board **FLASH EPROM** with the application program developed from the user. In this way the external EPROM programmer is not necessary to store definitely the program and it is possible to modify or to add code directly on the installed machine, through a portable PC.

This software tools is supplied on **FLASH EPROM** with MS-DOS **GET 188** floppy disk, some examples, utilities and the operating system documentation.

PASCAL 188

It is an efficient and complete **PASCAL Compiler** for **8086** family cards, with features similar to Release 3.0 of Borland **Turbo PASCAL**. It must work together with any **GDOS** version and it takes advantages of its RAM disk management in fact all the high level instructions for file system can be directly used. This compiler generates an optimized code that is saved and executed on EPROM or FLASH EPROM and it requires very small RAM data area.

The terminal emulation of **GET 80** program supports the typical full screen PASCAL Editor, including the attributes management.

This program is supplied saved on **GDOS EPROM** or **FLASH EPROM** and on MS-DOS floppy disk with some examples and manual.

GCTR 188F

Complete software tools for **GPC® 188F** cards that allows application program development by using a standard Borland C, C++ compiler and an external P.C. A very powerful remote symbolic and source debugger let the user download and test the compiled program with the same comfortable manners of an in circuit emulator, through a simple serial connection. At the end of debug phase the generated code can be saved in EPROM or FLASH EPROM, reducing the RAM usage.

This program is supplied saved on **EPROM** or **FLASH EPROM** and on MS-DOS floppy disks with some examples and a user manual.

FWR188

It is a really interesting utility program capable to save files saved on P.C. disks on the FLASH EPROM installed on the board. It is sufficient a serial connection with an external P.C., where **GET188** is executed, and the card is capable to burn its FLASH EPROM with the user selected code, data, configuration, etc. file.

This program is supplied saved on **FLASH EPROM** and on MS-DOS floppy disks with **GET188** program.

HI TECH C 86

Professional C Cross Compiler of Hi-Tech Software Inc. This Compiler is terribly fast and it generates a small quantity of code. This result is due to advanced techniques in optimizing the generated code based on Artificial Intelligence techniques which allow to get a very compact and very fast code. The package includes: IDE, Compiler, Code optimizer, Assembler, Linker, Remote Debugger and so on. This tool is Full ANSI/ISO Standard and Full Library Source Code. Once the porting of the Remote-Debugger module is done, this tool allows the user the software debugging directly on the hardware that he is experimenting. This type of specialization of the **Remote Debugger** is available from now and it is supplied with all **grifo®** CPU cards. This software package is on 3" 1/2 diskettes under MS-DOS format along with user manual. This version supports these following CPUs: 8088, 8086, 80186, 80188, 80286, V20, V30, V25, etc.

DDS MICRO C 86: low cost cross compiler for C source program. It is a powerful software tool that includes editor, C compiler (integer), assembler, optimizer, linker, library, and remote debugger, in one easy to use integrated development environment. There are also included the library sources and many utilities programs.

ADDRESSES AND MAPS

In this chapter are reported all informations about card use, related to software programming of **GPC® 188F**. For example the registers addresses, the memory and peripheral devices allocation are described below.

ON BOARD RESOURCES ALLOCATION

The card devices addresses are managed by a specific control logic, realized with programmable logic devices. This control logic allocates SRAM, EPROM and peripheral devices in a comfortable way for the User.

The control logic is capable to manage separately the memories mapping and the Input/Output peripherals mapping. CPU 80C188 can address directly up to 1M Bytes of memory and 64K I/O addresses, so the control logic has assigned the task to allocate the physical address of the memories inside this addressing space through a proper impagination.

Control logic of **GPC® 188F** board includes two different sections: a microprocessor inside section that generates control signals for memories and peripheral devices (/CS), and a memory management unit (MMU), external to the CPU. So the User can define size, address and mapping modalities of all the devices through the software programming of both these two sections. To let the User be able to configure correctly the **GPC® 188F** board by only the help of 80C188 technical manual (available in appendix B of this manual), in the following table the description of all the connections amongst peripheral devices and CPU signals is reported.

DEVICE	SIGNALS	ADDRESS
EPROM IC18 FLASH EPROM IC18 SRAM IC9 SRAM IC13	/UCS, /MCS0, /MCS1, /MCS2, /MCS3, /LCS, MMU (See MEMORY ADDRESSES)	-
EEPROM, WATCH DOG, MMU, LED, BT1	/PCS0	INDPCS0
SCC 85C30	/PCS1	INDPCS1
DMA	/PCS2	INDPCS2
A/D LM 12H458	/PCS3	INDPCS3
PPI 82C55	/PCS4	INDPCS4
RTC 72421	/PCS5	INDPCS5
WRITE PROTECT, DSW1	/PCS6	INDPCS6
ABACO® BUS	(See ABACO® BUS ADDRESSES)	INDBUS

FIGURE 33: PERIPHERAL DEVICES HARDWARE CONNECTIONS

Almost all the software packages developed for the **GPC® 188F** board takes care to program properly the control logic, according to the modalities described in their own documentation.

I/O ADDRESSES

The on board control logic manages the allocation of all the peripheral devices registers in the microprocessor I/O space through 7 /CS signals (/PCS0÷/PCS6) generated by the 80C188 when occur special addresses that the User can set programming the microprocessor itself. The on board peripheral devices are managed through specific registers allocated in the above mentioned I/O locations.

Next table shows names, addresses, meanings and directions of peripheral device registers (excluding the internal microprocessor ones). For a detailed description of the registers function, please refer to next chapter "PERIPHERAL DEVICES SOFTWARE DESCRIPTION".

DEVICE	REG.	ADDRESS	R/W	MEANING
W.DOG	RWD	INDPCS0+00H	R/W	Watch Dog retrigger
EEPROM	RE2	INDPCS0+00H	R/W	EEPROM serial access
MMU	MMU	INDPCS0+00H	R/W	MMU memory paging
LD3, 4	LED	INDPCS0+00H	R/W	Activity LEDs management register
BT1	BAT	INDPCS0+00H	R	Battery status acquisition register
SCC 85C30	RSB	INDPCS1+00H	R/W	Serial line B status register
	RDB	INDPCS1+01H	R/W	Serial line B data register
	RSA	INDPCS1+02H	R/W	Serial line A status register
	RDA	INDPCS1+03H	R/W	Serial line A data register
DMA	DMA	INDPCS2+00H	R/W	Disable DMA request register
A/D LM12458	IRL0÷7	INDPCS3+ 00H÷0EH (even)	R/W	Sequencer instruction register low 0÷7
	IRH0÷7	INDPCS3+ 01H÷0FH (odd)	R/W	Sequencer instruction register high 0÷7
	CNTL	INDPCS3+10H	R/W	Configuration register low
	CNTH	INDPCS3+11H	R/W	Configuration register high
	INTENL	INDPCS3+12H	R/W	Interrupt abilitation register low
	INTENH	INDPCS3+13H	R/W	Interrupt abilitation register high
	INTSTL	INDPCS3+14H	R	Interrupt status register low
	INTSTH	INDPCS3+15H	R	Interrupt status register high
	TMRL	INDPCS3+16H	R/W	Timer register low
	TMRH	INDPCS3+17H	R/W	Timer register high
	FIFOL	INDPCS3+18H	R	Conversions to FIFO register low
	FIFOH	INDPCS3+19H	R	Conversions to FIFO register high
	LIMSTL	INDPCS3+1AH	R	Limits status register low
	LIMSTH	INDPCS3+1BH	R	Limits status register high

FIGURE 34: I/O ADDRESSES TABLE - PART 1

DEVICE	REG.	ADDRESS	R/W	MEANING
PPI 82C55	PA	INDPCS4+00H	R/W	Port A data register
	PB	INDPCS4+01H	R/W	Port B data register
	PC	INDPCS4+02H	R/W	Port C data register
	RC	INDPCS4+03H	R/W	Control and command register
RTC 62421	S1	INDPCS5+00H	R/W	Units of seconds data register
	S10	INDPCS5+01H	R/W	Decines of seconds data register
	MI1	INDPCS5+02H	R/W	Units of minutes data register
	MI10	INDPCS5+03H	R/W	Decines of minutes data register
	H1	INDPCS5+04H	R/W	Units of hours data register
	H10	INDPCS5+05H	R/W	Decines of hours data register; AM/PM
	D1	INDPCS5+06H	R/W	Units of day data register
	D10	INDPCS5+07H	R/W	Decines of day data register
	MO1	INDPCS5+08H	R/W	Units of month data register
	MO10	INDPCS5+09H	R/W	Decines of month data register
	Y1	INDPCS5+0AH	R/W	Units of year data register
	Y10	INDPCS5+0BH	R/W	Decines of year data register
	W	INDPCS5+0CH	R/W	Day of week data register
	REGD	INDPCS5+0DH	R/W	D control and status register
	REGE	INDPCS5+0EH	R/W	E control and status register
REGF	INDPCS5+0FH	R/W	F control and status register	
WRITE PROTECT	WRP	INDPCS6+00H	W	SRAM write protection register
DIP SWITCH	DSW1	INDPCS6+00H	R	Dip Switch acquisition register

FIGURE 35: I/O ADDRESSES TABLE - PART 2

Please remark that the previous tables report the only description of external peripheral devices registers, for further informations about microprocessor internal registers please refer to the manufacturer documentation or to appendix B of this manual.

The values of the base addresses (INPCS_n) is set by the User through the programming of the control signals generation section. The documentation of software tools and packages configured to work with **GPC® 188F** report the previous table with absolute addresses.

For a detailed description of the registers function, please refer to next chapter "PERIPHERAL DEVICES SOFTWARE DESCRIPTION".

ABACO® BUS ADDRESSES

The **GPC® 188F** control logic defines **ABACO®** BUS addresses and only these addresses must be used to manage correctly the BUS. The addresses range **0000H÷FFFFH** is available for **ABACO®** BUS, except for the seven addresses taken by the /CS signals (/PCS0÷/PSC6). Any I/O operations at each one of these addresses enables the /IORQ signal and the other control signals of K1 connector. The last four K Bytes (F000H÷FFFFH) on the total of 64K Bytes are reserved to CPU internal registers = control block. To avoid conflicts the User will have to program properly the "relocation" internal register to map such internal register into that addresses range (for example the default setting for this register, 20FFH, allocates the control block from FF00H). For further information please refer to appendix B of this manual.

MEMORY ADDRESSES

The total 2056K Byte of memory supported by the card are divided this way:

- Up to 1024K Byte of EPROM or 512K Byte of FLASH EPROM allocated in the memory space;
- Up to 512K Byte of SRAM installed on IC9 allocated in the memory space;
- Up to 512K Byte of SRAM installed on IC13 allocated in the memory space;
- Up to 8K Byte of serial EEPROM allocated in the I/O space.

GPC® 188F can directly manage up to 1024K bytes of memory that is the microprocessor logic addressable space. On the board this physical space can be divided in several segments: each one of these segments has software programmable dimension and start/end address. The MMU circuitry divides the space directly managed by the microprocessor into 768K Bytes long pages directly manageable by the CPU through the 6 signals /LCS, /MCS0÷/MCS3, /UCS; these signals are managed by the microprocessor inside control signals generation circuit, whose task is to divide the CPU directly addressable memory space in segments and to allocate them in the physical devices addressable space. Programming properly the specific registers of these two circuits, the User can address directly the memories installed on IC9, IC13 and IC18, even deciding the access modalities and timings.

The following figures describe available memory configurations; for further information on control signals generation circuitry, please refer to appendix B, while for further information about MMU circuitry please refer to the following paragraph.

When a power on or a reset occurs, the control logic is programmed to allocate a segment 1K Byte long at the end of the CPU physical addressing space, so the board starts executing the code located at the logical address FFFF0H that coincides with the last 16 Byte long segment of EPROM or FLASH EPROM installed on IC18.

Some software packages, like GDOS and GCTR, manage in autonomy the memory management circuits without bothering the User.

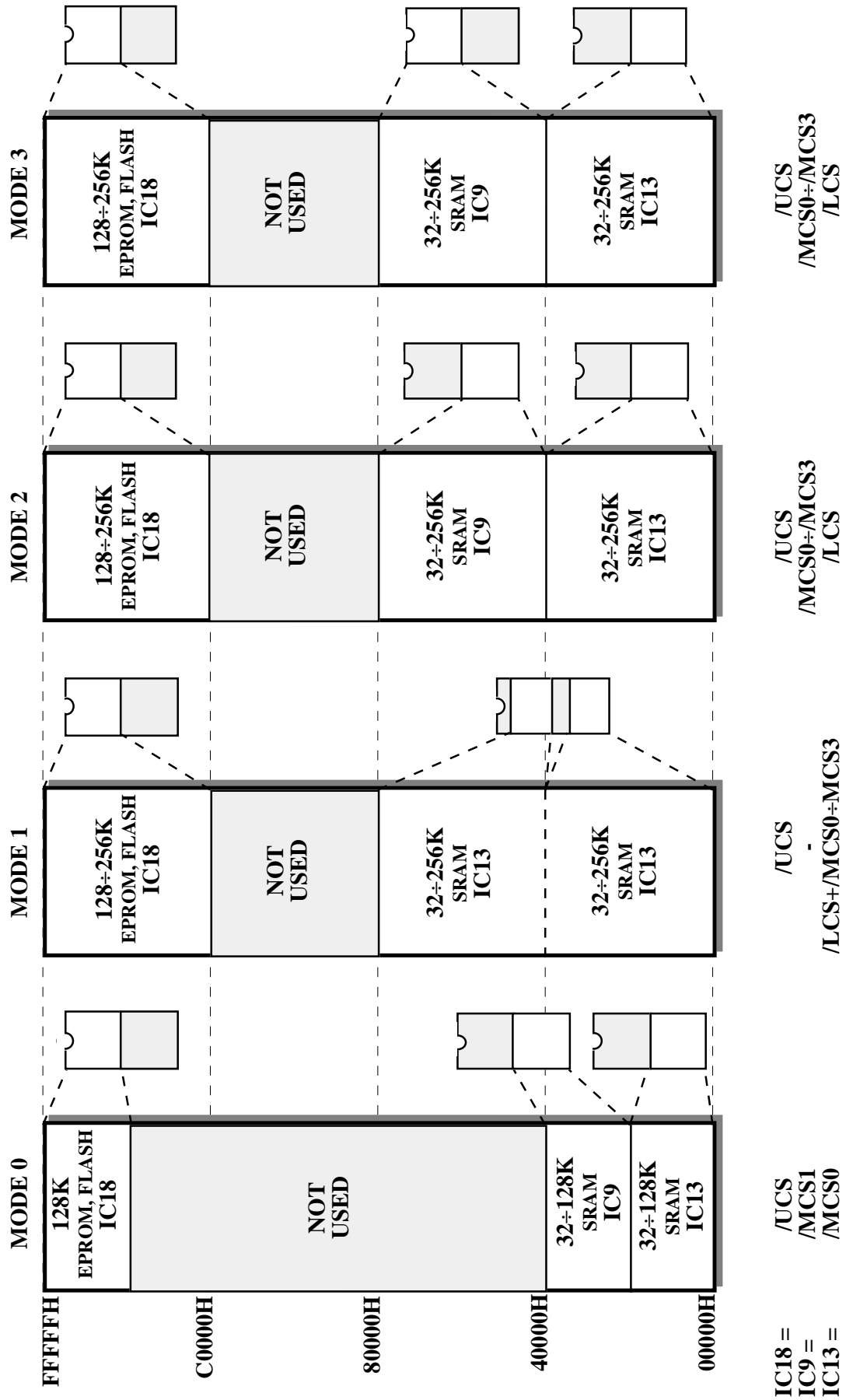


FIGURE 36: MEMORY ADDRESSES - PART 1

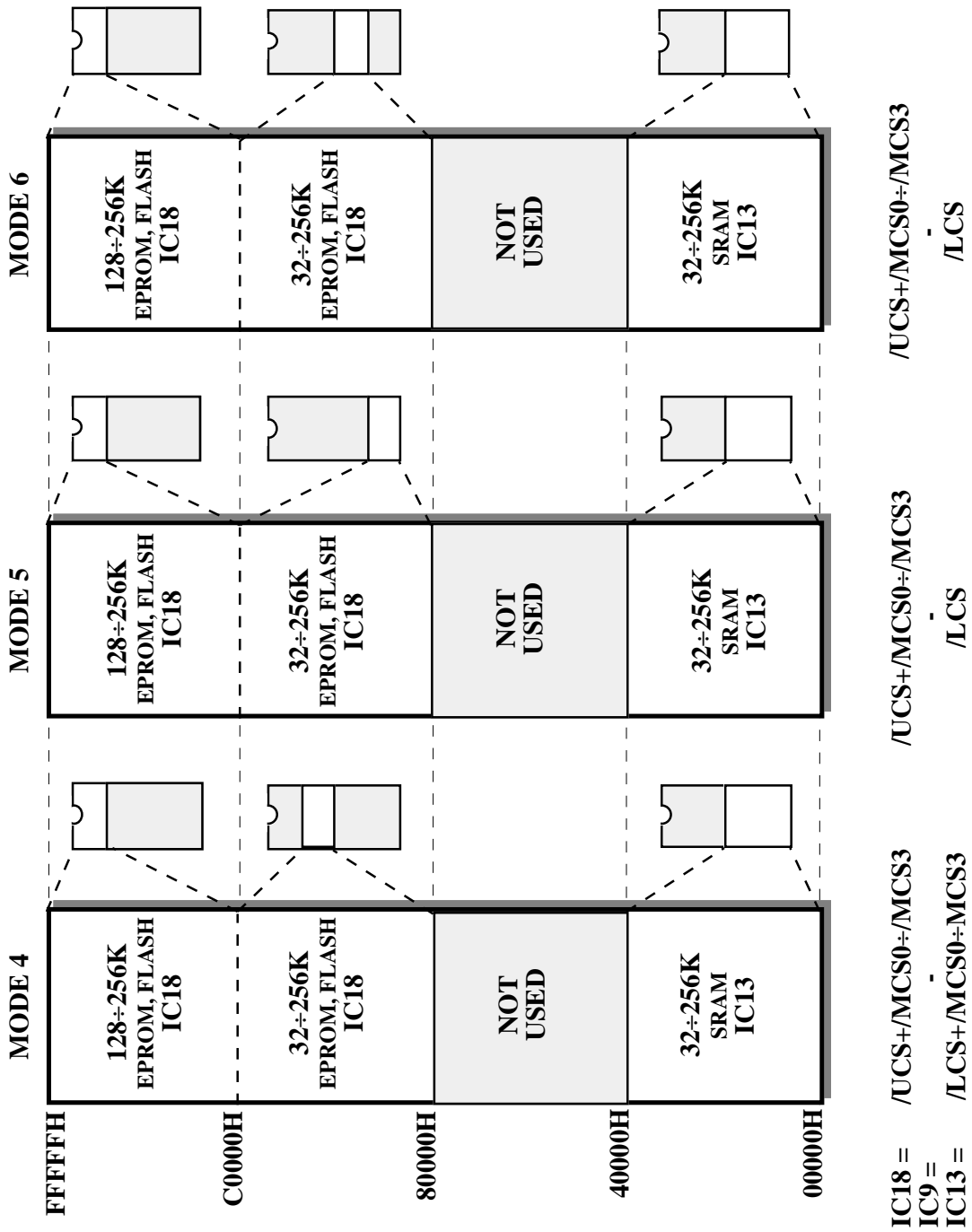


FIGURE 37: MEMORY ADDRESSES - PART 2

PERIPHERAL DEVICES SOFTWARE DESCRIPTION

In the previous paragraphs are described the external registers addresses, while in this one there is a specific description of registers meaning and function (please refer to I/O addresses table, for the registers names and addresses values). For a more detailed description of the devices, please refer to manufacturing company documentation; for microprocessor internal peripheral devices, not described in this paragraph, refer to appendix B. In the following paragraphs the **D7÷D0** and **.0÷7** indications denote the eight bits of the combination used in I/O operations.

MEMORY MANAGEMENT UNIT

An efficient MMU circuitry takes care to allocate in the CPU addressing space all the memory devices that can be installed on **GPC® 188F**. The task of this section is to page the memory to be able to allocate different physical devices area in the same CPU logical area, improving the addressing capability. The paging modalities have been decided by **grifo®** to satisfy the needs of the software packages that can be used with **GPC® 188F** board and cannot be changed unless for specific User requirements.

The section can be programmed through the specific register called MMU which is allocated into the I/O space:

MMU.5	->	first paging signal	M0
MMU.6	->	second paging signal	M1
MMU.7	->	third paging signal	M2

where:

M2	M1	M0	= Select the paging mode
0	0	0	-> Modo 0
0	0	1	-> Modo 1
0	1	0	-> Modo 2
0	1	1	-> Modo 3
1	0	0	-> Modo 4
1	0	1	-> Modo 5
1	1	0	-> Modo 6
1	1	1	-> Not used (available for future expansions)

Please refer to figures 36 and 37 for further informations about the 7 memory configuration modalities that the MMU section can manage.

When a reset or a Power On occur all the bits of MEM register are reset (all bits 0); this means to program the MMU section in mode 0 where the first 32÷128K addressed by the CPU coincide with the first 32÷128K of SRAM installed on IC13, the second 32÷128K addressed by the CPU coincide with the first 32÷128K of SRAM installed on IC9 and the last 128K addressed by the CPU coincide with the last 128K of EPROM or FLASH EPROM installed on IC18.

The status of the paging signals can be also acquired by software performing an input operation from the MMU register and examining the corresponding bits.

A/D CONVERTER

Please refer to appendix B where the software description of A/D Converter LM 12H458 is reported. Should these informations be still insufficient it is suggested to consult the manufacturer technical documentation.

WATCH DOG

Retrigger operation of **GPC® 188F** external Watch Dog circuit is performed with an input and/or output operation at the address of register RWD. To avoid Watch Dog activation it is indispensable to perform retrigger operations at regular time periods and the duration of these periods must be smaller than intervention time. If retrigger doesn't happen as before described and jumper J5 is connected in position 2-3, when intervention time is elapsed, the card is reset. By default the intervention time is about 1.4 s and jumper J5 does NOT connect external Watch Dog to reset circuitry.

Please remark that data read from or written to the RWD register does not influence the Watch Dog itself but may influence the other devices whose registers are allocated at the same address.

SERIAL EEPROM

For informations about the management of serial EEPROM module installed on IC23, please refer to the specific documentation of the component. This technical manual reports no further informations about the serial EEPROM management because this activity employs a very deep knowledge of the device itself. For this, its complete management is affordable through the high level instructions of the software package being used.

Please remark that the first 32 bytes (0÷31) are reserved so the User should avoid to modify their value. Control logic allows serial EEPROM software management through some bits of **RE2** register, these are the connections;

RE2.0	->	DATA signal	(SDA)
RE2.1	->	CLOCK signal	(SCL)

Known the serial EEPROM management circuitry hardware implementation, please remark that signals **A0,A1,A2** of this device's slave address are all set to logic **0**. Bit logic status 0 corresponds to low logic status (=0V) of the corresponding signal, while bit logic status 1 corresponds to high logic status (=1V).

The remaining six bits of RE2 register are used to manage MMU, activity LED and battery status, so every output operation to this register must respect the previous programming of these devices.

When a reset or a Power On occur all the bits of RE2 register are reset (all bits 0), so after one of these events the EEPROM signals are low.

The status of EEPROM management signals can be acquired by software performing an input operation from the RE2 register and examining the corresponding bits.

BATTERY STATUS

Status of on board battery BT1 installed on **GPC® 188F** can be acquired by software, performing a simple input operation from the address of BT register and extracting bit D4, that has the following meaning:

BAT.4 = 0	->	battery discharged	(<2.265 V)
BAT.4 = 1	->	battery charged	(> 2.265 V)

For further informations about on board battery and back up circuitry please refer to proper previous paragraphs.

CONFIGURATION INPUTS

GPC® 188F is provided with 12 software acquirable User settable configuration inputs divided as follows.

Dip Switch DSW1 can be acquired by software, performing a simple input operation from the address of DSW1 register. This is the correspondance between Dip Switch signals and DSW1 bits:

DSW1.7	->	Dip Switch 8
DSW1.6	->	Dip Switch 7
DSW1.5	->	Dip Switch 6
DSW1.4	->	Dip Switch 5
DSW1.3	->	Dip Switch 4
DSW1.2	->	Dip Switch 3
DSW1.1	->	Dip Switch 2
DSW1.0	->	Dip Switch 1

The signals are in complemented logic, this means that a dip **ON** gives a logic status **0** on the corresponding bit, while a dip **OFF** gives a logic status **1**.

Configuration jumpers J16÷J19 are connected to four hardware handshake signals of SCC 85C30 and can be acquired by software performing an input operation from the status registers of the two sections (RSA and RSB); this is the correspondance:

J16	->	RSA.3
J17	->	RSB.3
J18	->	RSA.4
J19	->	RSB.4

When a jumper is **not connected** it gives logic status **0** to the corresponding bit, when the jumper is **connected** it gives logic status **1**. For further informations about the acquisition of SCC 85C30 status registers, please refer to the proper paragraph.

Jumper J18 (RUN/DEBUG) works as selector of RUN modality (not connected) or DEBUG modality (connected). This feature is used by some of **grifo®** software packages.

ACTIVITY LEDS

On board control logic allows the management of two activity LEDs, called LD3 and LD4, through as many bits of the LED register:

LED.2 -> set LD4 status
LED.3 -> set LD3 stats

A LED can be lit performing an output operation to the corresponding bit of **LED** register resetting it to logical **0**. Of course, the LED can be turned off through the same output operation with the corresponding bit set to logical **1**.

Please remark that register LED has the same allocation address of registers MMU, BAT and RE2 so every write operation to the bits of this register must consider the effects on the other devices. Register LED is reset (all bits 0) when a reset or a power on occur, so after one of such events LEDs are turned on.

The status of the activity LED can be acquired by software performing an input operation at the address of register **LED** and examining the corresponding bits.

WRITE PROTECTION

SRAM installed on IC9 can be write protected. When write protection is enabled every output operation to this device must be performed like this:

- perform an output to the allocation address of register WRP, to disable protection;
- perform the output to the memory.

The data to write into the WRP register is meaningless, in fact the output operation itself is enough to disable the protection. Whenever any output operation to memory on IC9 is performed the protection is restored, to warrant maximum safety against accidental write operations.

Please remark that jumper J2 enables or disables write protection circuitry so when J2 is not connected memory on IC9 is not write protected and can be managed as normal SRAM.

Some software packages, like **GDOS**, manage in autonomy the write protection circuit without bother in the User.

After a reset or a power on write protection is enabled.

DISABLE DMA REQUEST

Data to transmit through serial line B can be transferred by CPU DMA channel 1, to avoid stealing time to the application program. By performing an input or output to allocation address of DMA register the SCC 85C30 data transfer request is disabled. So by software it is possible to disable the data transfer from memory to serial line B when no data need to be transferred without disabling CPU DMA channel 1 also.

After a reset or a power on this DMA request is disabled.

REAL TIME CLOCK

This peripheral is allocated in 16 consecutives I/O addresses, 3 of which correspond to status registres while the remaining 13 are for datas. Data registers are used both for read operations (of the current time and date) and write operations (to initialize the time and date) just like the status registers which are used in write operations (to program the operative mode) and in read operations (to acquire the RTC status). Here follows a list of the Real Time Clock data registers' meanings:

SEC1	- Units of seconds	- 4 least significant bits of SEC1.3÷SEC1.0
SEC10	- Decines of secondi	- 3 least significant bits of SEC10.2÷SEC10.0
MIN1	- Units of minutes	- 4 least significant bits of MIN1.3÷MIN1.0
MIN10	- Decines of minutes	- 3 least significant bits of MIN10.2÷MIN10.0
HOU1	- Units of hours	- 4 least significant bits of HOU1.3÷HOU1.0
HOU10	- Decines of hours	- 2 least significant bits of HOU10.1÷HOU10.0 The third bit of HOU10.2 indicates AM/PM
DAY1	- Units of day number	- 4 least significant bits of DAY1.3÷DAY1.0
DAY10	- Decines of day number	- 2 least significant bits of DAY10.1÷DAY10.0
MON1	- Units of month	- 4 least significant bits of MON1.3÷MON1.0
MON10	- Decines of month	- 1 least significant bit of MON10.0
YEA1	- Units of year	- 4 least significant bits of YEA1.3÷YEA1.0
YEA10	- Decines of year	- 4 least significant bits of YEA10.3÷YEA10.0
WEE	- Day of the week	- 3 least significant bits of WEE.2÷WEE.0

For this last register the three least significant bits mean:

WEE.2	WEE.1	WEE.0	Day of the week
0	0	0	Sunday
0	0	1	Monday
0	1	0	Tuesday
0	1	1	Wednesday
1	0	0	Thursday
1	0	1	Friday
1	1	0	Saturday

The meaning of the three control registers is:

bit 7 6 5 4 3 2 1 0

REG D = NU NU NU NU 30S IF B H

where:

NU = Not used.

30S = If high (1) it allows a 30 seconds correction of the time. Once set the RTC seconds are reset and the minutes increased, if the previous seconds was equal or greater than 30.

IF = It manages the RTC interrupt status. When read it shows the current interrupt status (1 active and viceversa), when reset to 0 it disables the RTC interrupt signal if the interrupt mode is selected.

B = Indicates whether R/W operations can be performed on the registers:

1 -> operations are not permitted and viceversa.

H = If high (1) it stores the written time and date.

bit 7 6 5 4 3 2 1 0
REG E = NU NU NU NU T1 T0 I M

where:

NU = Not used.

T1 T0 = Determin the duration of the internal counters interrupt cycle.

0 0 -> 1/64 second

0 1 -> 1 second

1 0 -> 1 minute

1 1 -> 1 hour

I = It defines the interrupt operating mode:

1 -> it selects interrupt mode: when the selected duration elapses the interrupt is enabled and then disabled only with a reset of bit IF of control register D;

0 -> it selects the standard mode: when the selected duration elapses the interrupt is enabled and then disabled only after 7,8 msec.

M = It mask the interrupt status:

1 -> interrupt masked: the RTC interrupt signal is always disabled;

0 -> interrupt not masked: the RTC interrupt signal reflects interrupt status.

bit 7 6 5 4 3 2 1 0
REG F = NU NU NU NU T 24/12 S R

where:

NU = Not used.

T = It determines from which internal counter to take the counting signal:

1 -> main counter (fast counter for test);

0 -> 15th counter.

24/12 = It determines the hours counting mode:

1 -> 0÷23;

0 -> 1-12 with AM/PM.

S = If high (1) it stops the clock time counting until the next enabling (0).

R = If high (1) it resets all the internal counters.

After a reset or a power on occur, the RTC is not reinitialized, in order to warrant the conservation of its content, after such events, through the back up circuitry.

SCC 85C30

This peripheral manages two independent serial lines, called A and B. Communication mode can be:

- Synchronous
- Asynchronous
- SDLC/HDLC

This peripheral is managed through 4 registers allocated in the I/O addressing space. Two of these registers, RSA and RSB, are used to set and get the peripheral status (one for each line), while registers RDA and RDB are used for data transfer. Both status and data registers can be accessed in input (to acquire the peripheral status or received data) and in output (to program the peripheral or to data to send). SCC 85C30 uses an indirect addressing mode to its internal registers; this consists in writing proper values to the status register to reach the whole set of internal registers. These latter are 16 writable and 9 readable, and are described in the following pages .

Write register 0 (Command Register)

This register is used for controlling the peripheral, resetting certain of its states and pointing to other internal registers:

D7 D6 D5 D4 D3 D2 D1 D0

WR0 = CRC1 CRC0 CD2 CD1 CD0 P2 P1 P0

where:

CRC1 CRC0 = CRC reset codes 0 and 1

0 0 -> Null code

0 1 -> Reset receive CRC checker

1 0 -> Reset transmit CRC generator

1 1 -> Reset transmit CRC Underrun/End of transmission latch

CD2 CD1 CD0 = Command code selection

0 0 0 -> Null code

0 0 1 -> Point to high registers WR8÷WR15, RD8÷RD15

0 1 0 -> Reset external/status interrupts

0 1 1 -> Send Abort in SDLC mode

1 0 0 -> Enable interrupt on next received char

1 0 1 -> Reset transmission interrupt pending

1 1 0 -> Error reset

1 1 1 -> Reset highest priority Interrupt Under Service (IUS)

P2 P1 P0 = Register selection code, used in conjunction with "Point to high registers" command code (see above)

0 0 0 -> WR0, RD0, WR8, RD8

0 0 1 -> WR1, RD1, WR9

0 1 0 -> WR2, RD2, WR10, RD10

0 1 1 -> WR3, RD3, WR11

1 0 0 -> WR4, WR12, RD12

1 0 1 -> WR5, WR13, RD13

1 1 0 -> WR6, WR14

1 1 1 -> WR7, WR15, RD15

Write register 1 (Interrupt and data transfer mode definition)

Transmit /Receive interrupt and data transfer mode definition; Wait/Ready mode definition:

D7 D6 D5 D4 D3 D2 D1 D0

WR1 = AWR W/R R/T IM1 IM0 P AIT AIE

where:

AWR = Wait/Request pin enable: AWR=0 -> disabled

W/R = Wait/Request function enable: W/R=0 -> /Wait function

R/T = Wait/Rquest function on receive or transmit: R/T=0 -> transmit

IM1 IM0 = Receive interrupt modes selection

0 0 -> Receive interrupts disabled

0 1 -> Receive interrupt on first character or special condition

1 0 -> Receive interrupt on all characters or special condition

1 1 -> Receive interrupt on first character or special condition

P -> Enable parity not matching as special condition: P=1 -> enabled

AIT -> Enable interrupt when transmit buffer becomes empty: AIT=1 -> enabled

AIE -> External/status master interrupt enable: AIE=1 -> enabled

Write register 2 (Interrupt vector)

Only one interrupt vector exists but it can be accessed through either channels:

D7 D6 D5 D4 D3 D2 D1 D0

WR2 = V7 V6 V5 V4 V3 V2 V1 V0

where:

Vn = Interrupt vector n-th bit

Write register 3 (Receive parameters and control)

This register contains control bits and parameter for receiver logic:

D7 D6 D5 D4 D3 D2 D1 D0

WR3 = R1 R0 AA IF AR RI CS A

where:

R1 R0 = Number of bits per character

0 0 -> 5 bit

0 1 -> 6 bit

1 0 -> 7 bit

1 1 -> 8 bit

AA = Handshake enables: AA=1 -> enabled

IF = Enable hunt mode for synchronization: IF=1 -> enabled

AR = Receiver CRC enable: AR=1 -> enabled

RI = Enable address search mode for SDLC: RI=1 -> enabled

CS = SYNC character load inhibit: CS=1 -> inhibited

A = Receiver enable (set as after all other bits): A=1 -> enabled

Write register 4 (Miscellaneous parameters)

Receiver/transmitter miscellaneous parameters and modes:

D7 D6 D5 D4 D3 D2 D1 D0

WR4 = VC1 VC0 MS1 MS0 BS1 BS0 P/D P

where:

VC1 VC0 = Clock rate 1 and 0

0 0 -> Data rate = clock rate (BITRATE=1)

0 1 -> Data rate = 1/16 fclock rate (BITRATE=16)

1 0 -> Data rate = 1/32 clock rate (BITRATE=32)

1 1 -> Data rate = 1/64 clock rate (BITRATE=64)

MS1 MS0 = SYNC modes 1 and 0

0 0 -> 6 or 8 bits character synchronization (see Sync bit in WR10)

0 1 -> 12 or 16 bits character synchronization (see Sync bit in WR10)

1 0 -> SDLC mode (flag to WR7 is 01111110)

1 1 -> External sync mode

BS1 BS0 = Stop bits 1 and 0 in asynchronous mode

0 0 -> Synchronous modes selection

0 1 -> 1 stop bit per character

1 0 -> 1+1/2 stop bits per character

1 1 -> 2 stop bits per character

P/D = Even or odd parity: P/D=1 -> even parity

P = Parity check enable: P=1 -> enabled

Write register 5 (Transmit parameters and control)

This register contains the bits that influence the transmitter behaviour (C/S influences also the receiver):

D7 D6 D5 D4 D3 D2 D1 D0

WR5 = DTR BC1 BC0 IB AT C/S RTS A

where:

DTR	= /DTR pin enable: DTR=1 -> /DTR is active when low
BC1 BC0	=Number of bits per character in transmission selection
0 0	-> 5 or less bits
0 1	-> 7 bits
1 0	-> 6 bits
1 1	-> 8 bits
IB	= Send break: IB=1 -> send
AT	= Transmit enable: AT=1 -> enabled
C/S	= Polynomial CRC selection: C/S=1 -> CRC 16 polynomial C/S=0 -> SDLC polynomial
RTS	= /RTS pin enable: RTS=1 -> /RTS is active when low
A	= Transmit CRC enable: A=1 -> enable

Write register 6 (Sync characters or SDLC address field)

This register contains the Sync character in monosync mode, the first byte of 16-bits sync character in external sync mode or the secondary address field used in SDLC mode:

D7 D6 D5 D4 D3 D2 D1 D0

WR6 = S7 S6 S5 S4 S3 S2 S1 S0

where:

S7	S6	S5	S4	S3	S2	S1	S0	
SYNC7	SYNC6	SYNC5	SYNC4	SYNC3	SYNC2	SYNC1	SYNC0	-> Monosync 8 bits
SYNC1	SYNC0	SYNC5	SYNC4	SYNC3	SYNC2	SYNC1	SYNC0	-> Monosync 6 bits
SYNC7	SYNC6	SYNC5	SYNC4	SYNC3	SYNC2	SYNC1	SYNC0	-> Bisync 16 bits
SYNC3	SYNC2	SYNC1	SYNC0	1	1	1	1	-> Bisync 12 bits
ADR7	ADR6	ADR5	ADR4	ADR3	ADR2	ADR1	ADR0	-> SDLC
ADR7	ADR6	ADR5	ADR4	X	X	X	X	-> SDLC add. range

Write register 7 (SYNC character of SDLC flag)

This register contains the rest of the synchronization informations:

D7 D6 D5 D4 D3 D2 D1 D0

WR7 = S15 S14 S13 S12 S11 S10 S9 S8

where:

S15	S14	S13	S12	S11	S10	S9	S8	
SYNC7	SYNC6	SYNC5	SYNC4	SYNC3	SYNC2	SYNC1	SYNC0	-> Monosync 8 bits
SYNC5	SYNC4	SYNC3	SYNC2	SYNC1	SYNC0	X	X	-> Monosync 6 bits
SYNC15	SYNC14	SYNC13	SYNC12	SYNC11	SYNC10	SYNC9	SYNC8	-> Bisync 16 bits
SYNC11	SYNC10	SYNC9	SYNC8	SYNC7	SYNC6	SYNC5	SYNC4	-> Bisync 12 bits
0	1	1	1	1	1	1	0	-> SDLC

Write register 8 (Transmit buffer)

This is the transmit buffer register:

D7 D6 D5 D4 D3 D2 D1 D0

WR8 = TX7 TX6 TX5 TX4 TX3 TX2 TX1 TX0

where:

TXn = n-th bit of data to transmit

Write register 9 (Master interrupt control)

This register contains the interrupt control bits and allows to reset the USRT channels:

D7 D6 D5 D4 D3 D2 D1 D0

WR9 = R1 R0 0 SH/L MIE DLC NV VIS

where:

R1 R0 = UART reset command bits
 0 0 -> No reset
 0 1 -> Channel B reset
 1 0 -> Channel A reset
 1 1 -> Force hardware reset
 0 = Not used (must be zero)
 SH/L = Which vector bits modify to indicate status: SH/L=1 -> modify V6÷V4
 SH/L=0 -> modify V3÷V1
 MIE = Master interrupt enable: MIE=1 -> interrupts enabled
 DLC = Disable lower daisy chain: DLC=1 -> disabled
 NV = Disable interrupt vector output : NV=1 -> disabled
 VIS = Interrupt vector includes status bits: VIS=1 -> variable vector

Write register 10 (Miscellaneous control bits)

Miscellaneous Transmitter/Receiver control bits:

D7 D6 D5 D4 D3 D2 D1 D0

WR10 = CRC FM1 FM0 GP MFI AFU LM S

where:

CRC = CRC presets: CRC=1 -> CRC preset to 1
 CRC=0 -> CRC preset to 0
 FM1 FM0 = Data encoding 1 and 2
 0 0 -> Modo NRZ
 0 1 -> Modo NRZI
 1 0 -> Modo FM1 (transition high)
 1 1 -> Modo FM0 (transition low)
 GP = Go active on POLL: GP=1 -> enabled
 MFI = SDLC idle line condition: MFI=1 -> send "1s"
 MFI=0 -> send flags
 AFU = Enable SDLC send abort on transmit underrun : AFU=1 enabled
 LM = Enable loop mode: LM=1 -> enabled
 S = Select SYNC char length: S=1 -> 6 bits
 S=0 -> 8 bits

Write register 11 (Clock mode control)

This register allows to select the source of both the receive and transmit clock:

D7 D6 D5 D4 D3 D2 D1 D0

WR11 = XT RC1 RC0 TC1 TC0 TR TR1 TR0

where:

XT clock source on pin /RTxC: XT=1 -> source is quartz crystal
 XT=0 -> source is TTL-compatible signal

RC1 RC0 = Receiver clock 1 and 0
 0 0 -> Receive clock = pin /RTxC
 0 1 -> Receive Clock = pin /TRxC
 1 0 -> Receive Clock = baud rate generator output
 1 1 -> Receive Clock = DPLL output

TC1 TC0 = Transmit clock 1 and 0
 0 0 -> Transmit Clock = pin /RTxC
 0 1 -> Transmit Clock = pin /TRxC
 1 0 -> Transmit Clock = baud rate generator output
 1 1 -> Transmit Clock = DPLL output

TR = Select direction of pin /TRxC: TR=0 -> input
 TR=1 -> output

TR1 TR0 = /TRxC output source
 0 0 -> XTAL oscillator output
 0 1 -> transmit clock
 1 0 -> baud rate generator output
 1 1 -> DPLL output

Write registers 12 and 13 (Baud rate generator time constant)

These registers contain the baud rate generator time constant:

D7 D6 D5 D4 D3 D2 D1 D0

WR12 = TC7 TC6 TC5 TC4 TC3 TC2 TC1 TC0

WR13 = TC15 TC14 TC13 TC12 TC11 TC10 TC9 TC8

The time constant to write in registers 12 and 13 can be calculated using the following formula:

$$TC = (11059200 / (2 * BITRATE * BAUDRATE)) - 2$$

where BAUDRATE is the desired rate in bits per second and BITRATE is the value written in write register 4. Suggested value is 16.

Write register 14 (Miscellaneous control bits)

This register contains miscellaneous control bits:

D7 D6 D5 D4 D3 D2 D1 D0

WR14 = C2 C1 C0 LL AE DTR BRS BRE

where:

C2 C1 C0	= Digital palse-locked loop (DPLL) command bits
0 0 0	-> Null command
0 0 1	-> Enter search mode
0 1 0	-> Reset missing clock
0 1 1	-> Disable DPLL
1 0 0	-> Set baud rate generator as clock source
1 0 1	-> Set pin /RTxC as clock source
1 1 0	-> Set FM mode
1 1 1	-> Set NRZI mode
LL	= Enable local loopback: LL=1 -> enabled
AE	= Enable auto echo: AE=1 -> enabled
DTR	= Enable /DTR: DTR=1 -> enabled
BRS	= select baud rate generator source: BRS=1 -> pin PCLK BRS=0 -> /RTxC or XTAL
BRE	= Enable baud rate generator: BRE=1 -> enable

Write register 15 (External/status interrupt control)

This register selects special interrupt sources:

D7 D6 D5 D4 D3 D2 D1 D0

WR15 = BA TU CTS SU DCD 0 ZC 0

where:

BA	= Generates interrupt on break/abort
TU	= Generates interrupt on transmitter underrun/EOM
CTS	= Generates interrupt on /CTS status variation
SH	= Generates interrupt on SYNC pin or hunt bit variation
DCD	= Generates interrupt on /DCD pin variation
0	= Not used (must be zero)
ZC	= Generates interrupt when baud rate generator counter reaches 0

The above described states are referred to a logic status "1" of the corresponding bit.

Read register 0 (Buffer and external status)

This register contains bits that report receiver and transmitter buffer status and external status:

D7 D6 D5 D4 D3 D2 D1 D0

RD0 = BA TU CTS SU DCD TBE ZC RCA

where:

BA	= Break/abort occurred: BA=1 -> occurred
TU	= Transmission underrun/EOM: TU=1 -> occurred
CTS	= pin /CTS status
SH	= pin /SYNC or hunt status
DCD	= pin /DCD status
TBE	= Transmission buffer status: TBE=1 -> empty
ZC	= Baud rate generator zero count: ZC=1 -> zero reached
RCA	= Receive character available: RCA=1 -> available

Read register 1

This register contains special receive condition status bits and the residue code for the I-field in SDLC mode:

D7 D6 D5 D4 D3 D2 D1 D0

RD1 = EOF CRC ROE PE RC0 RC1 RC2 AS

where:

EOF = End of frame (SDLC)
 CRC = CRC or framing error
 ROE = Receiver overrun error
 PE = Receive parity error
 RC0 = SDLC residue code 0
 RC1 = SDLC residue code 1
 RC2 = SDLC residue code 2
 AS = Everithing sent

The above described states are referred to a logic status "1" of the corresponding bit.

Read register 2

This register contains the value of the interrupt vector:

D7 D6 D5 D4 D3 D2 D1 D0

RD2 = V7 V6 V5 V4 V3 V2 V1 V0

where:

Vn = n-th bit of the interrupt vector

Read register 3

This is the interrupt pending register. It exists only on channel A but contains informations about both the channels. In channel B it always returns zero:

D7 D6 D5 D4 D3 D2 D1 D0

RD3 = 0 0 CAR CAT CAE CBR CBT CBE

where:

0 = Not used (always return a zero)
 CAR = Channel A reception interrupt pending
 CAT = Channel A transmission interrupt pending
 CAE = Channel A external/status variation interrupt pending
 CBR = Channel B reception interrupt pending
 CBT = Channel B transmission interrupt pending
 CBE = Channel B external/status variation interrupt pending

The above described states are referred to a logic status "1" of the corresponding bit.

Read register 8

This is the receive data register:

D7 D6 D5 D4 D3 D2 D1 D0

RD8 = RX7 RX6 RX5 RX4 RX3 RX2 RX1 RX0

where:

RXn = n-thbit of received data

Read register 10

This register contains some miscellaneous status bits:

D7 D6 D5 D4 D3 D2 D1 D0

RD10 = 1CM 2CM 0 LS 0 0 OL 0

where:

1CM = One clock missing
 2CM = Two clocks missing
 0 = Not used (always return a zero)
 LS = Loop sending
 OL = On loop

The above described states are referred to a logic status "1" of the corresponding bit.

Read register 12 and 13

These registers return the value stored in WR12 and WR13, that is the time constant for the baud rate generator.

D7 D6 D5 D4 D3 D2 D1 D0

RD12 = TC7 TC6 TC5 TC4 TC3 TC2 TC1 TC0
RD13 = TC15 TC14 TC13 TC12 TC11 TC10 TC9 TC8

Read register 15

This register reflects the value stored in write register 15, the external/status interrupt enable bits. For the meanings of such bits please refer to the description of write register 15. The two unused bits always return zero.

D7 D6 D5 D4 D3 D2 D1 D0

RD15 = BA TU CTS SU DCD 0 ZC 0

After a reset or a power on SCC 85C30 disables both the serial lines, just like their hardware signals.

PPI 82C55

This external peripheral device is managed through 4 registers: one status register (CNT) and three data registers (PDA, PDB, PDC). The data registers are available both for input operation (to obtain signal status) and for output operation (to set signal status) with the correspondence described in figure 35. The PPI 82C55 can work in three different modes:

MODE 0 = it provides two 8 bits bidirectional ports (A,B) and two 4 bits bidirectional ports (C LOW, C HIGH); output signals are latched and input signals are not latched; no handshake signals are provided.

MODE 1 = it provides two 12 bits ports (A+C LOW and B+C HIGH) where ports A and B are used as 8 I/O lines and port C are used as 4 handshake lines. Both inputs and outputs are latched.

MODE 2 = it provides a 13 bits port (A+C \div 7) where port A is used as 8 I/O lines and the 5 bits of port C are used as handshake, and a 11 bits port (B+C0 \div 2) where port B is used as 8 I/O lines and the 3 bits of port C are used as handshakes. Both inputs and outputs are latched.

The device is programmed writing an 8 bits word in the status register CNT, with the following bits meaning:

CNT = SF M1 M2 A CH M3 B CL

where:

SF = mode Set Flag: if activated (1) the device is enabled for standard I/O operation

M1 M2 = mode selection:

0 0 = mode 0

0 1 = mode 1

1 X = mode 2

A = port A direction: 1=input; 0=output

CH = port C HIGH direction: 1=input; 0=output

M3 = mode selection: 1=mode 1; 0=mode 0

B = port B direction: 1=input; 0=output

CL = port C LOW direction: 1=input; 0=output

After Reset or power on PPI 82C55 is programmed in mode 0 with all three ports in input; in this way any connection of PPI 82C55 signals can be used without conflict problems.

CPU INTERNAL DEVICES

For further informations about CPU internal devices (Timer Counter, DMA, interrupt controller, control signals generation, etc.) please refer to technical documentation on appendix B of this manual. Should these informations be still insufficient it is suggested to consult the manufacturer technical documentation.

EXTERNAL CARDS

GPC® 188F can be connected to a wide range of block modules and operator interface system produced by **grifo®**, or to many system of other companies. The on board resources can be expanded with a simple connection to the numerous peripheral **grifo®** boards, both intelligent and not, thanks to its standard **ABACO®** BUS connector. Even cards with **ABACO®** I/O BUS can be connected, by using the proper mother boards.

Hereunder some of these cards are briefly described; ask the detailed information directly to **grifo®**, if required.

KDL X24 - KDF 224

Keyboard Display LCD 2,4 rows 24 keys

Interface with Fluorescent or LCD display, LEDs backlit, 20x2 or 20x4 characters; up to 24 keys matrix keyboard connector. It is directly driven by a 20 pins **ABACO®** I/O standard connector; High level languages supported; standard pinout for telephone keyboard.

QTP 24 - QTP 24P

Quick Terminal Panel 24 keys with Parallel interface

Intelligent user panel equipped with Fluorescent or LCD display, LEDs backlit, 20x2 or 20x4 characters; RS 232, RS 422, RS 485 or current loop serial line; serial E2 for set up and message. Possibility of re-naming keys, LEDs and panel name by inserting label with new name into the proper slot; 24 Keys and 16 LEDs with blinking attribute and buzzer manageable by software; built in power supply; RTC option, reader of magnetic badge and relay. The QTP 24P is low cost no intelligent (passive) version. It is directly driven from 16 TTL I/O lines; high level languages supported.

QTP G28

Quick Terminal Panel - LCD Graphic, 28 keys

LCD display 240x128 pixels, CFC backlit; Optocoupled RS 232 line and additional RS 232/422/485/ C. L. line; CAN line controller; E² for set up; RTC and RAM lithium backed; primary graphic object; possibility of re-naming keys, LEDs and panel name; 28 keys and 16 LEDs with blinking attribute and buzzer manageable by software; Buzzer; built-in power supply; reader of magnetic badge and relay option.

ABB 03

ABACO® Block BUS 3 slots

3 slots **ABACO®** mother board; 4 TE pitch connectors; **ABACO®** I/O BUS connector; screw terminal for power supply; connection for DIN C type and Ω rails.

ABB 05

ABACO® Block BUS 5 slots

5 slots **ABACO®** mother board with power supply. Double power supply built in; 5Vdc 2,5A section for powering the on board logic; second section at 24Vdc 400mA galvanically coupled, for the optocoupled input lines. Auxiliary connector for **ABACO®** I/O BUS. Connection for DIN Ω rails.

MCI 64

Memory Cards Interfaces 64 MBytes

Interfacing card for managing 68 pins PCMCIA memory cards, it is directly driven from any **ABACO®** I/O standard connector; High level languages GDOS supported.

IAC 01

Interface Adapter Centronics

Interface between **ABACO**[®] standard I/O 20 pins connector and D 25 pins connector with Centronics standard pin out.

IBC 01

Interface Block Comunication

Conversion card for serial communication, 2 RS 232 lines; 1 RS 422-485 line; 1 optical fibre line; selectable DTE/DCE interface; quick connection for DIN 46277-1 and 3 rails.

OBI N8 - OBI P8

Opto BLOCK Input NPN-PNP

Interface between 8 NPN, PNP optocoupled and displayed input lines, with screw terminal and **ABACO**[®] standard I/O 20 pins connector; power supply section; connection for DIN Ω rails.

TBO 01 - TBO 08

Transistor BLOCK Output

Interface for **ABACO**[®] standard I/O 20 pins connector; 16 or 8 transistor output lines 45 Vdc 3 A open collector; screw terminal; optocoupled and displayed lines; connection for DIN 247277-1 and 3 rails.

RBO 08 - RBO 16

Relé BLOCK Output

Interface for **ABACO**[®] standard I/O 20 pins connector; 8 or 16 displayed Relays 3A with MOV; screw terminal; connection for DIN Ctype and Ω rails.

FBC 20 - FBC 120

Flat Block Contact 20 vie

Interfaccia tra 2 o 1 connettori a perforazione di isolante (scatolino da 20 vie maschi) e la filatura da campo (morsettiere a rapida estrazione). Attacco rapido per guide tipo DIN 46277-1 e 3.

DEB 01

Didactic Experimental Board

Supporting card for 16 TTL I/O lines use. It includes: 16 keys, 16 LEDs, 4 digits, 16 keys matrix keyboard, Centronics printer interface, LCD display and fluorescent display interface, **GPC**[®] 68 I/O connector, field connection with screw terminal.

IAL 42

Interface Adapter LCD

Interface between 16 I/O TTL available on I/O **ABACO**[®] standard connector and 14 pins lowprofile male connector featuring standard pin-out for fluorescent LCD displays management.

XBI 01

miXed BLOCK Input Output

Interface for **ABACO**[®] standard I/O 20 pins connector; 8 transistor output lines 45 Vdc 3A; 8 input lines; screw terminal; optocoupled and displayed I/O lines; connection for DIN 247277-1 and 3 rails.

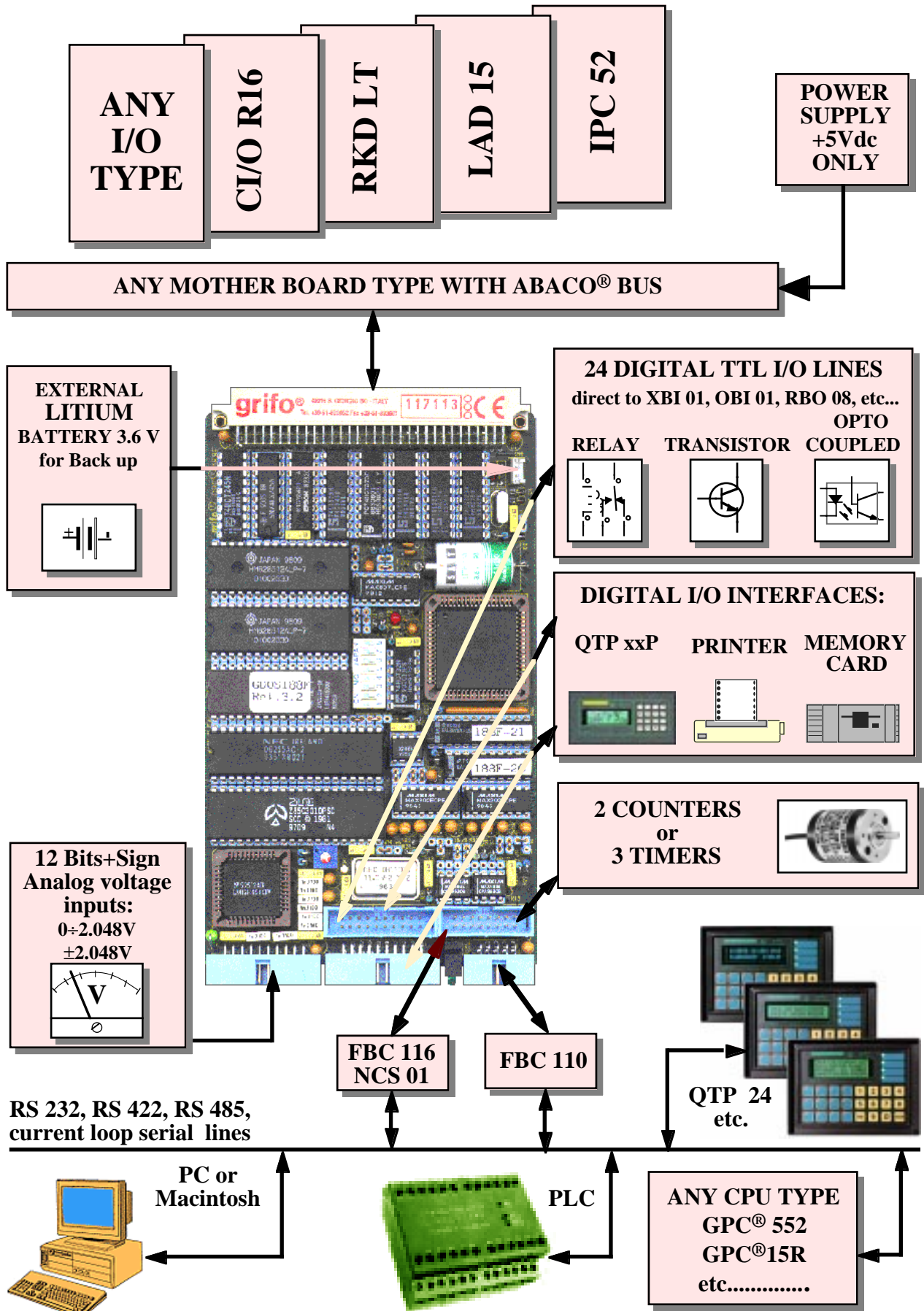


FIGURE 38: POSSIBLE CONECTIONS DIAGRAM



XBI R4 - XBI T4

miXed BLOCK Input-Output

Interface for **ABACO**[®] standard I/O 20 pins connector; 4 Relays 3A with MOV or 4 optocoupled Transistors 3A open collectors; 4 input lines optocoupled; screw terminal; connection for DIN Ctype and Ω rails.

C/O R16

16 Coupled Input Output with relays

16 optocoupled inputs with low frequency filter; standard rate +24 Vdc input voltage; 16 microrelays 1 A output lines; 24 Vac noise suppressor, type MOV; I/O displayed through LEDs.

IPC 52

Intelligent Peripheral Controller, 24 analogic input

This intelligent peripheral card acquires 24 independent analogic input lines: 8 PT 100 or PT 1000 sensors, 8 J,K,S,T termocouples, 8 analog input ± 2 Vdc or $4 \div 20$ mA; 16 bits + sign A/D section; 0.1 °C resolution; 32K RAM for local data logging; buzzer; 16 TTL I/O lines; 5 or 8 conversion per second; facility of networking up to 127 IPC 52 cards using serial line. BUS interfacing or through RS 232, RS 422, RS 485 or current loop line. Only 5Vdc power supply.

DAC 16

Digital to Analog Converter 16 bits

2 Digital to Analog converter, 16 bits galvanically insulated; programmed data displayed; ± 10 Vdc output; gain and offset setting; 8 bit Bus; standard addressing.

RKD LT

Remote Keyboard Display controller

Video terminal able to manage many different graphic LCD or alphanumeric fluorescent LCD or displays; matrix keyboard input; BUS or serial interfacing; 1 RS 232 line; additional RS 232, RS 422-485 or Current Loop line; serial EEPROM for set-up; primary graphic object; LEDs driving; Buzzer.

UCC A2

UART Communication Cards, 2 lines

2 Independent RS 232, RS 422, RS 485 or Current Loop lines. Each line: 3 characters buffer; Asynchron communication from 50 to 115K baud. Parity, bit stop and data length is software programmable.

PCI 01

Peripheral Coupled 32 Inputs

32 optocoupled input lines displayed through LEDs with Pi-Greek filter; standard rate 24 Vdc input voltage; 8/16 bits Bus extended addressing.

JMS 34

Jumbo Multifunction Support for Axis control

Generic peripheral axis control card. 3 optocoupled acquisition channels, with 16 bits bidirectional counter, for incremental encoder. 4 12bits ± 10 Vdc D/A channels. 8 Opto-in; 8 NPN Opto-output 40Vdc 500 mA. All I/O lines displayed with LEDs.

BIBLIOGRAPHY

Here follows a list of manuals that can be a source of further information about the devices installed on **GPC® 188F**.

TEXAS INSTRUMENTS Manual:	<i>The TTL Data Book - SN54/74 Families</i>
TEXAS INSTRUMENTS Manual:	<i>RS-422 and RS-485 Interface Circuits</i>
HEWLETT PACKARD Manual:	<i>Optoelectronics Designer's Catalog</i>
NEC Manual:	<i>Microprocessors and Peripherals - Volume 3</i>
NEC Manual:	<i>Memory Products</i>
AMD Manual:	<i>Flash Memory Products</i>
SGS-THOMSON Manual:	<i>Programmable Logic Manual GAL Products</i>
MAXIM Manual:	<i>New Releases Data Book - Volume IV</i>
MAXIM Manual:	<i>New Releases Data Book - Volume V</i>
XICOR Manual:	<i>Data Book</i>
ZILOG Manual:	<i>Z80 Microprocessor Family User's Manual</i>
NATIONAL SEMICONDUCTOR Manual:	<i>LM12458 12-Bit + Sign Data Acquisition System</i>
SEIKO EPSON Data Sheet:	<i>RTC-62421 Real Time Clock module</i>
ATMEL Manual:	<i>Serial Data FLASH</i>

Please connect to the manufacturers' Web sites to get the latest version of all manuals and data sheets.



APPENDIX A: ELECTRIC DIAGRAMS

This chapter shows the electric diagram of the most frequently used interfaces for **GPC® 188F**. Every one of these interfaces can be made by the User in autonomy, while only few of them are **grifo®** standard boards and can be ordered.

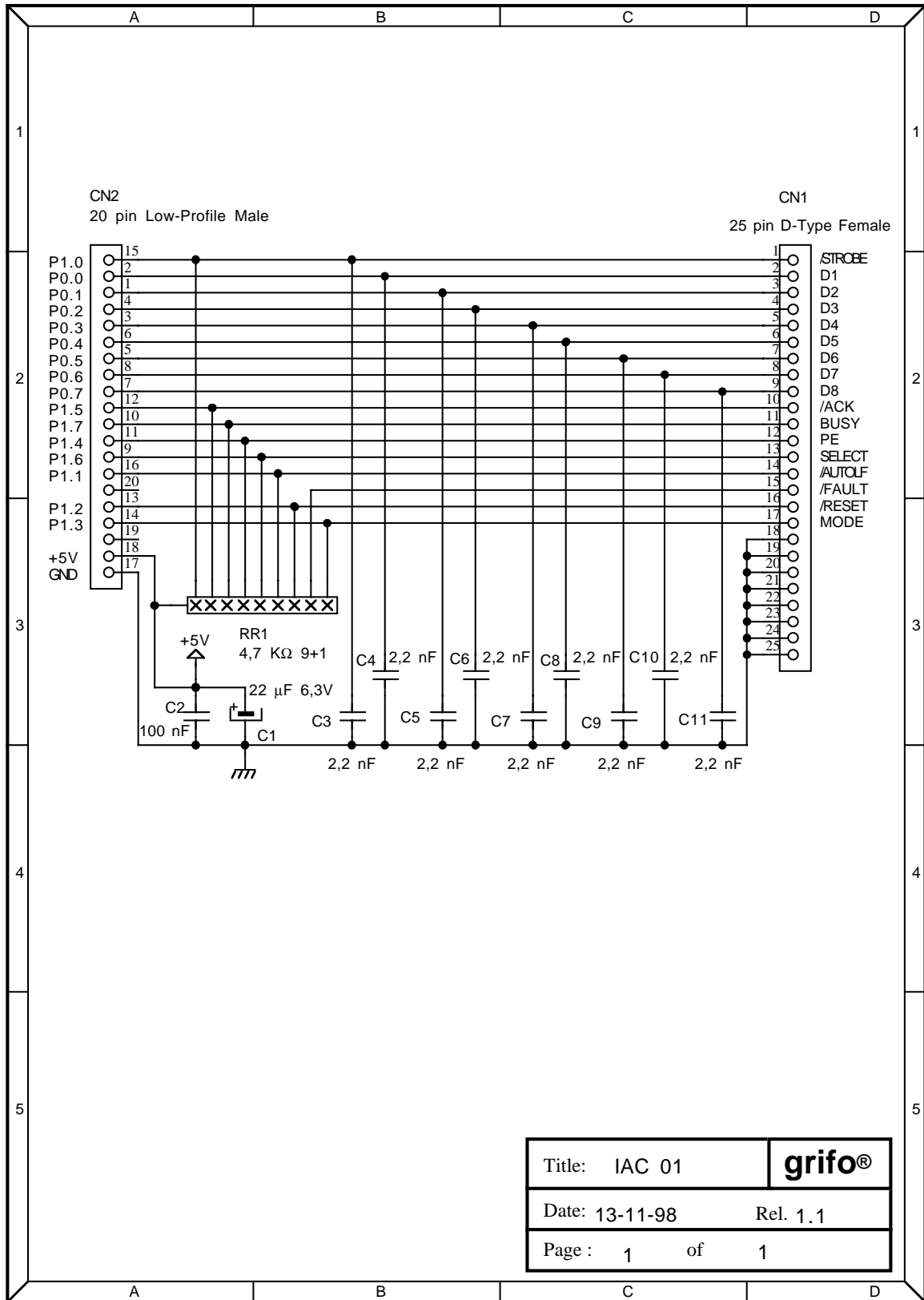


FIGURE A1: IAC 01 ELECTRIC DIAGRAM



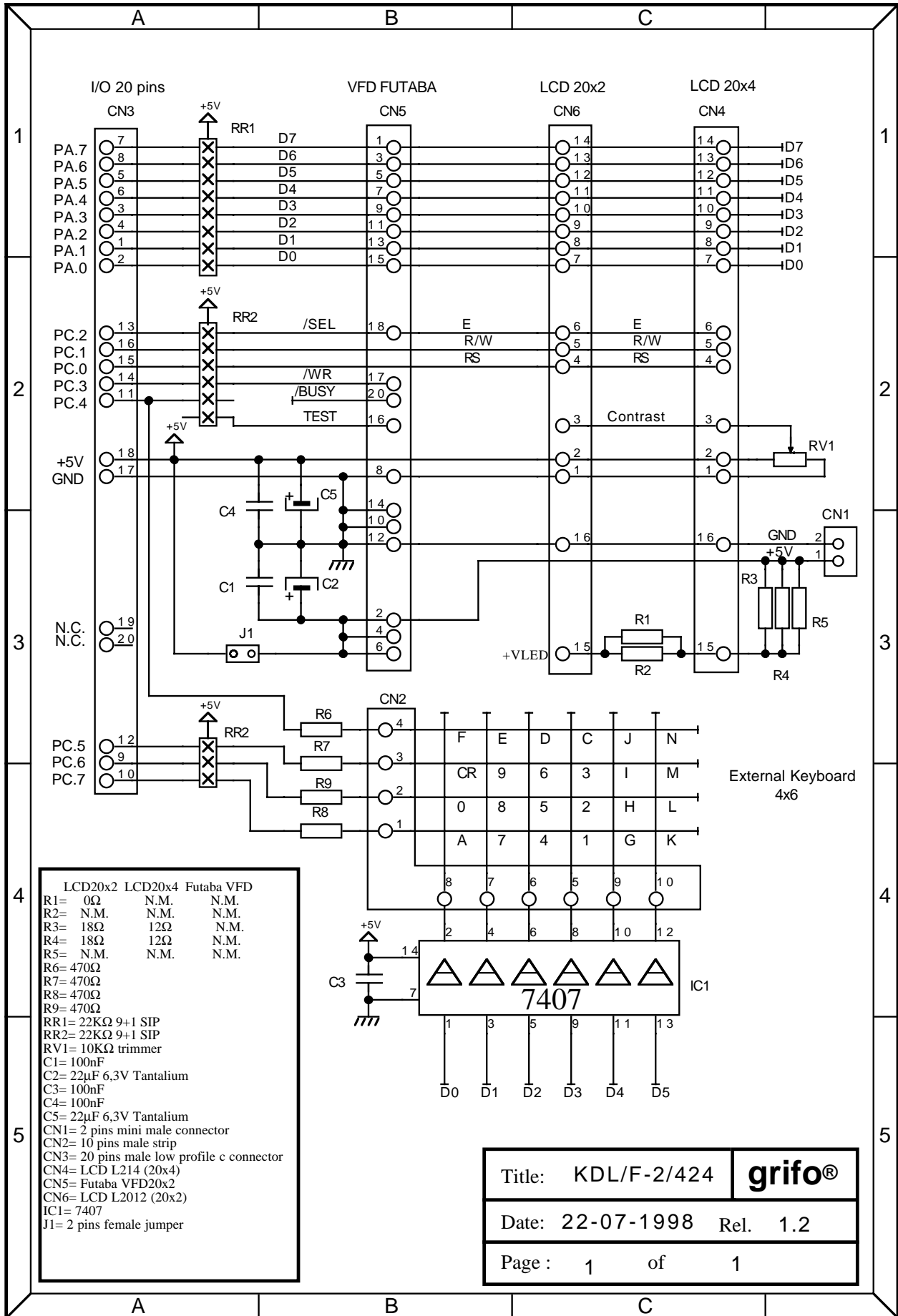


FIGURE A2: KDX x24 ELECTRIC DIAGRAM



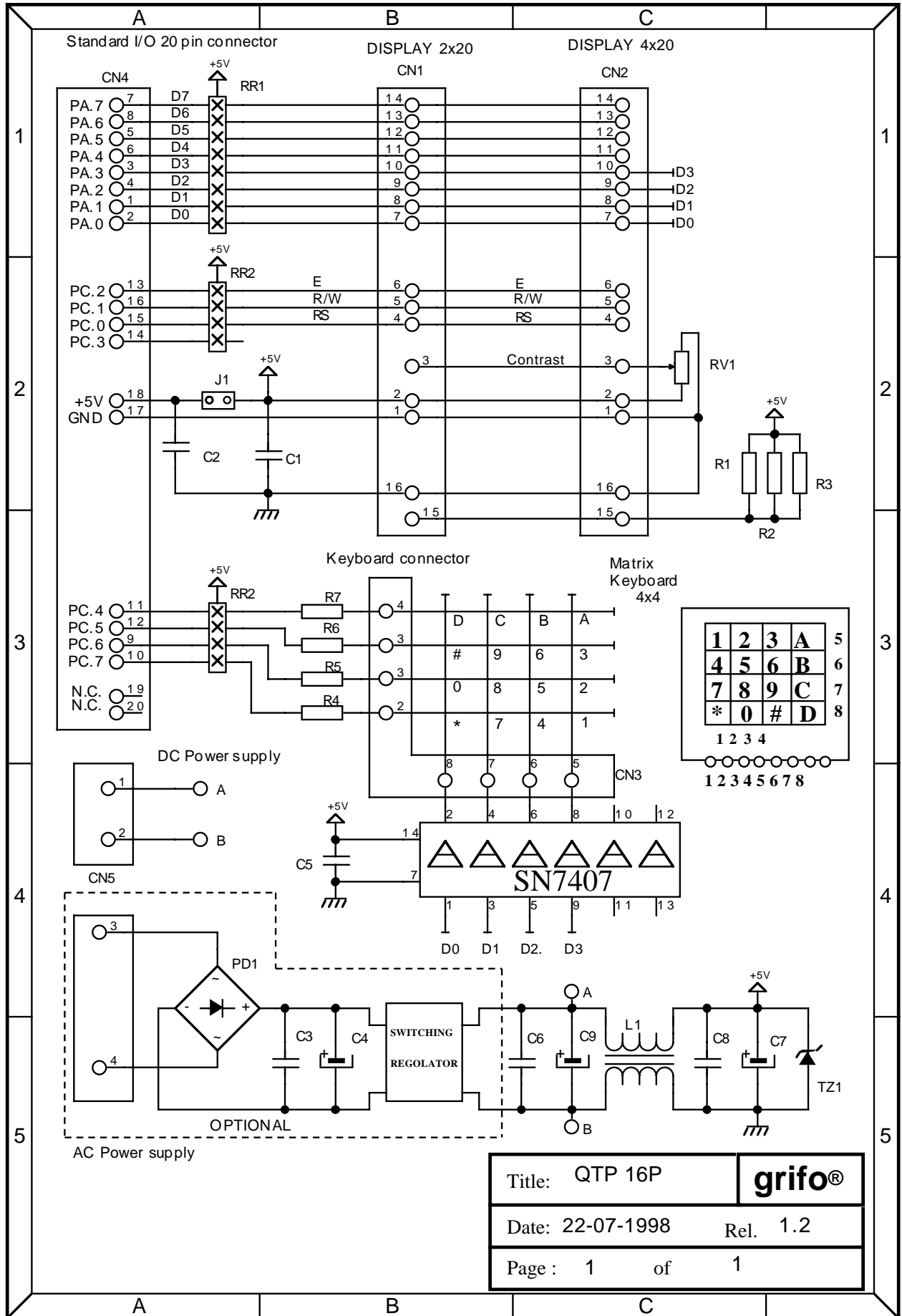


FIGURE A3: QTP 16P ELECTRIC DIAGRAM



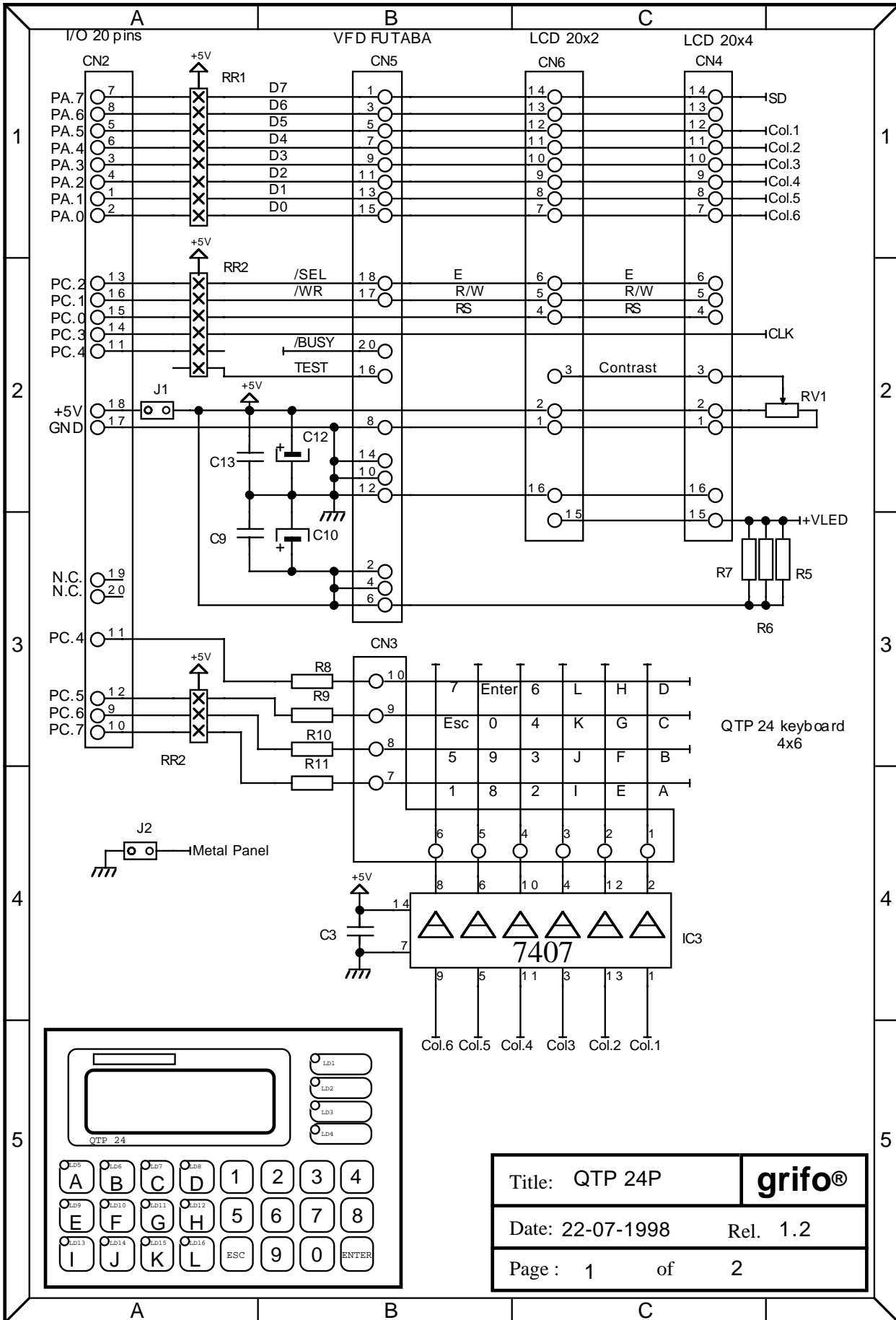
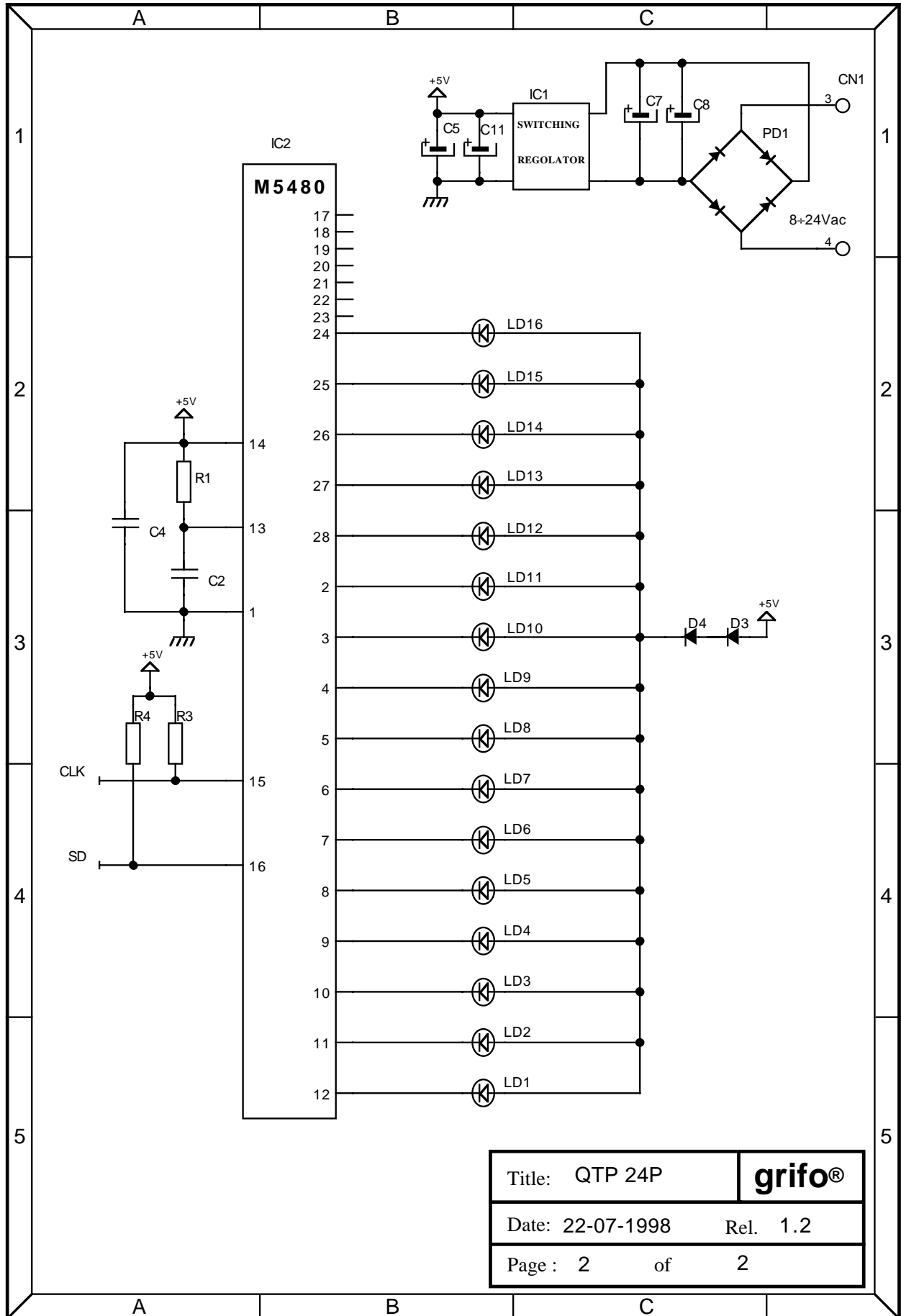


FIGURE A4: QTP 24P ELECTRIC DIAGRAM - PART 1



Title: QTP 24P	grifo®
Date: 22-07-1998	Rel. 1.2
Page : 2	of 2

FIGURE A5: QTP 24P ELECTRIC DIAGRAM - PART 2



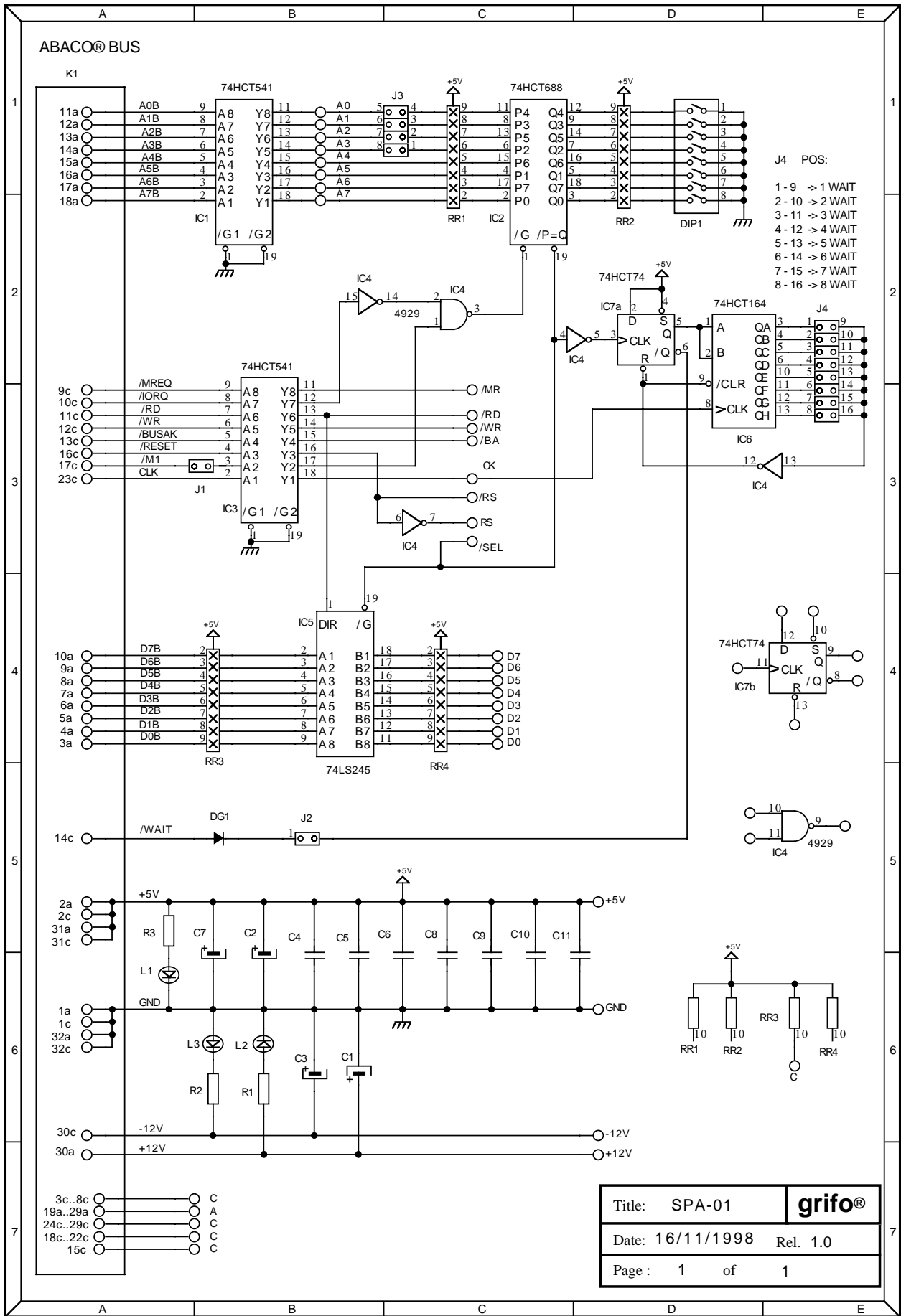


FIGURE A6: SPA 01 ELECTRIC DIAGRAM



APPENDIX B: ON BOARD COMPONENTS DESCRIPTION

CPU 80C188



M80C186
CHMOS HIGH INTEGRATION 16-BIT MICROPROCESSOR
Military

- **Operation Modes Include:**
 - Enhanced Mode Which Has
 - DRAM Refresh
 - Power-Save Logic
 - Direct Interface to New CMOS Numerics Coprocessor
 - Compatible Mode
 - NMOS M80186 Pin-for-Pin Replacement for Non-Numerics Applications
- **Integrated Feature Set**
 - Enhanced M80C86/C88 CPU
 - Clock Generator
 - 2 Independent DMA Channels
 - Programmable Interrupt Controller
 - 3 Programmable 16-Bit Timers
 - Dynamic RAM Refresh Control Unit
 - Programmable Memory and Peripheral Chip Select Logic
 - Programmable Wait State Generator
 - Local Bus Controller
 - Power Save Logic
 - System-Level Testing Support (High Impedance Test Mode)
- **Available in 10 MHz and 12.5 MHz Versions**
- **Direct Addressing Capability to 1 Mbyte and 64 Kbyte I/O**
- **Completely Object Code Compatible with All Existing M8086/M8088 Software and Also Has 10 Additional Instructions over M8086/M8088**
- **Complete System Development Support**
 - All M8086 and NMOS M80186 Software Development Tools Can Be Used for M80C186 System Development
 - Assembler, PL/M, Pascal, Fortran, and System Utilities
 - In-Circuit-Emulator (ICE™-C186)
- **Available in 68-Pin Ceramic Pin Grid Array (PGA) and 68-Lead Ceramic Quad Flat Pack**
 (See Packaging Outlines and Dimensions, Order # 231369)
- **Available in Two Product Grades:**
 - MIL-STD-883, -55°C to +125°C (T_C)
 - Military Temperature Only (MTO), -55°C to +125°C (T_C)

The Intel M80C186 is a CHMOS high integration microprocessor. It has features which are new to the M80186 family which include a DRAM refresh control unit, power-save mode and a direct numerics interface. When used in "compatible" mode, the M80C186 is 100% pin-for-pin compatible with the NMOS M80186 (except for M8087 applications). The "enhanced" mode of operation allows the full feature set of the M80C186 to be used. The M80C186 is upward compatible with M8086 and M8088 software and fully compatible with M80186 and M80188 software.

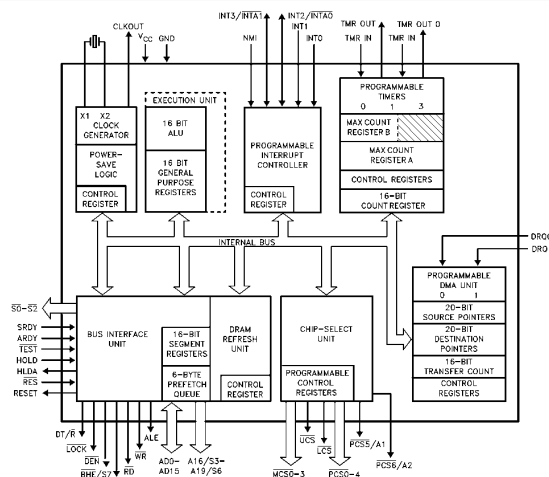


Figure 1. M80C186 Block Diagram



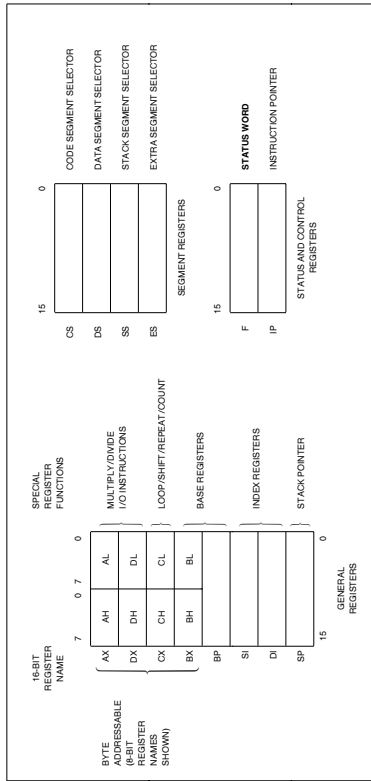


Figure 3a. M80C186 Register Set

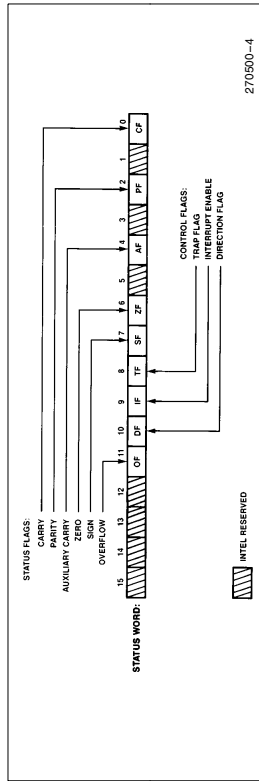


Figure 3b. Status Word Format

these (AX, BX, CX, and DX) can be used as 16-bit registers or split into pairs of separate 8-bit registers.

Segment Registers

Four 16-bit special purpose registers select, at any given time, the segments of memory that are immediately addressable for code, stack, and data. (For usage, refer to Memory Organization.)

Base and Index Registers

Four of the general purpose registers may also be used to determine offset addresses of operands in memory. These registers may contain base addresses or indexes to particular locations within a segment. The addressing mode selects the specific registers for operand and address calculations.

Status and Control Registers

Two 16-bit special purpose registers record or alter certain aspects of the M80C186 processor state. These are the Instruction Pointer Register, which contains the offset address of the next sequential instruction to be executed, and the Status Word Register, which contains status and control flag bits (see Figures 3a and 3b).

Status Word Description

The Status Word records specific characteristics of the result of logical and arithmetic instructions (bits 0, 2, 4, 6, 7, and 11) and controls the operation of the M80C186 within a given operating mode (bits 8, 9, and 10). The Status Word Register is 16-bits wide. The function of the Status Word bits is shown in Table 2.

FUNCTIONAL DESCRIPTION

Introduction

The following Functional Description describes the base architecture of the M80C186. This architecture is common to the M8086, M8088, M80186 and M80286 microprocessor families as well. The M80C186 is a very high integration 16-bit microprocessor. It combines 15-20 of the most common microprocessor system components onto one chip. The M80C186 is object code compatible with the M8086/M8088 microprocessors and adds 10 new instruction types to the existing M8086/M8088 instruction set.

The M80C186 has two major modes of operation, Compatible and Enhanced. In Compatible Mode the M80C186 is completely compatible with NMOS M80186, with the exception of M8087 support. All pin functions, timings, and drive capabilities are identical. The Enhanced mode adds three new features to the system design. These are Power-Save control, Dynamic RAM refresh, and an asynchronous Numerics Co-processor interface.

M80C186 BASE ARCHITECTURE

The M8086, M8088, M80186, and M80286 family all contain the same basic set of registers, instructions, and addressing modes. The M80C186 processor is upward compatible with the M8086, M8088, and M80286 CPUs.

Register Set

The M80C186 base architecture has fourteen registers as shown in Figures 3a and 3b. These registers are grouped into the following categories.

General Registers

Eight 16-bit general purpose registers may be used to contain arithmetic and logical operands. Four of



Table 2. Status Word Bit Functions

Bit Position	Name	Function
0	CF	Carry Flag—Set on high-order bit carry or borrow, cleared otherwise
2	PF	Parity Flag—Set if low-order 8 bits of result contain an even number of 1-bits; cleared otherwise
4	AF	Set on carry from or borrow to the low order four bits of AL; cleared otherwise
6	ZF	Zero Flag—Set if result is zero; cleared otherwise
7	SF	Sign Flag—Set equal to high-order bit of result (0 if positive, 1 if negative)
8	TF	Single Step Flag—Once set, a single step interrupt occurs after the next instruction executes. TF is cleared by the single step interrupt.
9	IF	Interrupt-Enable Flag—When set, maskable interrupts will cause the CPU to transfer control to an interrupt vector specified location.
10	DF	Direction Flag—Causes string instructions to auto decrement the appropriate index register when set. Clearing DF causes auto increment.
11	OF	Overflow Flag—Set if the signed result cannot be expressed within the number of bits in the destination operand; cleared otherwise

Instruction Set

The instruction set is divided into seven categories: data transfer, arithmetic, shift/rotate/logical, string manipulation, control transfer, high-level instructions, and processor control. These categories are summarized in Figure 4.

An M80C186 instruction can reference anywhere from zero to several operands. An operand can reside in a register, in the instruction itself, or in memory. Specific operand addressing modes are discussed later in this data sheet.

Memory Organization

Memory is organized in sets of segments. Each segment is a linear contiguous sequence of up to 64K (2¹⁶) 8-bit bytes. Memory is addressed using a two-component address (a pointer) that consists of a 16-bit base segment and a 16-bit offset. The 16-bit base values are contained in one of four internal segment registers (code, data, stack, extra). The physical address is calculated by shifting the base value LEFT by four bits and adding the 16-bit offset value to yield a 20-bit physical address (see Figure 5). This allows for a 1 MByte physical address size.

All instructions that address operands in memory must specify the base segment and the 16-bit offset value. For speed and compact instruction encoding, the segment register used for physical address generation is implied by the addressing mode used (see Table 3). These rules follow the way programs are written (see Figure 6) as independent modules that require areas for code and data, a stack, and access to external data areas.

Special segment override instruction prefixes allow the implicit segment register selection rules to be overridden for special cases. The stack, data, and extra segments may coincide for simple programs.

MOVMS	GENERAL PURPOSE	MOVMS	LOGICALS
MOV	Move byte or word	INS	"Not" byte or word
PUSH	Push word onto stack	OUTS	"And" byte or word
POP	Pop word off stack	CMPS	"Inclusive or" byte or word
PUSHA	Push all registers on stack	SCAS	"Exclusive or" byte or word
POPA	Pop all registers from stack	LDS	"Test" byte or word
XCHG	Exchange byte or word	STOS	Shift logical/arithmetic left byte or word
XLAT	Translate byte	STOSB	Shift logical right byte or word
IN	Input byte or word	REP	Shift arithmetic right byte or word
OUT	Output byte or word	REPNE/REPZ	Repeat while equal/not zero
LEA	Load effective address	NOT	Repeat while not equal/not zero
LDS	Load pointer using DS	AND	"Not" byte or word
LES	Load pointer using ES	OR	"And" byte or word
LAHF	Load AH register from flags	XOR	"Exclusive or" byte or word
SAHF	Store AH register in flags	TEST	"Test" byte or word
PUSHF	Push flags onto stack	SHL/SAL	Shift logical/arithmetic left byte or word
POPF	Pop flags off stack	SHR	Shift logical right byte or word
ADD	Add byte or word	SAR	Shift arithmetic right byte or word
ADC	Add byte or word with carry	ROL	Rotate left byte or word
INC	Increment byte or word by 1	ROR	Rotate right byte or word
AAA	ASCII adjust for addition	RCL	Rotate through carry left byte or word
DAA	Decimal adjust for addition	RCR	Rotate through carry right byte or word
SUB	Subtract byte or word	STC	Set carry flag
SBB	Subtract byte or word with borrow	CLC	Clear carry flag
DEC	Decrement byte or word by 1	CMC	Complement carry flag
NEG	Negate byte or word	STD	Set direction flag
CMP	Compare byte or word	CLD	Clear direction flag
AAS	ASCII adjust for subtraction	STI	Set interrupt enable flag
DAS	Decimal adjust for subtraction	CLI	Clear interrupt enable flag
MUL	Multiply byte or word unsigned	HLT	Halt until interrupt or reset
IMUL	Integer multiply byte or word	WAIT	Wait for TEST pin active
AAM	ASCII adjust for multiply	ESC	Escape to extension processor
DIV	Divide byte or word unsigned	LOCK	Lock bus during next instruction
IDIV	Integer divide byte or word	NOP	No operation
AAD	ASCII adjust for division	ENTER	Format stack for procedure entry
CBW	Convert byte to word	LEAVE	Restore stack for procedure exit
CWD	Convert word to doubleword	BOUND	Detects values outside prescribed range

Figure 4. M80C186 Instruction Set



CONDITIONAL TRANSFERS			
JAE/JNBE	Jump if above/not below nor equal	JO	Jump if overflow
JAE/JNB	Jump if above or equal/not below	JP/JPE	Jump if parity/parity even
JB/JNAE	Jump if below/not above nor equal	JS	Jump if sign
JBE/JNA	Jump if below or equal/not above	UNCONDITIONAL TRANSFERS	
JC	Jump if carry	CALL	Call procedure
JE/JZ	Jump if equal/zero	RET	Return from procedure
JG/JNLE	Jump if greater/not less nor equal	JMP	Jump
JGE/JNL	Jump if greater or equal/not less	ITERATION CONTROLS	
JL/JNGE	Jump if less/not greater nor equal	LOOP	Loop
JLE/JNG	Jump if less or equal/not greater	LOOPE/LOOPZ	Loop if equal/zero
JNC	Jump if not carry	LOOPNE/LOOPNZ	Loop if not equal/not zero
JNE/JNZ	Jump if not equal/not zero	JCXZ	Jump if register CX = 0
JNO	Jump if not overflow	INTERRUPTS	
JNP/JPO	Jump if not parity/parity odd	INT	Interrupt
JNS	Jump if not sign	INTO	Interrupt if overflow
		IRET	Interrupt return

Figure 4. M80C186 Instruction Set (Continued)

To access operands that do not reside in one of the four immediately available segments, a full 32-bit pointer can be used to reload both the base (segment) and offset values.

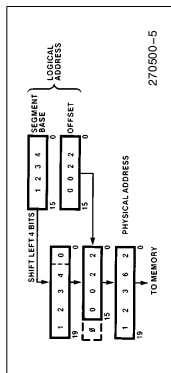
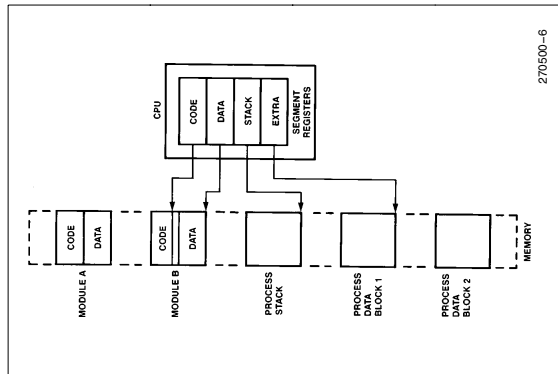


Figure 5. Two Component Address

Table 3. Segment Register Selection Rules

Memory Reference Needed	Segment Register Used	Implicit Segment Selection Rule
Instructions	Code (CS)	Instruction prefetch and immediate data.
Stack	Stack (SS)	All stack pushes and pops; any memory references which use BP Register as a base register.
External Data (Global)	Extra (ES)	All string instruction references which use the DI register as an index.
Local Data	Data (DS)	All other data references.

Figure 6. Segmented Memory Helps Structure Software



Addressing Modes

The M80C186 provides eight categories of addressing modes to specify operands. Two addressing modes are provided for instructions that operate on register or immediate operands:

- **Register Operand Mode:** The operand is located in one of the 8- or 16-bit general registers.
- **Immediate Operand Mode:** The operand is included in the instruction.

Six modes are provided to specify the location of an operand in a memory segment. A memory operand address consists of two 16-bit components: a segment base and an offset. The segment base is supplied by a 16-bit segment register either implicitly chosen by the addressing mode or explicitly chosen by a segment override prefix. The offset, also called the effective address, is calculated by summing any combination of the following three address elements:

- the **displacement** (an 8- or 16-bit immediate value contained in the instruction);
- the **base** (contents of either the BX or BP base registers); and
- the **index** (contents of either the SI or DI index registers).

Any carry out from the 16-bit addition is ignored. Eight-bit displacements are sign extended to 16-bit values.

Combinations of these three address elements define the six memory addressing modes, described below.

- **Direct Mode:** The operand's offset is contained in the instruction as an 8- or 16-bit displacement element.
- **Register Indirect Mode:** The operand's offset is in one of the registers SI, DI, BX, or BP.
- **Based Mode:** The operand's offset is the sum of an 8- or 16-bit displacement and the contents of a base register (BX or BP).
- **Indexed Mode:** The operand's offset is the sum of an 8- or 16-bit displacement and the contents of an index register (SI or DI).
- **Based Indexed Mode:** The operand's offset is the sum of the contents of a base register and an index register.
- **Based Indexed Mode with Displacement:** The operand's offset is the sum of a base register's contents, an index register's contents, and an 8- or 16-bit displacement.

Data Types

The M80C186 directly supports the following data types:

- **Integer:** A signed binary numeric value contained in an 8-bit byte or a 16-bit word. All operations assume a 2's complement representation. Signed 32- and 64-bit integers are supported using a Numeric Data Coprocessor with the M80C186.
- **Ordinal:** An unsigned binary numeric value contained in an 8-bit byte or a 16-bit word.
- **Pointer:** A 16- or 32-bit quantity, composed of a 16-bit offset component or a 16-bit segment base component in addition to a 16-bit offset component.
- **String:** A contiguous sequence of bytes or words. A string may contain from 1 to 64K bytes.
- **ASCII:** A byte representation of alphanumeric and control characters using the ASCII standard of character representation.
- **BCD:** A byte (unpacked) representation of the decimal digits 0-9.
- **Packed BCD:** A byte (packed) representation of two decimal digits (0-9). One digit is stored in each nibble (4-bits) of the byte.
- **Floating Point:** A signed 32-, 64-, or 80-bit real number representation. (Floating point operands are supported using a Numeric Data Coprocessor with the M80C186.)

In general, individual data elements must fit within defined segment limits. Figure 7 graphically represents the data types supported by the M80C186.

I/O Space

The I/O space consists of 64K 8-bit or 32K 16-bit ports. Separate instructions address the I/O space with either an 8-bit port address, specified in the instruction, or a 16-bit port address in the DX register. 8-bit port addresses are zero extended such that A15-A8 are LOW. I/O port addresses 00FF(H) through 00FF(H) are reserved.

Interrupts

An interrupt transfers execution to a new program location. The old program address (CS:IP) and machine state (Status Word) are saved on the stack to allow resumption of the interrupted program. Interrupts fall into three classes: hardware initiated, INT instructions, and instruction exceptions. Hardware initiated interrupts occur in response to an external input and are classified as non-maskable or maskable.



M80C186

Table 4. M80C186 Interrupt Vectors

Interrupt Name	Vector Type	Default Priority(4)	Related Instructions
Divide Error Exception	0	1(1)	DIV, IDIV
Single Step Interrupt	1	12(2)	All
NMI	2	1	All
Breakpoint Interrupt	3	1(1)	INT
INT0 Detected Overflow Exception	4	1(1)	INT0
Array Bounds Exception	5	1(1)	BOUND
Unused-Opcode Exception	6	1(1)	Undefined OpCodes
ESC Opcode Exception	7	1(1), (5)	ESC OpCodes
Timer 0 Interrupt	8	2A(3)	
Timer 1 Interrupt	18	2B(3)	
Timer 2 Interrupt	19	2C(3)	
Reserved	9	3	
DMA 0 Interrupt	10	4	
DMA 1 Interrupt	11	5	
INT1 Interrupt	12	6	
INT2 Interrupt	13	7	
INT3 Interrupt	14	8	
INT4 Interrupt	15	9	

ARRAY BOUNDS EXCEPTION (TYPE 5)

Generated during a BOUND instruction if the array index is outside the array bounds. The array bounds are located in memory at a location indicated by one of the instruction operands. The other operand indicates the value of the index to be checked.

UNUSED OP CODE EXCEPTION (TYPE 6)

Generated if execution is attempted on undefined opcodes.

ESCAPE OP CODE EXCEPTION (TYPE 7)

Generated if execution is attempted of ESC opcodes (DBH-DFFH). In compatible mode operation, ESC opcodes will always generate this exception. In enhanced mode operation, the exception will be generated only if a bit in the relocation register is set. The return address of this exception will point to the ESC instruction causing the exception. If a segment override prefix preceded the ESC instruction, the return address will point to the segment override prefix.

Hardware-generated interrupts are divided into two groups: maskable interrupts and non-maskable interrupts. The M80C186 provides maskable hardware interrupt request pins INT0-INT3. In addition, maskable interrupts may be generated by the M80C186 integrated DMA controller and the integrated timer unit. The vector types for these interrupts is shown in Table 4. Software enables these inputs by setting the interrupt flag bit (IF) in the Status Word. The interrupt controller is discussed in the peripheral section of this data sheet.

Further maskable interrupts are disabled while servicing an interrupt because the IF bit is reset as part of the response to an interrupt or exception. The saved Status Word will reflect the enable status of the processor prior to the interrupt. The interrupt flag will remain zero unless specifically set. The interrupt return instruction restores the Status Word, thereby restoring the original status of IF bit. If the interrupt return re-enables interrupts, and another interrupt is pending, the M80C186 will immediately service the highest-priority interrupt pending. I.e., no instructions of the main line program will be executed.

Non-Maskable Interrupt Request (NMI)

A non-maskable interrupt (NMI) is also provided. This interrupt is serviced regardless of the state of the IF bit. A typical use of NMI would be to activate a power failure routine. The activation of this input causes an interrupt with an internally supplied vector value of 2. No external interrupt acknowledge sequence is performed. The IF bit is cleared at the beginning of an NMI interrupt to prevent maskable interrupts from being serviced.

M80C186

if the prefix was present. In all other cases, the return address from an exception will point at the instruction immediately following the instruction causing the exception.

A table containing up to 256 pointers defines the proper interrupt service routine for each interrupt. Interrupts 0-31, some of which are used for instruction exceptions, are reserved. Table 4 shows the M80C186 predefined types and default priority levels. For each interrupt, an 8-bit vector must be supplied to the M80C186 which identifies the appropriate table entry. Exceptions supply the interrupt vector internally. In addition, internal peripherals and non-cascaded external interrupts will generate their own vectors through the internal interrupt controller. INT instructions contain or imply the vector and allow access to all 256 interrupts. Maskable hardware initiated interrupts supply the 8-bit vector to the CPU during an interrupt acknowledge bus sequence. Non-maskable hardware interrupts use a predefined internally supplied vector.

Interrupt Sources

The M80C186 can service interrupts generated by software or hardware. The software interrupts are generated by specific instructions (INT, ESC, unmasked OP, etc.) or the results of conditions specified by instructions (array bounds check, INTO, DIV, IDIV, etc.). All interrupt sources are serviced by an indirect call through an element of a vector table. This vector table is indexed by using the interrupt vector type (Table 4), multiplied by four. All hardware-generated interrupts are sampled at the end of each instruction. Thus, the software interrupts will begin service first. Once the service routine is entered and interrupts are enabled, any hardware source of sufficient priority can interrupt the service routine in progress. The software generated M80C186 interrupts are described below.

DIVIDE ERROR EXCEPTION (TYPE 0)

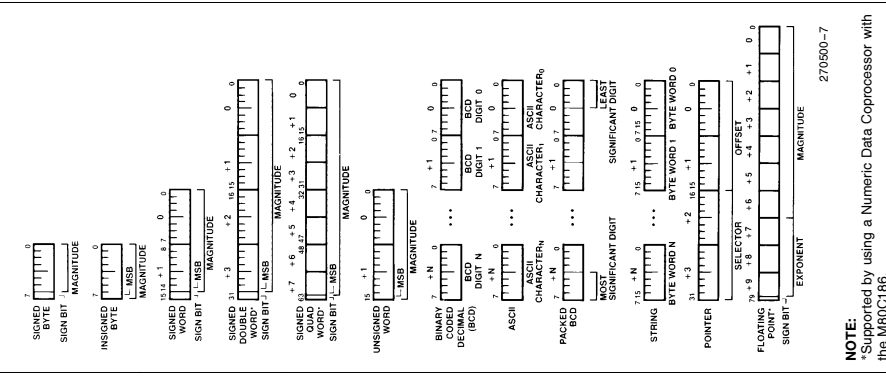
Generated when a DIV or IDIV instruction quotient cannot be expressed in the number of bits in the destination.

SINGLE-STEP INTERRUPT (TYPE 1)

Generated after most instructions if the TF flag is set. Interrupts will not be generated after prefix instructions (e.g. REP, instructions which modify segment registers (e.g., POP DS), or the WAIT instruction).

NON-MASKABLE INTERRUPT—NMI (TYPE 2)

An external interrupt source which cannot be masked.



NOTE: *Supported by using a Numeric Data Coprocessor with the M80C186.

Figure 7. M80C186 Supported Data Types

Programs may cause an interrupt with an INT instruction. Instruction exceptions occur when an unusual condition, which prevents further instruction processing, is detected while attempting to execute an instruction. If the exception was caused by executing an ESC instruction with the ESC trap bit set in the relocation register, the return instruction will point to the ESC instruction, or to the segment override prefix immediately preceding the ESC instruction.



Single-Step Interrupt

The M80C186 has an internal interrupt that allows programs to execute one instruction at a time. It is called the single-step interrupt and is controlled by the single-step flag bit (TF) in the Status Word. Once this bit is set, an internal single-step interrupt will occur after the next instruction has been executed. The interrupt clears the TF bit and uses an internally supplied vector of 1. The IRET instruction is used to set the TF bit and transfer control to the next instruction to be single-stepped.

Initialization and Processor Reset

Processor initialization or startup is accomplished by driving the RES input pin LOW. RES forces the M80C186 to terminate all execution and local bus activity. No instruction or bus activity will occur as long as RES is active. After RES becomes inactive and an internal processing interval elapses, the M80C186 begins execution with the instruction at physical location FFFF(H). RES also sets some registers to predefined values as shown in Table 5.

Table 5. M80C186 Initial Register State after RESET

Status Word	F002(H)
Instruction Pointer	0000(H)
Code Segment	FFFF(H)
Data Segment	0000(H)
Extra Segment	0000(H)
Stack Segment	0000(H)
Relocation Register	20FF(H)
UMCS	FFFB(H)

M80C186 CLOCK GENERATOR

The M80C186 provides an on-chip clock generator for both internal and external clock generation. The clock generator features a crystal oscillator, a divide-by-two counter, synchronous and asynchronous ready inputs, and reset circuitry.

Oscillator

The M80C186 oscillator circuit is designed to be used with either a parallel resonant fundamental or third-overtone mode crystal, depending upon the frequency range of the application as shown in Figure 8a. This is used as the time base for the M80C186. The crystal frequency chosen should be twice the required processor frequency. Use of an LC or RC circuit is not recommended.

The oscillator output is not directly available external to the M80C186. The two recommended crys-

tal configurations are shown in Figures 8b and 8c. When used in third-overtone mode, the tank circuit shown in Figure 8b is recommended for stable operation. The sum of the stray capacitances and load capacitors should equal the values shown. It is advisable to limit stray capacitance between the X1 and X2 pins to less than 10 pF. While a fundamental-mode circuit will require approximately 1 ms for start-up, the third-overtone arrangement may require 1 ms to 3 ms to stabilize.

Alternately, the oscillator pins may be driven from an external source as shown in Figure 8d or Figure 8e. The configuration shown in Figure 8f is not recommended.

The following parameters may be used for choosing a crystal:

- Temperature Range: -55°C to +125°C
- ESR (Equivalent Series Resistance): 40.0 max
- C₀ (Shunt Capacitance of Crystal): 7.0 pF max
- C₁ (Load Capacitance): 20 pF ± 2 pF
- Drive Level: 1 mW max

Clock Generator

The M80C186 clock generator provides the 50% duty cycle processor clock for the M80C186. It does this by dividing the oscillator output by 2 forming the symmetrical clock. If an external oscillator is used, the state of the clock generator will change on the falling edge of the oscillator signal. The CLKOUT pin provides the processor clock signal for use outside the M80C186. This may be used to drive other system components. All timings are referenced to the output clock.

READY Synchronization

The M80C186 provides both synchronous and asynchronous ready inputs. Asynchronous ready synchronization is accomplished by circuitry which samples ARDY in the middle of T₂, T₃ and again in the middle of each T_w until ARDY is sampled HIGH. One-half CLKOUT cycle of resolution time is used. Full synchronization is performed only on the rising edge of ARDY, i.e., the falling edge of ARDY must be synchronized to the CLKOUT signal if it will occur during T₂, T₃, or T_w. High-to-LOW transitions of ARDY must be performed synchronously to the CPU clock.

A second ready input (SRDY) is provided to interface with externally synchronized ready signals. This input is sampled at the end of T₂, T₃ and again at the end of each T_w until it is sampled HIGH. By using this input rather than the asynchronous ready input, the half-clock cycle resolution time penalty is eliminated.

This input must satisfy set-up and hold times to guarantee proper operation of the circuit.

In addition, the M80C186, as part of the integrated chip-select logic, has the capability to program WAIT states for memory and peripheral blocks. This is discussed in the Chip Select/Ready Logic description.

RESET Logic

The M80C186 provides both a RES input pin and a synchronized RESET pin for use with other system components. The RES input pin on the M80C186 is provided with hysteresis in order to facilitate power-on Reset generation via an RC network. RESET is guaranteed to remain active for at least five clocks given a RES input of at least six clocks. RESET may be delayed up to two and one-half clocks behind RES.

Multiple M80C186 processors may be synchronized through the RES input pin, since this input resets both the processor and divide-by-two internal counter in the clock generator. In order to insure that the divide-by-two counters all begin counting at the same time, the active going edge of RES must satisfy a 25 ns setup time before the falling edge of the M80C186 clock input. In addition, in order to insure that all CPUs begin executing in the same clock cycle, the reset must satisfy a 15 ns setup time before the rising edge of the CLKOUT signal of all the processors.

LOCAL BUS CONTROLLER

The M80C186 provides a local bus controller to generate the local bus control signals. In addition, it employs a HOLD/HLDA protocol for relinquishing the local bus to other bus masters. It also provides control lines that can be used to enable external buffers and to direct the flow of data on and off the local bus.

Memory/Peripheral Control

The M80C186 provides ALE, RD, and WR bus control signals. The RD and WR signals are used to strobe data from memory to the M80C186 or to strobe data from the M80C186 to memory. The ALE line provides a strobe to address latches for the multiplexed address/data bus. The M80C186 local bus controller does not provide a memory/I/O signal. If this is required, the user will have to use the S2 signal (which will require external latching), make the memory and I/O spaces nonoverlapping, or use only the integrated chip-select circuitry.

Transceiver Control

The M80C186 generates two control signals to be connected to external transceiver chips. This capability allows the addition of transceivers for extra buffering without adding external logic. These control lines, DT/R and DEN, are generated to control the flow of data through the transceivers. The operation of these signals is shown in Table 6.

Table 6. Transceiver Control Signals Description

Pin Name	Function
DEN (Data Enable)	Enables the output drivers of the transceivers. It is active LOW during memory, I/O, or INTA cycles.
DT/R (Data Transmit/Receive)	Determines the direction of travel through the transceivers. A HIGH level directs data away from the processor during write operations, while a LOW level directs data toward the processor during a read operation.

Local Bus Arbitration

The M80C186 uses a HOLD/HLDA system of local bus exchange. This provides an asynchronous bus utilizing the same bus can operate at separate clock frequencies. The M80C186 provides a single HOLD/HLDA pair through which all other bus masters may gain control of the local bus. This requires external circuitry to arbitrate which external device will gain control of the bus from the M80C186 when there is more than one alternate local bus master. When the M80C186 relinquishes control of the local bus, it floats DEN, RD, WR, SO-S2, LOCK, AD0-AD15, A16-A19, BHE, and DT/R to allow another master to drive these lines directly.

The M80C186 HOLD latency time, i.e., the time between HOLD request and HOLD acknowledge, is a function of the activity occurring in the processor when the HOLD request is received. A HOLD request is the highest-priority activity request which the processor may receive: higher than instruction fetching or internal DMA cycles. However, if a DMA cycle is in progress, the M80C186 will complete the transfer before relinquishing the bus. This implies that if a HOLD request is received just as a DMA transfer begins, the HOLD latency time can be as great as 4 bus cycles. This will occur if a DMA word transfer operation is taking place from an odd ad-





M80C186

dress to an odd address. This is a total of 16 clocks or more, if WAIT states are required. In addition, if locked transfers are performed, the HOLD latency time will be increased by the length of the locked transfer.

Local Bus Controller and Reset

Upon receipt of a RESET pulse from the \overline{RES} input, the local bus controller will perform the following action:

- Drive \overline{DEN} , \overline{RD} , and \overline{WR} HIGH for one clock cycle, then float.

NOTE:

- \overline{RD} is also provided with an internal pull-up device to prevent the processor from inadvertently entering Queue Status mode during reset.
- Drive $\overline{S0}$ – $\overline{S2}$ to the passive state (all HIGH) and then float.
 - Drive \overline{LOCK} HIGH and then float.
 - Float $A00$ – $A15$, $A16$ – $A19$, BHE , DT/\overline{R} .
 - Drive ALE LOW (ALE is never floated).
 - Drive HLDA LOW.

INTERNAL PERIPHERAL INTERFACE

All the M80C186 integrated peripherals are controlled via 16-bit registers contained within an internal 256-byte control block. This control block may be mapped into either memory or I/O space. Internal logic will recognize the address and respond to the bus cycle. During bus cycles to internal registers, the bus controller will signal the operation externally (i.e., the \overline{RD} , \overline{WR} , status, address, data, etc., lines will be driven as in a normal bus cycle), but $D15$ – 0 , \overline{SRDY} , and \overline{ARDY} will be ignored. The base address of the control block must be on an even 256-byte boundary (i.e., the lower 8 bits of the base address are all zeros). All of the defined registers within this control block may be read or written by the M80C186 CPU at any time. The location of any register contained within the 256-byte control block is determined by the current base address of the control block.

The control block base address is programmed via a 16-bit relocation register contained within the control block at offset FEH from the base address of the control block (see Figure 9). It provides the upper 12 bits of the base address of the control block. The control block is effectively an internal chip select range and must abide by all the rules concerning chip selects (the chip select circuitry is discussed later in this data sheet). Any access to the 256 bytes of the control block activates an internal chip select.



Other chip selects may overlap the control block only if they are programmed to zero wait states and ignore external ready. In addition, bit 12 of this register determines whether the control block will be mapped into I/O or memory space. If this bit is 1, the control block will be located in memory space, whereas if the bit is 0, the control register block will be located in I/O space. If the control register block is mapped into I/O space, the upper 4 bits of the base address must be programmed as 0 (since I/O addresses are only 16 bits wide).

In addition to providing relocation information for the control block, the relocation register contains bits which place the interrupt controller into slave mode, and cause the CPU to interrupt upon encountering ESC instructions. At RESET, the relocation register is set to 20FFH. This causes the control block to start at FF00H in I/O space. An offset map of the 256-byte control register block is shown in Figure 10.

The integrated M80C186 peripherals operate autonomously from the CPU. Access to them for the most part is via software read/write of the control block. Most of these registers can be both read and written. A few dedicated lines, such as interrupts and DMA request, provide real-time communication between the CPU and peripherals as in a more conventional system utilizing discrete peripheral blocks. The overall interaction and function of the peripheral blocks has not substantially changed.

CHIP-SELECT/READY GENERATION LOGIC

The M80C186 contains logic which provides programmable chip-select generation for both memories and peripherals. In addition, it can be programmed to provide READY (or WAIT state) generation. It can also provide latched address bits A1 and A2. The chip-select lines are active for all memory and I/O cycles in their programmed areas, whether they be generated by the CPU or by the integrated DMA unit.

Memory Chip Selects

The M80C186 provides 6 memory chip select outputs for 3 address areas: upper memory, lower memory, and midrange memory. One each is provided for upper memory and lower memory, while four are provided for midrange memory.

The range for each chip select is user-programmable and can be set to 2K, 4K, 8K, 16K, 32K, 64K, 128K (plus 1K and 256K for upper and lower chip selects). In addition, the beginning or base address

M80C186

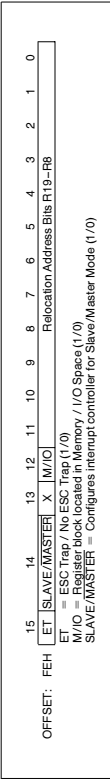


Figure 9. Relocation Register

The upper limit of memory defined by this chip select is always FFFFH, while the lower limit is programmable. By programming the lower limit, the size of the select block is also defined. Table 7 shows the relationship between the base address selected and the size of the memory block obtained.

Table 7. UMCS Programming Values

Starting Address (Base Address)	Memory Block Size	UMCS Value (Assuming R0=R1=R2=0)
FFC00	1K	FFF8H
FF800	2K	FFB8H
FF000	4K	FF38H
FE000	8K	FE38H
FC000	16K	FC38H
F8000	32K	F838H
F0000	64K	F038H
E0000	128K	E038H
C0000	256K	C038H

The lower limit of this memory block is defined in the UMCS register (see Figure 11). This register is at offset A0H in the internal control block. The legal values for bits 6–13 and the resulting starting address and memory block sizes are given in Table 7. Any combination of bits 6–13 not shown in Table 7 will result in undefined operation. After reset, the UMCS register is programmed for a 1K area. It must be reprogrammed if a larger upper memory area is desired.

of the midrange memory chip select may also be selected. Only one chip select may be programmed to be active for any memory location at a time. M80C186 memory is arranged in words but chip selects are sized in bytes. If sixteen 64K x 1 memories are used then the memory block size will be 128K, not 64K.

Upper Memory CS

The M80C186 provides a chip select, called UCS, for the top of memory. The top of memory is usually used as the system memory because after reset the M80C186 begins executing at memory location FFFF0H.

Lower Memory CS

The M80C186 provides a chip select for low memory called LCS. The bottom of memory contains the interrupt vector table, starting at location 00000H.

Figure 10. Internal Register Map

Offset	Register Name
FEH	Relocation Register
DAH	DMA Descriptors Channel 1
D0H	DMA Descriptors Channel 0
CAH	Chip-Select Control Registers
C0H	Time 2 Control Registers
A6H	Time 1 Control Registers
A0H	Time 0 Control Registers
66H	Interrupt Controller Registers
60H	
5EH	
58H	
56H	
50H	
3EH	
20H	



The lower limit of memory defined by this chip select is always 0H, while the upper limit is programmable. By programming the upper limit, the size of the memory block is also defined. Table 8 shows the relationship between the upper address selected and the size of the memory block obtained.

Table 8. LMCS Programming Values

Upper Address	Memory Block Size	LMCS Value (Assuming R0 = R1 = R2 = 0)
003FFH	1K	0039H
007FFH	2K	0078H
00FFFH	4K	00F8H
01FFFH	8K	01F8H
03FFFH	16K	03F8H
07FFFH	32K	07F8H
0FFFFH	64K	0FF8H
1FFFFH	128K	1FF8H
3FFFFH	256K	3FF8H

The upper limit of this memory block is defined in the LMCS register (see Figure 12). This register is at offset A2H in the internal control block. The legal values for bits 6-15 and the resulting upper address and memory block sizes are given in Table 8. Any combination of bits 6-15 not shown in Table 8 will result in undefined operation. After reset, the LMCS register value is undefined. However, the LCS chip-select line will not become active until the LMCS register is accessed.

Any internally generated 20-bit address whose upper 16 bits are less than or equal to LMCS (with bits 0-5 "1") will cause LCS to be active. LMCS register bits R2-R0 are used to specify the READY mode for the area of memory defined by this chip-select register.

Mid-Range Memory CS

The M80C186 provides four MCS lines which are active within a user-locatable memory block. This block can be located within the M80C186 1M byte memory address space exclusive of the areas defined by UCS and LCS. Both the base ad-

dress and size of this memory block are programmable.

The size of the memory block defined by the mid-range select lines, as shown in Table 9, is determined by bits 8-14 of the MPCS register (see Figure 13). This register is at location A8H in the internal control block. One and only one of bits 8-14 must be set at a time. Unpredictable operation of the MCS lines will otherwise occur. Each of the four chip-select lines is active for one of the four equal contiguous divisions of the mid-range block. Thus, if the total block size is 32K, each chip select is active for 8K of memory with MCS0 being active for the first range and MCS3 being active for the last range.

The EX and MS in MPCS relate to peripheral functionality as described in a later section.

Table 9. MPCS Programming Values

Total Block Size	Individual Select Size	MPCS Bits 14-8
8K	2K	0000001B
16K	4K	0000010B
32K	8K	0000100B
64K	16K	0001000B
128K	32K	0010000B
256K	64K	0100000B
512K	128K	1000000B

The base address of the mid-range memory block is defined by bits 15-9 of the MMCS register (see Figure 14). This register is at offset A6H in the internal control block. These bits correspond to bits A12-A10 of the 20-bit memory address. Bits A12-A10 of the base address are always 0. The base address may be set at any integer multiple of the size of the total memory block selected. For example, if the mid-range block size is 32K (or the size of the block for which each MCS line is active is 8K), the block could be located at 10000H or 18000H, but not at 14000H, since the first few integer multiples of a 32K memory block are 0H, 8000H, 10000H, 18000H, etc. After reset, the contents of both of these registers is undefined. However, none of the MCS lines will be active until both the MMCS and MPCS registers are accessed.

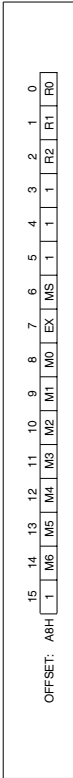


Figure 13. MPCS Register

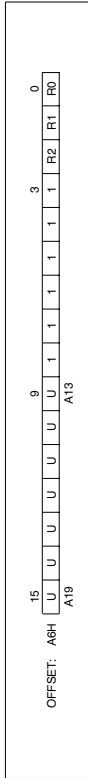


Figure 14. MMCS Register

MMCS bits R2-R0 specify READY mode of operation for all mid-range chip selects. All devices in mid-range memory must use the same number of WAIT states.

however it can only be a multiple of 1K bytes, i.e., the least significant 10 bits of the starting address are always 0.

The 512K block size for the mid-range memory chip selects is a special case. When using 512K, the base address would have to be at either locations 00000H or 80000H. If it were to be programmed at 00000H when the LCS line was programmed, there would be an internal conflict between the LCS ready generation logic and the MCS ready generation logic. Likewise, if the base address were programmed at 80000H, there would be a conflict with the UCS ready generation logic. Since the LCS chip-select line does not become active until programmed, while the UCS line is active at reset, the memory base can be set only at 00000H. If this base address is selected, however, the LCS range must not be programmed.

PCS6 and PCS6 can also be programmed to provide latched address bits A1, A2. If so programmed, they cannot be used as peripheral selects. These outputs can be connected directly to the A0, A1 pins used for selecting internal registers of 8-bit peripheral chips. This scheme simplifies the hardware interface because the 8-bit registers of peripherals are simply treated as 16-bit registers located on even boundaries in I/O space or memory spaces where only the lower 8-bits of the register are significant; the upper 8-bits are "don't cares."

The starting address of the peripheral chip-select block is defined by the PACS register (see Figure 15). This register is located at offset A4H in the internal control block. Bits 15-6 of this register correspond to bits 19-10 of the 20-bit Programmable Base Address (PBA) of the peripheral chip-select block. Bits 9-0 of the PBA of the peripheral chip-select block are all zeros. If the chip-select block is located in I/O space, bits 12-15 must be programmed zero, since the I/O address is only 16 bits wide. Table 10 shows the address range of each peripheral chip select with respect to the PBA contained in PACS register.

Peripheral Chip Selects

The M80C186 can generate chip selects for up to seven peripheral devices. These chip selects are active for seven contiguous blocks of 128 bytes above a programmable base address. This base address may be located in either memory or I/O space.

Seven CS lines called PCS0-6 are generated by the M80C186. The base address is user-programmable;



Figure 15. PACS Register

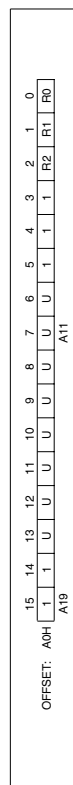


Figure 11. UMCS Register



Figure 12. LMCS Register



DMA CHANNELS

The M80C186 DMA controller provides two independent high-speed DMA channels. Data transfers can occur between memory and I/O spaces (e.g., Memory to I/O) or within the same space (e.g., Memory to Memory or I/O to I/O). Data can be transferred either in bytes (8 bits) or in words (16 bits) to or from even or odd addresses. Each DMA channel maintains both a 20-bit source and destination pointer which can be optionally incremented or decremented after each data transfer (by one or two depending on byte or word transfers). Each data transfer consumes 2 bus cycles (a minimum of 8 clocks), one cycle to fetch data and the other to store data.

DMA Operation

Each channel has six registers in the control block which define each channel's specific operation. The control registers consist of a 20-bit Source pointer (2

words), a 20-bit destination pointer (2 words), a 16-bit Transfer Counter, and a 16-bit Control Word. The format of the DMA Control Blocks is shown in Table 13. The Transfer Counter Register (TC) specifies the number of DMA transfers to be performed. Up to 64K byte or word transfers can be performed with automatic termination. The Control Word defines the channel's operation (see Figure 17). All registers may be modified or altered during any DMA activity. Any changes made to these registers will be reflected immediately in DMA operation.

Table 13. DMA Control Block Format

Register Name	Register Address	
	Ch. 0	Ch. 1
Control Word	CAH	DAH
Transfer Counter	C8H	D8H
Destination Pointer (upper 4 bits)	C6H	D6H
Destination Pointer	C4H	D4H
Source Pointer (upper 4 bits)	C2H	D2H
Source Pointer	C0H	D0H

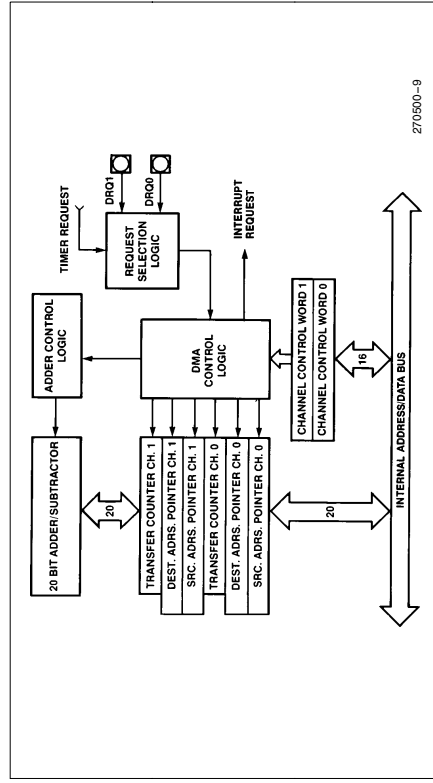


Figure 16. DMA Unit Block Diagram

Table 12. READY Bits Programming

R2	R1	R0	Number of WAIT States Generated
0	0	0	0 wait states, external RDY also used.
0	0	1	1 wait state inserted, external RDY also used.
0	1	0	2 wait states inserted, external RDY also used.
0	1	1	3 wait states inserted, external RDY also used.
1	0	0	0 wait states, external RDY ignored.
1	0	1	1 wait state inserted, external RDY ignored.
1	1	0	2 wait states inserted, external RDY ignored.
1	1	1	3 wait states inserted, external RDY ignored.

The internal ready generator operates in parallel with external READY, not in series if the external READY is used (R2 = 0). This means, for example, if the internal generator is set to insert two wait states, but activity on the external READY lines will insert four wait states, the processor will only insert four wait states, not six. This is because the two wait states generated by the internal generator overlapped the first two wait states generated by the external ready signal. Note that the external ARDY and SRDY lines are always ignored during cycles accessing internal peripherals.

R2-R0 of each control word specifies the READY mode for the corresponding block, with the exception of the peripheral chip selects: R2-R0 of PACS set the PCS0-3 READY mode, R2-R0 of MPACS set the PCS4-6 READY mode.

Chip Select/Ready Logic and Reset

Upon reset, the Chip-Select/Ready Logic will perform the following actions:

- All chip-select outputs will be driven HIGH.
- Upon leaving RESET, the UCS line will be programmed to provide chip selects to a 1K block with the accompanying READY control bits set at 011 to allow the maximum number of internal wait states in conjunction with external Ready consideration (i.e., UMCS resets to FFFBH).
- No other chip select or READY control registers have any predefined values after RESET. They will not become active until the CPU accesses their control registers. Both the PACS and MPACS registers must be accessed before the PCS lines will become active.

The user should program bits 15-6 to correspond to the desired peripheral base location. PACS bits 0-2 are used to specify READY mode for PCS0-PCS3.

Table 10. PCS Address Ranges

PCS Line	Active between Locations
PCS0	PBA —PBA + 127
PCS1	PBA + 128—PBA + 255
PCS2	PBA + 256—PBA + 383
PCS3	PBA + 384—PBA + 511
PCS4	PBA + 512—PBA + 639
PCS5	PBA + 640—PBA + 767
PCS6	PBA + 768—PBA + 895

The mode of operation of the peripheral chip selects is defined by the MPACS register (which is also used to set the size of the mid-range memory chip-select block, see Figure 13). This register is located at offset A8H in the internal control block. Bit 7 is used to select the function of PCS5 and PCS6, while bit 6 is used to select whether the peripheral chip selects are mapped into memory or I/O space. Table 11 describes the programming of these bits. After reset, the contents of both the MPACS and the PACS registers are undefined, however none of the PCS lines will be active until both of the MPACS and PACS registers are accessed.

Table 11. MS, EX Programming Values

Bit	Description
MS	1 = Peripherals mapped into memory space.
	0 = Peripherals mapped into I/O space.
EX	0 = 5 PCS lines. A1, A2 provided.
	1 = 7 PCS lines. A1, A2 are not provided.

MPACS bits 0-2 are used to specify READY mode for PCS4-PCS6 as outlined below.

READY Generation Logic

The M80C186 can generate a "READY" signal internally for each of the memory or peripheral CS lines. The number of WAIT states to be inserted for each peripheral or memory is programmable to provide 0-3 wait states for all accesses to the area for which the chip select is active. In addition, the M80C186 may be programmed to either ignore external READY for each chip-select range individually or to factor external READY with the integrated ready generator.

READY control consists of 3 bits for each CS line or group of lines generated by the M80C186. The interpretation of the ready bits is shown in Table 12.



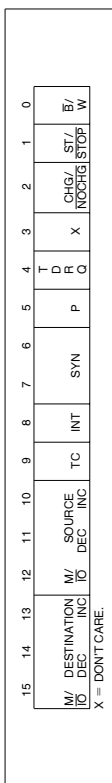


Figure 17. DMA Control Register

DMA Channel Control Word Register

Each DMA Channel Control Word determines the mode of operation for the particular M80C186 DMA channel. This register specifies:

- the mode of synchronization;
- whether bytes or words will be transferred;
- whether interrupts will be generated after the last transfer;
- whether DMA activity will cease after a programmed number of DMA cycles;
- the relative priority of the DMA channel with respect to the other DMA channel;
- whether the source pointer will be incremented, decremented, or maintained constant after each transfer;
- whether the source pointer addresses memory or I/O space;
- whether the destination pointer will be incremented, decremented, or maintained constant after each transfer; and
- whether the destination pointer will address memory or I/O space.

The DMA channel control registers may be changed while the channel is operating. However, any changes made during operation will affect the current DMA transfer.

DMA Control Word Bit Descriptions

- B/W:** Byte/Word (0/1) Transfers.
ST/STOP: Start/stop (1/0) Channel.
CHG/NOCHG: Change/Do not change (1/0) ST/STOP bit. If this bit is set when writing to the control word, the ST/STOP bit will be programmed by the write to the control word. If this bit is cleared when writing the control word, the ST/STOP bit will not be altered. This bit is not stored; it will always be a 0 on read.
INT: Enable Interrupts to CPU on Transfer Count termination.
TC: If set, DMA will terminate when the contents of the Transfer Count

If both INC and DEC are specified for the same pointer, the pointer will remain constant after each cycle.

DMA Destination and Source Pointer Registers

Each DMA channel maintains a 20-bit source and a 20-bit destination pointer. Each of these pointers takes up two full 16-bit registers in the peripheral control block. The lower four bits of the upper register contain the upper four bits of the 20-bit physical address (see Figure 18). These pointers may be individually incremented or decremented after each transfer. If word transfers are performed the pointer is incremented or decremented by two. Each pointer may point into either memory or I/O space. Since the DMA channels can perform transfers to or from odd addresses, there is no restriction on values for the pointer registers. Higher transfer rates can be obtained if all word transfers are performed to even addresses, since this will allow data to be accessed in a single memory access.

DMA Transfer Count Register

Each DMA channel maintains a 16-bit transfer count register (TC). This register is decremented after every DMA cycle, regardless of the state of the TC bit in the DMA Control Register. If the TC bit in the DMA control word is set or if unsynchronized transfers are programmed, however, DMA activity will terminate when the transfer count register reaches zero.

DMA Requests

Data transfers may be either source or destination synchronized, that is either the source of the data or the destination of the data may request the data transfer. In addition, DMA transfers may be unsynchronized; that is, the transfer will take place continually until the correct number of transfers has occurred. When source or unsynchronized transfers are performed, the DMA channel may begin another DMA transfer. This allows a complete transfer to take place every 2 bus cycles or eight clock cycles (assuming no wait states). No preflashing occurs when destination synchronization is performed, however. Data will not be fetched from the source address until the destination device signals that it is ready to receive it. When destination synchronized transfers are requested, the DMA controller will relinquish control of the bus after every transfer. If no other bus activity is initiated, another DMA cycle will begin after two processor clocks. This is done to allow the destination device time to remove its request if another transfer is not desired. Since the DMA controller will relinquish the bus, the CPU can initiate a bus cycle. As a result, a complete bus cycle will often be inserted between destination synchronized transfers. These lead to the maximum DMA transfer rates shown in Table 14.

Table 14. Maximum DMA Transfer Rates

Type of Synchronization Selected	CPU Running	CPU Halted
Unsynchronized	2.5MBytes/sec	2.5MBytes/sec
Source Synchron	2.5MBytes/sec	2.5MBytes/sec
Destination Synchron	1.7MBytes/sec	2.0MBytes/sec

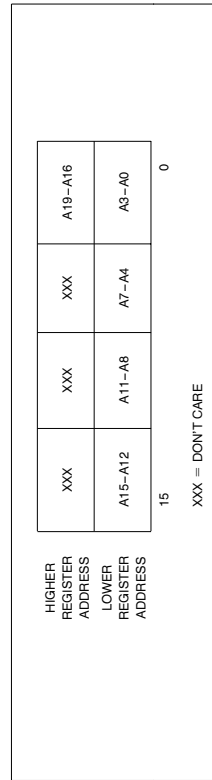


Figure 18. DMA Memory Pointer Register Format



DMA Acknowledge

No explicit DMA acknowledge pulse is provided. Since both source and destination pointers are maintained, a read from a requesting source, or a write to a requesting destination, should be used as the DMA acknowledge signal. Since the chip-select lines can be programmed to be active for a given block of memory or I/O space, and the DMA pointers can be programmed to point to the same given block, a chip-select line could be used to indicate a DMA acknowledge.

DMA Priority

The DMA channels may be programmed such that one channel is always given priority over the other, or they may be programmed such as to alternate cycles when both have DMA requests pending. DMA cycles always have priority over internal CPU cycles except between locked memory accesses or word accesses to odd memory locations; however, an external bus hold takes priority over an internal DMA cycle. Because an interrupt request cannot suspend a DMA operation and the CPU cannot access memory during a DMA cycle, interrupt latency time will suffer during sequences of continuous DMA cycles. An NMI request, however, will cause all internal DMA activity to halt. This allows the CPU to quickly respond to the NMI request.

DMA Programming

DMA cycles will occur whenever the ST/STOP bit of the Control Register is set. If synchronized transfers



are programmed, a DRQ must also have been generated. Therefore the source and destination transfer pointers, and the transfer count register (if used) must be programmed before this bit is set.

Each DMA register may be modified while the channel is operating. If the CHG/NOCHG bit is cleared when the control register is written, the ST/STOP bit of the control register will not be modified by the write. If multiple channel registers are modified, it is recommended that a LOCKED string transfer be used to prevent a DMA transfer from occurring between updates to the channel registers.

DMA Channels and Reset

Upon RESET, the DMA channels will perform the following actions:

- The Start/Stop bit for each channel will be reset to STOP.
- Any transfer in progress is aborted.

TIMERS

The M80C186 provides three internal 16-bit programmable timers (see Figure 19). Two of these are highly flexible and are connected to four external pins (2 per timer). They can be used to count external events, time external events, generate nonrepetitive waveforms, etc. The third timer is not connected to any external pins, and is useful for real-time coding and time delay applications. In addition, this third timer can be used as a prescaler to the other two, or as a DMA request source.

Timer Operation

The timers are controlled by 11 16-bit registers in the internal peripheral control block. The configuration of these registers is shown in Table 15. The control register contains the current value of the timer. It can be read or written at any time independent of whether the timer is running or not. The value of this register will be incremented for each timer event. Each of the timers is equipped with a MAX COUNT register, which defines the maximum count the timer will reach. After reaching the MAX COUNT register value, the timer count value will reset to zero during that same clock, i.e., the maximum count value is never stored in the count register itself. Timers 0 and 1 are, in addition, equipped with a second MAX COUNT register, which enables the timers to alternate their count between two different MAX COUNT values programmed by the user. If a single MAX COUNT register is used, the timer output pin will switch LOW for a single clock, 1 clock after the maximum count value has been reached. In the dual MAX COUNT register mode, the output pin will indicate which MAX COUNT register is currently in use, thus allowing nearly complete freedom in selecting waveform duty cycles. For the timers with two MAX COUNT registers, the RIU bit in the control register determines which is used for the comparison.

Each timer gets serviced every fourth CPU-clock cycle, and thus can operate at speeds up to one-quarter the internal clock frequency (one-eighth the crystal rate). External clocking of the timers may be done at up to a rate of one-quarter of the internal CPU-clock rate. Due to internal synchronization and pipelining of the timer circuitry, a timer output may take up to 6 clocks to respond to any individual clock or gate input.



Since the count registers and the maximum count registers are all 16 bits wide, 16 bits of resolution are provided. Any Read or Write access to the timers will add one wait state to the minimum four-clock bus cycle, however. This is needed to synchronize and coordinate the internal data flows between the internal timers and the internal bus.

- The timers have several programmable options.
- All three timers can be set to halt or continue on a terminal count.
 - Timers 0 and 1 can select between internal and external clocks, alternate between MAX COUNT registers and be set to retrigger on external events.
 - The timers may be programmed to cause an interrupt on terminal count.

These options are selectable via the timer mode/control word.

Timer Mode/Control Register

The mode/control register (see Figure 20) allows the user to program the specific mode of operation or check the current programmed status for any of the three integrated timers.

Table 15. Timer Control Block Format

Register Name	Register Offset		
	Tmr. 0	Tmr. 1	Tmr. 2
Mode/Control Word	56H	5EH	66H
Max Count B	54H	5CH	not present
Max Count A	52H	5AH	62H
Count Register	50H	58H	60H

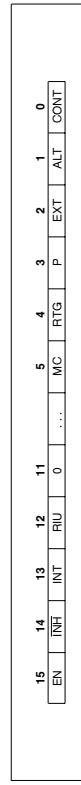


Figure 20. Timer Mode/Control Register

DMA Acknowledge

No explicit DMA acknowledge pulse is provided. Since both source and destination pointers are maintained, a read from a requesting source, or a write to a requesting destination, should be used as the DMA acknowledge signal. Since the chip-select lines can be programmed to be active for a given block of memory or I/O space, and the DMA pointers can be programmed to point to the same given block, a chip-select line could be used to indicate a DMA acknowledge.

DMA Priority

The DMA channels may be programmed such that one channel is always given priority over the other, or they may be programmed such as to alternate cycles when both have DMA requests pending. DMA cycles always have priority over internal CPU cycles except between locked memory accesses or word accesses to odd memory locations; however, an external bus hold takes priority over an internal DMA cycle. Because an interrupt request cannot suspend a DMA operation and the CPU cannot access memory during a DMA cycle, interrupt latency time will suffer during sequences of continuous DMA cycles. An NMI request, however, will cause all internal DMA activity to halt. This allows the CPU to quickly respond to the NMI request.

DMA Programming

DMA cycles will occur whenever the ST/STOP bit of the Control Register is set. If synchronized transfers

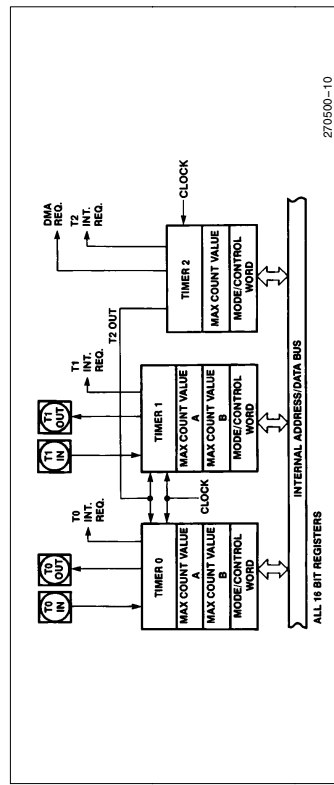


Figure 19. Timer Block Diagram



ALT:

The ALT bit determines which of two MAX COUNT registers is used for count comparison. If ALT = 0, register A for that timer is always used, while if ALT = 1, the comparison will alternate between register A and register B when each maximum count is reached. This alternation allows the user to change one MAX COUNT register while the other is being used, and thus provides a method of generating non-repetitive waveforms. Square waves and pulse outputs of any duty cycle are a subset of available signals obtained by not changing the final count registers. The ALT bit also determines the function of the timer output pin. If ALT is zero, the output pin will go LOW for one clock, the clock after the maximum count is reached. If ALT is one, the output pin will reflect the current MAX COUNT register being used (0/1 for B/A).

CONT:

Setting the COUNT bit causes the associated timer to run continuously, while resetting it causes the timer to halt upon maximum count. If COUNT = 0 and ALT = 1, the timer will count to the MAX COUNT register A value, reset, count to the register B value, reset, and halt.

EXT:

The external bit selects between internal and external clocking for the timer. The external signal may be asynchronous with respect to the M80C186 clock. If this bit is set, the timer will count LOW-to-HIGH transitions on the input pin. If cleared, it will count an internal clock while using the input pin for control. In this mode, the function of the external pin is defined by the RTG bit. The maximum input to output transition latency time may be as much as 6 clocks. However, clock inputs may be pipelined as closely together as every 4 clocks without losing clock pulses.

P:

The prescaler bit is ignored unless internal clocking has been selected (EXT = 0). If the P bit is a zero, the timer will count at one-fourth the internal CPU clock rate. If the P bit is a one, the output of timer 2 will be used as a clock for the timer. Note that the user must initialize and start timer 2 to obtain the prescaled clock.

RTG:

Retrigger bit is only active for internal clocking (EXT = 0). In this case it determines the control function provided by the input pin.

If RTG = 0, the input level gates the internal clock on and off. If the input pin is HIGH, the timer will count; if the input pin is LOW, the timer will hold its value. As indicated previously, the input signal may be asynchronous with respect to the M80C186 clock.

When RTG = 1, the input pin detects LOW-to-HIGH transitions. The first such transition starts the timer running, clearing the timer value to zero on the first clock, and then incrementing thereafter. Further transitions on the input pin will again reset the timer to zero, from which it will start counting up again. If COUNT = 0, when the timer has reached maximum count, the EN bit will be cleared, inhibiting further timer activity.

EN:

The enable bit provides programmer control over the timer's RUN/HALT status. When set, the timer is enabled to increment subject to the input pin constraints in the internal clock mode (discussed previously). When cleared, the timer will be inhibited from counting. All input pin transitions during the time EN is zero will be ignored. If COUNT is zero, the EN bit is automatically cleared upon maximum count.

INH:

The inhibit bit allows for selective updating of the enable (EN) bit. If INH is a one during the write to the mode/control word, then the state of the EN bit will be modified by the write. If INH is a zero during the write, the EN bit will be unaffected by the operation. This bit is not stored; it will always be a 0 on a read.

INT:

When set, the INT bit enables interrupts from the timer, which will be generated on every terminal register mode, an interrupt will be generated each time the value in MAX COUNT register A is reached, and each time the value in MAX COUNT register B is reached. If this enable bit is cleared after the interrupt request has been generated, but before a pending interrupt is serviced, the interrupt request will still be in force. (The request is latched in the Interrupt Controller).

MC:

The Maximum Count bit is set whenever the timer reaches its final maximum count value. If the timer is configured in dual MAX COUNT register mode, this bit will be set each time the value in MAX COUNT register A is reached, and each time the value in MAX COUNT register B is reached. This bit is set

regardless of the timer's interrupt-enable bit. The MC bit gives the user the ability to monitor timer status through software instead of through interrupts.

Programmer intervention is required to clear this bit.

RIU:

The Register In Use bit indicates which MAX COUNT register is currently being used for comparison to the timer count value. A zero value indicates register A. The RIU bit cannot be written, i.e., its value is not affected when the control register is written. It is always cleared when the ALT bit is zero.

Not all mode bits are provided for timer 2. Certain bits are hardwired as indicated below:

ALT = 0, EXT = 0, P = 0, RTG = 0, RIU = 0

Count Registers

Each of the three timers has a 16-bit count register. The current contents of this register may be read or written by the processor at any time. If the register is written into while the timer is counting, the new value will take effect in the current count cycle.

Max Count Registers

Timers 0 and 1 have two MAX COUNT registers, while timer 2 has a single MAX COUNT register. These contain the number of events the timer will count. In timers 0 and 1, the MAX COUNT register used can alternate between the two max count values whenever the current maximum count is reached. The condition which causes a timer to reset is equivalent between the current count value and the max count being used. This means that if the count is changed to be above the max count value, or if the max count value is changed to be below the current value, the timer will not reset to zero, but rather will count to its maximum value, "wrap around" to zero, then count until the max count is reached.

Timers and Reset

Upon RESET, the Timers will perform the following actions:

- All EN (Enable) bits are reset preventing timer counting.
- All SEL (Select) bits are reset to zero. This selects MAX COUNT register A, resulting in the Timer Out pins going HIGH upon RESET.

INTERRUPT CONTROLLER

The M80C186 can receive interrupts from a number of sources, both internal and external. The internal interrupt controller serves to merge these requests on a priority basis, for individual service by the CPU.

Internal interrupt sources (Timers and DMA channels) can be disabled by their own control registers or by mask bits within the interrupt controller. The M80C186 interrupt controller has its own control register that set the mode of operation for the controller.

The interrupt controller will resolve priority among requests that are pending simultaneously. Nesting is provided so interrupt service routines for lower priority interrupts may themselves be interrupted by higher priority interrupts. A block diagram of the interrupt controller is shown in Figure 21.

The M80C186 has a special slave mode in which the internal interrupt controller acts as a slave to an external master. The controller is programmed into this mode by setting bit 14 in the peripheral control block relocation register. (See Slave Mode section.)

MASTER MODE OPERATION
Interrupt Controller External Interface

For external interrupt sources, five dedicated pins are provided. One of these pins is dedicated to NMI, non-maskable interrupt. This is typically used for power-fail interrupts, etc. The other four pins may function either as four interrupt input lines with internally generated interrupt vectors, as an interrupt line and an interrupt acknowledge line (called the "cascade mode") along with two other input lines with internally generated interrupt vectors, or as two interrupt input lines and two dedicated interrupt acknowledge output lines. When the interrupt lines are configured in cascade mode, the M80C186 interrupt controller will not generate internal interrupt vectors.

External sources in the cascade mode use externally generated interrupt vectors. When an interrupt is acknowledged, two INTA cycles are initiated and the vector is read into the M80C186 on the second cycle. The capability to interface to external M82C59A programmable interrupt controllers is thus provided when the inputs are configured in cascade mode.

Interrupt Controller Modes of Operation

The basic modes of operation of the interrupt controller in master mode are similar to the M82C59A. The interrupt controller responds identically to internal interrupts in all three modes; the difference is only in the interpretation of function of the four external interrupt pins. The interrupt controller is set into one of these three modes by programming the control bits in the INT0 and INT1 control registers. The modes of interrupt controller operation are as follows:

Fully Nested Mode

When in the fully nested mode four pins are used as direct interrupt requests as in Figure 22. The vectors for these four inputs are generated internally. An in-service bit is provided for every interrupt source. If a lower-priority device requests an interrupt while the in-service bit (IS) is set, no interrupt will be generated by the interrupt controller. In addition, if another interrupt request occurs from the same interrupt source while the in-service bit is set, no interrupt will be generated by the interrupt controller. This allows interrupt service routines to operate with interrupts enabled without being themselves interrupted by lower-priority interrupts. Since interrupts are enabled, higher-priority interrupts will be serviced.

When a service routine is completed, the proper IS bit must be reset by writing the proper pattern to the EOI register. This is required to allow subsequent interrupts from this interrupt source and to allow servicing of lower-priority interrupts. An EOI command

is issued at the end of the service routine just before the issuance of the return from interrupt instruction. If the fully nested structure has been upheld, the next highest-priority source with its IS bit set is then serviced.

Cascade Mode

The M80C186 has four interrupt pins and two of them have dual functions. In the fully nested mode the four pins are used as direct interrupt inputs and the corresponding vectors are generated internally. In the cascade mode, the four pins are configured into interrupt input-dedicated acknowledge signal pairs. The interconnection is shown in Figure 23. INT0 is an interrupt input interfaced to an M82C59A, while INT2/INTA0 serves as the dedicated interrupt acknowledge signal to that peripheral. The same is true for INT1 and INT3/INTA1. Each pair can selectively be placed in the cascade or non-cascade mode by programming the proper value into INT0 and INT1 control registers. The use of the dedicated acknowledge signals eliminates the need for the use of external logic to generate INTA and device select signals.

The primary cascade mode allows the capability to serve up to 128 external interrupt sources through the use of external master and slave M82C59As. Three levels of priority are created, requiring priority resolution in the M80C186 interrupt controller, if an master M82C59As, and the slave M82C59As. If an external interrupt is serviced, one IS bit is set at each of these levels. When the interrupt service routine is completed, up to three end-of-interrupt commands must be issued by the programmer.

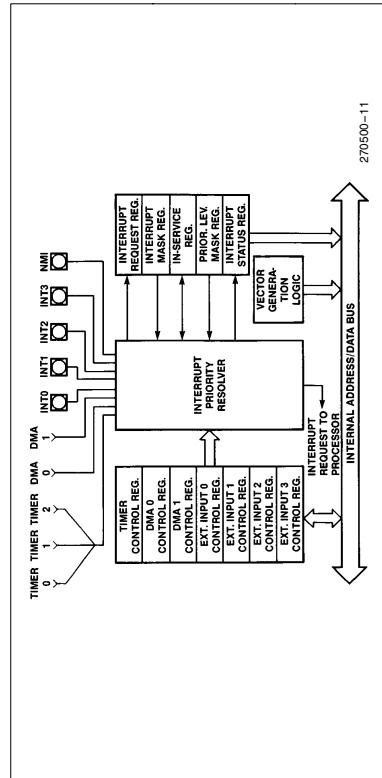


Figure 21. Interrupt Controller Block Diagram

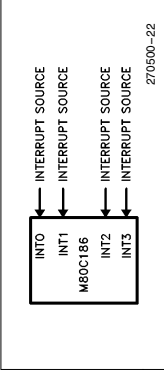


Figure 22. Fully Nested (Direct) Mode Interrupt Controller Connections

Special Fully Nested Mode

This mode is entered by setting the SFNM bit in INT0 or INT1 control register. It enables complete nestability with external M82C59A masters. Normally, an interrupt request from an interrupt source will not be recognized unless the in-service bit for that source is reset. If more than one interrupt source is connected to an external interrupt controller, all of the interrupts will be tunneled through the same M80C186 interrupt request pin. As a result, if the external interrupt controller receives a higher-priority interrupt, its interrupt will not be recognized by the M80C186 controller until the M80C186 in-service bit is reset. In special fully nested mode, the M80C186 interrupt controller will allow interrupts from an external pin regardless of the state of the in-service bit for an interrupt source in order to allow multiple interrupts from a single pin. An in-service bit will continue to be set, however, to inhibit interrupts from other lower-priority M80C186 interrupt sources.

Special procedures should be followed when resetting IS bits at the end of interrupt service routines. Software polling of the external master's IS register is required to determine if there is more than one bit set. If so, the IS bit in the M80C186 remains active and the next interrupt service routine is entered.

Operation in a Polled Environment

The controller may be used in a polled mode if interrupts are undesirable. When polling, the processor disables interrupts and then polls the interrupt controller whenever it is convenient. Polling the interrupt controller is accomplished by reading the Poll Word (Figure 32). Bit 15 in the poll word indicates to the processor that an interrupt of high enough priority is requesting service. Bits 0-4 indicate to the processor the type vector of the highest-priority source requesting service. Reading the Poll Word causes the In-Service bit of the highest priority source to be set.

It is desirable to be able to read the Poll Word information without guaranteeing service of any pending

interrupt, i.e., not set the indicated in-service bit. The M80C186 provides a Poll Status Word in addition to the conventional Poll Word to allow this to be done. Poll Word information is duplicated in the Poll Status Word, but reading the Poll Status Word does not set the associated in-service bit. These words are located in two adjacent memory locations in the register file.

Master Mode Features

Programmable Priority

The user can program the interrupt sources into any of eight different priority levels. The programming is done by placing a 3-bit priority level (0-7) in the control register of each interrupt source. (A source with a priority level of 4 has higher priority over all priority levels from 5 to 7. Priority registers containing values lower than 4 have greater priority). All interrupt sources have preprogrammed default priority levels (see Table 4).

If two requests with the same programmed priority level are pending at once, the priority ordering scheme shown in Table 4 is used. If the serviced interrupt routine reenables interrupts, it allows other requests to be serviced.

End-of-Interrupt Command

The end-of-interrupt (EOI) command is used by the programmer to reset the In-Service (IS) bit when an interrupt service routine is completed. The EOI command is issued by writing the proper pattern to the EOI register. There are two types of EOI commands, specific and nonspecific. The nonspecific command does not specify which IS bit is reset. When issued, the interrupt controller automatically resets the IS bit of the highest priority source with an active service routine. A specific EOI command requires that the programmer send the interrupt vector type to the interrupt controller indicating which source's IS bit is to be reset. This command is used when the fully nested structure has been disturbed or the highest priority IS bit that was set does not belong to the service routine in progress.

Trigger Mode

The four external interrupt pins can be programmed in either edge- or level-trigger mode. The control register for each external source has a level-trigger mode (LTM) bit. All interrupt inputs are active HIGH. In the edge sense mode or the level-trigger mode, the interrupt request must remain active (HIGH) until the interrupt request is acknowledged by the



M80C186 CPU. In the edge-sense mode, if the level remains high after the interrupt is acknowledged, the input is disabled and no further requests will be generated. The input level must go LOW for at least one clock cycle to reenable the input. In the level-trigger mode, no such provision is made: holding the interrupt input HIGH will cause continuous interrupt requests.

Interrupt Vectoring

The M80C186 Interrupt Controller will generate interrupt vectors for the integrated DMA channels and the integrated Timers. In addition, the Interrupt Controller will generate interrupt vectors for the external interrupt lines if they are not configured in Cascade or Special Fully Nested Mode. The interrupt vectors generated are fixed and cannot be changed (see Table 4).

Interrupt Controller Registers

The Interrupt Controller register model is shown in Figure 24. It contains 15 registers. All registers can both be read or written unless specified otherwise.

In-Service Register

This register can be read from or written into. The format is shown in Figure 25. It contains the In-Service bit for each of the interrupt sources. The In-Service bit is set to indicate that a source's service routine is in progress. When an In-Service bit is set, the interrupt controller will not generate interrupts to the CPU when it receives interrupt requests from devices with a lower programmed priority level. The TMR bit is the In-Service bit for all three timers; the D0 and D1 bits are the In-Service bits for the two DMA channels; the I0-I3 are the In-Service bits for the external interrupt pins. The IS bit is set when the

processor acknowledges an interrupt request either by an interrupt acknowledge or by reading the poll register. The IS bit is reset at the end of the interrupt service routine by an end-of-interrupt command issued by the CPU.

Interrupt Request Register

The internal interrupt sources have interrupt request bits inside the interrupt controller. The format of this register is shown in Figure 25. A read from this register yields the status of these bits. The TMR bit is the logical OR of all timer interrupt requests. D0 and D1 are the interrupt request bits for the DMA channels.

The state of the external interrupt input pins is also indicated. The state of the external interrupt pins is not a stored condition inside the interrupt controller, therefore the external interrupt bits cannot be written. The external interrupt request bits show exactly when an interrupt request is given to the interrupt controller, so if edge-triggered mode is selected, the bit in the register will be HIGH only after an inactive-to-active transition. For internal interrupt sources, the register bits are set when a request arrives and are reset when the processor acknowledges the requests.

Writes to the interrupt request register will affect the D0 and D1 interrupt request bits. Setting either bit will cause the corresponding interrupt request while clearing either bit will remove the corresponding interrupt request. All other bits in the register are read-only.

Mask Register

This is a 16-bit register that contains a mask bit for each interrupt source. The format for this register is shown in Figure 25. A one in a bit position corre-

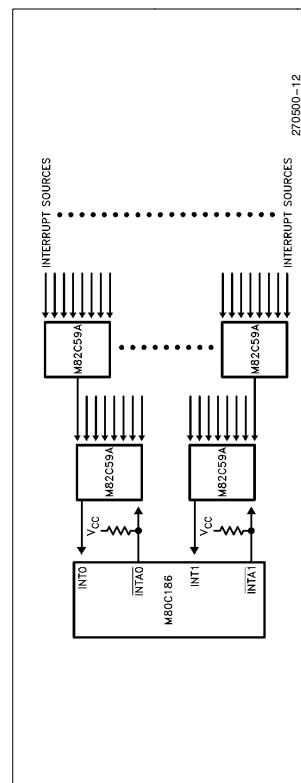


Figure 23. Cascade and Special Fully Nested Mode Interrupt Controller Connections

ponding to a particular source serves to mask the source from generating interrupts. These mask bits are the exact same bits which are used in the individual control registers; programming a mask bit using the mask register will also change this bit in the individual control registers, and vice versa.

	OFFSET
INT3 CONTROL REGISTER	3EH
INT2 CONTROL REGISTER	3CH
INT1 CONTROL REGISTER	3AH
INT0 CONTROL REGISTER	38H
DMA 1 CONTROL REGISTER	36H
DMA 0 CONTROL REGISTER	34H
TIMER CONTROL REGISTER	32H
INTERRUPT STATUS REGISTER	30H
INTERRUPT REQUEST REGISTER	2EH
IN-SERVICE REGISTER	2CH
PRIORITY/MASK REGISTER	2AH
MASK REGISTER	28H
POLL STATUS REGISTER	26H
POLL REGISTER	24H
EOI REGISTER	22H

Figure 24. Interrupt Controller Registers (Master Mode)

Priority Mask Register

This register is used to mask all interrupts below particular interrupt priority levels. The format of this register is shown in Figure 26. The code in the lower three bits of this register inhibits interrupts of priority lower (a higher priority number) than the code specified. For example, 100 written into this register masks interrupts of level five (101), six (110), and seven (111). The register is reset to seven (111) upon RESET so no interrupts are masked due to priority number.

Interrupt Status Register

This register contains general interrupt controller status information. The format of this register is shown in Figure 27. The bits in the status register have the following functions:

DHLT: DMA Halt Transfer; setting this bit halts all DMA transfers. It is automatically set whenever a non-maskable interrupt occurs, and it is reset when an IRET instruction is executed. The purpose of this bit is to allow prompt service of all non-maskable interrupts. This bit may also be set by the programmer.

IRTX: These three bits represent the individual timer interrupt request bits. These bits are used to differentiate the timer interrupts, since the timer IR bit in the interrupt request register is the "OR" function of all timer interrupt requests. Note that setting any one of these three bits initiates an interrupt request to the interrupt controller.

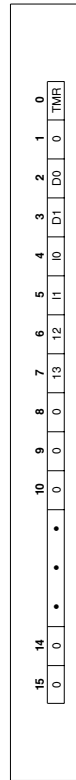


Figure 25. In-Service, Interrupt Request, and Mask Register Formats

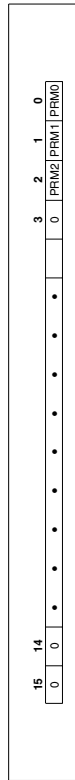


Figure 26. Priority Mask Register Format

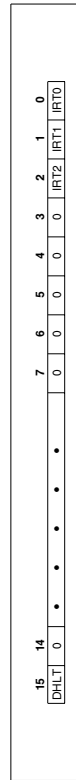


Figure 27. Interrupt Status Register Format (Master Mode)

M80C186

Timer, DMA 0, 1; Control Register

These registers are the control words for all the internal interrupt sources. The format for these registers is shown in Figure 28. The three bit positions PRO, PRI1, and PRI2 represent the programmable priority level of the interrupt source. The MSK bit inhibits interrupt requests from the interrupt source. The MSK bits in the individual control registers are the exact same bits as are in the Mask Register; modifying them in the individual control registers will also modify them in the Mask Register, and vice versa.

INT0-INT3 Control Registers

These registers are the control words for the four external input pins. Figure 29 shows the format of the INT0 and INT1 Control registers; Figure 30 shows the format of the INT2 and INT3 Control registers. In cascade mode or special fully nested mode, the control words for INT2 and INT3 are not used.

The bits in the various control registers are encoded as follows:
 PRO-2: Priority programming information. Highest Priority = 000, Lowest Priority = 111
 LTM: Level-trigger mode bit. 1 = level-triggered; 0 = edge-triggered. Interrupt input levels are active high. In level-triggered mode, an interrupt is generated whenever the external line is high. In edge-triggered mode, an interrupt will be generated only when this



level is preceded by an inactive-to-active transition on the line. In both cases, the level must remain active until the interrupt is acknowledged.

- MSK: Mask bit, 1 = mask; 0 = non-mask.
- C: Cascade mode bit, 1 = cascade; 0 = direct
- SFNM: Special fully nested mode bit, 1 = SFNM

EOI Register

The end of the interrupt register is a command register which can only be written into. The format of this register is shown in Figure 30. It initiates an EOI command when written to by the M80C186 CPU.

The bits in the EOI register are encoded as follows:
 Sx: Encoded information that specifies an interrupt source vector type as shown in Table 4. For example, to reset the In-Service bit for DMA channel 0, these bits should be set to 01010, since the vector type for DMA channel 0 is 10.

NOTE:

To reset the single In-Service bit for any of the three timers, the vector type for timer 0 (8) should be written in this register.
 NSPEC: A bit that determines the type of EOI command. Nonspecific = 1, Specific = 0.



Figure 28. Timer/DMA Control Registers Formats

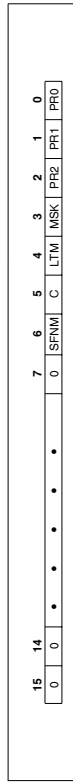


Figure 29. INT0/INT1 Control Register Formats



Figure 30. INT2/INT3 Control Register Formats



Poll and Poll Status Registers

These registers contain polling information. The format of these registers is shown in Figure 32. They can only be read. Reading the Poll register constitutes a software poll. This will set the IS bit of the highest priority pending interrupt. Reading the poll status register will not set the IS bit of the highest priority pending interrupt; only the status of pending interrupts will be provided.

Encoding of the Poll and Poll Status register bits are as follows:

- Sx: Encoded information that indicates the vector type of the highest priority interrupting source. Valid only when INTREQ = 1.
- INTREQ: This bit determines if an interrupt request is present. Interrupt Request = 1; no Interrupt Request = 0.

SLAVE MODE OPERATION

When slave mode is used, the internal M80C186 interrupt controller will be used as a slave controller to an external master interrupt controller. The internal M80C186 resources will be monitored by the internal interrupt controller, while the external controller functions as the system master interrupt controller.

M80C186

Upon reset, the M80C186 will be in master mode. To provide for slave mode operation bit 14 of the location register should be set.

Because of pin limitations caused by the need to interface to an external M82C59A master, the internal interrupt controller will no longer accept external inputs. There are however, enough M80C186 interrupt controller inputs (internally) to dedicate one to each timer. In this mode, each timer interrupt source has its own mask bit, IS bit, and control word.

In slave mode each peripheral must be assigned a unique priority to ensure proper interrupt controller operation. Therefore, it is the programmer's responsibility to assign correct priorities and initialize interrupt control registers before enabling interrupts.

Slave Mode External Interface

The configuration of the M80C186 with respect to an external M82C59A master is shown in Figure 33. The INTO (Pin 45) input is used as the M80C186 CPU interrupt input, INT3 (Pin 41) functions as an output to send the M80C186 slave-interrupt-request to one of the 8 master-PIC-inputs.

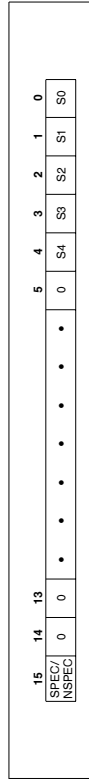


Figure 31. EOI Register Format

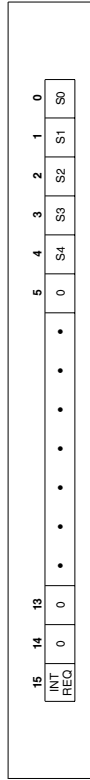


Figure 32. Poll and Poll Status Register Format



The bits of the Control Registers are encoded as follows:

3-bit encoded field indicating a priority level for the source; note that each source must be programmed at specified levels.

mask bit for the priority level indicated by pr_x bits.

LEVEL	CONTROL REGISTER	OFFSET
LEVEL 5	CONTROL REGISTER (TIMER 2)	3AH
LEVEL 4	CONTROL REGISTER (TIMER 1)	38H
LEVEL 3	CONTROL REGISTER (DMA 1)	36H
LEVEL 2	CONTROL REGISTER (DMA 0)	34H
LEVEL 0	CONTROL REGISTER (TIMER 0)	32H
	INTERRUPT STATUS REGISTER	30H
	INTERRUPT-REQUEST REGISTER	2EH
	IN-SERVICE REGISTER	2CH
	PRIORITY-LEVEL MASK REGISTER	2AH
	MASK REGISTER	28H
	SPECIFIC EOI REGISTER	22H
	INTERRUPT VECTOR REGISTER	20H

Figure 34. Interrupt Controller Registers (Slave Mode)

In-Service Register

This register can be read from or written into. It contains the in-service bit for each of the internal interrupt sources. The format for this register is shown in Figure 36. Bit positions 2 and 3 correspond to the DMA channels; positions 0, 4, and 5 correspond to the integral timers. The source's IS bit is set when the processor acknowledges its interrupt request.

Interrupt Request Register

This register indicates which internal peripherals have interrupt requests pending. The format of this register is shown in Figure 36. The interrupt request bits are set when a request arrives from an internal source, and are reset when the processor acknowledges the request. As in master mode, D0 and D1 are read/write; all other bits are read only.

Mask Register

The register contains a mask bit for each interrupt source. The format for this register is shown in Figure 36. If the bit in this register corresponding to a particular interrupt source is set, any interrupts from that source will be masked. These mask bits are exactly the same bits which are used in the individual control registers, i.e., changing the state of a mask bit in this register will also change the state of the mask bit in the individual interrupt control register corresponding to the bit.

Control Registers

These registers are the control words for all the internal interrupt sources. The format of these registers is shown in Figure 37. Each of the timers and both of the DMA channels have their own Control Register.

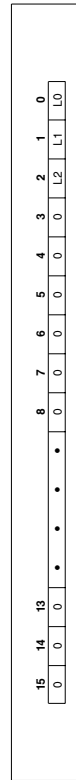


Figure 35. Specific EOI Register Format

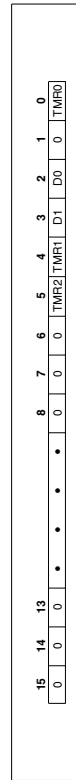


Figure 36. In-Service, Interrupt Request, and Mask Register Format

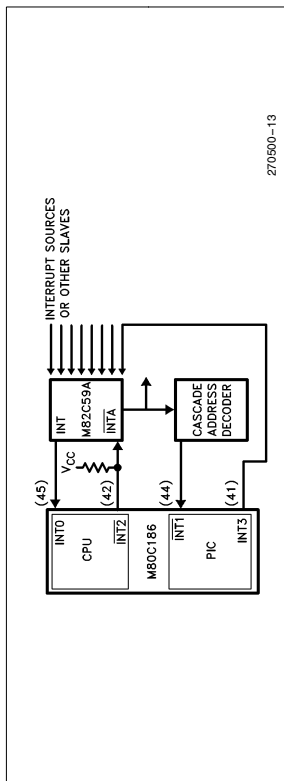


Figure 33. Slave Mode Interrupt Controller Connections

Correct master-slave interface requires decoding of the slave addresses (CASO-2). Slave M82C59As do this internally. Because of pin limitations, the M80C186 slave address will have to be decoded externally. INT1 (Pin 44) is used as a slave-select input. Note that the slave vector address is translated internally, but the READY input must be supplied externally.

INT2 (Pin 42) is used as an acknowledge output, suitable to drive the INTA input of an M82C59A.

Interrupt Nesting

Slave mode operation allows nesting of interrupt requests. When an interrupt is acknowledged, the priority logic masks off all priority levels except those with equal or higher priority.

Vector Generation in the Slave Mode

Vector generation in slave mode is exactly like that of an M82C59A slave. The interrupt controller generates an 8-bit vector which the CPU multiplies by four and uses as an address into a vector table. The significant five bits of the vector are user-programmable while the lower three bits are generated by the priority logic. These bits represent the encoding of the priority level requesting service. The significant five bits of the vector are programmed by writing to the Interrupt Vector register at offset 20H.

Interrupt Controller Registers in the Slave Mode

All control and command registers are located inside the internal peripheral control block. Figure 34 shows the offsets of these registers.

End-of-Interrupt Register

The end-of-interrupt register is a command register which can only be written. The format of this register is shown in Figure 35. It initiates an EOI command when written by the M80C186 CPU.

The bits in the EOI register are encoded as follows:

L_x : Encoded value indicating the priority of the IS bit to be reset.

M80C186

Interrupt Vector Register

This register provides the upper five bits of the interrupt vector address. The format of this register is shown in Figure 38. The interrupt controller itself provides the lower three bits of the interrupt vector as determined by the priority level of the interrupt request.

The format of the bits in this register is:
 fx: 5-bit field indicating the upper five bits of the vector address.

Priority-Level Mask Register

This register indicates the lowest priority-level interrupt which will be serviced.

The encoding of the bits in this register is:

mx: 3-bit encoded field indicating priority-level value. All levels of lower priority will be masked.

Interrupt Status Register

This register is defined as in master mode except that DHLT is not implemented (see Figure 27).

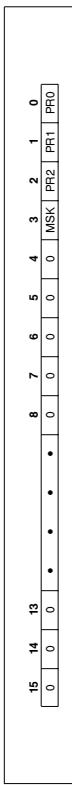


Figure 37. Control Word Format

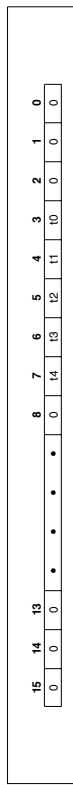


Figure 38. Interrupt Vector Register Format

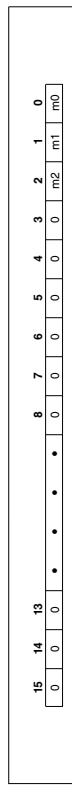


Figure 39. Priority Level Mask Register



Enhanced Mode Operation

In Enhanced Mode, the M80C186 will operate with Power-Save, DRAM refresh, and numerics coprocessor support in addition to all the Compatible Mode features.

In Compatible Mode the M80C186 operates with all the features of the NMOS M80186, with the exception of M8087 support (i.e. no numeric coprocessing is possible in Compatible Mode). Queue-Status information is still available for design purposes other than M8087 support.

All the Enhanced Mode features are completely masked when in Compatible Mode. A write to any of the Enhanced Mode registers will have no effect, while a read will not return any valid data.

Entering Enhanced Mode

If connected to a numerics coprocessor, this mode will be invoked automatically. Without a NPX, this mode can be entered by tying the RESET output signal from the M80C186 to the TEST/BUSY input.

Queue-Status Mode

The queue-status mode is entered by strapping the RD pin low. RD is sampled at RESET and if LOW, the M80C186 will reconfigure the ALE and WR pins to be QS0 and QS1 respectively. This mode is available on the M80C186 in both Compatible and Enhanced Modes and is identical to the NMOS M80186.

M80C186

DRAM Refresh Control Unit Description

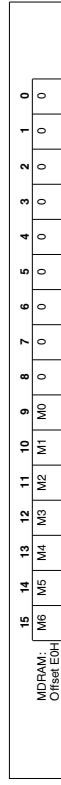
The Refresh Control Unit (RCU) automatically generates DRAM refresh bus cycles. The RCU operates in Enhanced Mode. After a programmable period of time, the RCU generates a memory read request to the BIU. If the address generated during a refresh bus cycle is within the range of a properly programmed chip select, that chip select will be activated when the BIU executes the refresh bus cycle. The ready logic and wait states programmed for that region will also be in force. If no chip select is activated, then external ready is automatically required to terminate the refresh bus cycle.

If the HLDA pin is active when a DRAM refresh request is generated (indicating a bus hold condition), then the M80C186 will deactivate the HLDA pin in order to perform a refresh cycle. The circuit external to the M80C186 must remove the HOLD signal in order to execute the refresh cycle. The sequence of HLDA going inactive while HOLD is being held active can be used to signal a pending refresh request.

All registers controlling DRAM refresh may be read and written in Enhanced Mode. When the processor is operating in Compatible Mode, they are deselected and are therefore inaccessible. Some fields of these registers cannot be written and are always read as zeros.

DRAM Refresh Addresses

The address generated during a refresh cycle is determined by the contents of the MDRAM register (see Figure 40) and the contents of a 9-bit counter. Figure 41 illustrates the origin of each bit.



Bits 0-8: Reserved, read back as 0.

Bits 9-15: M0-M6, are address bits A13-A19 of the 20-bit memory refresh address. These bits should correspond to the chip select address to be activated for the DRAM partition. These bits are set to 0 on RESET.

Figure 40. Memory Partition Register

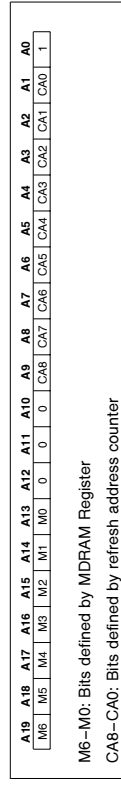


Figure 41. Addresses Generated by RCU



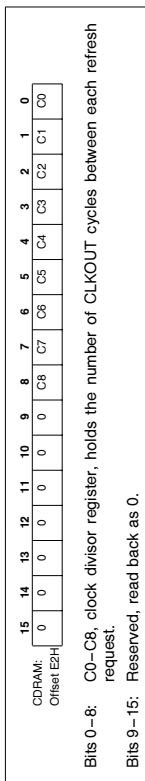


Figure 42. Clock Pre-Scaler Register

Bits 0-8: C0-C8, clock divisor register, holds the number of CLKOUT cycles between each refresh request.
 Bits 9-15: Reserved, read back as 0.

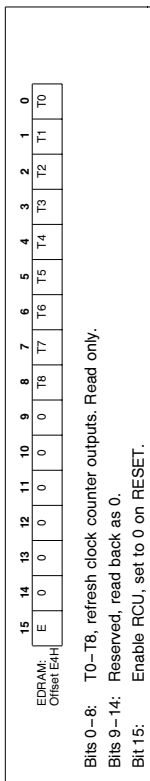


Figure 43. Enable RCU Register

Bits 0-8: T0-T8, refresh clock counter outputs. Read only.
 Bits 9-14: Reserved, read back as 0.
 Bit 15: Enable RCU, set to 0 on RESET.

Refresh Control Unit Programming and Operation

After programming the MDRAM and the CDRAM registers (Figures 40 and 42), the RCU is enabled by setting the "E" bit in the EDRAM register (Figure 43). The clock counter (T0-T8 of EDRAM) will be loaded from C0-C8 or CDRAM during T3 of instruction cycle that sets the "E" bit. The clock counter is then decremented at each subsequent CLKOUT.

A refresh is requested when the value of the counter has reached 1 and the counter is reloaded from CDRAM. In order to avoid missing refresh requests, the value in the CDRAM register should always be at least 18 (12H). Clearing the "E" bit at anytime will clear the counter and stop refresh requests, but will not reset the refresh address counter.

POWER-SAVE CONTROL

Power Save Operation

The M80C186, when in Enhanced Mode, can enter a power saving state by internally dividing the clock-in frequency by a programmable factor. This divided

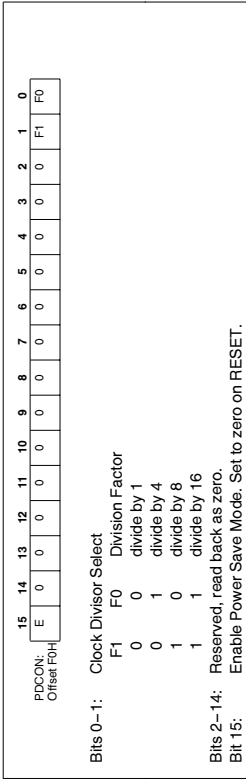


Figure 44. Power-Save Control Register

Bits 0-1: Clock Divisor Select
 F1 F0 Division Factor
 0 0 divide by 1
 0 1 divide by 4
 1 0 divide by 8
 1 1 divide by 16
 Bits 2-14: Reserved, read back as zero.
 Bit 15: Enable Power-Save Mode. Set to zero on RESET.

Numeric Coprocessor (NPX) Extension

Three of the mid-range memory chip selects are re-defined according to Table 16 when using the numeric coprocessor extension. The fourth chip select, MCS2 functions as in compatible mode, and may be programmed for activity with ready logic and wait states accordingly. As in compatible mode, MCS2 will function for one-fourth a programmed block size.

Table 16. MCS Assignments

Compatible Mode	Enhanced Mode
MCS0	PEREQ Processor Extension Request
MCS1	ERROR NPX Error
MCS2	MCS2 Mid-Range Chip Select
MCS3	NPS Numeric Processor Select

Four port addresses are assigned to the NPX for 16-bit reads and writes by the M80C186. Table 17 shows the port definitions. These ports are not accessible by using the M80C186 I/O instructions. However, numerics operations will cause a PCS line to be activated if it is properly programmed for this I/O range.

Table 17. Numerics Coprocessor I/O Port Assignments

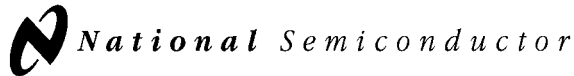
I/O Address	Read Definition	Write Definition
00F8H	Status/Control Data	Opcode Data
00FAH	reserved	CS,IP,DS,EA reserved
00FCH	Opcode Status	
00FEH	Opcode Status	

"ONCE" Test Mode

To facilitate testing and inspection of devices when fixed into a target system, the M80C186 has a test mode available which allows all pins to be placed in a high-impedance state. "ONCE" stands for "ON Circuit Emulation". When placed in this mode, M80C186 will put all pins in the high-impedance state until RESET.

The ONCE mode is selected by tying the UCS and the LCS LOW during RESET. These pins are sampled on the low-to-high transition of the RES pin. The UCS and the LCS pins have weak internal pull-up resistors similar to the RD and TEST/BUSY pins to guarantee proper normal operation.



A/D CONVERTER LM 12H458

July 1999

LM12454/LM12458/LM12H458

12-Bit + Sign Data Acquisition System with Self-Calibration

General Description

The LM12454, LM12458, and LM12H458 are highly integrated Data Acquisition Systems. Operating on just 5V, they combine a fully-differential self-calibrating (correcting linearity and zero errors) 13-bit (12-bit + sign) analog-to-digital converter (ADC) and sample-and-hold (S/H) with extensive analog functions and digital functionality. Up to 32 consecutive conversions, using two's complement format, can be stored in an internal 32-word (16-bit wide) FIFO data buffer. An internal 8-word RAM can store the conversion sequence for up to eight acquisitions through the LM12(H)458's eight-input multiplexer. The LM12454 has a four-channel multiplexer, a differential multiplexer output, and a differential S/H input. The LM12454 and LM12(H)458 can also operate with 8-bit + sign resolution and in a supervisory "watchdog" mode that compares an input signal against two programmable limits.

Programmable acquisition times and conversion rates are possible through the use of internal clock-driven timers. The reference voltage input can be externally generated for absolute or ratiometric operation or can be derived using the internal 2.5V bandgap reference.

All registers, RAM, and FIFO are directly addressable through the high speed microprocessor interface to either an 8-bit or 16-bit databus. The LM12454 and LM12(H)458 include a direct memory access (DMA) interface for high-speed conversion data transfer.

An evaluation/interface board is available. Order number LM12458EVAL.

Additional applications information can be found in applications notes AN-906, AN-947 and AN-949.

Key Specifications

($f_{CLK} = 5 \text{ MHz}; 8 \text{ MHz, H}$)

■ Resolution	12-bit + sign or 8-bit + sign
■ 13-bit conversion time	8.8 μs , 5.5 μs (H) (max)
■ 9-bit conversion time	4.2 μs , 2.6 μs (H) (max)
■ 13-bit Through-put rate	88k samples/s (min), 140k samples/s (H) (min)
■ Comparison time ("watchdog" mode)	2.2 μs (max), 1.4 μs (H) (max)
■ ILE	$\pm 1 \text{ LSB}$ (max)
■ V_{IN} range	GND to V_{A+}
■ Power dissipation	30 mW, 34 mW (H) (max)
■ Stand-by mode	50 μW (typ)
■ Single supply	3V to 5.5V

Features

- Three operating modes: 12-bit + sign, 8-bit + sign, and "watchdog"
- Single-ended or differential inputs
- Built-in Sample-and-Hold and 2.5V bandgap reference
- Instruction RAM and event sequencer
- 8-channel (LM12(H)458), 4-channel (LM12454) multiplexer
- 32-word conversion FIFO
- Programmable acquisition times and conversion rates
- Self-calibration and diagnostic mode
- 8- or 16-bit wide databus microprocessor or DSP interface

Applications

- Data Logging
- Instrumentation
- Process Control
- Energy Management
- Inertial Guidance

TRI-STATE® is a registered trademark of National Semiconductor Corporation.
AT® is a registered trademark of International Business Machines Corporation.



Application Information

1.0 Functional Description

The LM12454 and LM12(H)458 are multi-functional Data Acquisition Systems that include a fully differential 12-bit-plus-sign self-calibrating analog-to-digital converter (ADC) with a two's-complement output format, an 8-channel (LM12(H)458) or a 4-channel (LM12454) analog multiplexer, an internal 2.5V reference, a first-in-first-out (FIFO) register that can store as many as eight instructions to be sequentially executed. The LM12454 also has a differential multiplexer output and a differential S/H input. All of this circuitry operates on only a single +5V power supply.

The LM12(H)454/8 have three modes of operation:

- 12-bit + sign with correction
 - 8-bit + sign without correction
 - 8-bit + sign comparison mode ("watchdog" mode)
- The fully differential 12-bit-plus-sign ADC uses a charge redistribution topology that includes calibration capabilities. Charge re-distribution ADCs use a capacitor ladder in place of a resistor ladder to form an internal DAC. The DAC is used by a successive approximation register to generate intermediate voltages between the voltages applied to V_{REF-} and V_{REF+} . These intermediate voltages are compared against the sampled analog input voltage as each bit is generated. The number of intermediate voltages and comparisons equals the ADC's resolution. The correction of each bit's accuracy is accomplished by calibrating the capacitor ladder used in the ADC.

Two different calibration modes are available; one compensates for offset voltage, or zero error, while the other corrects both offset error and the ADC's linearity error.

When correcting offset only, the offset error is measured once and a correction coefficient is created. During the full calibration, the offset error is measured eight times, averaged, and a correction coefficient is created. After completion of either calibration mode, the offset correction coefficient is stored in an internal offset correction register.

The LM12(H)454/8's overall linearity correction is achieved by connecting the internal DAC's capacitor mismatch. Each smaller value capacitors and any errors are averaged. A correction coefficient is then created and stored in one of the thirteen internal linearity correction registers. An internal state machine, using patterns stored in an internal 16 x 8-bit ROM, executes each calibration algorithm.

Once calibrated, an internal arithmetic logic unit (ALU) uses the offset correction coefficient and the 13 linearity correction coefficients to reduce the converter's offset error and linearity error, in the background, during the 12-bit + sign conversion. The 8-bit + sign conversion and comparison modes use only the offset coefficient. The 8-bit + sign mode performs a conversion in less than half the time used by the 12-bit + sign conversion mode.

The LM12(H)454/8's "watchdog" mode is used to monitor a single-ended or differential signals' amplitude. Each sampled signal has two limits. An interrupt can be generated if the input signal is above or below either of the two limits. This allows interrupts to be generated when analog voltage inputs are "inside the window" or, alternatively, "outside the window". After a "watchdog" mode interrupt, the processor can then request a conversion on the input signal and read the signal's magnitude.

The analog input multiplexer can be configured for any combination of single-ended or fully differential operation. Each input is referenced to ground when a multiplexer channel operates in the single-ended mode. Fully differential analog input channels are formed by pairing any two channels together.

The LM12454's multiplexer outputs and S/H inputs (MUXOUT+, MUXOUT- and S/H IN+, S/H IN-) provide the option for additional analog signal processing. Fixed-gain amplifiers, programmable-gain amplifiers, filters, and other processing circuits can operate on the signal applied to the selected multiplexer channel(s). If external processing is not used, connect MUXOUT+ to S/H IN+ and MUXOUT- to S/H IN-.

The LM12(H)454/8's internal S/H is designed to operate at its minimum acquisition time (1.13 μ s, 12 bits) when the source impedance, $R_{S, IS}$ is $\leq 60\Omega$ ($f_{CLK} \leq 8$ MHz). When $60\Omega < R_{S, IS} \leq 4.17$ k Ω , the internal S/H's acquisition time can be increased to a maximum of 4.88 μ s (12 bits, $f_{CLK} = 8$ MHz). See Section 2.1 (Instruction RAM "00") Bits 12-15 for more information.

An internal 2.5V bandgap reference output is available at pin 44. This voltage can be used as the ADC reference for ratio-metric conversion or as a virtual ground for front-end analog conditioning circuits. The V_{REFOUT} pin should be bypassed to ground with a 100 μ F capacitor.

Microprocessor overhead is reduced through the use of the internal conversion FIFO. Thirty-two consecutive conversions can be completed and stored in the FIFO without any microprocessor intervention. The microprocessor can, at any time, interrogate the FIFO and retrieve its contents. It can also wait for the LM12(H)454/8 to issue an interrupt when the FIFO is full or after any number (≤ 32) of conversions have been stored.

Conversion sequencing, internal timer, interval, multiplexer configuration, and many other operations are programmed and set in the Instruction RAM.

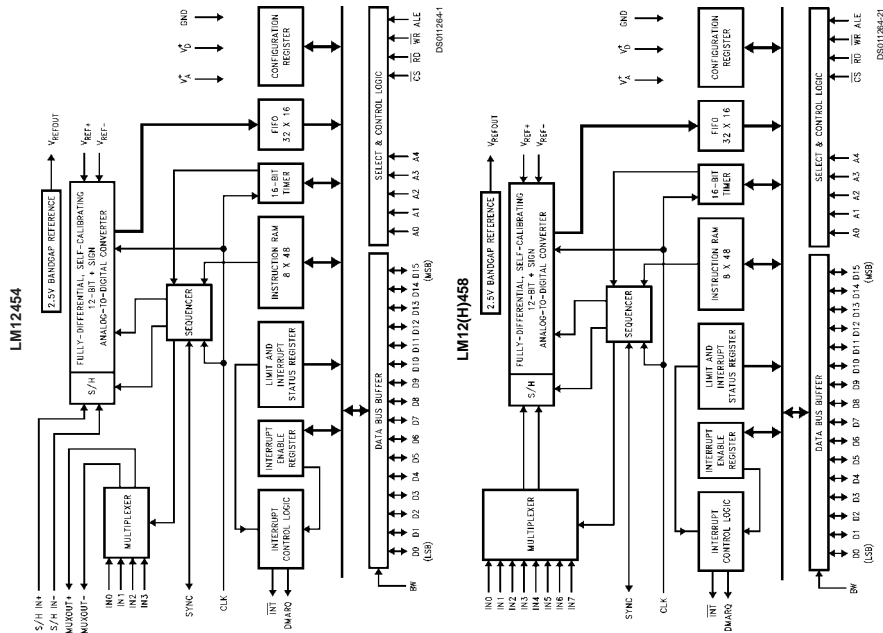
A diagnostic mode is available that allows verification of the LM12(H)458's operation. The diagnostic mode is disabled in the LM12454. This mode internally connects the voltages present at the V_{REFOUT} , V_{REF+} , V_{REF-} , and GND pins to the internal V_{IN+} and V_{IN-} S/H inputs. This mode is activated by setting the Diagnostic bit (Bit 11) in the Configuration register to a "1". More information concerning this mode of operation can be found in Section 2.2.

2.0 Internal User-Programmable Registers

INSTRUCTION RAM

The instruction RAM holds up to eight sequentially executable instructions. Each 48-bit long instruction is divided into three 16-bit sections. READ and WRITE operations can be issued to each 16-bit section using the instruction's address and the 2-bit "RAM pointer" in the Configuration register. The eight instructions are located at addresses 0000 through 0111 (A4-A1, BW = 0) when using a 16-bit wide data bus or at addresses 00000 through 01111 (A4-A0, BW = 1) when using an 8-bit wide data bus. They can be accessed and programmed in random order.

Functional Diagrams



2.0 Internal User-Programmable Registers (Continued)

FIGURE 13. LM12(H)45/48 Memory Map for 16-Bit Wide Databus (BW = "0", Test Bit = "0" and A0 = Don't Care)

Note 21: LM12(H)458 only. Must be set to "0" for the LM12454.
Note 20: LM12454 (Refer to Table 2).

Address	Type	Purpose	RAM (Pointer = 00)	RAM (Pointer = 01)	RAM (Pointer = 10)	Config	Interrupt Enable	Register	Interrupt Status	Timer	Conversion	Limit #2: Status	Limit #1: Status
A4 A3A2A1	R/W	Instruction RAM	0 0 0	0 0 0	0 0 0	1 0 0 0	1 0 0 1	1 0 1 0	1 0 1 0	1 0 1 1	1 1 0 0	1 1 0 1	1 1 0 1
D15	R/W	Acquisition Time	1 1 1	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0
D14	R/W	Watch-dog	1 1 1	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0
D13	R/W	Time	1 1 1	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0
D12	R/W	Acquisition Time	1 1 1	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0
D11	R/W	Watch-dog	1 1 1	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0
D10	R/W	Time	1 1 1	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0
D9	R/W	Timer Sync	1 1 1	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0
D8	R/W	Timer Sync	1 1 1	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0
D7	R/W	V _{IN-} (MUXOUT-)	1 1 1	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0
D6	R/W	V _{IN+} (MUXOUT+)	1 1 1	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0
D5	R/W	V _{IN+} (MUXOUT+)	1 1 1	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0
D4	R/W	V _{IN+} (MUXOUT+)	1 1 1	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0
D3	R/W	V _{IN+} (MUXOUT+)	1 1 1	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0
D2	R/W	V _{IN+} (MUXOUT+)	1 1 1	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0
D1	R/W	V _{IN+} (MUXOUT+)	1 1 1	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0
D0	R/W	V _{IN+} (MUXOUT+)	1 1 1	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0

2.0 Internal User-Programmable Registers (Continued)

Any instruction RAM READ or WRITE can affect the sequencer's operation. The Sequencer should be stopped by setting the RESET bit to a "1" or by resetting the START bit in the Configuration Register and waiting for the current instruction to finish execution before any instruction RAM READ or WRITE is initiated.

A soft RESET should be issued by writing a "1" to the Configuration Register's RESET bit after any READ or WRITE to the Instruction RAM. The three sections in the Instruction RAM are selected by the Configuration Register's 2-bit "RAM Pointer", bits D8 and D9. The first 16-bit instruction RAM section is selected with the RAM Pointer equal to "00". This section provides multiplexer channel selection, as well as resolution, acquisition time, etc. The second 16-bit section holds "watchdog" limit #1, its sign, and an indicator that shows that an interrupt can be generated if the input signal is greater or less than the programmed limit. The third 16-bit section holds "watchdog" limit #2, its sign, and an indicator that shows that an interrupt can be generated if the input signal is greater or less than the programmed limit.

Instruction RAM "00"
Bit 0 is the LOOP bit. It indicates the last instruction to be executed in any instruction sequence when it is set to a "1". The next instruction to be executed will be instruction 0.
Bit 1 is the PAUSE bit. This controls the Sequencer's operation. When the PAUSE bit is set ("1"), the Sequencer will stop after reading the current instruction and before executing it, and the start bit in the Configuration register is automatically reset to a "0". Setting the PAUSE also causes an interrupt to be issued. The Sequencer is restarted by placing a "1" in the Configuration register's Bit 0 (Start bit).
After the instruction RAM has been programmed and the RESET bit is set to "1", the Sequencer retrieves instruction 000, decodes it, and waits for a "1" to be placed in the Configuration's START bit. The START bit value of "0" overrides the action of instruction 000's PAUSE bit when the Sequencer is started. Once started, the Sequencer executes instruction 000 and retrieves, decodes, and executes each of the remaining instructions. No PAUSE interrupt (INT 5) is generated the first time the Sequencer executes instruction 000 having a PAUSE bit set to "1". When the Sequencer encounters a LOOP bit or completes all eight instructions, instruction 000 is retrieved and decoded. A set PAUSE bit in instruction 000 now halts the Sequencer before the instruction is executed.

Bits 2-4 select which of the eight input channels ("000" to "111" for IN0-IN7) will be configured as non-inverting inputs to the LM12(H)458's ADC. (See Page 27, Table 1.) They select which of the four input channels ("000" to "011" for IN0-IN4) will be configured as non-inverting inputs to the LM12454's ADC. (See Page 27, Table 2).
Bits 5-7 select which of the seven input channels ("001" to "111" for IN1 to IN7) will be configured as inverting inputs to the LM12(H)458's ADC. (See Page 27, Table 1.) They select which of the three input channels ("001" to "011" for IN1-IN4) will be configured as inverting inputs to the LM12454's ADC. (See Page 27, Table 2.) Fully differential operation is created by selecting two multiplexer channels, one operating in the



2.0 Internal User-Programmable Registers (Continued)

mand ("1") disconnects the external clock from the internal circuitry, decreases the LM12(H)454/8's internal analog circuitry power supply current, and preserves all internal RAM contents. After writing a "0" to the Standby bit, the LM12(H)454/8 returns to an operating state identical to that caused by exercising the RESET bit. A Standby completion interrupt is issued after a power-up completion delay that allows the analog circuitry to settle. The Sequencer should be restarted only after the Standby completion is issued. The Instruction RAM can still be accessed through read and write operations while the LM12(H)454/8 are in Standby Mode.

Bit 5 is the Channel Address Mask. If Bit 5 is set to a "1", Bits 13–15 in the conversion FIFO will be equal to the sign bit (Bit 12) of the conversion data. Resetting Bit 5 to a "0" causes conversion data Bits 13 through 15 to hold the instruction pointer value of the instruction to which the conversion data belongs.

Bit 6 is used to select a "short" auto-zero correction for every conversion. The Sequencer automatically inserts an auto-zero before every conversion or "watchdog" comparison if Bit 6 is set to "1". No automatic correction will be performed if Bit 6 is reset to "0".

The LM12(H)454/8's offset voltage, after calibration, has a typical drift of 0.1 LSB over a temperature range of -40°C to +85°C. This small drift is less than the variability of the change in offset that can occur when using the auto-zero correction with each conversion. This variability is the result of using only one sample of the offset voltage to create a correction value. This variability decreases when using the full calibration mode because eight samples of the offset voltage are taken, averaged, and used to create a correction value.

Bit 7 is used to program the SYNC pin (29) to operate as either an input or an output. The SYNC pin becomes an output when Bit 7 is a "1" and an input when Bit 7 is a "0". With SYNC programmed as an input, the rising edge of any logic signal applied to pin 29 will start a conversion or "watchdog" comparison. Programmed as an output, the logic level at pin 29 will go high at the start of a conversion or "watchdog" comparison and remain high until either have finished. See Instruction RAM "00", Bit 8.

Bits 8 and 9 form the RAM Pointer that is used to select each of a 48-bit instruction's three 16-bit sections during read or write actions. A "00" selects Instruction RAM section one, "01" selects section two, and "10" selects section three.

Bit 10 activates the Test mode that is used only during production testing. Leave this bit reset to "0".

Bit 11 is the Diagnostic bit and is available only in the LM12(H)458. It can be activated by setting it to a "1" (the Test bit must be reset to a "0"). The Diagnostic mode, along with a correctly chosen instruction, allows verification that the LM12(H)458's ADC is performing correctly. When activated, the inverting and non-inverting inputs are connected as shown in Table 1. As an example, an instruction with "001" for both V_{IN+} and V_{IN-} , while using the Diagnostic mode typically results in a full-scale output.

2.3 INTERRUPTS

The LM12454 and LM12(H)458 have eight possible interrupts, all with the same priority. Any of these interrupts will cause a hardware interrupt to appear on the INT pin (31) if

they are not masked (by the Interrupt Enable register). The Interrupt Status register is then read to determine which of the eight interrupts has been issued.

TABLE 1. LM12(H)458 Input Multiplexer Channel Configuration Showing Normal Mode and Diagnostic Mode

Channel Selection Data	Normal Mode		Diagnostic Mode	
	V_{IN+}	V_{IN-}	V_{IN+}	V_{IN-}
000	IN0	GND	V_{REFOUT}	GND
001	IN1	IN1	V_{REF+}	V_{REF-}
010	IN2	IN2	IN2	IN2
011	IN3	IN3	IN3	IN3
100	IN4	IN4	IN4	IN4
101	IN5	IN5	IN5	IN5
110	IN6	IN6	IN6	IN6
111	IN7	IN7	IN7	IN7

TABLE 2. LM12454 Input Multiplexer Channel Configuration

Channel Selection Data	MUX+	MUX-
000	IN0	GND
001	IN1	IN1
010	IN2	IN2
011	IN3	IN3
1XX	OPEN	OPEN

The Interrupt Status register, 1010 (A4–A1, BW = 0) or 1010x (A4–A0, BW = 1) must be cleared by reading it after writing to the Interrupt Enable register. This removes any spurious interrupts on the INT pin generated during an Interrupt Enable register access.

Interrupt 0 is generated whenever the analog input voltage on a selected multiplexer channel crosses a limit while the LM12(H)454/8 are operating in the "watchdog" comparison mode. Two sequential comparisons are made when the LM12(H)454/8 are executing a "watchdog" instruction. Depending on the logic state of Bit 9 in the Instruction RAM's second and third sections, an interrupt will be generated either when the input signal's magnitude is greater than or less than the programmable limits. (See the Instruction RAM, Bit 9 description.) The Limit Status register will indicate which preprogrammed limit, #1 or #2, and which instruction was executing when the limit was crossed.

Interrupt 1 is generated when the Sequencer reaches the instruction counter value specified in the Interrupt Enable register's bits 8–10. This flag appears before the instruction's execution.

Interrupt 2 is activated when the Conversion FIFO holds a number of conversions equal to the programmable value stored in the Interrupt Enable register's Bits 11–15. This value ranges from 0001 to 1111, representing 1 to 31 conversions stored in the FIFO. A user-programmed value of 0000 has no meaning. See Section 3.0 for more FIFO information. The completion of the short, single-sampled auto-zero calibration generates **Interrupt 3**.

2.0 Internal User-Programmable Registers (Continued)

The completion of a full auto-zero and linearity self-calibration generates **Interrupt 4**. **Interrupt 5** is generated when the Sequencer encounters an instruction that has its Pause bit (Bit 1 in Instruction RAM "00") set to "1".

The LM12(H)454/8 issues **Interrupt 6** whenever it senses that its power supply voltage is dropping below 4V (typ). This interrupt indicates the potential corruption of data returned by the LM12(H)454/8.

Interrupt 7 is issued after a short delay (10 ms typ) while the LM12(H)454/8 returns from Standby mode to active operation using the Configuration register's Bit 4. This short delay allows the internal analog circuitry to settle sufficiently, ensuring accurate conversion results.

2.4 INTERRUPT ENABLE REGISTER

The Interrupt Enable register at address location 1001 (A4–A1, BW = 0) or 1001x (A4–A0, BW = 1) has READ/WRITE capability. An individual interrupt's ability to produce an external interrupt at pin 31 (INT) is accomplished by placing a "1" in the appropriate bit location. Any of the internal interrupt-producing operations will set their corresponding bits to "1" in the Interrupt Status register regardless of the state of the associated bit in the Interrupt Enable register. See Section 2.3 for more information about each of the eight internal interrupts.

Bit 0 enables an external interrupt when an internal "watchdog" comparison limit interrupt has taken place.

Bit 1 enables an external interrupt when the Sequencer has reached the address stored in Bits 8–10 of the Interrupt Enable register.

Bit 2 enables an external interrupt when the Conversion FIFO's limit, stored in Bits 11–15 of the Interrupt Enable register, has been reached.

Bit 3 enables an external interrupt when the single-sampled auto-zero calibration has been completed.

Bit 4 enables an external interrupt when a full auto-zero and linearity self-calibration has been completed.

Bit 5 enables an external interrupt when an internal Pause interrupt has been generated.

Bit 6 enables an external interrupt when a low power supply condition ($V_{A+} < 4V$) has generated an internal interrupt.

Bit 7 enables an external interrupt when the LM12(H)454/8 return from power-down to active mode.

Bits 8–10 form the storage location of the user-programmable value against which the Sequencer's address is compared. When the Sequencer reaches an address that is equal to the value stored in Bits 8–10, an internal interrupt is generated and appears in Bit 1 of the Interrupt Status register. If Bit 1 of the Interrupt Enable register is set to "1", an external interrupt will appear at pin 31 (INT).

The value stored in bits 8–10 ranges from 000 to 111, representing 0 to 7 instructions stored in the Instruction RAM. After the Instruction RAM has been programmed and the RESET bit is set to "1", the Sequencer is started by placing a "1" in the Configuration register's START bit. Setting the INT 1 trigger value to 000 **does not generate** an INT 1 the first time the Sequencer retrieves and decodes Instruction 000. The Sequencer generates INT 1 (by placing a "1" in the Interrupt Status register's Bit 1) the **second time** and after the Sequencer encounters Instruction 000. It is important to re-

member that the Sequencer continues to operate even if an instruction interrupt (INT 1) is internally or externally generated. The only mechanisms that stop the Sequencer are an instruction with the PAUSE bit set to "1" (halts before instruction execution), placing a "0" in the Configuration register's START bit, or placing a "1" in the Configuration register's RESET bit.

Bits 11–15 hold the number of conversions that must be stored in the Conversion FIFO in order to generate an internal interrupt. This internal interrupt appears in Bit 2 of the Interrupt Status register. If Bit 2 of the Interrupt Enable register is set to "1", an external interrupt will appear at pin 31 (INT).

2.5 INTERRUPT STATUS REGISTER

This read-only register is located at address 1010 (A4–A1, BW = 0) or 1010x (A4–A0, BW = 1). The corresponding flag in the Interrupt Status register goes high ("1") any time that an interrupt condition takes place, whether an interrupt is enabled or disabled in the Interrupt Enable register. Any of the active ("1") Interrupt Status register flags are reset to "0" whenever this register is read or a device reset is issued (see Bit 1 in the Configuration Register).

Bit 0 is set to "1" when a "watchdog" comparison limit interrupt has taken place.

Bit 1 is set to "1" when the Sequencer has reached the address stored in Bits 8–10 of the Interrupt Enable register.

Bit 2 is set to "1" when the Conversion FIFO's limit, stored in Bits 11–15 of the Interrupt Enable register, has been reached.

Bit 3 is set to "1" when the single-sampled auto-zero has been completed.

Bit 4 is set to "1" when an auto-zero and full linearity self-calibration has been completed.

Bit 5 is set to "1" when a Pause interrupt has been generated.

Bit 6 is set to "1" when a low-supply voltage condition ($V_{A+} < 4V$) has taken place.

Bit 7 is set to "1" when the LM12(H)454/8 return from power-down to active mode.

Bits 8–10 hold the Sequencer's actual instruction address while it is running.

Bits 11–15 hold the actual number of conversions stored in the Conversion FIFO while the Sequencer is running.

2.6 LIMIT STATUS REGISTER

The read-only register is located at address 1101 (A4–A1, BW = 0) or 1101x (A4–A0, BW = 1). This register is used in tandem with the Limit #1 and Limit #2 registers in the Instruction RAM. Whenever a given instruction's input voltage exceeds the limit set in its corresponding Limit register (#1 or #2), a bit, corresponding to the instruction number, is set in the Limit Status register. Any of the active ("1") Limit Status flags are reset to "0" whenever this register is read or a device reset is issued (see Bit 1 in the Configuration Register). This register holds the status of limits #1 and #2 for each of the eight instructions.

Bits 0–7 show the Limit #1 status. Each bit will be set high ("1") when the corresponding instruction's input voltage exceeds the threshold stored in the instruction's Limit #1 register. When, for example, instruction 3 is a "watchdog" operation (Bit 11 is set high) and the input for instruction 3 meets the magnitude and/or polarity data stored in instruction 3's Limit #1 register, Bit 3 in the Limit Status register will be set to a "1".



2.0 Internal User-Programmable Registers (Continued)

Bits 8–15 show the Limit #2 status. Each bit will be set high ("1") when the corresponding instruction's input voltage exceeds the threshold stored in the instruction's Limit #2 register. When, for example, the input to instruction 6 meets the value stored in instruction 6's Limit #2 register, Bit 14 in the Limit Status register will be set to a "1".

2.7 TIMER

The LM12(H)45/48 have an on-board 16-bit timer that includes a 5-bit pre-scaler. It uses the clock signal applied to pin 23 as its input. It can generate time intervals of 0 through 2²¹ clock cycles in steps of 2⁵. This time interval can be used to delay the execution of instructions. It can also be used to slow the conversion rate when converting slowly changing signals. This can reduce the amount of redundant data stored in the FIFO and retrieved by the controller.

The user-defined timing value used by the Timer is stored in the 16-bit READ/WRITE Timer register at location 1011 (A4–A1, BW = 0) or 1011x (A4–A0, BW = 1) and is pre-loaded automatically. Bits 0–7 hold the preset value's low byte and Bits 8–15 hold the high byte. The Timer is activated by the Sequencer only if the current instruction's Bit 9 is set ("1"), if the equivalent decimal value "N" ($0 \leq N \leq 2^{16} - 1$) is written inside the 16-bit Timer register and the Timer is enabled by setting an instruction's bit 9 to a "1", the Sequencer will delay the same instruction's execution by halting at state 3 (S3), as shown in Figure 15, for $32 \times N + 2$ clock cycles.

2.8 DMA

The DMA works in tandem with Interrupt 2. An active DMA Request on pin 32 (DMARQ) requires that the FIFO interrupt be enabled. The voltage on the DMARQ pin goes high when the number of conversions in the FIFO equals the 5-bit value stored in the Interrupt Enable register (bits 11–15). The voltage on the INT pin goes low at the same time as the voltage on the DMARQ pin goes high. The voltage on the DMARQ pin goes low when the FIFO is emptied. The Interrupt Status register must be read to clear the FIFO interrupt flag in order to enable the next DMA request.

DMA operation is optimized through the use of the 16-bit databus connection (a logic "0" applied to the BW pin). Using this bus width allows DMA controllers that have single address Read/Write capability to easily unload the FIFO. Using DMA on an 8-bit databus is more difficult. Two read operations (low byte, high byte) are needed to retrieve each conversion result from the FIFO. Therefore, the DMA controller must be able to repeatedly access two constant addresses when transferring data from the LM12(H)45/48 to the host system.

3.0 FIFO

The result of each conversion stored in an internal read-only FIFO (First-In, First-Out) register. It is located at 1100 (A4–A1, BW = 0) or 1100x (A4–A0, BW = 1). This register has 32 16-bit wide locations. Each location holds 13-bit data. Bits 0–3 hold the four LSBs in the 12 bits + sign mode or "1110" in the 8 bits + sign mode. Bits 4–11 hold the eight MSBs and Bit 12 holds the sign bit. Bits 13–15 can hold either the sign bit, extending the register's two's complement data format to a full sixteen bits or the instruction address

4.0 Sequencer

The Sequencer uses a 3-bit counter (Instruction Pointer, or IP, in Figure 9) to retrieve the programmable conversion instructions stored in the Instruction RAM. The 3-bit counter is reset to 000 during chip reset or if the current executed instruction has its Loop bit (Bit 1 in any Instruction RAM "00") set high ("1"). It increments at the end of the currently executed instruction and points to the next instruction. It will continue to increment up to 111 unless an instruction's Loop bit is set. If this bit is set, the counter resets to "000" and execution begins again with the first instruction. If all instructions have their Loop bit reset to "0", the Sequencer will execute all eight instructions continuously. Therefore, it is important to realize that if less than eight instructions are programmed, the Loop bit on the last instruction must be set. Leaving this bit reset to "0" allows the Sequencer to execute "unprogrammed" instructions, the results of which may be unpredictable.

The Sequencer's Instruction Pointer value is readable at any time and is found in the Status register at Bits 8–10. The Sequencer can go through eight states during instruction execution:

State 0: The current instruction's first 16 bits are read from the Instruction RAM "00". This state is one clock cycle long.
State 1: Checks the state of the Calibration and Start bits. This is the "rest" state whenever the Sequencer is stopped using the reset, a Pause command, or the Start bit is reset low ("0"). When the Start bit is set to a "1", this state is one clock cycle long.

State 2: Perform calibration. If bit 2 or bit 6 of the Configuration register is set to a "1", state 2 is 76 clock cycles long. If the Configuration register's bit 3 is set to a "1", state 2 is 4944 clock cycles long.

State 3: Run the internal 16-bit Timer. The number of clock cycles for this state varies according to the value stored in the Timer register. The number of clock cycles is found by using the expression below

$$\text{where } 0 \leq T \leq 2^{16} - 1, \quad 32T + 2$$

State 7: Run the acquisition delay and read Limit #1's value if needed. The number of clock cycles for 12-bit + sign mode varies according to

$$9 + 2D$$

where D is the user-programmable 4-bit value stored in bits 12–15 of Instruction RAM "00" and is limited to $0 \leq D \leq 15$. The number of clock cycles for 8-bit + sign or "watchdog" mode varies according to

$$2 + 2D$$

where D is the user-programmable 4-bit value stored in bits 12–15 of Instruction RAM "00" and is limited to $0 \leq D \leq 15$.

State 6: Perform first comparison. This state is 5 clock cycles long.

State 4: Read Limit #2. This state is 1 clock cycle long.

State 5: Perform a conversion or second comparison. This state takes 44 clock cycles when using the 12-bit + sign mode or 21 clock cycles when using the 8-bit + sign mode. The "watchdog" mode takes 5 clock cycles.

that generated the conversion and the resulting data. These modes are selected according to the logic state of the Configuration register's Bit 5.

The FIFO status should be read in the Interrupt Status register (Bits 11–15) to determine the number of conversion results that are held in the FIFO before retrieving them. This will help prevent conversion data corruption that may take place if the number of reads are greater than the number of conversion results contained in the FIFO. Trying to read the FIFO when it is empty may corrupt new data being written into the FIFO. Writing more than 32 conversion data into the FIFO by the ADC results in loss of the first conversion data. Therefore, to prevent data loss, it is recommended that the LM12(H)45/48's interrupt capability be used to inform the system controller that the FIFO is full.

The lower portion (A0 = 0) of the data word (Bits 0–7) should be read first followed by a read of the upper portion (A0 = 1) when using the 8-bit bus width (BW = 1). Reading the upper portion first causes the data to shift down, which results in loss of the lower byte.

Bits 0–12 hold 12-bit + sign conversion data. **Bits 0–3** will be 1110 (LSB) when using 8-bit plus sign resolution.

Bits 13–15 hold either the instruction responsible for the associated conversion data or the sign bit. Either mode is selected with Bit 5 in the Configuration register.

Using the FIFO's full depth is achieved as follows. Set the value of the Interrupt Enable register's Bits 11–15 to 1111 and the Interrupt Enable register's Bit 2 to a "1". This generates an external interrupt when the 31st conversion is stored in the FIFO. This gives the host processor a chance to send a "0" to the LM12(H)45/48's Start bit (Configuration register) and halt the ADC before it completes the 32nd conversion. The Sequencer halts after the current (32) conversion is completed. The conversion data is then transferred to the FIFO and occupies the 32nd location. FIFO overflow is avoided if the Sequencer is halted before the start of the 32nd conversion by placing a "0" in the Start bit (Configuration register). It is important to remember that the Sequencer continues to operate even if a FIFO interrupt (INT 2) is internally or externally generated. The only mechanisms that stop the Sequencer are an instruction with the PAUSE bit set to "1" (halts before instruction execution), placing a "0" in the Configuration register's START bit, or placing a "1" in the Configuration register's RESET bit.



5.0 Analog Considerations

5.1 REFERENCE VOLTAGE

The difference in the voltages applied to the V_{REF+} and V_{REF-} defines the analog input voltage span (the difference between the voltages applied between two multiplexer inputs or the voltage applied to one of the multiplexer inputs and analog ground), over which 4095 positive and 4096 negative codes exist. The voltage sources driving V_{REF+} or V_{REF-} must have very low output impedance and noise.

The ADC can be used in either ratiometric or absolute reference applications. In ratiometric systems, the analog input voltage is proportional to the voltage used for the ADC's reference voltage. When this voltage is the system power supply, the V_{REF+} pin is connected to V_{A+} and V_{REF-} is connected to GND. This technique relaxes the system reference stability requirements because the analog input voltage and the ADC reference voltage move together. This maintains the same output code for given input conditions.

For absolute accuracy, where the analog input voltage varies between very specific voltage limits, a time and temperature stable voltage source can be connected to the reference inputs. Typically, the reference voltage's magnitude will require an initial adjustment to null reference voltage induced full-scale errors.

When using the LM12(H)454/8's internal 2.5V bandgap reference, a parallel combination of a 100 μ F capacitor and a 0.1 μ F capacitor connected to the V_{REFOUT} pin is recommended for low noise operation. When left unconnected, the reference remains stable without a bypass capacitor. However, ensure that stray capacitance at the V_{REFOUT} pin remains below 50 pF.

5.2 INPUT RANGE

The LM12(H)454/8's fully differential ADC and reference voltage inputs generate a two's-complement output that is found by using the equation below.

$$\text{output code} = \frac{V_{IN+} - V_{IN-}}{V_{REF+} - V_{REF-}} (4096) - \frac{1}{2} \quad \text{(12-bit)}$$

$$\text{output code} = \frac{V_{IN+} - V_{IN-}}{V_{REF+} - V_{REF-}} (256) - \frac{1}{2} \quad \text{(8-bit)}$$

Round up to the next integer value between -4096 to 4095 for 12-bit resolution and between -256 to 255 for 8-bit resolution if the result of the above equation is not a whole number. As an example, $V_{REF+} = 2.5V$, $V_{REF-} = 1V$, $V_{IN+} = 1.5V$ and $V_{IN-} = GND$. The 12-bit + sign output code is positive full-scale, or 0.1111,1111,1111. If $V_{REF+} = 5V$, $V_{REF-} = 1V$, $V_{IN+} = 3V$, and $V_{IN-} = GND$, the 12-bit + sign output code is 0.11100,0000,0000.

5.3 INPUT CURRENT

A charging current flows into or out of (depending on the input voltage polarity) the analog input pins. IN0-IN7 at the start of the analog input acquisition time (t_{ACQ}). This current's peak value will depend on the actual input voltage applied.

5.4 INPUT SOURCE RESISTANCE

For low impedance voltage sources (<100 Ω for 5 MHz operation and <60 Ω for 8 MHz operation), the input charging current will decay before the end of the SH's acquisition time, to a value that will not introduce any conversion errors. For higher source impedances, the SH's acquisition time

can be increased. As an example, operating with a 5 MHz clock frequency and maximum acquisition time, the LM12(H)454/8's analog inputs can handle source impedance as high as 6.67 k Ω . When operating at 8 MHz and maximum acquisition time, the LM12(H)454/8's analog inputs can handle source impedance as high as 4.17 k Ω . Refer to Section 2.1, Instruction RAM "00". Bits 12-15 for further information.

5.5 INPUT BYPASS CAPACITANCE

External capacitors (0.01 μ F-0.1 μ F) can be connected between the analog input pins, IN0-IN7, and analog ground to filter any noise caused by inductive pickup associated with long input leads. It will not degrade the conversion accuracy.

5.6 NOISE

The leads to each of the analog multiplexer input pins should be kept as short as possible. This will minimize input noise and clock frequency coupling that can cause conversion errors. Input filtering can be used to reduce the effects of the noise sources.

5.7 POWER SUPPLIES

Noise spikes on the V_{A+} and V_{D+} supply lines can cause conversion errors; the comparator will respond to the noise. The ADC is especially sensitive to any power supply spikes that occur during the auto-zero or linearity correction. Low inductance tantalum capacitors of 10 μ F or greater paralleled with 0.1 μ F monolithic ceramic capacitors are recommended for supply bypassing. Separate bypass capacitors should be used for the V_{A+} and V_{D+} supplies and placed as close as possible to these pins.

5.8 GROUNDING

The LM12(H)454/8's nominal high resolution performance can be maximized through proper grounding techniques. These include the use of separate analog and digital ground planes. The digital ground plane is placed under all components that handle digital signals, while the analog ground plane is placed under all analog signal handling circuitry. The digital and analog ground planes are connected at only one point, the power supply ground. This greatly reduces the occurrence of ground loops and noise.

It is recommended that stray capacitance between the analog inputs or outputs (LM12(H)454: IN0-IN3, MUXOUT+, MUXOUT-, SH IN+, SH IN-, LM12(H)458: IN0-IN7, V_{REF+} , and V_{REF-}) be reduced by increasing the clearance (+1/16th inch) between the analog signal and reference pins and the ground plane.

5.9 CLOCK SIGNAL LINE ISOLATION

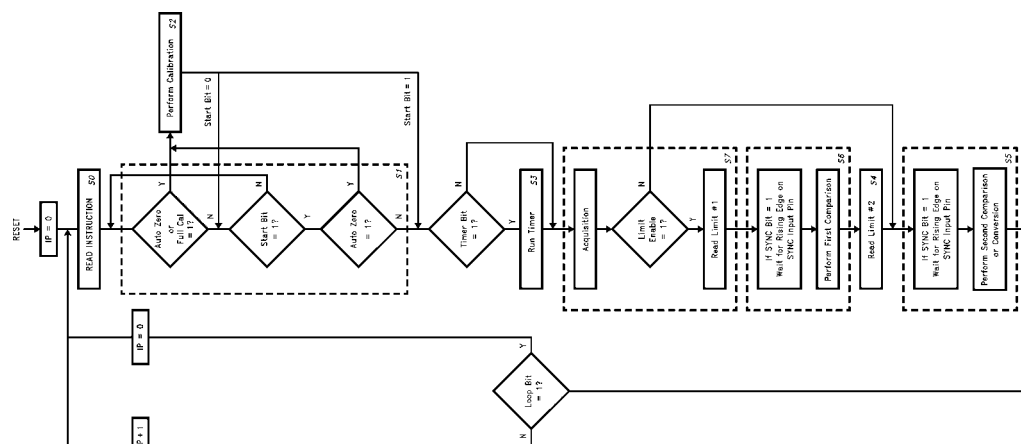
The LM12(H)454/8's performance is optimized by routing the analog input/output and reference signal conductors (pins 34-44) as far as possible from the conductor that carries the clock signal to pin 23. Ground traces parallel to the clock signal trace can be used on printed circuit boards to reduce clock signal interference on the analog input/output pins.

6.0 Application Circuits

PC EVALUATION/INTERFACE BOARD

Figure 16 is the schematic of an evaluation/interface board designed to interface the LM12(H)454 or LM12(H)458 with an XT or AT® style computer. The board can be used to de-

4.0 Sequencer (Continued)



DS91284-19

FIGURE 15. Sequencer Logic Flow Chart (IP = Instruction Pointer)



6.0 Application Circuits (Continued)

velop both software and hardware. The board hardwires the BW (Bus Width) pin to a logic high, selecting an 8-bit wide databus. Therefore, it is designed for an 8-bit expansion slot on the computer's motherboard.

The circuit operates on a single +5V supply derived from the computer's +12V supply using an LM340 regulator. This greatly attenuates noise that may be present on the computer's power supply lines. However, your application may only need an LC filter.

Figure 16 also shows the recommended supply (V_{A+} and V_{D+}) and reference input (V_{REF+} and V_{REF-}) bypassing. The digital and analog supply pins can be connected together to the same supply voltage. However, they need separate, multiple bypass capacitors. Multiple capacitors on the supply pins and the reference inputs ensures a low impedance bypass path over a wide frequency range.

All digital interface control signals (IOR, IOW, and AEN), data lines (DB0-DB7), address lines (A0-A9), and IRQ (interrupt request) lines (IRQ2, IRQ3, and IRQ5) connections are made through the motherboard slot connector. All analog signals applied to, or received by, the input multiplexer (IN0-IN7 for the LM12(H)488 and IN0-IN3, MUXOUT+, MUXOUT-, SH IN+, and SH IN- for the LM12(H)454), V_{REF+} , V_{REF-} , $V_{REFOUT+}$, and the SYNC signal input/output are applied through a DB-37 connector on the rear side of the board. Figure 16 shows that there are numerous analog ground connections available on the DB-37 connector.

The voltage applied to V_{REF-} and V_{REF+} is selected using two jumpers, JP1 and JP2. JP1 selects between the voltage applied to the DB-37's pin 24 or GND and applies it to the LM12(H)454's V_{REF-} input. JP2 selects between the LM12(H)454's internal reference output, $V_{REFOUT+}$, and the voltage applied to the DB-37's pin 22 and applies it to the LM12(H)454's V_{REF+} input.

TABLE 3. LM12(H)454/8 Evaluation/Interface Board SW DIP-8 Switch Settings for Available I/O Memory Locations

Hexidecimal I/O Memory Base Address	SW DIP-8			
	SW1 (SEL0)	SW2 (SEL1)	SW3 (SEL2)	SW4 (SEL3)
100	ON	ON	ON	ON
120	OFF	ON	ON	ON
140	ON	OFF	ON	ON
160	OFF	OFF	ON	ON
180	ON	ON	OFF	ON
1A0	OFF	ON	OFF	ON
1C0	ON	OFF	OFF	ON
300	OFF	OFF	OFF	ON
340	ON	ON	ON	OFF
280	OFF	ON	ON	OFF
2A0	ON	OFF	ON	OFF

The board allows the use of one of three Interrupt Request (IRQ) lines IRQ2, IRQ3, and IRQ5. The individual IRQ line can be selected using switches 5, 6, and 7 of SW DIP-8. When using any of these three IRQs, the user needs to ensure that there are no conflicts between the evaluation board and any other boards attached to the computer's motherboard.

Switches 1-4, along with address lines A5-A9 are used as inputs to GAL16V8 Programmable Gate Array (LJ2). This device forms the interface between the computer's control and address lines and generates the control signals used by the LM12(H)454/8 for CS, WR, and RD. It also generates the signal that controls the data buffers. Several address ranges within the computer's I/O memory map are available. Refer to Table III for the switch settings that gives the desired I/O memory address range. Selection of an address range must be done so that there are no conflicts between the evaluation board and any other boards attached to the computer's motherboard. The GAL equations are shown in Figure 18. The GAL functional block diagram is shown in Figure 19. Figures 20, 21, 22, 23 show the layout of each layer in the 3-layer evaluation/Interface board plus the silk-screen layout showing parts placement. Figure 21 is the top or component side, Figure 22 is the middle or ground plane layer, Figure 23 is the circuit side, and Figure 20 is the parts layout.



APPENDIX C: ALPHABETICAL INDEX

A

A/D CONVERTER 2, 4, 8, 14, 26, 49
A/D FREQUENCY 8
A/D VOLTAGE REFERENCE 26
ABACO® 2, 4, 8, 24, 63
ABACO® BUS ADDRESSES 45
ACTIVITY LEDS 51
ADDRESSES AND MAPS 42
ANALOG INPUTS 28
ANALOG INPUTS IMPEDANCE 9

B

BACK UP 2, 10, 34
BACK UP BATTERY 9
BACK UP CURRENT 9
BATTERY STATUS 50

C

CARD VERSION 1
CLOCK 4
CONFIGURATION INPUTS 35, 50
CONNECTIONS 10
CONNECTORS 8
 CN1 16
 CN2 12
 CN3 14
 CN4 18
 CN5 11
 CN6 10
 K1 24
CONSUMPTION ON 9
CPU 2, 3, 8
CPU CLOCK FREQUENCY 8
CPU INTERNAL DEVICES 62
CTC 8, 16
CURRENT LOOP 2, 3, 8, 18, 28, 36

D

DIGITAL I/O INTERFACES 29
DIGITAL I/O LINES 6
DIP SWITCH 2, 8, 35, 50
DISABLE DMA REQUEST 51
DMA 3, 4, 51
DMA TRANSFERS 38

E

ELECTRIC FEATURES 9
EPROM 2, 6, 8, 45

F

FLASH EPROM 2, 6, 8, 45

G

GDOS 40
GENERAL FEATURES 8
GENERAL INFORMATIONS 2
GET 40

I

I/O ADDRESSES 43
I/O CONNECTION 28
INSTALLATION 10
INTERRUPTS 35
INTRODUCTION 1

J

JUMPERS 30, 50
2 PINS JUMPERS 32
3 PINS JUMPERS 33
4 PINS JUMPER 32
5 PINS JUMPER 33

L

LED 2, 28, 51
LITHIUM BATTERY 34
LM 12H458 49

M

MEMORY 6
MEMORY ADDRESSES 45
MEMORY MANAGEMENT UNIT 48
MEMORY SELECTION 39
MMU 6, 42, 48

O

ON BOARD RESOURCES ALLOCATION 42

P

PERIPHERAL DEVICES SOFTWARE DESCRIPTION 48
PHYSICAL FEATURES 8
PIC 3
POWER FAILURE 34
POWER FAILURE 2
POWER FAILURE THRESHOLD 9
POWER SUPPLY 9
PPI 2, 6, 8, 11, 12, 61
PPI 82C55 61
PWM 2, 3

R

REAL TIME CLOCK 2, 3, 8, 52
RESET AND WATCH DOG 34
RS 232 2, 3, 8, 16, 28, 36
RS 422 2, 3, 8, 18, 28, 36
RS 485 2, 3, 8, 18, 28, 36
RS422, 485 TERMINATION NETWORK 9
RTC 34, 52
RUN/DEBUG 50
RV1 26

S

SCC 85C30 53
SCC CLOCK FREQUENCY 8
SCL 49
SDA 49
SDLC/HDLC 53
SERIAL COMMUNICATION 3
SERIAL COMMUNICATION SELECTION 36
SERIAL EEPROM 2, 6, 8, 39, 45, 49
SERIAL LINES 53
SIZE 8
SOFTWARE DESCRIPTION 40
SRAM 2, 6, 8, 34, 39, 45

T

TECHNICAL FEATURES 8
TEST POINT 26
TIMER COUNTER 2, 16
TP1 26
TRIMMER AND CALIBRATION 26

V

VISUAL FEEDBACK 28

VREF 26

W

WATCH DOG 2, 6, 8, 34, 49

WEIGHT 8

WRITE PROTECTION 51

