GPC® 324
General Purpose Controller 80C32,320 etc. 4 type

TECHNICAL MANUAL
GPC® 324
General Purpose Controller 80C32,320 etc. 4 type

TECHNICAL MANUAL

Intelligent module of the ABACO® BLOCK series, 100x50 mm size. Optional plastic mount for connection to DIN 46277-1 and DIN 46277-3 Ω rails. CPU 80c32 or 80c320 at 22 MHz and all the compatibles models. Moreover the In System Programming of the PHILIPS 89CRx+/2 is completely supported. Maximum memory configuration of 104K Bytes: 32K SRAM, socket for 32K EPROM, socket for 32K EPROM, SRAM, EEPROM or FLASH EPROM and serial EEPROM, from 256 to 8192 Bytes. Complete Real Time Clock capable to generate INTERRUPT. Back up circuit for SRAM and RTC, through on board and external LITHIUM battery. Watch dog software and hardware settable. 5 TTL I/O lines, three 16 Bits timer counters; jumper for RUN/DEBUG mode. 2 RS232 serial lines, one configurable in RS422, RS485 or current loop. Standard 26 pins expansion connector for ABACO® I/O BUS interface. Numerous interrupt sources, including an efficient Power Failure circuitry. Single 5Vdc, 115 mA power supply, with different power saving modes. On board logic protected against transients by TransZorb™. Wide range of development software such as: Monitor Debugger (MDP, FMO52, NOICE); Assembler (MCA51); GET51; C compilers (MCC51, HTC51, SYS51CW, DDS Micro C51); BASIC (BASIC 324, BXC51, BASCOM 8051); PASCAL compiler (SYS51PW); etc.
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 FEATURES ........................................................................................................... 2
  CPU .................................................................................................................................... 3
  ABACO® I/O BUS ............................................................................................................... 4
  SERIAL COMMUNICATION ............................................................................................... 4
  CONTROL LOGIC ............................................................................................................. 4
  RESET CONTACT ........................................................................................................... 4
  PERIPHERAL DEVICES .................................................................................................... 6
  MEMORY DEVICES .......................................................................................................... 7
  CLOCK ............................................................................................................................. 7
  POWER SUPPLY ............................................................................................................ 7

TECHNICAL FEATURES ...................................................................................................... 8
  GENERAL FEATURES .................................................................................................... 8
  PHYSICAL FEATURES .................................................................................................... 8
  ELECTRIC FEATURES ...................................................................................................... 9

INSTALLATION .................................................................................................................. 10
  CONNECTIONS ............................................................................................................. 10
    CN2 - EXTERNAL BACK UP BATTERY CONNECTOR .................................................... 10
    CN1 - ABACO® I/O BUS CONNECTOR ........................................................................ 11
    CN3A - SERIAL LINE A CONNECTOR ......................................................................... 12
    CN3B - SERIAL LINE B CONNECTOR ......................................................................... 14
    CN5 - CPU I/O LINES AND PROGRAMMING VOLTAGE CONNECTOR .................... 20
  I/O CONNECTION .......................................................................................................... 22
  DIGITAL I/O INTERFACES ............................................................................................... 22
  JUMPERS ......................................................................................................................... 24
  2 PINS JUMPERS .......................................................................................................... 26
  3 PINS JUMPERS .......................................................................................................... 27
  5 PINS JUMPERS .......................................................................................................... 27
  MEMORY SELECTION ..................................................................................................... 28
  SOLDER JUMPERS .......................................................................................................... 28
  INTERRUPTS MANAGEMENT .......................................................................................... 29
  RESET AND WATCH DOG ............................................................................................. 29
  SERIAL COMMUNICATION SELECTION ...................................................................... 30
  BACK UP ......................................................................................................................... 33
  POWER FAILURE ........................................................................................................... 33
  IN SYSTEM PROGRAMMING ......................................................................................... 33

SOFTWARE ....................................................................................................................... 34
## FIGURE INDEX

<table>
<thead>
<tr>
<th>FIGURE</th>
<th>DESCRIPTION</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>MICROPROCESSOR FEATURES</td>
<td>3</td>
</tr>
<tr>
<td>2</td>
<td>BLOCK DIAGRAM</td>
<td>5</td>
</tr>
<tr>
<td>3</td>
<td>CARD PHOTO</td>
<td>9</td>
</tr>
<tr>
<td>4</td>
<td>CN2 - EXTERNAL BACK UP BATTERY CONNECTOR</td>
<td>10</td>
</tr>
<tr>
<td>5</td>
<td>CN1 - ABACO® I/O BUS CONNECTOR</td>
<td>11</td>
</tr>
<tr>
<td>6</td>
<td>CN3A - SERIAL LINE A CONNECTOR</td>
<td>12</td>
</tr>
<tr>
<td>7</td>
<td>P1, MEMORIES, BATTERY, CONNECTORS LOCATION</td>
<td>13</td>
</tr>
<tr>
<td>8</td>
<td>CN3B - SERIAL LINE B CONNECTOR</td>
<td>14</td>
</tr>
<tr>
<td>9</td>
<td>SERIAL COMMUNICATION DIAGRAM</td>
<td>15</td>
</tr>
<tr>
<td>10</td>
<td>RS 232 POINT TO POINT CONNECTION EXAMPLE</td>
<td>16</td>
</tr>
<tr>
<td>11</td>
<td>RS 422 POINT TO POINT CONNECTION EXAMPLE</td>
<td>16</td>
</tr>
<tr>
<td>12</td>
<td>RS 485 POINT TO POINT CONNECTION EXAMPLE</td>
<td>16</td>
</tr>
<tr>
<td>13</td>
<td>RS 485 NETWORK CONNECTION EXAMPLE</td>
<td>17</td>
</tr>
<tr>
<td>14</td>
<td>4 WIRES CURRENT LOOP POINT TO POINT CONNECTION EXAMPLE</td>
<td>18</td>
</tr>
<tr>
<td>15</td>
<td>2 WIRES CURRENT LOOP POINT TO POINT CONNECTION EXAMPLE</td>
<td>18</td>
</tr>
<tr>
<td>16</td>
<td>CURRENT LOOP NETWORK CONNECTION EXAMPLE</td>
<td>19</td>
</tr>
<tr>
<td>17</td>
<td>CN5 - CPU I/O LINES AND PROGRAMMING VOLTAGE CONNECTOR</td>
<td>20</td>
</tr>
<tr>
<td>18</td>
<td>I/O LINES CONNECTION DIAGRAM</td>
<td>21</td>
</tr>
<tr>
<td>19</td>
<td>COMPONENTS MAPS (COMPONENTS SIDE)</td>
<td>23</td>
</tr>
<tr>
<td>20</td>
<td>COMPONENTS MAPS (SOLDERING SIDE)</td>
<td>23</td>
</tr>
<tr>
<td>21</td>
<td>JUMPERS SUMMARIZING TABLE</td>
<td>24</td>
</tr>
<tr>
<td>22</td>
<td>JUMPERS LOCATION (COMPONENT SIDE)</td>
<td>25</td>
</tr>
<tr>
<td>23</td>
<td>JUMPERS LOCATION (SOLDERING SIDE)</td>
<td>25</td>
</tr>
<tr>
<td>24</td>
<td>2 PINS JUMPERS TABLE</td>
<td>26</td>
</tr>
<tr>
<td>25</td>
<td>3 PINS JUMPERS TABLE</td>
<td>27</td>
</tr>
<tr>
<td>26</td>
<td>5 PINS JUMPERS TABLE</td>
<td>27</td>
</tr>
<tr>
<td>27</td>
<td>MEMORY SELECTION TABLE</td>
<td>28</td>
</tr>
<tr>
<td>28</td>
<td>SERIAL COMMUNICATION DRIVERS LOCATION</td>
<td>31</td>
</tr>
<tr>
<td>29</td>
<td>MODE 0 MEMORY CONFIGURATION</td>
<td>37</td>
</tr>
<tr>
<td>30</td>
<td>MODE 1 MEMORY CONFIGURATION</td>
<td>38</td>
</tr>
<tr>
<td>31</td>
<td>MODE 3 MEMORY CONFIGURATION</td>
<td>39</td>
</tr>
<tr>
<td>32</td>
<td>I/O ADDRESSES TABLE</td>
<td>40</td>
</tr>
<tr>
<td>33</td>
<td>GPC® 324 AVAILABLE CONNECTIONS DIAGRAM</td>
<td>45</td>
</tr>
<tr>
<td>A1</td>
<td>MODULE DIMENSION FOR PIGGY BACK MOUNTING</td>
<td>A-1</td>
</tr>
<tr>
<td>A2</td>
<td>PIGGY BACK MOUNTING</td>
<td>A-2</td>
</tr>
<tr>
<td>A3</td>
<td>WEIDMULLER RAIL MOUNTING</td>
<td>A-2</td>
</tr>
<tr>
<td>C1</td>
<td>PPI EXPANSION ELECTRIC DIAGRAM</td>
<td>C-1</td>
</tr>
<tr>
<td>C2</td>
<td>SPA 03 ELECTRIC DIAGRAM</td>
<td>C-2</td>
</tr>
<tr>
<td>C3</td>
<td>QTP 16P ELECTRIC DIAGRAM</td>
<td>C-3</td>
</tr>
<tr>
<td>C4</td>
<td>QTP 24P ELECTRIC DIAGRAM (1 OF 2)</td>
<td>C-4</td>
</tr>
<tr>
<td>C5</td>
<td>QTP 24P ELECTRIC DIAGRAM (2 OF 2)</td>
<td>C-5</td>
</tr>
<tr>
<td>C6</td>
<td>ABACO® I/O BUS INPUT OUTPUT ELECTRIC DIAGRAM</td>
<td>C-6</td>
</tr>
<tr>
<td>C7</td>
<td>BUS INTERFACE ELECTRIC DIAGRAM</td>
<td>C-7</td>
</tr>
<tr>
<td>C8</td>
<td>IAC 01 ELECTRIC DIAGRAM</td>
<td>C-8</td>
</tr>
</tbody>
</table>
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® 324 card release 110400 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 the CN5 connector and IC14 socket on the component side).
The GPC® 324 belongs to the CPUs 4 Serie 100x50 mm. size. It is a powerful control low cost module capable of operating in stand alone mode or as an intelligent peripheral, and/or remoted, in a wider telecontrol or aquisition network.

The GPC® 324 module can be secured in a plastic mount for connection to Omega rails DIN 46277-1 and DIN 46277-3, thereby dispensing with the need of a rack and allowing a less costly mounting direct to the electrical control panel. Thanks to this small size, the GPC® 324 put into the same plastic rails that contains the peripheral I/O card, i.e ZBR xxx, ZBT xxx, forming in this way an unique BLOCK element. The GPC® 324 can also be mounted as a macro CPU module on a card developed by the end user, in Piggy Back (stack through) mode; cards as SPA 03 and SPA 04 can be used to realize a quick prototype in this modality. Finally the presence of the ABACO® I/O BUS allows the connection to cards as CAN 14, ADC 812, DAC 212, etc, and through ABB 03 and ABB 05 mother boards even to ABACO® BUS peripheral cards.

The most interesting characteristic of GPC® 324 module is that it can be equipped with a wide range of µP. It is in fact possible to get it by the standard 80C32; by the fast DALLAS 80C320, by the new PHILIPS 89CRx+/2 and many of the compatibles models. The characteristics of the GPC® 324 module remain basically the same, but its performance changes according to the built in µP.

At present there are some developing software tools which allow the card to be used as developing system of itself both in asselmler and high level languages. Noteworthy among these are the numerous C compilers, PASCAL, the powerfull BASIC compilers and the handy BASIC 324. The latter is compatible with the widespread MCS® BASIC 52 of INTEL to which have been added new commands. Among these new ones is duitful to mention some of those referred to the serial EEPROM and 2nd serial line. By adding an external 82c55 a Fluorescent or LCD displays and a matrix keyboard are managed; for an immediate use of this new commands, KDx x24 boards are available, or if you need a finished object, there is the operator panel QTP xxP. This operator panel, offered in the open frame version, bears the same aesthetic as QTP xx, but, as the local intelligence is not furnished, it is driven directly by GPC® 324, allowing a notable cost reduction.

The BASIC 324 and BASCOM 8051 + FMO52 affords truly notable debug facilities, and allows to program directly the on board EEPROM or FLASH EPROM with the user program.

- Intelligent module of the ABACO® BLOCK series, 100x50 mm size.
- Optional plastic mount for connection to DIN 46277-1 and DIN 46277-3 Ω rails.
- CPU 80c32 or 80c320,at 22 MHz and all the compatibles models. Moreover the In System Programming of the PHILIPS 89CRx+/2 is completely supported.
- Maximum memory configuration of 104K Bytes: 32K SRAM, socket for 32K EPROM, socket for 32K EPROM, SRAM, EEPROM or FLASH EPROM and serial EEPROM, from 256 to 8192 Bytes.
- Complete Real Time Clock capable to generate INTerrupt.
- Back up circuit for SRAM and RTC, through on board and external LITHIUM battery.
- Watch dog software and hardware settable.
- 5 TTL I/O lines, three 16 Bits timer counters; jumper for RUN/DEBUG mode.
- 2 RS232 serial lines, one configurable in RS422, RS485 or current loop.
- Standard 26 pins expansion connector for ABACO® I/O BUS interface.
- Numerous interrupt sources, including an efficient Power Failure circuitry.
- Single 5Vdc, 115 mA power supply, with different power saving modes.
- On board logic protected against transients by TransZorb™
- Wide range of development software such as: Monitor Debugger (MDP, FMO52, NOICE); Assembler (MCA51); GET 51; C compilers (MCC51, HTC51, SYS51CW,
Here follows a description of the board's functional blocks, with an indication of the operations performed by each one. To easily locate these blocks and verify their connections please refer to figure 2.

**CPU**

The **GPC® 324** can use many of ’51 microprocessors family as 80C32, 80C52, 87C52, 89C52 (from INTEL and other secound sources), 89S8252 (from ATMEL), 89CRx+/2 (from PHILIPS) 80C320, 87C320 (manufactured by DALLAS) and all the interchangeable ones. These 8 bit microprocessors are code compatible with the world wide used 8051 INTEL and they have an extended instruction set, fast execution time, easy use of all kind of memory and an efficient interrupt management. The most important features of the described microprocessor, are:

<table>
<thead>
<tr>
<th>Microprocessor</th>
<th>80C32</th>
<th>89S8252</th>
<th>89CRx+/2</th>
<th>80C320</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data BUS width</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>8</td>
</tr>
<tr>
<td>Clock system cycle</td>
<td>12</td>
<td>12</td>
<td>6</td>
<td>4</td>
</tr>
<tr>
<td>Internal RAM (bytes)</td>
<td>256</td>
<td>256</td>
<td>256</td>
<td>256</td>
</tr>
<tr>
<td>Internal ROM (kbytes)</td>
<td>8</td>
<td>8</td>
<td>64</td>
<td>8</td>
</tr>
<tr>
<td>Internal EEPROM (kbytes)</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>External code area (kbytes)</td>
<td>64</td>
<td>64</td>
<td>64</td>
<td>64</td>
</tr>
<tr>
<td>External data area (kbytes)</td>
<td>64</td>
<td>64</td>
<td>64</td>
<td>64</td>
</tr>
<tr>
<td>I/O ports</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>16 bits Timer/Counters</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>Interrupt sources</td>
<td>6</td>
<td>9</td>
<td>7</td>
<td>13</td>
</tr>
<tr>
<td>Interrupt priority level</td>
<td>2</td>
<td>2</td>
<td>4</td>
<td>3</td>
</tr>
<tr>
<td>A/Syncronous serial line</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>Idle mode or Power down mode</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Power monitor and control section</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Internal watch dog timer</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>In system programming</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>In application programming</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
</tr>
</tbody>
</table>

**Figure 1: Microprocessor features**

For further information, please refer to specific documentation of the manufacturing company or to appendix B of this manual. Please remember that the previous table describes the general microprocessor features and some of them can be not supported by the card.

The user must specify the requested microprocessor in the order phase and in absence of any indication the card is supplied in its default condition with 80C32. The version with DALLAS 80C320 is instead denoted by the D suffix = **GPC® 324D**.
ABACO® I/O BUS

One of the most important features of GPC® 324 is its possibility to be interfaced to many others industrial cards. Thanks to its standard ABACO® I/O 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 and even custom boards designed to satisfy specific needs of the end user. Using ABB 03 or ABB 05 mother boards it is possible manage all the BUS ABACO® single EURO cards. So GPC® 324 becomes the right component for each industrial automation system, in fact ABACO® I/O BUS makes the card easily expandable with the best price/performance ratio.

SERIAL COMMUNICATION

An hardware serial line is always available on GPC® 324 (named serial A) and a second serial line (named serial B) is managed as below described:

µP 80C32: software serial line driven through two microprocessor I/O lines;
µP 80C320: hardware serial line driven by proper hardware section;

The hardware serial communication lines are completely software configurable for physical protocol and by simply programming some microprocessor registers, the user can set the baud rate, stop bits number, length of character and parity. For software serial line the physical protocol is completely defined by the software procedures. Some software tools (BASIC 324, BASCOM 8051, etc.) directly manages the software lines through high level instructions. The serial line B is always RS 232 buffered, while the serial line A can be configured in RS 232, RS 422, RS 485 or passive current loop thanks to comfortable on board jumpers. For further information about serial communication please refer to chapter "SERIAL COMMUNICATION SELECTION" and to the technical documentation of the microprocessor.

CONTROL LOGIC

The addresses of all peripheral device's registers and of memory devices on GPC® 324 are assigned from a specific control logic that allocates all these devices in the microprocessor addressing space. For further information please refer to paragraph "I/O ADDRESSES" and "MEMORY ADDRESSES" of this manual.

RESET CONTACT

P1 reset contact of the GPC® 324 allows the user to reset the board and restarting it in a general clearing condition. The main purpose of this contact is to come out of infinite loop conditions, useful especially during debug or to grant a particular initial state. Please see figure 7 for an easy localization of this contact.
FIGURE 2: BLOCK DIAGRAM

CPU
51 family

IC 9
SERIAL EEPROM

IC 8
EPROM

IC 5
EPROM, SRAM, FLASH, EEPROM

WATCH DOG

POWER FAILURE

JUMPER RUN/DEBUG

CONTROL LOGIC

CN1
ABACO® I/O BUS

CN3A
SERIAL LINE A
RS 232
RS 422
RS 485
CURRENT LOOP

CN3B
SERIAL LINE B
RS 232

CN2
BACK UP

RTC

IC7
SRAM

3V LITHIUM BATTERY

+ -

3V

LITHIUM BATTERY

CN5
I/O lines, Vpp

CN2
BACK UP
PERIPHERAL DEVICES

GPC® 324 is the right card to solve many control problems in automation fields, in fact it is equipped with some peripheral components that facilitate the connection and the management to external system. These peripherals are:

- **external watch dog** it is an astable circuit that can reset the card at programmable time intervals of about 1.4 sec, if not retriggered. By software the user can retrigger the circuit to prevent card reset, using specific register allocated in microprocessor addressing space. The external watch dog is used when the user want to exit from endless loops or to reset anomalous conditions not estimated by application program.

- **serial EEPROM** With the IC9 EEPROM module (range 256÷8192 bytes), there is the possibility to keep data also when power supply is failed; in this way the card is always able to maintain parameters, few logged data, system status and configuration, etc. without using backed RAM. This component has a default size of 512 bytes.

- **board configuration** Jumper J1 has been introduced expressly to make the board and in particular the application program configurable. The possibility to read by software the status of this jumper gives the user the ability to manage several working conditions through an unique program, without no need to employ more input singnals (typical applications are: language choice, program parameters definition, operational mode selection, etc.). Some software tools developed for the GPC® 324 board use jumper J1 to select between the operation modalities RUN and DEBUG, as described in the manuals of the tools themselves.

- **real time clock** GPC® 324 board is provided with a complete Real Time Clock device capable to manage hours, minutes, seconds, day of month, month, year and day of week in stand alone mode. The component is supplied by the back up circuitry to warrant data integrity in every working condition and is completely software programmable acting on 16 registers addressable in the CPU addressing space by the control logic. The RTC section can generate interrupts at a software programmable rate, for diverting the CPU from its normal tasks or awakening it from one of its low consumption working modes.

- **I/O lines** Five lines of the four 8 bits ports are connected to CN5 connector. The lines direction is software settable at bit level and interrupts can be generated. In this way an external status can obtain CPU control in any condition, with a fast response time. The ports are completely driven by software by programming the proper microprocessor internal registers.

For further information about peripheral devices please refer to chapter "PERIPHERAL DEVICE SOFTWARE DESCRIPTION" and APPENDIX B.
MEMORY DEVICES

On the card can be mounted 104K of memory divided with 32K EPROM on socket, 32K SRAM, 32K SRAM, EPROM, EEPROM or FLASH EPROM on socket and up to 8K of serial EEPROM. The GPC® 324 memory configuration must be chosen considering the application to realize or the specific requirements of the user. Normally the card is equipped with 32K byte of static RAM and 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 32K SRAM data (IC7), 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 based on the on board and external battery, as described in "BACK UP" paragraph. By mounting a proper device on IC5 socket the user can improve the on board backed memory capacity.

The addressing of memory devices is controlled by a specific control logic, that provides to allocate the devices in the microprocessor address space, this control logic automatically manages the different addressing mode and it satisfy the requests of each GPC® 324 software tools.

For further information about memory configuration, sockets description and jumpers connection, please refer to paragraphs "MEMORY ADDRESSES" and "MEMORY SELECTION".

CLOCK

GPC® 184 is provided with a circuitry that generates the CPU clock frequency (22.1184 MHz); this frequency is used also to generate the frequencies needed to the other sections of the board (Timer/Counters, serial lines, etc.). On some microprocessors it is possible to divide the clock frequency by software to reduce the power consumption.

POWER SUPPLY

The card must be powered only with +5 Vdc through the pin 25 (GND) and pin 26 (+5Vdc) of the CN1 connector. The power supply circuit generates all the necessary voltages for the card and it is designed for reducing the consumption (the microprocessor power down and idle mode are available) and for increasing the electrical noise immunity. An interesting power failure circuitry capable to detect the imminent power black out is installed on the board, so it can start a software intervent through a generated interrupt. Please remember that on board there is a protection circuit against voltage peaks by TransZorb™.
TECHNICAL FEATURES

GENERAL FEATURES

Devices:
- 5 programmable TTL input/output lines
- 3 timer counters (16 bits)
- 1 bidirectional RS 232 serial line (software/hardware)
- 1 bidirectional RS 232, RS 422, RS 485, current loop serial line
- 1 watch dog
- 1 local contact for reset
- 1 software readable user inputs
- 1 ABACO® I/O BUS expansion interface
- 1 backed Real Time Clock
- 1 power failure circuit

Memory:
- IC 8: 32K x 8 EPROM
- IC 7: 32K x 8 SRAM
- IC 5: 32K x 8 SRAM, EPROM, EEPROM, FLASH EPROM
- IC 9: serial EEPROM from 256 bytes to 8192 bytes

Memories access time: 70 nsec

CPU:
- INTEL 80C32 and compatible ones
- ATMEL 89S8252 and compatible ones
- PHILIPS 89CRx+/2 and compatible ones
- DALLAS 80C320 and compatible ones

Clock Frequency: 22.1184 MHz

External watch dog reset time: from 940 msec to 2060 msec (typical 1420 msec)

PHYSICAL FEATURES

Size:
- 100 x 50 x 25 mm (without container)
- 110 x 60 x 60 mm (with plastic container)

Weight:
- 75 g (without container)
- 130 g (with plastic container)

Connectors:
- CN1: 26 pins, male, vertical, low profile connector
- CN2: 2 pins, male, vertical, low profile connector
- CN3A: 6 pins PLUG connector
- CN3B: 6 pins PLUG connector
- CN5: 4+4 pins, male, vertical, strip connector

Temperature range: 0÷50 °C

Relative humidity: 20%÷90% (without condense)
ELECTRIC FEATURES

Power supply voltage: +5 Vdc

Consumption on 5 Vdc:
- 115 mA (default configuration)
- 160 mA (full and higher configuration)

On board back up battery: 3.0 Vdc; 180 mAh

External back up battery: 3.6÷5 Vdc

Back up current:
- 4.2 µA (on board battery)
- 5.5 µA (external 3.6 V battery)

RS 422, RS 485 termination network:
- Line termination = 120 Ω
- Positive pull-up resistor = 3.3 KΩ
- Negative pull-down resistor = 3.3 KΩ

Power failure threshold: 52 mV before reset activation

Figure 3: Card photo
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®324 module has 5 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 7) and some electrical diagrams that shows the on board circuit of each connector.

CN2 - EXTERNAL BACK UP BATTERY CONNECTOR

CN2 is a 2 pins, male, vertical, low profile connector with 2.54 mm pitch. Through CN2 the user back up the IC7 SRAM and the real time clock through an external battery when the power supply is switched off (for further information please refer to paragraphs "ELECTRIC FEATURES", "BACK UP" and "MEMORY SELECTION").

Signals description:

+Vbat = I - External back up battery positive pin
GND = - External back up battery negative pin
CN1 - ABACO® I/O BUS CONNECTOR

CN1 is a 26 pins, male, vertical, low profile connector with 2.54 mm pitch. Through CN1 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. All this connector signals are at TTL level.

![Diagram of CN1](image)

**Figure 5: CN1 - ABACO® I/O BUS CONNECTOR**

Signals description:
- **A0÷A7** = O - Address BUS.
- **D0÷D7** = I/O - Data BUS.
- **/INT BUS** = I - Interrupt request (open collector type).
- **/NMI BUS** = I - Non mascable interrupt.
- **/IORQ** = O - Input output request.
- **/RD** = O - Read cycle status.
- **/WR** = O - Write cycle status.
- **/RESET** = O - Reset.
- **+5 Vdc** = I - +5 Vdc power supply.
- **GND** = - Ground signal.
- **N.C.** = - Not connected.
CN3A - SERIAL LINE A CONNECTOR

CN3A is a 6 pins, female PLUG connector. On CN3A are available the buffered signals for RS 232, RS 422, RS 485, current loop serial line A that is physically connected to the hardware serial line 0 of the microprocessor. The electric protocol follows the CCITT normative and all the signals are placed in order to reduce interference and electrical noise and in order to simplify connection with other systems.

**Figure 6: CN3A - Serial line A connector**

Signals description:

- **RXA RS 232** = I - Serial line A RS 232 Receive Data.
- **TXA RS 232** = O - Serial line A RS 232 Transmit Data.
- **RXA- RS 422** = I - Receive Data Negative: Serial line A negative signal for RS 422 serial differential receive.
- **RXA+ RS 422** = I - Receive Data Positive: Serial line A positive signal for RS 422 serial differential receive.
- **TXA- RS 422** = O - Transmit Data Negative: Serial line A negative signal for RS 422 serial differential transmit.
- **TXA+ RS 422** = O - Transmit Data Positive: Serial line A positive signal for RS 422 serial differential transmit.
- **RXTXA- RS 485** = I/O- Receive Transmit Data Negative: Serial line A negative signal for RS 485 serial differential receive and transmit.
- **RXTXA+ RS 485** = I/O- Receive Transmit Data Positive: Serial line A positive signal for RS 485 serial differential receive and transmit.
- **+5 Vdc/GND**
RXTXB+ RS 485 = I/0- Receive Transmit Data Positive: Serial line A positive signal for RS 485 serial differential receive and transmit.

RXB- C.L. = I - Receive Data Negative: Serial line A negative signal for Current Loop serial bipolar receive.

RXB+ C.L. = I - Receive Data Positive: Serial line A positive signal for Current Loop serial bipolar receive.

TXB- C.L. = O - Transmit Data Negative: Serial line A negative signal for Current Loop serial bipolar transmit.

TXB+ C.L. = O - Transmit Data Positive: Serial line A positive signal for Current Loop serial bipolar transmit.

+5 Vdc = I - +5 Vdc or ground signal.

GND = - Ground signal.

**Figure 7:** P1, memories, battery, connectors location
CN3B - SERIAL LINE B CONNECTOR

CN3B is a 6 pins, female PLUG connector. On CN3B are available the buffered signals for RS 232 serial line B that is physically connected to the hardware serial line 1 of the microprocessor or to the software serial line. The electric protocol follows the CCITT normative and all the signals are placed in order to reduce interference and electrical noise and in order to simplify connection with other systems.

**Figure 8: CN3B - Serial Line B Connector**

Signals description:

- **RXB RS 232** = I - Serial line B RS 232 Receive Data.
- **TXB RS 232** = O - Serial line B RS 232 Transmit Data.
- **+5 Vdc** = I - +5 Vdc or ground signal.
- **GND** = - Ground signal.
- **N.C.** = - Not connected.
FIGURE 9: SERIAL COMMUNICATION DIAGRAM
**Figure 10:** RS 232 point to point connection example

**Figure 11:** RS 422 point to point connection example

**Figure 12:** RS 485 point to point 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 extrem, respectively near the master unit and the slave unit at the greatest distance from the master.

Forcing and terminating circuitry is installed on GPC® 324 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 14: 4 wires current loop point to point connection example

Figure 15: 2 wires current loop point to point connection example
Possible Current Loop connections are two: 2 wires and 4 wires. These connections are shown in figures 14 ÷ 16 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 16: Current Loop Network Connection Example**
CN5 - CPU I/O LINES AND PROGRAMMING VOLTAGE CONNECTOR

CN5 is a 4+4 pins, male, vertical, strip connector with 2.54 mm pitch. On CN5 are available some signals of microprocessor ports 1 and 3 that can be connected to external devices and the possible input for the voltage required by PHILIPS in system programmable microprocessors. Please note and remember that the lines P3.3 can be used only when jumper J2 is not connected.

![Figure 17: CN5 - CPU I/O lines and programming voltage connector](image)

**Signals description:**

| P1.n | = I/O - Digital line n of the CPU port 1. |
| P3.3 | = I/O - Digital line 3 of the CPU port 3 = /INT1. |
| Vpp  | = I - +12 Vdc or +5 Vdc programming voltage (for further information read paragraph “IN SYSTEM PROGRAMMING”). |
| +5 Vdc | = I - +5 Vdc power supply. |
| GND  | = - Ground signal. |
FIGURE 18: I/O LINES CONNECTION DIAGRAM
I/O CONNECTION

To prevent possible connecting problems between GPC® 324 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, RS 485 and current loop 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 and if an electric insulation is necessary, then an opto coupled interface must be connected. The 0 Vdc level corresponds to logic state "0", while 5Vdc level corresponds to logic state "1".

DIGITAL I/O INTERFACES

Through CN1 (ABACO® I/O BUS standard connector) the GPC® 324 card can be connected to all the numerous grifo® boards featuring the same standard pin out. It is really interesting the connection to ETI 324 card that expands the digital I/O lines and that allows a direct connection to many digital I/O interfaces, ready to use. Installation of these modules is very easy; in fact only a 20 pins flat cable (code FLT20+20) is required, while the software management of these interfaces is as easy; in fact most of the software packages available for GPC® 324 card are provided with the necessary procedures. Remarkable interfaces 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® 324 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. About software the developed drivers provide procedures to read and write data at a specified address, for the selected programming language.

- 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 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.
Figure 19: Components maps (components side)

Figure 20: Components maps (soldering side)
**JUMPERS**

On GPC® 324 there are 19 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. Here below is the jumpers list, location and function:

<table>
<thead>
<tr>
<th>JUMPERS</th>
<th>PIN N°</th>
<th>FUNCTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>J1</td>
<td>2</td>
<td>It selects the connection for RUN/DEBUG user input.</td>
</tr>
<tr>
<td>J2</td>
<td>2</td>
<td>It selects the memory map.</td>
</tr>
<tr>
<td>J3</td>
<td>3</td>
<td>It selects the memory map.</td>
</tr>
<tr>
<td>J4</td>
<td>5</td>
<td>It selects IC5 memory type.</td>
</tr>
<tr>
<td>J9</td>
<td>3</td>
<td>It selects the electric interface for the serial communication line A.</td>
</tr>
<tr>
<td>J10, J11</td>
<td>2</td>
<td>They connect the serial line A, buffered in RS 232, to CN3A connector.</td>
</tr>
<tr>
<td>J12</td>
<td>3</td>
<td>It selects the direction and operating mode for RS 422 and RS 485 serial line A communication line.</td>
</tr>
<tr>
<td>J13</td>
<td>3</td>
<td>It selects the connection for pin 32 (/PSEN) of the CPU.</td>
</tr>
<tr>
<td>J14</td>
<td>5</td>
<td>It selects the connection for pin 35 (/EA) of the CPU.</td>
</tr>
<tr>
<td>JS1, JS2</td>
<td>2</td>
<td>They connect termination and force circuit to RS 422 and RS 485 serial line A communication line.</td>
</tr>
<tr>
<td>JS3</td>
<td>3</td>
<td>It selects the connection for pin 1 of CN3A.</td>
</tr>
<tr>
<td>JS4</td>
<td>3</td>
<td>It selects the connection for pin 1 of CN3B.</td>
</tr>
<tr>
<td>JS10</td>
<td>2</td>
<td>It enables/disables the external watch dog circuit.</td>
</tr>
<tr>
<td>JS14</td>
<td>2</td>
<td>It connects the on board battery BT1 to the back up circuit.</td>
</tr>
<tr>
<td>JS19</td>
<td>2</td>
<td>It connects the power failure circuit to microprocessor interrupt.</td>
</tr>
<tr>
<td>JS23</td>
<td>2</td>
<td>It connects the real time clock to microprocessor interrupt.</td>
</tr>
</tbody>
</table>

**Figure 21: JUMPERS summarizing table**

The following tables describe all the right connections of GPC® 324 jumpers with their relative functions. To recognize these valid connections, please refer to the board printed diagram (serigraph) or to figures 19,20 of this manual, where the pins numeration is listed, while for recognizing jumpers location, please refer to figures 22,23.

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 22: Jumpers Location (Component Side)**

- J11
- J10
- J4
- J13
- J5
- J9
- J12

**Figure 23: Jumpers Location (Soldering Side)**

- JS4
- JS3
- JS2
- JS23
- JS10
- JS19
- JS1
### 2 PINS JUMPERS

<table>
<thead>
<tr>
<th>JUMPERS</th>
<th>CONNECTION</th>
<th>FUNCTION</th>
<th>DEF.</th>
</tr>
</thead>
<tbody>
<tr>
<td>J1</td>
<td>not connected</td>
<td>It sets RUN/DEBUG user input at logic level 1 (DEBUG mode).</td>
<td>*</td>
</tr>
<tr>
<td></td>
<td>connected</td>
<td>It sets RUN/DEBUG user input at logic level 0 (RUN mode).</td>
<td></td>
</tr>
<tr>
<td>J2</td>
<td>not connected</td>
<td>This jumper is used with J3 for the memory map selection. For further information please refer to chapter &quot;MEMORY ADDRESSES&quot;.</td>
<td>*</td>
</tr>
<tr>
<td></td>
<td>connected</td>
<td></td>
<td></td>
</tr>
<tr>
<td>J10, J11</td>
<td>not connected</td>
<td>The RS 232 serial line A is not connected to proper pins on CN3A.</td>
<td>*</td>
</tr>
<tr>
<td></td>
<td>connected</td>
<td>The RS 232 serial line A is connected to proper pins on CN3A.</td>
<td></td>
</tr>
<tr>
<td>JS1, JS2</td>
<td>not connected</td>
<td>The termination and force circuit is not connected to RS 422 and RS 485 serial line A.</td>
<td>*</td>
</tr>
<tr>
<td></td>
<td>connected</td>
<td>The termination and force circuit is connected to RS 422 and RS 485 serial line A.</td>
<td></td>
</tr>
<tr>
<td>JS10</td>
<td>not connected</td>
<td>It disables the watch dog circuit by hardware.</td>
<td>*</td>
</tr>
<tr>
<td></td>
<td>connected</td>
<td>It enables the watch dog circuit by hardware.</td>
<td></td>
</tr>
<tr>
<td>JS14</td>
<td>not connected</td>
<td>It doesn't connect the on board battery BT1 to the back up circuit.</td>
<td>*</td>
</tr>
<tr>
<td></td>
<td>connected</td>
<td>It connects the on board battery BT1 to the back up circuit.</td>
<td></td>
</tr>
<tr>
<td>JS19</td>
<td>not connected</td>
<td>It doesn't connect the power failure circuit to the microprocessor /INT0 interrupt.</td>
<td>*</td>
</tr>
<tr>
<td></td>
<td>connected</td>
<td>It connects the power failure circuit to the microprocessor /INT0 interrupt.</td>
<td></td>
</tr>
<tr>
<td>JS23</td>
<td>not connected</td>
<td>It doesn't connect the real time clock to the microprocessor /INT1 interrupt.</td>
<td>*</td>
</tr>
<tr>
<td></td>
<td>connected</td>
<td>It connects the power failure circuit to the microprocessor /INT1 interrupt.</td>
<td></td>
</tr>
</tbody>
</table>

**FIGURE 24: 2 PINS JUMPERS TABLE**
3 PINS JUMPERS

<table>
<thead>
<tr>
<th>JUMPERS</th>
<th>CONNECTION</th>
<th>FUNCTION</th>
<th>DEF.</th>
</tr>
</thead>
<tbody>
<tr>
<td>J3</td>
<td>not connected</td>
<td>This jumper is used with J2 for the memory map selection. For further information please refer to chapter &quot;MEMORY ADDRESSES&quot;.</td>
<td>*</td>
</tr>
<tr>
<td></td>
<td>position 1-2</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>position 2-3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>J9</td>
<td>position 1-2</td>
<td>It configures serial line A for RS 232 standard.</td>
<td>*</td>
</tr>
<tr>
<td></td>
<td>position 2-3</td>
<td>It configures serial line A for RS 422, RS 485 or current loop standard.</td>
<td></td>
</tr>
<tr>
<td>J12</td>
<td>position 1-2</td>
<td>It configures serial line A for RS 485 electric standard (2 wires).</td>
<td>*</td>
</tr>
<tr>
<td></td>
<td>position 2-3</td>
<td>It configures serial line A for RS 422 electric standard (4 wires).</td>
<td></td>
</tr>
<tr>
<td>J13</td>
<td>position 1-2</td>
<td>It connects pin 32 (/PSEN) of CPU to GND, to enable the microprocessor FLASH programming.</td>
<td>*</td>
</tr>
<tr>
<td></td>
<td>position 2-3</td>
<td>It connects pin 32 (/PSEN) of CPU to control logic of the card.</td>
<td></td>
</tr>
<tr>
<td>JS3</td>
<td>position 1-2</td>
<td>It connects pin 1 of CN3A to GND.</td>
<td>*</td>
</tr>
<tr>
<td></td>
<td>position 2-3</td>
<td>It connects pin 1 of CN3A to +5 Vdc.</td>
<td></td>
</tr>
<tr>
<td>JS4</td>
<td>position 1-2</td>
<td>It connects pin 1 of CN3B to GND.</td>
<td>*</td>
</tr>
<tr>
<td></td>
<td>position 2-3</td>
<td>It connects pin 1 of CN3B to +5 Vdc.</td>
<td></td>
</tr>
</tbody>
</table>

**FIGURE 25: 3 PINS JUMPERS TABLE**

5 PINS JUMPERS

<table>
<thead>
<tr>
<th>JUMPERS</th>
<th>CONNECTION</th>
<th>FUNCTION</th>
<th>DEF.</th>
</tr>
</thead>
<tbody>
<tr>
<td>J4</td>
<td>position 1-2 and 3-4</td>
<td>It configures IC5 for 32K FLASH EPROM.</td>
<td>*</td>
</tr>
<tr>
<td></td>
<td>position 2-3 and 4-5</td>
<td>It configures IC5 for 32K SRAM or EEPROM.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>position 3-4</td>
<td>It configures IC5 for 32K EPROM.</td>
<td></td>
</tr>
<tr>
<td>J14</td>
<td>position 1-2</td>
<td>It connects pin 35 (/EA) of CPU to Vpp from CN5, to provide the microprocessor FLASH programming voltage.</td>
<td>*</td>
</tr>
<tr>
<td></td>
<td>position 2-3 or 3-4</td>
<td>It connects pin 35 (/EA) of CPU to GND, to disable the microprocessor internal ROM.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>position 4-5</td>
<td>It connects pin 35 (/EA) of CPU to +5 Vdc, to enable the microprocessor internal ROM.</td>
<td></td>
</tr>
</tbody>
</table>

**FIGURE 26: 5 PINS JUMPERS TABLE**
MEMORY SELECTION

On GPC® 324 can be mounted 104K bytes of memory divided in several configurations, as described in the following table:

<table>
<thead>
<tr>
<th>IC</th>
<th>DEVICE</th>
<th>SIZE</th>
<th>JUMPERS CONFIGURATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>SRAM, EEpROM</td>
<td>32K Bytes</td>
<td>J4 in 2-3 and 4-5</td>
</tr>
<tr>
<td></td>
<td>FLASH EPROM</td>
<td>32K Bytes</td>
<td>J4 in 1-2 and 3-4</td>
</tr>
<tr>
<td></td>
<td>EPROM</td>
<td>32K Bytes</td>
<td>J4 in 3-4</td>
</tr>
<tr>
<td>7</td>
<td>SRAM</td>
<td>32K Bytes</td>
<td>-</td>
</tr>
<tr>
<td>8</td>
<td>EEpROM</td>
<td>32K Bytes</td>
<td>-</td>
</tr>
<tr>
<td>9</td>
<td>EEpROM</td>
<td>256-8192 Bytes</td>
<td>-</td>
</tr>
</tbody>
</table>

**Figure 27: Memory selection table**

The sockets IC 5 and IC 8 follow the JEDEC standard, so the mounted memory devices must have JEDEC pin outs. IC 9 is a serial memory device that must be specified at the moment of the order and can be mounted only by grifo® technicians. The jumpers configurations described on figure 27 only set the sockets for the indicated memory device, but there are some other jumpers that set the memory addressing map; for this information, please refer to "MEMORY ADDRESSES" paragraph. Normally GPC® 324 is supplied in its default configuration with 32K RAM on IC 7 and 512 byte EEPROM on IC 9; each different configurations can be defined during order phase or self mounted by the user. Below are reported the abbreviation of the possible memory options:

- .32K -> 32K x 8 SRAM
- .32KMOD -> 32K x 8 backed SRAM
- .32EE -> 32K x 8 parallel EEPROM
- .32KF -> 32K x 8 parallel FLASH EPROM
- .EE02 -> 2K bit (256 byte) serial EEPROM
- .EE08 -> 8K bit (1K byte) serial EEPROM
- .EE16 -> 16K bit (2K byte) serial EEPROM
- .EE32 -> 32K bit (4K byte) serial EEPROM
- .EE64 -> 64K bit (8K byte) serial EEPROM

For further information and prices please contact directly grifo®.

SOLDER JUMPERS

The solder jumpers called JSxx are connected by a thin copper connection on the solder side. So, if one of their configurations has to be changed, the user must first cut this connection using a sharpened cutter, then make the new connection using a low power soldering tool and some non corrosive tin.
INTERRUPTS MANAGEMENT

One of the most important GPC® 324 features is the powerful interrupts management. Here is a short description of how the board's hardware interrupt signals can be managed; a more complete description of the hardware interrupts can be found in the microprocessor data sheets or in appendix B of this manual.

- CPU inside devices -> Possible sources of internal interrupt events are: timer 0÷2; serial port 0, 1; External interrupts 0÷5; internal watch dog.
- I/O signals -> The P3.3 signal of CN5 is connected to pin /INT1 = P3.3 of CPU.
- Real Time Clock -> It is wire anded to pin /INT1 = P3.3 of CPU, according to JS23 connection.
- Power failure -> It is wire anded to pin /INT0 = P3.2 of CPU, according to JS19 connection.
- ABACO® I/O BUS -> The /INT BUS signal of CN1 is connected to pin /INT0 = P3.2 of CPU. The /NMI BUS signal of CN1 is connected to pin T2 = P1.0 of CPU.

The last described connection is really important for two different reasons: each activation of /NMI BUS signal can generate an interrupt or each /NMI BUS signal change can be counted. The /NMI BUS signal management is defined by software programmation of timer 2, so the user can select the favourite mode. This feature is really important especially when GPC® 324 is connected to external card as ZBT xxx and ZBR xxx, in fact optocoupled digital signals can be counted or they can generate standard interrupts.

The microprocessor features a programmable priority structure that manages the case of contemporary interrupts. The addresses of the interrupt response subroutines can be software programmed by the user placing them on the proper code areas while the interrupts priority level and activation are software programmable through internal CPU registers. So the user program has always the possibility to react promptly to every external event, deciding also the priority of interrupts.

RESET AND WATCH DOG

The watch dog circuit of GPC® 324 is really efficient and provided of easy software management. In details the most important features of this circuit are:
- astable functionality;
- intervent time of about 1420 msec;
- hardware enable;
- software retrigger;

With the astable mode when the intervent time elapses, the circuit becomes active, it stay active till the end of reset time (about200 msec) and after it is deactivated. Jumper JS10 connects the watch dog circuit to reset circuit so when it is connected the watch dog is enabled and viceversa. The watch dog retrigger operation is described in chapter "WATCH DOG".

After an activation and following deactivation of /RESET signal, the card resumes execution of the program saved on IC8 (at address 0000H) starting from a global reset status of all the on board peripheral devices.

Please remember that the /RESET signal is connected to CN1 connector and that on GPC® 324 are available other reset sources as the power good circuit and the contact P1. The two pins of P1 can be connected to a normally open contact (i.e. a push button) and when the contact is closed (shortcut of the two pins) the reset circuit is enabled.
SERIAL COMMUNICATION SELECTION

Please remember that if not differently specified during the order phase, the card is delivered in its default configuration with two RS 232 serial line.

The serial line A is available on connector CN3A and can be buffered in RS 232, RS 422, RS 485 or current loop electric standard. By hardware can be selected which one of these electric standard is used, through jumpers connection (as described in the previous table). By software the serial line can be programmed to operate with 8, 9 bits per character, no parity, 1 stop bits at standard or no standard baud rates, through some some CPU internal register setting.

Some components necessary for RS 422 and RS 485 communication are not mounted and not tested on the default configuration card, so the first not standard configuration must always be executed by grifo® technician; then the user can change himself the configuration, following the below description:

- SERIAL LINE A CONFIGURED IN RS 232 (default configuration)
  IC 1 = no component
  IC 2 = no component
  J9 = position 1-2
  J10, J11 = connected
  J12 = don't care
  JS1, JS2 = not connected
  IC 3 = no component
  IC 4 = no component
  IC 13 = MAX 202 driver

- SERIAL LINE A CONFIGURED IN CURRENT LOOP (.CLOOP option)
  IC 1 = no component
  IC 2 = HCPL 4200 driver
  J9 = position 2-3
  J10, J11 = not connected
  J12 = don't care
  JS1, JS2 = not connected
  IC 3 = no component
  IC 4 = HCPL 4100 driver
  IC 13 = don't care

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 as described in figures 14÷16.

- SERIAL LINE A CONFIGURED IN RS 422 (.RS422 option)
  IC 1 = SN 75176 driver
  IC 2 = no component
  J9 = position 2-3
  J10, J11 = not connected
  J12 = position 1-2
  JS1, JS2 = (*)
  IC 3 = no component
  IC 4 = SN 75176 driver
  IC 13 = don't care

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

  DIR = low level = logic state 0 -> transmitter enabled
  DIR = high level = logic state 1 -> transmitter disabled

In point to point connections, signal DIR can be always kept low (transmitter always enabled), while in multi point connections transmitter must be enabled only when a transmission is requested.
Serial A in RS 232  
Serial A in Current Loop  
Serial A in RS 422  
Serial A in RS 485

**Figure 28: Serial Communication Drivers Location**
- SERIAL LINE A CONFIGURED IN RS 485 (.RS485 option)

<table>
<thead>
<tr>
<th>J9</th>
<th>position 2-3</th>
<th>IC 1</th>
<th>SN 75176 driver</th>
</tr>
</thead>
<tbody>
<tr>
<td>J10, J11</td>
<td>not connected</td>
<td>IC 2</td>
<td>no component</td>
</tr>
<tr>
<td>J12</td>
<td>position 2-3</td>
<td>IC 3</td>
<td>no component</td>
</tr>
<tr>
<td>JS1, JS2</td>
<td>(*)</td>
<td>IC 4</td>
<td>no component</td>
</tr>
<tr>
<td></td>
<td></td>
<td>IC 13</td>
<td>don't care</td>
</tr>
</tbody>
</table>

In this modality the signals to use are pins 4 and 5 of connector CN3A, that become transmission or reception lines according to the status of signal DIR, managed by software, as follows:

- **DIR** = low level = logic state 0 -> transmitter enabled
- **DIR** = 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 success of transmission. In fact, any conflict on the line can be recognized by testing the received character after each transmission.

(*) 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 JS1 and JS2. 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 farthest boards, that is on the edges of the communication line.

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

The serial line B is available on connector CN3B and it can be buffered only in RS 232. By mounting a MAX 202 driver on IC 13 this second serial line is hardware enabled and by software it can be managed as below described:

- *µP 80C32 and compatible ones*
  
  The serial line B is a software serial line, managed through two I/O pins of the processor (pin 4 = P1.2 = RXB and pin 5 = P1.3 = TXB). The communication parameters (baud rate, stop bit, bit x char, etc.) are software defined through some timing and some sequence of management procedure. For further information, please refer to the software tools manuals.

- *µP 80C320 and compatible ones*
  
  The serial line B is an hardware serial line, managed through the dedicated pins of the processor (pin 4 = RXB and pin 5 = TXB). The communication parameters (baud rate, stop bit, bit x char, etc.) are software defined through proper microprocessor registers setting. For further information, please refer to microprocessor data sheets.

For further informations about serial communication please refer to the examples of figures 10÷16 and paragraph "RS 422-485 DIRECTION".
BACK UP

GPC® 324 has an on-board lithium battery BT1 for the backup of SRAM and RTC content when power supply is switched off. Jumper JS14 connects physically the battery so it can be disconnected to save its duration whenever backup is not needed. By CN2 connector it is possible to connect an external battery: configuration of jumper JS14 does not affect the working of this battery and it can replace BT1 completely. The user can order an external battery (2.1 Ah) ready to be connected to CN2 with the code: LITIO.

Please refer to the paragraph “ELECTRIC FEATURES” to choose the type of the external backup battery, to easily locate see figure 7.

POWER FAILURE

In addition to the CPU controlled power management circuitry, GPC® 324 card also features an efficient power failure circuitry. Through jumper JS19 this latter can be connected to the microprocessor /INT0 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 of 52 mV above the reset intervend.

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.

IN SYSTEM PROGRAMMING

One of the most important features of GPC® 324 is the possibility to use the new PHILIPS 89CRx+ /2 microprocessors that support the in system and in application programming (ISP). Below are listed the sequence of operations that must be performed by the user to use this features:

1) develop the application program through a proper software tools that generate an executable code;
2) connect jumper J13 in position 1-2 and J14 in position 1-2;
3) connect the programming voltage (according to used microprocessor: +12 V for 89CRx+ or +5 V for 89CRx2) to pin 8 of CN5, referred to pin 5 always of CN5;
4) connect RS 232 serial line A to a personal computer free COM line;
5) power on the card;
6) program the microprocessor internal FLASH EPROM by using the specific program supplied by PHILIPS: WINISP.
7) power off the card;
8) disconnect the programming voltage from CN5, connect J13 in position 2-3 and J14 in position 4-5;
9) power on the card: the programmed application program will start execution from internal ROM.

The ISP reduce the total application cost, in fact it eliminates the requirements of EPROM, EPROM programmer, external FLASH EPROM, etc. For further informations on in system programming please refer to specific technical documentation from PHILIPS.
A wide selection of software development tools can be obtained, allowing use of the module 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 51 family, can be used.

**BASIC 324:** complete development tools for MCS BASIC (interpreted BASIC language for industrial application). It needs a P.C. for console and program saving operations, while the debug, test and program operations are performed on the card. Special instructions which manage the on board peripheral devices have been added, plus the possibility to save the application on EEPROM.

**BXC51:** cross compiler for source files written in BASIC 324. It allows a considerable speed increment of the starting MCS BASIC program and it is completely executed on external P.C.

**FORTH:** complete software development tools to program the card with FORTH high level language. It needs a P.C. for user interface and it is really interesting for its fast execution and small size, of the generated code.

**MCC 51:** integer cross compiler for source files in standard ANSI C. It produces a source assembly file compatible with MCA 51 or with Intel macro relocatable assembler MCS 51.

**MCA 51:** macro cross assembler available for MS-DOS operating system in absolute and relocatable version. In this relocatable version is supplied with a linker and a library manager.

**MCS 51:** source level debugger and simulator. It is executed on P.C. without any additional hardware and it allows loading of HEX and SYMBOLIC files, breakpoint setting, instruction execution in trace mode, registers and memory dump, etc.

**MCK 51:** it is the sum of MCC 51 and MCA 51 and it is a complete C compiler with an output file type compatible with MCS 51.

**HI TECH C 51:** cross compiler for C source program. It is a powerful software tool that includes editor, C compiler, assembler, optimizer, linker, library, and remote symbolic debugger, in one easy to use integrated development environment.

**SYS51CW:** cross compiler for C source program. It is a powerful software tool that includes editor, C compiler, assembler, optimizer, linker, library, simulator and remote symbolic debugger, included in an easy to use integrated development environment for Windows.

**SYS51PW:** cross compiler for PASCAL source program. It is a powerful software tool that includes editor, PASCAL compiler, assembler, optimizer, linker, library, simulator and remote symbolic debugger, included in an easy to use integrated development environment for Windows.

**MDP:** monitor debugger program able to load and debug each HEX Intel files for 51 microprocessor family. It is provided of the standard commands available on in circuit emulator and requires only an external P.C. connected through a serial line.
DDS MICRO C 51: low cost ross 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.

SOFTICE: It is a remote symbolic debugger with cross assembler. It is provided of the standard commands available on in circuit emulator and requires only an external P.C. connected through a serial line. There is an high level user interface that can visualize all the processor status in a multiwindow visualization.

NOICE 51: It is a PC hosted debugger consists of a target specific DOS program, NOICExxx.EXE, and a target resident monitor program. The two programs communicate via RS 232. NOICE includes: source level debug; a disassembler; a file viewer; memory display and editing; a virtually unlimited number of breakpoints; hardware free single step; definition of symbols; the ability to record and play back files of commands; on line help.

OPEN 51/UNI: in circuit emulator for the 51 family. It is a powerful hardware and software tool that includes: source level debugging and symbolic debugging; project management; built-in multi-file editor; execution of external compilers; debugging of several modules at the same time; built-in disassembler; source level step and trace functions; animate functions; inserting and deleting of breakpoints on the source level; watching and modifying variables on symbol and absolute level.

BASCOM 8051: cross compiler for BASIC source program. It is a powerful software tool that includes editor, BASIC compiler and simulator included in an easy to use integrated development environment for Windows. Many memory models, data types and direct use of hardware resource instructions are available.

GET 51: it is a complete program with editor, communication driver and mass memory management for all '51 family cards. This program developed by grifo® allows to operate in the best conditions when BASIC 324, BXC51, MDP, FMO 52 software tools are used. The program is menu driven and mouse driven. It is designed to run undr MS-DOS but can run also in MACINTOSH environment with VIRTUAL-PC. It is delivered in MS-DOS 3”1/2 floppy disks.

FM052: monitor debugger program able to load and debug each HEX Intel files for 51 microprocessor family. It is provided of the standard commands available on in circuit emulator and requires only an external P.C. connected through a serial line. It is preconfigured to work directly with BASCOM 8051 and GET51 software tools. It can also program the on board FLASH EPROM with the user application program and then execute it in autotrun mode.
INTRODUCTION

In this chapter are reported all information about card use, related to hardware and software. For example, the registers addresses and the memory allocation are described below.

ADDRESSES

The card devices addresses are managed by a specific control logic, realized with programmable array logic. This control logic allocates memory and peripheral devices in a comfortable mode for the user. The available microprocessors address 64K bytes of code memory and 64K bytes of data memory and the control logic maps the on board memory and peripheral devices, inside these addresses spaces. Control logic sets size, type and addresses of memory devices through jumpers J2, J3 and J4; at the same time it allocates I/O addresses always in the upper 256 bytes of microprocessor data memory. Summarizing the control logic allocates:

- 32K bytes of EPROM on IC 8;
- 32K bytes of SRAM on IC 7;
- 32K bytes of SRAM, EPROM, EEPROM, FLASH EPROM on IC 5;
- ABACO® I/O BUS;
- RUN/DEBUG selector (J1 status);
- External watch dog retrigger;
- Real time clock

The addresses of all these devices are described in the following paragraphs and can't be set with different value. The other device serial EEPROM of IC 9, is managed always by control logic but it is not allocated in memory space, in fact this device is drived through CPU I/O lines with a synchronous communication.

MEMORY ADDRESSES

On the GPC® 324 three different memory configurations can be used. The configuration must be selected, with J2 and J3, both according to used software tools and user requests and/or application features. The following figures describe available memory configurations, with proper J2 and J3 setting.
The position 1-2 for jumper J3 is not described because it is provided for future expansion.
Figure 29: Mode 0 Memory Configuration

Used by software tools as: BASIC 324; BX51; HI TECH C; DDS MICRO C; SYS51CW; SYS51CW; BASCOM 8051; SOFTICE (J3 in 1-2); etc.
MEMORY CONFIGURATION 1

**FIGURE 30: MODE 1 MEMORY CONFIGURATION**

Used by software tools as: HI TECH C; DDS MICRO C; SYS51CW; SYS51CW; BASCOM 8051; etc.
MEMORY CONFIGURATION 3

Figure 31: Mode 3 Memory Configuration

Used by software tools as: MDP; LUCIFER HI TECH C; DDS MICRO C; SYS51CW; SYS51CW; BASCOM 8051; FMO 52, NO ICE 51, etc.
I/O ADDRESSES

I/O addresses are located in the last 256 bytes (128 for **ABACO® I/O BUS** and 128 bytes for **RUN/DEBUG** reading, watch dog retrigger and real time clock) of the 64K bytes data microprocessor addressing space. Next table shows name, addresses, meanings and direction of peripheral devices registers (only the external ones to microprocessor):

<table>
<thead>
<tr>
<th>DEVICES</th>
<th>REG.</th>
<th>ADDRESS</th>
<th>R/W</th>
<th>FUNCTION</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ABACO® I/O BUS</strong></td>
<td>I/O BUS</td>
<td>FF00H−FF7FH</td>
<td>R/W</td>
<td><strong>ABACO® I/O BUS addresses.</strong></td>
</tr>
<tr>
<td><strong>RUN/DEBUG</strong></td>
<td>RUNDEB</td>
<td>FF80H−FFFFFH</td>
<td>R</td>
<td>J1 status reading register.</td>
</tr>
<tr>
<td><strong>WATCH DOG</strong></td>
<td>RWD</td>
<td>FFC0H−FFFFFH</td>
<td>R</td>
<td>Retrigger external watch dog.</td>
</tr>
<tr>
<td><strong>REAL TIME</strong></td>
<td>SEC1</td>
<td>FFC0H</td>
<td>R/W</td>
<td>Data register for seconds units</td>
</tr>
<tr>
<td></td>
<td>SEC10</td>
<td>FFC1H</td>
<td>R/W</td>
<td>Data register for seconds decines</td>
</tr>
<tr>
<td></td>
<td>MIN1</td>
<td>FFC2H</td>
<td>R/W</td>
<td>Data register for minutes units</td>
</tr>
<tr>
<td></td>
<td>MIN10</td>
<td>FFC3H</td>
<td>R/W</td>
<td>Data register for minutes decines</td>
</tr>
<tr>
<td></td>
<td>HOU1</td>
<td>FFC4H</td>
<td>R/W</td>
<td>Data register for hours units</td>
</tr>
<tr>
<td></td>
<td>HOU10</td>
<td>FFC5H</td>
<td>R/W</td>
<td>Data register for hours decines and AM/PM</td>
</tr>
<tr>
<td></td>
<td>DAY1</td>
<td>FFC6H</td>
<td>R/W</td>
<td>Data register for day units</td>
</tr>
<tr>
<td></td>
<td>DAY10</td>
<td>FFC7H</td>
<td>R/W</td>
<td>Data register for day decines</td>
</tr>
<tr>
<td></td>
<td>MON1</td>
<td>FFC8H</td>
<td>R/W</td>
<td>Data register for month units</td>
</tr>
<tr>
<td></td>
<td>MON10</td>
<td>FFC9H</td>
<td>R/W</td>
<td>Data register for month decines</td>
</tr>
<tr>
<td></td>
<td>YEA1</td>
<td>FFCAH</td>
<td>R/W</td>
<td>Data register for year units</td>
</tr>
<tr>
<td></td>
<td>YEA10</td>
<td>FFCBH</td>
<td>R/W</td>
<td>Data register for year decines</td>
</tr>
<tr>
<td></td>
<td>WEE</td>
<td>FFCCH</td>
<td>R/W</td>
<td>Data register for week day</td>
</tr>
<tr>
<td></td>
<td>REGD</td>
<td>FFCDH</td>
<td>R/W</td>
<td>Control register D</td>
</tr>
<tr>
<td></td>
<td>REGE</td>
<td>FFCEH</td>
<td>R/W</td>
<td>Control register E</td>
</tr>
<tr>
<td></td>
<td>REGF</td>
<td>FFCFH</td>
<td>R/W</td>
<td>Control register F</td>
</tr>
</tbody>
</table>

**Figure 32: I/O ADDRESSES table**

For further information about register meanings, please refer to next chapter called "PERIPHERAL DEVICES SOFTWARE DESCRIPTION".
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 figure 32, for the registers name and addresses values). For microprocessor internal peripheral devices, not described in this paragraph, or for further information, please refer to manufacturing company documentation. In the following paragraphs the D7:D0 and .0:.7 indications denote the eight bits of the combination used in I/O operations.

CONFIGURATION INPUT: J1 (RUN/DEBUG)

The on board J1 status can be obtained by software, through a simple read operation at the RUNDEB register address. The correspondence between register bits and J1 status is as follows:

D0:D6  ->  Reserved
D7     ->  J1 status (RUN/DEBUG in some software tools)

Remember that if the jumper is connected the logic status is 0 and if the jumper is not connected the logic status is 1. This register shares the same address of other on board peripherals so the user must remember that the configuration input acquisition has effects also on other peripheral section. This jumper is normally used for system configuration (operating mode selection, card number programme inside a network system, firmware configuration, etc.).

EXTERNAL WATCH DOG

Retrigger operation of GPC® 324 external watch dog circuit is performed with a simple read operation at the address of register RWD. This register shares the same address of other on board peripherals, but no conflict are generated in fact retrigger operation is an input operation and the read data has no meaning. To avoid external watch dog activation is necessary to retrigger its circuit 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 JS10 is connected, when intervention time is elapsed, the card is reset. The default intervention time is about 1420 msec.

RS 422-485 DIRECTION

To manage the RS 485 line direction or the RS 422 transmission driver enabling, on GPC® 324 is used a proper digital I/O line of microprocessor, named DIR. This line is driven by CPU pin 3 (P1.1) and it has the following functionality:

- RS 485:
  - DIR = 0 -> RS 485 line transmitting
  - DIR = 1 -> RS 485 line receiving
- RS 422:
  - DIR = 0 -> RS 422 transmitter driver enabled
  - DIR = 1 -> RS 422 transmitter driver disabled

DIR signal is set (1) after reset or power on, maintaining enabled the RS 485 reception and maintaining disabled the RS 422 transmission; in this way each conflict is eliminated.
SERIAL EEPROM

For software management of serial EEPROM module of IC 9, please refer to specific documentation or to demo programs supplied with the card. The user must realize a serial communication with I²C bus standard protocol, through two I/O microprocessor pins. The only necessary information is the electric connection:

- DATA line (SDA) -> pin 16 (P3.4) of the CPU
- CLOCK line (SCL) -> pin 17 (P3.5) of the CPU
- A2 address line -> GND (0 logic state)
- A1 address line -> GND (0 logic state)
- A0 address line -> GND (0 logic state)

The first 30 bytes of serial EEPROM are reserved for software tools use, so they can't be neither read nor written by user program.

REAL TIME CLOCK

This peripherial 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 RTC data registres' 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
- 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:

<table>
<thead>
<tr>
<th>WEE.2</th>
<th>WEE.1</th>
<th>WEE.0</th>
<th>Day of the week</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>Sunday</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>Monday</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>Tuesday</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>Wednesday</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>Thursday</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>Friday</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>Saturday</td>
</tr>
</tbody>
</table>

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 vice versa), 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 vice versa.

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 = Determine 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 masks 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.

**SERIAL LINES; TIMER/COUNTER; I/O LINES; POWER MANAGEMENT**

For further information on microprocessor internal peripheral device, please refer to specific documentation of the manufacturing company or to appendix B of this manual.
EXTERNAL DEVICES FOR GPC® 324

GPC® 324 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.

ADC 812
Analog to Digital Converter, 12 bits, multi range
DAS (Data Acquisition System) multi range 8 channels 12 bit A/D conversion lines; track and hold; 6μs conversion time; range ±10, ±5, +10, +5Vdc or 0÷20, 4÷20mA; analog inputs connections through quick terminal screw connectors; ABACO® I/O BUS interface; direct mounting for DIN 247277-1 and 3 rails; 4 type dimension.

DAC 212
Digital to Analog Converter 12 bits, multi range
Digital to Analog converter; multi range 2 channels 12 bits ±10, +10 Vdc output; analog outputs connections through quick terminal screw connectors; ABACO® I/O BUS interface; direct mounting for DIN 247277-1 and 3 rails; 4 type dimension.

CAN 14
Control Area Network, 1 channel, galvanically insulated
UART CAN SJA1000; 1 serial channels galvanically insulated; ABACO® I/O BUS interface; 4 type dimension; support of CAN 2.0B protocol; transfer rate up to 1M bit/sec; direct mounting for DIN 247277-1 and 3 rails.

ETI 324
Encoder Timer I/O, 3 counters, 24 I/O
Three timers counters driven by 82C54; bidirectional optocoupled encoder input; direction identifier; phases multiplier; 24 digital lines driven by 82C55 on two standard I/O ABACO® connectors; ABACO® I/O BUS interface; direct mounting for DIN 247277-1 and 3 rails; 4 type dimension.

KDL xxx - KDF xxx
Keyboard Display interface - LCD or Fluorescent
Interface with Fluorescent or LCD display, LEDs backlit, 20x2 or 20x4 characters; up to 24 keys matrix keyboard connector. It is directly driven by 16 TTL I/O lines; High level languages supported.

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.
FIGURE 33: GPC® 324 AVAILABLE CONNECTIONS DIAGRAM

1 Hardware Serial Line RS 232, RS 422, RS 485, Current Loop

1 Hardware/Software Serial Line RS 232

5 DIGITAL TTL INPUT/OUTPUT
Vpp programming voltage
RBO 08, XBI R4, XBI T4, etc.

RELAY

OPTO COUPLED

TRANSISTOR

INPUTS

EXTERNAL LITIUM BATTERY 3,6 V

CONNECTION TO QTP 24P, QTP 16P, KDx xxx, MCI 64, etc.

ABACO® BUS

ZBx series

ABB 03 or ABB 05, etc.

ABACO® I/O BUS

ANY I/O TYPE
CI/O R16-T16, etc.

IPC 52, UAR 24, etc.

ETI 324

82C55

PC like or Macintosh

QTP G28

PLC

PC like or Macintosh

QTP 22

PLC

ABB 03 or ABB 05, etc.
QTP 16 - QTP 16P
Quick Terminal Panel 16 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 messages; buzzer manageable by software; 4 readable auxiliary opto in lines; power supply 5 Vdc. The QTP 16P is low cost no intelligent (passive) version. It is directly driven from 16 TTL I/O lines.

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/Current Loop 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.

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.

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.

FBC xxx
Flat Block Contactxxx pins
This interconnection system "wire to board" allows the connection to many type of flat cable connectors to terminal for external connections. Connection for DIN Ω rails.

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.

DEB 01
Didactis 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.
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.

ZBR xxx
Zipped BLOCK Relays xx Input + xx Output
Peripheral cards family, relays outputs, equipped with housing for Ω rails mounting. Double power supply built in; 5Vdc section for powering the on board logic and an external CPU cards; second section, galvanically coupled, for the optocoupled input lines. All I/O lines are displayed by LEDs. Relays contacts are 3A and are MOV protected. All I/O connections are availables on quick terminal connectors. Connector interface to ABACO® I/O BUS. The ZBR serie represent the ideal complement for cards 3 and 4 type. The ZBR cards can be directly driven, through the PC parallel port, by using the PCC A26 low cost interface card. ZBR 324 -> 32 opto in, 24 relays; ZBR 246 -> 24 opto in, 16 relays; ZBR 168 -> 16 opto in, 8 relays; ZBR 84 -> 8 opto in, 4 relays.

ZBT xxx
Zipped BLOCK Transistors xx Input + xx Output
Peripheral cards family having optocoupled outputs and 3A transistor in open collector. Cards are equipped with housing for Ω rails mounting. Double power supply built in; 5Vdc section for powering the on board logic and an external CPU cards; second section, galvanically coupled, for the optocoupled input lines. All I/O lines are displayed by LEDs. All output transistors are equipped with protection against inductive loads. All I/O connections are availables on easy quick terminal connectors. Connector interface to ABACO® I/O BUS. The ZBT serie represent the ideal complement for cards 3 and 4 type. The ZBT cards can be directly driven, through the PC parallel port, by using the PCC A26 low cost interface card. ZBT 324 -> 32 opto in, 24 transistors; ZBT 246 -> 24 opto in, 16 transistors; ZBT 168 -> 16 opto in, 8 transistors; ZBT 84 -> 8 opto in, 4 transistors.

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.

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 ±2Vdc or 4÷20mA; 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.
BIBLIOGRAPHY

In this chapter there is a complete list of technical books, where the user can find all the necessary documentations on the components mounted on GPC® 324.

Data book TEXAS INSTRUMENTS: The TTL Data Book - SN54/74 Families
Data book TEXAS INSTRUMENTS: RS-422 and RS-485 Interface Circuits

Data book NEC: Memory Products

Data book MAXIM: New Releases Data Book - Volume IV
Data book MAXIM: New Releases Data Book - Volume V

Data book XICOR: Data Book

Data book PHILIPS: 80C51 - Based 8 Bit Microcontrollers


Data book INTEL: 8 Bit Embedded Microcontrollers

Data book TOSHIBA: Mos Memory Products

Data sheet SEIKO EPSON: RTC-62421 Real Time Clock module

For further information and upgrades please refer to specific internet web pages of the manufacturing companies.
APPENDIX A: CARD MECHANICAL MOUNTING

The GPC® 324 can be physically mounted in two different manner. The first is the piggy back mounting (stack trough mode) that use the two connectors CN1 and CN5 for the interface with a user developed board. This connectors lead out of 7 mm on solder side and the user board must have proper female strip connectors (2,54 mm pitch) where the card can be plugged in, obtaining a single system.

The second mode expect a mounting inside a proper plastic container for a direct mounting on DIN 247277-1 and 3 Ω rails (order code BLOCK.100.50); if the card is used with some other peripheral cards (i.e. ZBR xxx or ZBT xxx), a single longer container can be used obtaining a single module. The described long plastic container code is 414487 type RS/100 by Weidmuller and it can be ordered to grifo® as .EXT-WMIII options, where III is the required length. By selecting this mounting the electric connection between GPC® 324 and other peripheral cards is performed with flat cables that must be really short, as the FLT.26+26 I/O for ABACO® I/O BUS signals.

In the following figures are described the module dimensions with the connector positions and some images that illustrate the connection modes.

![Figure A1: Module dimension for piggy back mounting](image-url)
FIGURE A2: PIGGY BACK MOUNTING

FIGURE A3: WEIDMULLER RAIL MOUNTING
APPENDIX B: ON BOARD DEVICES DESCRIPTION

µP 80C32

CMOS single-chip 8-bit microcontrollers
80C32/87C52

DESCRIPTION
The Philips 80C32/87C52 is a high-performance microcontroller fabricated with Philips high-density CMOS technology. The Philips CMOS technology combines the high speed and density characteristics of HMOS with the low power attributes of CMOS. Phillips epitaxial substrate minimizes latch-up sensitivity.

The 87C52 contains an 8k x 8 EPROM and the 80C32 is ROMless. Both contain a 256 x 8 RAM, 32 I/O lines, three 16-bit counter/timers, a six-source, two-priority level nested interrupt structure, a serial I/O port for either multi-processor communications, I/O expansion or full duplex UART, and on-chip oscillator and clock circuits.

In addition, the 80C32/87C52 has two software selectable modes of power reduction—idle mode and power-down mode. The idle mode freezes the CPU while allowing the RAM, timers, serial port, and interrupt system to continue functioning. The power-down mode saves the RAM contents but freezes the oscillator, causing all other chip functions to be inoperative.

See 80C52/80C54/80C58 datasheet for ROM device specifications.

FEATURES
• 80C51 based architecture
• 8032 compatible
  – 8k x 8 EPROM (87C52)
  – ROMless (80C32)
  – 256 x 8 RAM
  – Three 16-bit counter/timers
  – Full duplex serial channel
  – Boolean processor
• Memory addressing capability
  – 64k ROM and 64k RAM
• Power control modes:
  – Idle mode
  – Power-down mode
• CMOS and TTL compatible
• Three speed ranges:
  – 3.5 to 16MHz
  – 3.5 to 24MHz
  – 3.5 to 33MHz
• Five package styles
• Extended temperature ranges
• OTP package available
Timer 2 as a baud rate generator is shown in Figure 4. This figure is valid only if RCX = TCLK = 1 in T2CON. Note that a rollover in TH2 does not set TF2, and will not generate an interrupt. Therefore, the Timer 2 interrupt must be disabled when Timer 2 is in the baud rate generator mode. Notice that if EXEN2 is set, a 0-to-1 transition on T2EX will set EXF2 but will not cause a reload from (RCAP2H, RCAP2L) to (TH2, TL2). Thus when Timer 2 is used as a baud rate generator, T2EX can be used as an extra external interrupt, if desired.

It should be noted that when Timer 2 is running (TR2 = 1) in "timer" function in the baud rate generator mode, one should not try to read or write TH2 or TL2. Under these conditions, the timer is being incremented every state time, and the results of a read or write may not be accurate. The RCAP registers may be read, but should not be written to, because a write might overlap a reload and cause write and/or reload errors. Turn the timer off (clear TR2) before accessing the Timer 2 or RCAP registers, in this case.

Timer/Counter 2 Set-up

Except for the baud rate generator mode, the values given in T2CON do not include the setting of the TR2 bit. Therefore, bit TR2 must be set, separately, to turn the timer on. See Table 3 for set-up of Timer 2 as a timer. See Table 4 for set-up of Timer 2 as a counter.

Using Timer/Counter 2 to Generate Baud Rates

For this purpose, Timer 2 must be used in the baud rate generating mode. If Timer 2 is being clocked through pin T2 (P1.0) the baud rate is:

\[
\text{Baud Rate} = \frac{\text{Timer 2 Overflow Rate}}{16} \times \frac{1}{\text{Oscillator Frequency}}
\]

And if it is being clocked internally, the baud rate is:

\[
\text{Baud Rate} = \frac{32}{(\text{RCAP2H}, \text{RCAP2L})} \times \frac{1}{\text{Oscillator Frequency}}
\]

To obtain the reload values for RCAP2H and RCAP2L, the above equation can be rewritten as:

\[
\text{RCAP2H, RCAP2L} = \frac{32 \times \text{Baud Rate}}{(\text{Oscillator Frequency})}
\]
activations are skipped during each access to external data

Table 1. 8XC52 Special Function Registers

| SYMBOL | DESCRIPTION | DIRECT ADDRESS | BIT ADDRESS, SYMBOL, OR ALTERNATIVE PORT FUNCTION | MSB | LSB | MSB | LSB | MSB | LSB | MSB | LSB |
|--------|-------------|----------------|--------------------------------------------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| ACC*   | Accumulator | E6H            | E7 E6 E5 E4 E3 E2 E1 E0                     | 0   | 3   | 0   | 3   | 0   | 3   | 0   | 3   |
| B*     | Bit register| F7H            | F8 F7 F6 F5 F4 F3 F2 F1 F0                     | 0   | 3   | 0   | 3   | 0   | 3   | 0   | 3   |
| DPTR   | Data pointer| 83H            | AF AE AD AC AB AA All                        | 0   | 3   | 0   | 3   | 0   | 3   | 0   | 3   |
| SFRH   | Data pointer high | 84H | AF AE AD AC AB AA All | 0   | 3   | 0   | 3   | 0   | 3   | 0   | 3   |
| SFRL   | Data pointer low | 85H | AF AE AD AC AB AA All | 0   | 3   | 0   | 3   | 0   | 3   | 0   | 3   |
| IE*    | Interrupt enable | B6H | BF BE BD BC BB BA B0 | 0   | 3   | 0   | 3   | 0   | 3   |
| IP*    | Interrupt priority | B7H | B7 B6 B5 B4 B3 B2 B1 B0 | 0   | 3   | 0   | 3   | 0   | 3   |

**Example**: Port 0 is an open-drain, bidirectional I/O port. Port 0 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 0 pins that are externally pulled low will source current because of the strong internal pull-ups. During accesses to external data memory that use 8-bit addresses (MOV @Ri), port 0 emits the contents of the P0 special function register.
DIFFERENCES FROM THE 80C51

Special Function Registers

The special function register space is the same as the 80C51 except that the 80C32/87C52 contains the additional special function registers T2CON, RCAP2L, RCAP2H, TL2, and TH2. Since the standard 80C51 on-chip functions are identical in the 80C52, the SPH locations, bit locations, and operation are likewise identical. The only exceptions are in the interrupt mode and interrupt priority SPH (see Table 1).

Timer/Counters

In addition to timer/counter 0 and 1 of the 80C51, the 80C32/87C52 contains timer/counter 2. Like timers 0 and 1, timer 2 can operate as either an event timer or as an event counter. This is selected by bit C/T2 in the special function register T2CON (see Figure 1). It has three operating modes: capture, auto-reload, and baud rate generator, which are selected by bits in the T2CON as shown in Table 2.

In the Capture Mode there are two options which are selected by bit EXEN2 in T2CON. If EXEN2 = 0, then Timer 2 is a 16-bit timer or counter which upon overflowing sets bit TF2, the Timer 2 overflow bit, which can be used to generate an interrupt. If EXEN2 = 1, then Timer 2 still does the above, but with the added feature that a 1-to-0 transition at an external input T2EX will also trigger the 16-bit reload and set EXF2. This auto-reload mode is illustrated in Figure 3. The baud rate generation mode is selected by RCLK = 1 and/or TCLK = 1. It will be described in conjunction with the serial port.

Serial Port

The serial port of the 80C52 is identical to that of the 80C51 except that counter/timer 2 can be used as a baud rate generator by setting TCLK and/or RCLK in T2CON (see Figure 1). Note that the baud rate for transmit and receive can be simultaneously different. Setting RCLK and/or TCLK puts Timer 2 into its baud rate generator mode, as shown in Figure 4. The baud rate generation mode is similar to the auto-reload mode, in that a rollover in Timer 2 will cause the Timer 2 registers to be reloaded with the 16-bit value in registers RCAP2H and RCAP2L, which are preset by software.

Now, the baud rates in Modes 1 and 3 are determined by Timer 2’s overflow rate as follows:

Modes 1, 3 Baud Rate Timer 2 Overflow Rate

<table>
<thead>
<tr>
<th>Mode</th>
<th>Oscillator Frequency</th>
<th>Baud Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>32 MHz (RCAP2H, RCAP2L)</td>
<td>16 bits</td>
</tr>
<tr>
<td>3</td>
<td>16 MHz (RCAP2H, RCAP2L)</td>
<td>32 bits</td>
</tr>
</tbody>
</table>

Note that in both these modes, the baud rate is dependent on the oscillator frequency and the content of RCAP2H and RCAP2L. In Mode 1, the baud rate is always twice the oscillator frequency, whereas in Mode 3, the baud rate is half the oscillator frequency. In that case, the baud rate is given by the formula:

\[
\text{Baud Rate} = \frac{\text{Oscillator Frequency}}{2 \times (\text{RCAP2H} + \text{RCAP2L})}
\]

Table 1. Symbol Position, Name, and Significance

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Position</th>
<th>Name</th>
<th>Significance</th>
</tr>
</thead>
<tbody>
<tr>
<td>TF2</td>
<td>T2CON.7</td>
<td>Timer 2 overflow flag set when a Timer 2 overflow occurs and can be cleared by software. TF2 will not be set in the next rollover if RCLK = 1.</td>
<td></td>
</tr>
<tr>
<td>EXF2</td>
<td>T2CON.6</td>
<td>Timer 2 external flag set when either a capture or reload is caused by a negative transition on T2EX and EXEN2 = 1. When Timer 2 overflow occurs while EXF2 = 0, the CPU does not interrupt the Timer 2 overflow event.</td>
<td></td>
</tr>
<tr>
<td>RCLK</td>
<td>T2CON.5</td>
<td>Receive clock flag. When set, causes the CPU to use Timer 2 overflow pulses for the receive clock in modes 1 and 3. RCLK = 0 causes Timer 2 overflows to be used for the receive clock.</td>
<td></td>
</tr>
<tr>
<td>TCLK</td>
<td>T2CON.4</td>
<td>Transmit clock flag. When set, causes the CPU to use Timer 2 overflow pulses for the transmit clock in modes 1 and 3. TCLK = 0 causes Timer 1 overflows to be used for the transmit clock.</td>
<td></td>
</tr>
<tr>
<td>EXEN2</td>
<td>T2CON.3</td>
<td>Timer 2 external enable flag. When set, allows Timer 2 to operate as a timer when Timer 2 rollover flag (TF2) is not set. EXEN2 = 1 causes Timer 2 to operate as a timer.</td>
<td></td>
</tr>
<tr>
<td>TR2</td>
<td>T2CON.2</td>
<td>Start/stop control for Timer 2. A logic 1 starts the timer.</td>
<td></td>
</tr>
<tr>
<td>C/T2</td>
<td>T2CON.1</td>
<td>Capture/Reload flag. When set, Timer 2 will capture on negative transitions at T2EX and auto-reload on Timer 2 rollover.</td>
<td></td>
</tr>
<tr>
<td>GPC</td>
<td>T2CON.0</td>
<td>Capture/Reload flag. When set, Timer 2 will capture on negative transitions at T2EX and auto-reload on Timer 2 rollover.</td>
<td></td>
</tr>
</tbody>
</table>

Figure 1. Timer/Counter 2 (T2CON) Control Register

Figure 2. Timer 2 in Capture Mode

Figure 3. Timer 2 in Auto-Reload Mode

Figure 4. Timer 2 in Baud Rate Mode
Timer 2 as a baud rate generator is shown in Figure 4. This figure is valid only if RCLK + TCLK = 1 in T2CON. Note that a rollover in TH2 does not set TF2, and will not generate an interrupt. Therefore, the Timer 2 interrupt does not have to be disabled when Timer 2 is in the baud rate generator mode. Note too, that if EXEN2 is set, a 1-to-0 transition in T2EX will set EXF2 but will not cause a reload from (RCAP2H, RCAP2L) to (TH2, TL2). Thus when Timer 2 is in use as a baud rate generator, T2EX can be used as an extra external interrupt, if desired.

It should be noted that when Timer 2 is running (TR2 = 1) in "timer" function in the baud rate generator mode, one should not try to read or write TH2 or TL2. Under these conditions, the timer is being incremented every state time, and the results of a read or write may not be accurate. The RCAP registers may be read, but should not be written to, because a write might overlap a reload and cause unwanted reload errors. Turn the timer off (clear TR2) before accessing the Timer 2 or RCAP registers, in this case.

Exception/Counter 2 Set-up

Except for the baud rate generator mode, the value given for T2CON do not include the setting of the TR2 bit. Therefore, bit TR2 must be set separately to turn the timer on. See Table 3 for setup of timer 2 as a counter.

Using Timer/Counter 2 to Generate Baud Rates

For this purpose, Timer 2 must be used in the baud rate generating mode. If Timer 2 is being clocked through pin T2 (P1.1)pin except when Timer 2 is used in the baud rate generator mode.

The 80C32/87C52 has 6 interrupt sources. All except TF2 and EXF2 are identical sources to those in the 80C51.

Interruptions

The Interrupts

The Interrupt Enable Register and the Interrupt Priority Register are modified to include the additional 80C32/87C52 interrupt sources.

The operation of these registers is identical to the 80C51.

In the 80C32/87C52, the Timer 2 Interrupt is generated by the logical OR of TF2 and EXF2. Neither of these flags is cleared by hardware when the service routine is entered. In fact, the service routine may have to determine whether it was TF2 or EXF2 that generated the interrupt, and the bit will have to be cleared in software.

All of the bits that generate interrupts can be set or cleared by software with the same result as though it has been set or cleared by hardware. That is, interrupts can be generated and pending interrupts can be cleared in software.

The interrupt vector addresses and the interrupt priority for requests in the same priority level are given in the following:

<table>
<thead>
<tr>
<th>Source</th>
<th>Vector Address</th>
<th>Priority Within Line</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. IE0</td>
<td>0003H</td>
<td>(highest)</td>
</tr>
<tr>
<td>2. TF0</td>
<td>0005H</td>
<td></td>
</tr>
<tr>
<td>3. IE1</td>
<td>0013H</td>
<td></td>
</tr>
<tr>
<td>4. TF1</td>
<td>0015H</td>
<td></td>
</tr>
<tr>
<td>5. T2 + TF</td>
<td>0033H</td>
<td></td>
</tr>
<tr>
<td>6. TF + EXF2</td>
<td>0035H</td>
<td>(lowest)</td>
</tr>
</tbody>
</table>

Note that they are identical to those in the 80C51 except for the addition of the Timer 2 (TF2 and EXF2) interrupt at 0035H and at the lowest priority within a level.

Table 3. Timer 2 as a Timer

<table>
<thead>
<tr>
<th>MODE</th>
<th>T2CON</th>
</tr>
</thead>
<tbody>
<tr>
<td>16-bit Auto-Reload</td>
<td>00H</td>
</tr>
<tr>
<td>16-bit Capture</td>
<td>01H</td>
</tr>
<tr>
<td>Baud rate generator receive and transmit same baud rate</td>
<td>34H</td>
</tr>
<tr>
<td>Transmit only</td>
<td>14H</td>
</tr>
</tbody>
</table>

Table 4. Timer 2 as a Counter

<table>
<thead>
<tr>
<th>MODE</th>
<th>TMOD</th>
</tr>
</thead>
<tbody>
<tr>
<td>16-bit</td>
<td>02H</td>
</tr>
<tr>
<td>Auto-Reload</td>
<td>03H</td>
</tr>
</tbody>
</table>

NOTES:

1. Capture/Reload occurs only on timer/counter overflow.
2. Capture/Reload occurs on timer/counter overflow and a 1-to-0 transition on T2EX (P1.1) pin except when Timer 2 is used in the baud rate generator mode.

To obtain the reload value for RCAP2H and RCAP2L, the above equation can be rewritten as:

RCAP2H, RCAP2L = 32 * Oscillator Frequency / Baud Rate
OSCILLATOR CHARACTERISTICS

XTAL1 and XTAL2 are the input and output, respectively, of an inverting amplifier. The pins can be configured for use as an on-chip oscillator, as shown in the Logic Symbol, page 4. To drive the device from an external clock source, XTAL1 should be driven while XTAL2 is left unconnected. There are no requirements on the duty cycle of the external clock signal, because the input to the internal clock circuitry is through a divide-by-two flip-flop. However, minimum and maximum high and low times specified in the data sheet must be observed.

RESET

A reset is accomplished by holding the RST pin high for at least two machine cycles (24 oscillator periods), while the oscillator is running. To insure a good power-up reset, the RST pin must be high long enough to allow the oscillator time to start up (normally a few milliseconds) plus two machine cycles.

IDLE MODE

In idle mode, the CPU puts itself to sleep while all of the on-chip peripherals stay active. The instruction to invoke the idle mode is the last instruction executed in the normal operating mode before the idle mode is activated. The CPU contents, the on-chip RAM, and all of the special function registers remain intact during this mode. The idle mode can be terminated either by any enabled interrupt (at which time the process is picked up at the interrupt service routine and continued) or by a hardware reset which restarts the processor in the same manner as a power-on reset.

POWER-DOWN MODE

In the power-down mode, the oscillator is stopped and the instruction to invoke power-down is the last instruction executed. Only the contents of the on-chip RAM are preserved. A hardware reset is the only way to terminate the power-down mode; the control bits for the reduced power modes are in the special function register PCON.

DESIGN CONSIDERATIONS

At power-on, the voltage on VCC and RST must come up at the same time for a proper start-up. Table 5 shows the state of I/O ports during low current operating modes. As a precaution to coming out of an unexpected power down, INT0 and INT1 should be disabled prior to entering a power down.

<table>
<thead>
<tr>
<th>MODE</th>
<th>PROGRAM MEMORY</th>
<th>ALE</th>
<th>TEST</th>
<th>PORT 0</th>
<th>PORT 1</th>
<th>PORT 2</th>
<th>PORT 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Idle</td>
<td>Internal</td>
<td>1</td>
<td>1</td>
<td>Data</td>
<td>Data</td>
<td>Data</td>
<td>Data</td>
</tr>
<tr>
<td>Idle</td>
<td>Internal</td>
<td>1</td>
<td>1</td>
<td>Float</td>
<td>Address</td>
<td>Data</td>
<td>Data</td>
</tr>
<tr>
<td>Power-on</td>
<td>Internal</td>
<td>0</td>
<td>0</td>
<td>Data</td>
<td>Data</td>
<td>Data</td>
<td>Data</td>
</tr>
<tr>
<td>Power-on</td>
<td>External</td>
<td>0</td>
<td>0</td>
<td>Float</td>
<td>Data</td>
<td>Data</td>
<td>Data</td>
</tr>
</tbody>
</table>
**FEATURES**

- **80C32–Compatible**
  - Pin–compatible
  - Standard 8051 instruction set
  - Four 8–bit I/O ports
  - Three 16–bit timer/counters
  - 256 bytes scratchpad RAM
  - Multiplexed address/data bus
  - Addresses 64KB ROM and 64KB RAM

- **High–speed architecture**
  - 4 clocks/machine cycle (8032=12)
  - Wasted cycles removed
  - Runs DC to 33 MHz clock rates
  - Single–cycle instruction in 121 ns
  - Uses less power for equivalent work
  - Dual data pointer
  - Optional variable length MOVX to access fast/slow RAM/peripherals

- **High integration controller includes:**
  - Power–fail reset
  - Programmable Watchdog timer
  - Early–warning power–fail interrupt

- **Two full–duplex hardware serial ports**

- **13 total interrupt sources with six external**

- **Available in 40–pin DIP, 44–pin PLCC and TQFP**

**DESCRIPTION**

The DS80C320 is a fast 80C31/80C32–compatible microcontroller. Wasted clock and memory cycles have been removed using a redesigned processor core. As a result, every 8051 instruction is executed between 1.5 and 3 times faster than the original for the same crystal speed. Typical applications will see a speed improvement of 2.5 times using the same code and same crystal. The DS80C320 offers a maximum crystal rate of 33 MHz, resulting in apparent execution speeds of 82.5 MHz (approximately 2.5X).
The DS80C320 is pin compatible with all three packages of the standard 80C32 and offers the same timer/counters, serial port, and I/O ports. In short, the DS80C320 is extremely familiar to 8051 users but provides the speed of a 16-bit processor.

The DS80C320 provides several extras in addition to greater speed. These include a second full hardware serial port, seven additional interrupts, programmable watchdog timer, power-fail interrupt and reset. The DS80C320 also provides dual data pointers (DPTRs) to speed I/O data memory moves. It can also adjust the speed of off-chip data memory access to between two and nine machine cycles for flexibility in selecting memory and peripherals.

**ORDERING INFORMATION**

<table>
<thead>
<tr>
<th>PART NUMBER</th>
<th>PACKAGE</th>
<th>MAX CLOCK SPEED</th>
<th>TEMPERATURE RANGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>DS80C320–MC</td>
<td>40-pin plastic DIP</td>
<td>25 MHz</td>
<td>0°C to +70°C</td>
</tr>
<tr>
<td>DS80C320–QC</td>
<td>44-pin PLCC</td>
<td>25 MHz</td>
<td>0°C to +70°C</td>
</tr>
<tr>
<td>DS80C320–EC</td>
<td>44-pin TQFP</td>
<td>25 MHz</td>
<td>0°C to +70°C</td>
</tr>
<tr>
<td>DS80C320–NG</td>
<td>40-pin plastic DIP</td>
<td>25 MHz</td>
<td>-40°C to +85°C</td>
</tr>
<tr>
<td>DS80C320–QG</td>
<td>44-pin PLCC</td>
<td>25 MHz</td>
<td>-40°C to +85°C</td>
</tr>
<tr>
<td>DS80C320–EN</td>
<td>44-pin TQFP</td>
<td>25 MHz</td>
<td>-40°C to +85°C</td>
</tr>
<tr>
<td>DS80C320–MC</td>
<td>40-pin plastic DIP</td>
<td>33 MHz</td>
<td>0°C to +70°C</td>
</tr>
<tr>
<td>DS80C320–EC</td>
<td>44-pin TQFP</td>
<td>33 MHz</td>
<td>0°C to +70°C</td>
</tr>
<tr>
<td>DS80C320–MN</td>
<td>40-pin plastic DIP</td>
<td>33 MHz</td>
<td>-40°C to +85°C</td>
</tr>
<tr>
<td>DS80C320–QN</td>
<td>44-pin PLCC</td>
<td>33 MHz</td>
<td>-40°C to +85°C</td>
</tr>
<tr>
<td>DS80C320–EN</td>
<td>44-pin TQFP</td>
<td>33 MHz</td>
<td>-40°C to +85°C</td>
</tr>
</tbody>
</table>

**HIGH-SPEED OPERATION**

The DS80C320 is built around a high-speed 80C32 compatible core. Higher speed comes not just from increasing the clock frequency, but from a newer, more efficient design.

In this updated core, dummy memory cycles have been eliminated. In a conventional 80C32, machine cycles are generated by dividing the clock frequency by 12. In the DS80C320, the same machine cycle is performed in 4 clocks. Thus the fastest instruction, 1 machine cycle, is executed three times faster for the same crystal frequency. Note that these are identical instructions. A comparison of the timing differences is shown in Figure 2. The majority of instructions on the DS80C320 will see the full 3 to 1 speed improvement. Some instructions will get between 1.5 and 2.4 X improvement. Note that all instructions are faster than the original 80C51. Table 2 below shows a summary of the instruction set including the speed.

The numerical analysis of all opcodes is approximately a 2.5 to 1 speed improvement. Individual programs will be affected differently, depending on the actual instructions used. Speed sensitive applications would make the most use of instructions that are three times faster. However, the sheer number of 3 to 1 improved opcodes makes dramatic speed improvements likely for any code. When these architecture improvements are combined with 0.8 μm CMOS, the result is a single cycle instruction execution in 160 ns. The Dual Data Pointer feature also allows the user to eliminate wasted instructions when moving blocks of memory.

**INSTRUCTION SET SUMMARY**

All instructions in the DS80C320 perform the same functions as their 80C32 counterparts. Their affect on bits, flags, and other status functions is identical. However, the timing of each instruction is different. This applies both in absolute and relative number of clocks.

For absolute timing of real-time events, the timing of software loops will need to be calculated using the table below. However, counter/timers default to run at the older 12 clocks per increment. Therefore, while software runs at higher speed, timer-based events need no modification to operate as before. Timers can be set to run at 4 clocks per increment cycle to take advantage of higher speed operation.

The relative time of two instructions might be different in the new architecture than it was previously. For example, in the original architecture, the "MOV A, @DPTR" instruction and the "MOV direct, direct" instruction executes in two machine cycles or 24 oscillator cycles. Therefore, they required the same amount of time. In the DS80C320, the MOVX instruction can be done in two machine cycles or eight oscillator cycles but the "MOV direct, direct" uses three machine cycles or 12 oscillator cycles. While both are faster than their original counterparts, they now have different execution times from each other. This is because in most cases, the DS80C320 uses one cycle per byte. The user concerned with precise program timing should examine the timing of each instruction for familiarity with the changes. Note that a machine cycle now requires just four clocks, and provides one ALE pulse per cycle. Many instructions require only one cycle, but some require five. In the original architecture, all were one or two cycles except for MUL and DIV.
### Arithmetic Instructions:

- **ADD A, Rn**
- **ADD A, direct**
- **ADD A, @Ri**
- **ADD A, #data**
- **ADDC A, Rn**
- **ADDC A, direct**
- **ADDC A, @Ri**
- **ADDC A, #data**
- **SUBB A, Rn**
- **SUBB A, direct**
- **SUBB A, @Ri**
- **SUBB A, #data**

### Logical Instructions:

- **ANL A, Rn**
- **ANL A, direct**
- **ANL A, @Ri**
- **ANL A, #data**
- **ANL direct, A**
- **ANL direct, #data**
- **ORL A, Rn**
- **ORL A, direct**
- **ORL A, @Ri**
- **ORL A, #data**
- **ORL direct, A**
- **ORL direct, #data**

### Data Transfer Instructions:

- **MOV A, Rn**
- **MOV A, direct**
- **MOV A, @Ri**
- **MOV A, #data**
- **MOV Rn, A**
- **MOV Rn, direct**
- **MOV Rn, @Ri**
- **MOV Rn, #data**
- **MOV direct, Rn**
- **MOV direct, direct**
- **MOV direct, @Ri**
- **MOV direct, #data**
- **MOV @Ri, A**
- **MOV @Ri, direct**
- **MOV @Ri, @Ri**
- **MOV @Ri, #data**
- **MOV @Ri, #data**
- **MOV DPTR, #data**

### Bit Manipulation Instructions:

- **CL R C**
- **CLR bit**
- **SETB C**
- **SETB bit**
- **CLR C**
- **CLR bit**
- **MOV C, bit**
- **MOV bit, C**

### Program Branching Instructions:

- **ACALL addr**
- **LCALL addr**
- **RET**
- **RETI**
- **AJMP addr**
- **LJMP addr**
- **JB bit, rel**
- **JNC rel**
- **JNZ rel**
- **DJNZ Rn, rel**
- **DJNZ direct, rel**

### Memory Access

- The DS80C320 contains no on-chip ROM and 256 bytes of scratchpad RAM. Off-chip memory is accessed using the multiplexed address/data bus on P0 and the MSB address on P2. A typical memory connection is shown in Figure 3. Timing diagrams are provided in the Electrical Specifications. Program memory (ROM) is accessed at a fixed rate determined by the crystal frequency and the actual instructions. As mentioned above, an instruction cycle requires four clocks.

### Speed Advantage Summary

<table>
<thead>
<tr>
<th>Opcodes</th>
<th>Speed Improvement</th>
</tr>
</thead>
<tbody>
<tr>
<td>159</td>
<td>3.0 x</td>
</tr>
<tr>
<td>51</td>
<td>1.5 x</td>
</tr>
<tr>
<td>43</td>
<td>2.0 x</td>
</tr>
<tr>
<td>...</td>
<td>2.4 x</td>
</tr>
<tr>
<td>255</td>
<td>Average: 2.5</td>
</tr>
</tbody>
</table>

*User Selectable*
64 BYTE BLOCK MOVE WITH DUAL DATA POINTER

; DH and DL are high and low byte source address.
; DS is the data pointer select. Reset condition is DS=0. DPTR0 is selected.

EQU DPS, #86h ; TELL ASSEMBLER ABOUT DPS
MOV R5, #64 ; NUMBER OF BYTES TO MOVE
MOV DPTR, #DHDL ; LOAD DESTINATION ADDRESS
INC DPS ; CHANGE ACTIVE DPTR
MOV DPTR, #SHSL ; LOAD SOURCE ADDRESS

MOVE: ; THIS LOOP IS PERFORMED THE NUMBER OF TIMES LOADED INTO R5, IN THIS EXAMPLE 64
MOV R5, #64d ; NUMBER OF BYTES TO MOVE

END:

64 BYTE BLOCK MOVE WITHOUT DUAL DATA POINTER

; DH and DL are high and low byte source address.
; DS is the data pointer select. Reset condition is DS=0. DPTR0 is selected.

EQU DPS, #86h ; TELL ASSEMBLER ABOUT DPS
MOV R5, #64 ; NUMBER OF BYTES TO MOVE
MOV DPTR, #DHDL ; LOAD DESTINATION ADDRESS
INC DPS ; CHANGE ACTIVE DPTR
MOV DPTR, #SHSL ; LOAD SOURCE ADDRESS

MOVE: ; THIS LOOP IS PERFORMED THE NUMBER OF TIMES LOADED INTO R5, IN THIS EXAMPLE 64
MOV A, @DPTR ; READ SOURCE DATA BYTE
MOV R1, DPL ; SAVE NEW SOURCE POINTER
MOV R2, DPH ; SAVE HIGH BYTE OF DESTINATION

END:
scheme on a reset. This allows existing code with real-time dependencies such as baud rates to operate properly. If an application needs higher speed timers or serial baud rates, the timers can be set to run at the 4 clock rate.

The Clock Control register (CKCON – 8Eh) determines these timer speeds. When the relevant CKCON bit is a logic 1, the DS80C320 uses 4 clocks per cycle to generate timer speeds. When the control bit is set to a 0, the DS80C320 uses 12 clocks for timer speeds. The reset condition is a 0. CKCON.5 selects the speed of Timer2, CKCON.4 selects Timer 1 and CKCON.3 selects Timer 0. Note that unless a user desires very fast timing, it is unnecessary to alter these bits. Note that the timer controls are independent.

POWER FAIL RESET

The DS80C320 incorporates a precision band-gap voltage reference to determine when VCC is out-of-tolerance. While powering up, internal circuits will hold the DS80C320 in a reset state until VCC rises above the VREF reset threshold. Once VCC is above this level, the oscillator will begin running. An internal reset circuit will then count 65536 clocks to allow time for power and the oscillator to stabilize. The microcontroller will then exit the reset condition. No external components are needed to generate a power on reset. During power down or during a severe power glitch, as VCC falls below VREF the microcontroller will also generate its own reset. It will hold the reset condition as long as power remains below the threshold. This reset will occur automatically, needing no action from the user or from the software. Refer to the Electrical Specifications for the exact value of VREF.

POWER FAIL INTERRUPT

The same reference that generates a precision reset threshold can also generate an optional early warning Power-fail Interrupt (PFI). When enabled by the application software, this interrupt always has the highest priority. On detecting that the VCC has dropped below VREF and that the PFI is enabled, the processor will vector to ROM address 0033h. The PFI enable is located in the Watchdog Control SFR (WDCON – D8h). Setting WDCON.5 to a logic one will enable the PFI. The application software can also read a flag at WDCON.4. This bit is set when a PFI condition has occurred. The flags are independent of the interrupt enable and software must manually clear it.

WATCHDOG TIMER

For applications that cannot afford to run out-of-control, the DS80C320 incorporates a programmable Watchdog Timer circuit. Reset the uC if software fails to reset the Watchdog before the selected time period has elapsed. The user selects one of four time-out values. After enabling the Watchdog, software must reset the timer prior to expiration of the interval, or the CPU will be reset. Both the Watchdog Enable and the Watchdog Reset bit are protected by a "Timed Access" circuit. This prevents accidentally clearing the Watchdog. Time-out values are precise since they are related to the crystal frequency as shown in Table 4. For reference, the time periods at 25MHz are also shown.

The DS80C320 Watchdog also provides a useful option for systems that may not require a reset. If enabled, then 512 clocks before giving a reset, the Watchdog will give an interrupt. The interrupt can also serve as a convenient time-base generator, or be used to wake-up the processor from Idle mode. The Watchdog function is controlled in the Clock Control (CKCON – 8Eh), Watchdog Control (WDCON – D8h), and Extended Interrupt Enable (EIE – E8h) SFRs. CKCON.7 and CKCON.6 are called WD1 and WD0 respectively and are used to select the Watchdog time-out period as shown in Table 4.

As shown above, the Watchdog Timer uses the crystal frequency as a time base. A user selects one of four counter values to determine the time-out. These clock counter lengths are $2^{17} = 131,072$ clocks; $2^{20} = 1,048,576$; $2^{23} = 8,388,608$ clocks; or $2^{26} = 67,108,864$ clocks. The times shown in Table 4 above are with a 25 MHz crystal frequency. Note that once the counter chain has expired, the natural precedence given below determines which is a acted upon. Except for the PFI, all interrupts that are new to the 8051 family have a lower priority than the originals.

INTERRUPTS

The DS80C320 provides 13 sources of interrupt with three priority levels. The Power-fail Interrupt (PFI), if enabled, always has the highest priority. There are two remaining user selectable priorities: high and low. If two interrupts that have the same priority occur simultaneously, the natural precedence given below determines which is a acted upon. Except for the PFI, all interrupts that are new to the 8051 family have a lower natural priority than the original.

POWER MANAGEMENT

The DS80C320 provides the standard Idle and power down (Stop) that are available on the standard 80C32. However, the DS80C320 has enhancements that make these modes more useful, and allow more power saving.

The Idle mode is invoked by setting the LS1B of the Power Control register (POCON – 87h). Idle will leave internal clocks, serial port and timer running. No memory access will be performed so power is dramatically reduced. Since clocks are running, the Idle power consumption is related to crystal frequency. It should be approximately 1/2 of the operational power. The CPU can exit the Idle state with any interrupt or a reset.

The power-down or Stop modes are invoked by setting the PCON.1 bit. Stop mode is a lower power state than Idle since it turns off all internal clocking. The PCON.1 bit is specified in the Electrical Specifications. The CPU will exit Stop mode from an external interrupt or a reset condition.
Note that internally generated interrupts (timer, serial port, watchdog) are not useful since they require clocking activity.

**IDLE MODE ENHANCEMENTS**

A simple enhancement to idle mode makes it substantially more useful. The innovation involves not the idle mode itself, but the watchdog timer. As mentioned above, the watchdog timer provides an optional interrupt capability. This interrupt can provide a periodic interval timer to bring the DS80C320 out of idle mode. This can be useful even if the watchdog is not normally used. By enabling the watchdog timer and its interrupt prior to entering idle, a user can periodically come out of idle perform an operation, then return to idle until the next operation. This will lower the overall power consumption. When using the watchdog interrupt to cancel the idle state, make sure to restart the Watchdog Timer or it will cause a reset.

**STOP MODE ENHANCEMENTS**

The DS80C320 provides two enhancements to the Stop mode. As documented above, the DS80C320 provides an internal band-gap reference to determine Power-fail Interrupt and Reset thresholds. The default state is that the band-gap reference is off when Stop mode is invoked. This allows the extremely low power state mentioned above. A user can optionally choose to have the band-gap enabled during Stop mode. This means that PFI and power-fail reset will be activated and are valid means for leaving Stop mode.

In Stop mode with the band-gap on, VDD will be approximately 50 µA compared with 1 µA with the band-gap off. If a user does not require a Power-fail Reset or Interrupt while in Stop mode, the band-gap can remain turned off. Note that the only power sensitive applications should turn off the band-gap, as this results in an uncontrolled power down condition.

The control of the band-gap reference is located in the Extended Interrupt Flag register (EXIF - 91h). Setting BGS (EXIF.0) to a one will leave the band-gap reference enabled during Stop mode. The default or reset condition is with the bit at a logic 0. This results in the band-gap being turned off during Stop mode. Note that this bit has no control of the reference during full power or idle modes.

The second feature allows an additional power saving option. This is the ability to start instantly when exiting Stop mode. It is accomplished using an internal ring oscillator that can be used when exiting Stop mode in response to an interrupt. The benefit of the ring oscillator is as follows.

Using Stop mode turns off the crystal oscillator and all internal clocks to save power. This requires that the oscillator be restarted when exiting Stop mode. Actual start-up time is crystal dependent, but it is normally at least 4 ms. A common recommendation is 10 ms. In an application that will wake-up, perform a short operation, then return to sleep, the crystal start-up can be longer than the real transaction. However, the ring oscillator will start instantly. The user can perform a simple operation and return to sleep before the crystal has even stabilized. If the ring is used to start and the processor remains running, hardware will automatically switch to the crystal once a power-on reset interval (65536 clocks) has elapsed. This value is used to guarantee stability even though power is not being cycled.

If the user returns to Stop mode prior to switching of crystal, then all clocks will be turned off again. The ring oscillator runs at approximately 4 MHz but will not be a precision value. No real-time precision operations (including serial communication) should be conducted during this ring period. Figure 7 shows how the operation would compare when using the ring, and when starting up normally. The default states is to come out of Stop mode without using the ring oscillator.

This function is controlled using the RGSL – Ring Select bit at EXIF.1 (EXIF - 91h). When EXIF.1 is set, the ring oscillator will be used to come out of Stop mode quickly. As mentioned above, the processor will automatically switch from the ring to the crystal after a delay of 65536 crystal clocks. For a 3.57 MHz crystal, this is approximately 18 ms. The processor sets a flag called RGMID – Ring Mode to tell software that the ring is being used. This bit at EXIF.2 will be a logic 1 when the ring is in use. No serial communication or precision timing should be attempted while this bit is set, since the operating frequency is not precise.

The Timed Access procedure prevents an errant write of a protected bit be preceded by the following instructions:

1. MOV 0C7h, #0AAh
2. MOV 0C7h, #55h

By writing an AAh followed by a 55h to the Timed Access register (location C7h), the hardware opens a two cycle window that allows software to modify one of the protected bits. If the instruction that sets or modifies the protected bit is not immediately preceded by these instructions, the write will not take effect. The protected bits are:

- EXIF.0
- BGS Band-gap Select
- WDCON.6 POR Power-on Reset flag
- WDCON.1 EWT Enable Watchdog
- WDCON.0 RWTS Reset Watchdog
- WDCON.3 WDIF Watchdog Interrupt Flag

**SPECIAL FUNCTION REGISTERS**

Most special features of the DS80C320 or 80C32 are controlled by bits in special function registers (SFRs). This allows the DS80C320 to add many features but use the same instruction set. When writing software to use a new feature, the SFR must be defined to an assembler or compiler using an equate statement. This is the only change needed to access the new feature. The DS80C320 duplicates the SFRs that are contained in the standard 80C32. Table 6 shows the register addresses and bit locations. Many are standard 80C32 registers. The High-Speed Microcontroller User's Guide describes all SFRs.
### SPECIAL FUNCTION REGISTER LOCATIONS

**Table 6**

<table>
<thead>
<tr>
<th>REGISTER</th>
<th>BIT 7</th>
<th>BIT 6</th>
<th>BIT 5</th>
<th>BIT 4</th>
<th>BIT 3</th>
<th>BIT 2</th>
<th>BIT 1</th>
<th>BIT 0</th>
<th>ADDRESS</th>
</tr>
</thead>
<tbody>
<tr>
<td>SP</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>81h</td>
</tr>
<tr>
<td>DPL</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>82h</td>
</tr>
<tr>
<td>DPH</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>83h</td>
</tr>
<tr>
<td>DPLU</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>84h</td>
</tr>
<tr>
<td>DPHR</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>85h</td>
</tr>
<tr>
<td>DP0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>SEL</td>
</tr>
<tr>
<td>PCON</td>
<td>SMOD_0</td>
<td>SMOD0</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>TC0N</td>
<td>TF1</td>
<td>TR1</td>
<td>TR0</td>
<td>IE1</td>
<td>T11</td>
<td>IE0</td>
<td>T00</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>TMOD</td>
<td>GATE</td>
<td>C7</td>
<td>M7</td>
<td>M0</td>
<td>GATE</td>
<td>C7</td>
<td>M1</td>
<td>M0</td>
<td>89h</td>
</tr>
<tr>
<td>TLS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>8Ah</td>
</tr>
<tr>
<td>TL1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>8Bh</td>
</tr>
<tr>
<td>TH0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>8Ch</td>
</tr>
<tr>
<td>TH1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>8Dh</td>
</tr>
<tr>
<td>OKCON</td>
<td>WD1</td>
<td>WD0</td>
<td>TM2</td>
<td>TM1</td>
<td>TM0</td>
<td>MD2</td>
<td>MD1</td>
<td>MD0</td>
<td>8Eh</td>
</tr>
<tr>
<td>P1</td>
<td>P1.7</td>
<td>P1.6</td>
<td>P1.5</td>
<td>P1.4</td>
<td>P1.3</td>
<td>P1.2</td>
<td>P1.1</td>
<td>P1.0</td>
<td>90h</td>
</tr>
<tr>
<td>ECPF</td>
<td>IES</td>
<td>IE4</td>
<td>IE3</td>
<td>IE2</td>
<td>–</td>
<td>RGM3</td>
<td>RSSL</td>
<td>BGS</td>
<td>91h</td>
</tr>
<tr>
<td>SCON0</td>
<td>SM2FE_0</td>
<td>SM2_0</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>SIUF0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>98h</td>
</tr>
<tr>
<td>P2</td>
<td>P2.0</td>
<td>P2.6</td>
<td>P2.5</td>
<td>P2.4</td>
<td>P2.3</td>
<td>P2.2</td>
<td>P2.1</td>
<td>P2.0</td>
<td>AOh</td>
</tr>
<tr>
<td>E</td>
<td>E1A</td>
<td>E1S</td>
<td>ET2</td>
<td>ES0</td>
<td>ET1</td>
<td>EX1</td>
<td>E10</td>
<td>EX0</td>
<td>AOh</td>
</tr>
<tr>
<td>SAD0R0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>A0h</td>
</tr>
<tr>
<td>SAD0R1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>A1h</td>
</tr>
<tr>
<td>SAD0E0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>99h</td>
</tr>
<tr>
<td>SAD0E1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>9Ah</td>
</tr>
<tr>
<td>SCONF1</td>
<td>SM1FE_0</td>
<td>SM1_0</td>
<td>SM2_0</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>SBUF1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>9Ch</td>
</tr>
<tr>
<td>STATUS</td>
<td>PIP</td>
<td>MIP</td>
<td>UP</td>
<td>T</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>C0h</td>
</tr>
<tr>
<td>TA</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>C0h</td>
</tr>
<tr>
<td>T2CON</td>
<td>TF2</td>
<td>EXF2</td>
<td>RC10</td>
<td>T2L</td>
<td>EXE12</td>
<td>T2R2</td>
<td>C17</td>
<td>C1R1</td>
<td>C1h</td>
</tr>
<tr>
<td>T2MOD</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>T2C0P2L</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>C0h</td>
</tr>
<tr>
<td>T2C0P2H</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>C0h</td>
</tr>
<tr>
<td>T2L</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>C0h</td>
</tr>
<tr>
<td>TH0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>C0h</td>
</tr>
<tr>
<td>PSW</td>
<td>CY</td>
<td>AC</td>
<td>F0</td>
<td>RS1</td>
<td>RS0</td>
<td>CV</td>
<td>FL</td>
<td>JP</td>
<td>D0h</td>
</tr>
<tr>
<td>WDCON</td>
<td>SMOD_1</td>
<td>SMD0</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>D1h</td>
</tr>
<tr>
<td>ACC</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>D2h</td>
</tr>
<tr>
<td>B</td>
<td></td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>D3h</td>
</tr>
<tr>
<td>BIP</td>
<td></td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>F0h</td>
</tr>
<tr>
<td>SIE</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>F1h</td>
</tr>
</tbody>
</table>
Memory Organization

The 80C51 has separate address spaces for program and data memory. The Program memory can be up to 64k bytes long. The lower 4k can reside on-chip. Figure 1 shows a map of the 80C51 program memory.

The 80C51 has 64k bytes of data memory to the chip. The MOVX instruction is used to access the external data memory. The 80C51 has 128 bytes of on-chip RAM, plus a number of Special Function Registers (SFRs). The lower 128 bytes of RAM can be accessed either by direct addressing (MOV data addr) or by indirect addressing (MOV QP). Figure 2 shows the Data Memory organization.

Direct and Indirect Address Area

The 128 bytes of RAM which can be accessed by both direct and indirect addressing can be divided into three segments as listed below and shown in Figure 3.

1. Register Banks 0-3: Locations 0 through 1FH (32 bytes). The device after reset defaults to register bank 0. To use the other register banks, the user must select them in software. Each register bank contains eight 1-byte registers 0 through 7. Reset initializes the stack pointer to location 00H, and it is incremented once to start from location 01H, which is the first register (R0) of the second register bank. Thus, in order to use more than one register bank, the SP should be initialized to a different location of the RAM where it is not used for data storage, e.g., the higher part of the RAM.

2. Bit Addressable Area: 16 bytes have been assigned for this segment, 20H-2FH. Each one of the 128 bits of this segment can be directly addressed (0-7FH). The bits can be referred to in two ways, both of which are acceptable by most assemblers. One way is to refer to their address (i.e., 0-7FH). The other way is with reference to bits 00H to 7FH. Thus, bits 0-7 can also be referred to as bits 20.0-20.7, and bits 8-15 are the same as 21.0-21.7, and so on. Each of the 16 bytes in this segment can also be addressed as a byte.

3. Scratch Pad Area: 30H through 7FH are available to the user as data RAM. However, if the stack pointer has been initialized to this area, enough bytes should be left aside to prevent SP data destruction.

Figure 2 shows the different segments of the on-chip RAM.
### Table 1. 80C51 Special Function Registers

<table>
<thead>
<tr>
<th>SYMBOL</th>
<th>DESCRIPTION</th>
<th>DIRECT ADDRESS</th>
<th>BIT ADDRESS, SYMBOL, OR ALTERNATIVE PORT FUNCTION</th>
<th>LSB</th>
<th>RESET VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACC*</td>
<td>Accumulator</td>
<td>E0H</td>
<td>E7  E6  E5  E4  E3  E2  E1  E0</td>
<td>00H</td>
<td>00H</td>
</tr>
<tr>
<td>B*</td>
<td>B register</td>
<td>F0H</td>
<td>F7  F6  F5  F4  F3  F2  F1  F0</td>
<td>00H</td>
<td>00H</td>
</tr>
<tr>
<td>DPTR</td>
<td>Data pointer (2-bytes)</td>
<td>A0H</td>
<td>E1H  SC  A0H  DPH  DPL  A9  A8  A7  A6  A5  A4  A3  A2  A1  A0  P0  P1</td>
<td>00H</td>
<td>00H</td>
</tr>
<tr>
<td>IP*</td>
<td>Interrupt priority</td>
<td>B0H</td>
<td>B9H  B8H  B7H  B6H  B5H  B4H  B3H  B2H  B1H  B0H</td>
<td>00H</td>
<td>00H</td>
</tr>
<tr>
<td>IE*</td>
<td>Interrupt enable</td>
<td>C0H</td>
<td>C9H  C8H  C7H  C6H  C5H  C4H  C3H  C2H  C1H  C0H</td>
<td>00H</td>
<td>00H</td>
</tr>
<tr>
<td>P0*</td>
<td>Port 0</td>
<td>D0H</td>
<td>D9  D8  D7  D6  D5  D4  D3  D2  D1  D0</td>
<td>FFH</td>
<td>FFH</td>
</tr>
<tr>
<td>P1*</td>
<td>Port 1</td>
<td>E0H</td>
<td>E9  E8  E7  E6  E5  E4  E3  E2  E1  E0</td>
<td>FFH</td>
<td>FFH</td>
</tr>
<tr>
<td>P2*</td>
<td>Port 2</td>
<td>F0H</td>
<td>F9  F8  F7  F6  F5  F4  F3  F2  F1  F0</td>
<td>FFH</td>
<td>FFH</td>
</tr>
<tr>
<td>P3*</td>
<td>Port 3</td>
<td>G0H</td>
<td>G9  G8  G7  G6  G5  G4  G3  G2  G1  G0</td>
<td>FFH</td>
<td>FFH</td>
</tr>
<tr>
<td>PCON†</td>
<td>Power control</td>
<td>H0H</td>
<td>H9H  H8H  H7H  H6H  H5H  H4H  H3H  H2H  H1H  H0H</td>
<td>00H</td>
<td>00H</td>
</tr>
<tr>
<td>SCON*</td>
<td>Serial controller</td>
<td>B0H</td>
<td>B9H  B8H  B7H  B6H  B5H  B4H  B3H  B2H  B1H  B0H</td>
<td>00H</td>
<td>00H</td>
</tr>
<tr>
<td>SBUF*</td>
<td>Serial data buffer</td>
<td>D0H</td>
<td>D9H  D8H  D7H  D6H  D5H  D4H  D3H  D2H  D1H  D0H</td>
<td>00H</td>
<td>00H</td>
</tr>
<tr>
<td>SBUF*</td>
<td>Serial data buffer</td>
<td>D0H</td>
<td>D9H  D8H  D7H  D6H  D5H  D4H  D3H  D2H  D1H  D0H</td>
<td>00H</td>
<td>00H</td>
</tr>
</tbody>
</table>

NOTES:  
* Bit addressable  
† Bits G1, G0, P0, and P1 of the PCON register are not implemented on the NMOS 8051/8031.
Those SFRs that have their bits assigned for various functions are listed in this section. A brief description of each bit is provided for quick reference. For more detailed information refer to the Architecture Chapter of this book.

**PSW: PROGRAM STATUS WORD. BIT ADDRESSABLE.**

<table>
<thead>
<tr>
<th>CY</th>
<th>AC</th>
<th>F0</th>
<th>RS1</th>
<th>RS0</th>
<th>OV</th>
<th>–</th>
<th>P</th>
</tr>
</thead>
<tbody>
<tr>
<td>CY</td>
<td>PSW.7 Carry Flag.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AC</td>
<td>PSW.6 Auxiliary Carry Flag.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>F0</td>
<td>PSW.5 Flag 0 available to the user for general purpose.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RS1</td>
<td>PSW.4 Register Bank selector bit 1 (SEE NOTE 1).</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RS0</td>
<td>PSW.3 Register Bank selector bit 0 (SEE NOTE 1).</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>OV</td>
<td>PSW.2 Overflow Flag.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>–</td>
<td>PSW.1 Usable as a general purpose flag.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>P</td>
<td>PSW.0 Parity flag. Set/cleared by hardware each instruction cycle to indicate an odd/even number of '1' bus in the accumulator.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**NOTE:**

1. The value represented by RS0 and RS1 selects the corresponding register bank.

**REGISTER BANK ADDRESS**

<table>
<thead>
<tr>
<th>RS1</th>
<th>RS0</th>
<th>ADDRESS</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>00H-07H</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>08H-0FH</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>10H-17H</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>18H-1FH</td>
</tr>
</tbody>
</table>

**PCON: POWER CONTROL REGISTER. NOT BIT ADDRESSABLE.**

<table>
<thead>
<tr>
<th>SMOD</th>
<th>–</th>
<th>–</th>
<th>GF1</th>
<th>GF0</th>
<th>PD</th>
<th>IDL</th>
</tr>
</thead>
<tbody>
<tr>
<td>SMOD</td>
<td>Double baud rate bit. If Timer 1 is used to generate baud rate and SMOD = 1, the baud rate is doubled when the Serial Port is used in modes 1, 2, or 3.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>–</td>
<td>Not implemented, reserved for future use.*</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>–</td>
<td>Not implemented, reserved for future use.*</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>–</td>
<td>Not implemented, reserved for future use.*</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GF1</td>
<td>General purpose flag bit.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GF0</td>
<td>General purpose flag bit.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PD</td>
<td>Power Down Bit. Setting this bit activates Power Down operation in the 80C51. (Available only in CMOS.)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>IDL</td>
<td>Idle mode bit. Setting this bit activates Idle Mode operation in the 80C51. (Available only in CMOS.)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**IE: INTERRUPT ENABLE REGISTER. BIT ADDRESSABLE.**

If the bit is 0, the corresponding interrupt is disabled. If the bit is 1, the corresponding interrupt is enabled.

**Interrupts:**

To use any of the interrupts in the 80C51 Family, the following three steps must be taken.

1. Set the EA (enable all) bit in the IE register to 1.
2. Set the corresponding individual interrupt enable bit in the IE register to 1.
3. Begin the interrupt service routine at the corresponding Vector Address of that interrupt. See Table below.

**Interrupt Source** | **Vector Address**
--- | ---
IE0 | 0003H
TF0 | 000BH
IE1 | 0013H
PF1 | 001BH
RI & TI | 0023H

In addition, for external interrupts, pins INT0 and INT1 (P3.2 and P3.3) must be set to 1, and depending on whether the interrupt is to be level or transition activated, bits IT0 or IT1 in the TCON register may need to be set to 1.

ITx = 0 level activated
ITx = 1 transition activated

**NOTE:**

* User software should not write 1 to reserved bits. These bits may be used in future 80C51 products to invoke new features.
ASSIGNING HIGHER PRIORITY TO ONE OR MORE INTERRUPTS:
In order to assign higher priority to an interrupt the corresponding bit in the IP register must be set to 1.
Remember that while an interrupt service is in progress, it cannot be interrupted by a lower or same level interrupt.

PRIORITY WITHIN LEVEL:
Priority within level is only to resolve simultaneous requests of the same priority level.
From high to low, interrupt sources are listed below:
IE0
TF0
IE1
TF1
RI or TI

IP: INTERRUPT PRIORITY REGISTER. BIT ADDRESSABLE.
If the bit is 0, the corresponding interrupt has a lower priority and if the bit is 1 the corresponding interrupt has a higher priority.

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>–</td>
<td>–</td>
<td>PS</td>
<td>PT1</td>
<td>PX1</td>
<td>PT0</td>
<td>PX0</td>
<td>–</td>
</tr>
</tbody>
</table>

* User software should not write 1 to reserved bits. These bits may be used in future 80C51 products to invoke new features.

TCON: TIMER/COUNTER CONTROL REGISTER. BIT ADDRESSABLE.

<table>
<thead>
<tr>
<th>TF1</th>
<th>TR1</th>
<th>TP0</th>
<th>TR0</th>
<th>IE1</th>
<th>IT1</th>
<th>IE0</th>
<th>IT0</th>
</tr>
</thead>
</table>
| TF1 TCON.7: Timer 1 overflow flag. Set by hardware when the Timer/Counter 1 overflows. Cleared by hardware as processor vectors to the interrupt service routine.
| TR1 TCON.6: Timer 1 run control bit. Set/cleared by software to turn Timer/Counter 1 ON/OFF.
| TP0 TCON.5: Timer 0 overflow flag. Set by hardware when the Timer/Counter 0 overflows. Cleared by hardware as processor vectors to the service routine.
| TR0 TCON.4: Timer 0 run control bit. Set/cleared by software to turn Timer/Counter 0 ON/OFF.
| IE1 TCON.3: External Interrupt 1 edge flag. Set by hardware when External Interrupt edge is detected. Cleared by hardware when interrupt is processed.
| IT1 TCON.2: Interrupt 1 type control bit. Set/cleared by software to specify falling edge/low level triggered External Interrupt.
| IE0 TCON.1: External Interrupt 0 edge flag. Set by hardware when External Interrupt edge detected. Cleared by hardware when interrupt is processed.
| IT0 TCON.0: Interrupt 0 type control bit. Set/cleared by software to specify falling edge/low level triggered External Interrupt.

TMOD: TIMER/COUNTER MODE CONTROL REGISTER. NOT BIT ADDRESSABLE.

<table>
<thead>
<tr>
<th>GATE</th>
<th>C/T</th>
<th>M1</th>
<th>M0</th>
</tr>
</thead>
</table>
| GATE | When TRx (in TCON) is set and GATE = 1, TIMER/COUNTER will run only while INTx pin is high (hardware control). When GATE = 0, TIMER/COUNTER will run only while TRx = 1 (software control).
| C/T | Timer or Counter selector. Cleared for Timer operation (input from internal system clock). Set for Counter operation (input from Tx input pin).
| M1 | Mode selector bit. (NOTE 1)
| M0 | Mode selector bit. (NOTE 1)

NOTE 1:

<table>
<thead>
<tr>
<th>M1</th>
<th>M0</th>
<th>Operating Mode</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>13-bit Timer (8048 compatible)</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>16-bit Timer/Counter</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>8-bit Auto-Reload Timer/Counter</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>(Timer 0) T0 is an 8-bit Timer/Counter controlled by the standard Timer 0 control bits. TH0 is an 8-bit Timer and is controlled by Timer 1 control bits.</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>(Timer 1) T1 stopped.</td>
</tr>
</tbody>
</table>
### TIMER/COUNTER 0

**Table 2. As a Timer:**

<table>
<thead>
<tr>
<th>MODE</th>
<th>INTERNAL FUNCTION</th>
<th>TMOD</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>13-bit Timer</td>
<td>08H</td>
</tr>
<tr>
<td>1</td>
<td>16-bit Timer</td>
<td>09H</td>
</tr>
<tr>
<td>2</td>
<td>8-bit Auto-Reload</td>
<td>0AH</td>
</tr>
<tr>
<td>3</td>
<td>Two 8-bit Timers</td>
<td>0BH</td>
</tr>
</tbody>
</table>

**Table 3. As a Counter:**

<table>
<thead>
<tr>
<th>MODE</th>
<th>INTERNAL FUNCTION</th>
<th>TMOD</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>13-bit Timer</td>
<td>09H</td>
</tr>
<tr>
<td>1</td>
<td>16-bit Timer</td>
<td>0AH</td>
</tr>
<tr>
<td>2</td>
<td>8-bit Auto-Reload</td>
<td>0BH</td>
</tr>
<tr>
<td>3</td>
<td>One 8-bit Counter</td>
<td>0CH</td>
</tr>
</tbody>
</table>

### TIMER/COUNTER 1

**Table 4. As a Timer:**

<table>
<thead>
<tr>
<th>MODE</th>
<th>INTERNAL FUNCTION</th>
<th>TMOD</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>13-bit Timer</td>
<td>08H</td>
</tr>
<tr>
<td>1</td>
<td>16-bit Timer</td>
<td>09H</td>
</tr>
<tr>
<td>2</td>
<td>8-bit Auto-Reload</td>
<td>0AH</td>
</tr>
<tr>
<td>3</td>
<td>Does not run</td>
<td>0BH</td>
</tr>
</tbody>
</table>

**Table 5. As a Counter:**

<table>
<thead>
<tr>
<th>MODE</th>
<th>INTERNAL FUNCTION</th>
<th>TMOD</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>13-bit Timer</td>
<td>09H</td>
</tr>
<tr>
<td>1</td>
<td>16-bit Timer</td>
<td>0AH</td>
</tr>
<tr>
<td>2</td>
<td>8-bit Auto-Reload</td>
<td>0BH</td>
</tr>
<tr>
<td>3</td>
<td>Not available</td>
<td>–</td>
</tr>
</tbody>
</table>

**NOTES:**
1. The timer is turned ON/OFF by setting/clearing bit TRx in the software.
2. The timer is turned ON/OFF by the 1-to-0 transition on INT0 (P3.2) when TR0 = 1 (hardware control).
**SCON: SERIAL PORT CONTROL REGISTER. BIT ADDRESSABLE.**

<table>
<thead>
<tr>
<th>SM0</th>
<th>SM1</th>
<th>SM2</th>
<th>REN</th>
<th>TB8</th>
<th>RB8</th>
<th>TI</th>
<th>RI</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

**SERIAL PORT SET-UP:**

<table>
<thead>
<tr>
<th>MODE</th>
<th>SCON</th>
<th>SM2 VARIATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>18H</td>
<td>Single Processor</td>
</tr>
<tr>
<td>1</td>
<td>59H</td>
<td>Single Processor</td>
</tr>
<tr>
<td>2</td>
<td>99H</td>
<td>Multiprocessor (SM2 = 0)</td>
</tr>
<tr>
<td>3</td>
<td>D9H</td>
<td>Multiprocessor (SM2 = 1)</td>
</tr>
<tr>
<td>0</td>
<td>NA</td>
<td>None</td>
</tr>
<tr>
<td>1</td>
<td>79H</td>
<td>Multiprocessor</td>
</tr>
<tr>
<td>2</td>
<td>69H</td>
<td>Single Processor</td>
</tr>
<tr>
<td>3</td>
<td>59H</td>
<td>Single Processor</td>
</tr>
</tbody>
</table>

**GENERATING BAUD RATES**

**Serial Port in Mode 0:**

Mode 0 has a fixed baud rate which is 1/12 of the oscillator frequency. To run the serial port in this mode none of the Timer/Counters need to be set up. Only the SCON register needs to be defined.

<table>
<thead>
<tr>
<th>Baud Rate</th>
<th>Osc Freq</th>
</tr>
</thead>
<tbody>
<tr>
<td>384</td>
<td>384</td>
</tr>
<tr>
<td>192</td>
<td>768</td>
</tr>
</tbody>
</table>

**Serial Port in Mode 1:**

Mode 1 has a variable baud rate. The baud rate is generated by Timer 1.

**Using Timer/Counter 1 to Generate Baud Rates:**

For this purpose, Timer 1 is used in mode 2 (Auto-Reload). Refer to Timer Setup section of this chapter.

**Baud Rate**

K = Osc Freq

If SMOD = 0, then K = 1.
If SMOD = 1, then K = 2 (SMOD is in the PCON register).

The baud rate is fixed in this mode and is 1/32 or 1/64 of the oscillator frequency, depending on the value of the SMOD bit in the PCON register.

In this mode none of the Timer/Counters are used and the clock comes from the internal phase 2 clock.

SMOD = 1, Baud Rate = 1/32 Osc Freq
SMOD = 0, Baud Rate = 1/64 Osc Freq

To set the SMOD bit: ORL PCON,#80H. The address of PCON is 87H.

**SERIAL PORT IN MODE 2:**

The baud rate is fixed in this mode and is 1/32 or 1/64 of the oscillator frequency, depending on the value of the SMOD bit in the PCON register.

In this mode none of the Timer/Counters are used and the clock comes from the internal phase 2 clock.

SMOD = 1, Baud Rate = 1/32 Osc Freq
SMOD = 0, Baud Rate = 1/64 Osc Freq

To set the SMOD bit: ORL PCON,#80H. The address of PCON is 87H.

**SERIAL PORT IN MODE 3:**

The baud rate in mode 3 is variable and sets up exactly the same as in mode 1.
### Table 7. 80C51 Instruction Set Summary

<table>
<thead>
<tr>
<th>MNEMONIC</th>
<th>DESCRIPTION</th>
<th>BYTE</th>
<th>OSCILOSCOPE PERIOD</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ARITHMETIC OPERATIONS</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ADD</td>
<td>A,Rn Add register to Accumulator</td>
<td>1</td>
<td>12</td>
</tr>
<tr>
<td>ADD</td>
<td>A,direct Add direct byte to Accumulator</td>
<td>2</td>
<td>12</td>
</tr>
<tr>
<td>ADD</td>
<td>A,@Ri Add indirect RAM to Accumulator</td>
<td>1</td>
<td>12</td>
</tr>
<tr>
<td>ADD</td>
<td>A,#data Add immediate data to Accumulator</td>
<td>2</td>
<td>12</td>
</tr>
<tr>
<td>ADDC</td>
<td>A,Rn Add register to Accumulator with carry</td>
<td>1</td>
<td>12</td>
</tr>
<tr>
<td>ADDC</td>
<td>A,direct Add direct byte to Accumulator with carry</td>
<td>2</td>
<td>12</td>
</tr>
<tr>
<td>ADDC</td>
<td>A,@Ri Add indirect RAM to Accumulator with carry</td>
<td>1</td>
<td>12</td>
</tr>
<tr>
<td>ADDC</td>
<td>A,#data Add immediate data to Accumulator with carry</td>
<td>2</td>
<td>12</td>
</tr>
<tr>
<td>SUBB</td>
<td>A,Rn Subtract Register from Accumulator with borrow</td>
<td>1</td>
<td>12</td>
</tr>
<tr>
<td>SUBB</td>
<td>A,direct Subtract direct byte from Accumulator with borrow</td>
<td>2</td>
<td>12</td>
</tr>
<tr>
<td>SUBB</td>
<td>A,@Ri Subtract indirect RAM from Accumulator with borrow</td>
<td>1</td>
<td>12</td>
</tr>
<tr>
<td>SUBB</td>
<td>A,#data Subtract immediate data from Accumulator with borrow</td>
<td>2</td>
<td>12</td>
</tr>
<tr>
<td>INC</td>
<td>A Increment Accumulator</td>
<td>1</td>
<td>12</td>
</tr>
<tr>
<td>INC</td>
<td>Rn Increment register</td>
<td></td>
<td></td>
</tr>
<tr>
<td>INC</td>
<td>direct Increment direct byte</td>
<td>2</td>
<td>12</td>
</tr>
<tr>
<td>DEC</td>
<td>A Decrement Accumulator</td>
<td>1</td>
<td>12</td>
</tr>
<tr>
<td>DEC</td>
<td>Rn Decrement register</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DEC</td>
<td>direct Decrement direct byte</td>
<td>2</td>
<td>12</td>
</tr>
<tr>
<td>DEC</td>
<td>@Ri Decrement indirect RAM</td>
<td>1</td>
<td>12</td>
</tr>
<tr>
<td>INC</td>
<td>DPTR Increment Data Pointer</td>
<td>1</td>
<td>24</td>
</tr>
<tr>
<td>MUL</td>
<td>AB Multiply A and B</td>
<td>1</td>
<td>48</td>
</tr>
<tr>
<td>DIV</td>
<td>AB Divide A by B</td>
<td>1</td>
<td>48</td>
</tr>
<tr>
<td>DA</td>
<td>A Decimal Adjust Accumulator</td>
<td>1</td>
<td>12</td>
</tr>
<tr>
<td>LOGICAL OPERATIONS</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ANL</td>
<td>A,Rn AND Register to Accumulator</td>
<td>1</td>
<td>12</td>
</tr>
<tr>
<td>ANL</td>
<td>A,direct AND direct byte to Accumulator</td>
<td>2</td>
<td>12</td>
</tr>
<tr>
<td>ANL</td>
<td>A,@Ri AND indirect RAM to Accumulator</td>
<td>1</td>
<td>12</td>
</tr>
<tr>
<td>ANL</td>
<td>A,#data AND immediate data to Accumulator</td>
<td>2</td>
<td>12</td>
</tr>
<tr>
<td>ANL</td>
<td>direct,A AND Accumulator to direct byte</td>
<td>2</td>
<td>12</td>
</tr>
<tr>
<td>ANL</td>
<td>direct,#data AND immediate data to direct byte</td>
<td>3</td>
<td>24</td>
</tr>
<tr>
<td>ORL</td>
<td>A,Rn OR register to Accumulator</td>
<td>1</td>
<td>12</td>
</tr>
<tr>
<td>ORL</td>
<td>A,direct OR direct byte to Accumulator</td>
<td>2</td>
<td>12</td>
</tr>
<tr>
<td>ORL</td>
<td>A,@Ri OR indirect RAM to Accumulator</td>
<td>1</td>
<td>12</td>
</tr>
<tr>
<td>ORL</td>
<td>A,#data OR immediate data to Accumulator</td>
<td>2</td>
<td>12</td>
</tr>
<tr>
<td>ORL</td>
<td>direct,A OR Accumulator to direct byte</td>
<td>2</td>
<td>12</td>
</tr>
<tr>
<td>ORL</td>
<td>direct,#data OR immediate data to direct byte</td>
<td>3</td>
<td>24</td>
</tr>
<tr>
<td>XRL</td>
<td>A,Rn Exclusive-OR register to Accumulator</td>
<td>1</td>
<td>12</td>
</tr>
<tr>
<td>XRL</td>
<td>A,direct Exclusive-OR direct byte to Accumulator</td>
<td>2</td>
<td>12</td>
</tr>
<tr>
<td>XRL</td>
<td>A,@Ri Exclusive-OR indirect RAM to Accumulator</td>
<td>1</td>
<td>12</td>
</tr>
<tr>
<td>XRL</td>
<td>A,#data Exclusive-OR immediate data to Accumulator</td>
<td>2</td>
<td>12</td>
</tr>
<tr>
<td>XRL</td>
<td>direct,A Exclusive-OR Accumulator to direct byte</td>
<td>2</td>
<td>12</td>
</tr>
<tr>
<td>XRL</td>
<td>direct,#data Exclusive-OR immediate data to direct byte</td>
<td>3</td>
<td>24</td>
</tr>
<tr>
<td>CLR</td>
<td>A Clear Accumulator</td>
<td>1</td>
<td>12</td>
</tr>
<tr>
<td>CPL</td>
<td>A Complement Accumulator</td>
<td>1</td>
<td>12</td>
</tr>
<tr>
<td>RL</td>
<td>A Rotate Accumulator left</td>
<td>1</td>
<td>12</td>
</tr>
<tr>
<td>RLC</td>
<td>A Rotate Accumulator left through the carry</td>
<td>1</td>
<td>12</td>
</tr>
<tr>
<td>RR</td>
<td>A Rotate Accumulator right</td>
<td>1</td>
<td>12</td>
</tr>
<tr>
<td>RRC</td>
<td>A Rotate Accumulator right through the carry</td>
<td>1</td>
<td>12</td>
</tr>
<tr>
<td>SWAP</td>
<td>A Swap nibbles within the Accumulator</td>
<td>1</td>
<td>12</td>
</tr>
<tr>
<td>DATA TRANSFER</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MOV</td>
<td>A,Rn Move register to Accumulator</td>
<td>1</td>
<td>12</td>
</tr>
<tr>
<td>MOV</td>
<td>A,direct Move direct byte to Accumulator</td>
<td>2</td>
<td>12</td>
</tr>
<tr>
<td>MOV</td>
<td>A,@Ri Move indirect RAM to Accumulator</td>
<td>1</td>
<td>12</td>
</tr>
</tbody>
</table>

Notes on instruction set and addressing modes:
- **Rn** Register R7-R0 of the currently selected Register Bank.
- **direct** 8-bit internal data location's address. This could be an Internal Data RAM location (0-127) or a SFR [i.e., I/O port, control register, status register, etc. (128-255)].
- **@Ri** 8-bit internal data RAM location (0-255) addressed indirectly through register R1 or R0.
- **#data** 8-bit constant included in the instruction.
- **#data 16** 16-bit constant included in the instruction.
- **addr 16** 16-bit destination address. Used by LCALL and LJMP. A branch can be anywhere within the 64k-byte Program Memory address space.
- **addr 11** 11-bit destination address. Used by ACALL and AJMP. The branch will be within the same 2k-byte page of program memory as the first byte of the following instruction.
- **rel** Signed (two's complement) 8-bit offset byte. Used by SJMP and all conditional jumps. Range is –128 to +127 bytes relative to first byte of the following instruction.
- **bit** Direct Addressed bit in Internal Data RAM or Special Function Register.

All mnemonics copyrighted © Intel Corporation 1980
<table>
<thead>
<tr>
<th>MNEMONIC</th>
<th>DESCRIPTION</th>
<th>BYTE</th>
<th>OSCILLATOR PERIOD</th>
</tr>
</thead>
<tbody>
<tr>
<td>MOV A,#data</td>
<td>Move immediate data to Accumulator</td>
<td>2</td>
<td>12</td>
</tr>
<tr>
<td>MOV A,Rn</td>
<td>Move Accumulator to register</td>
<td>1</td>
<td>12</td>
</tr>
<tr>
<td>MOV Rn direct</td>
<td>Move direct byte to register</td>
<td>2</td>
<td>24</td>
</tr>
<tr>
<td>MOV Rn,Rn</td>
<td>Move Accumulator to register</td>
<td>2</td>
<td>12</td>
</tr>
<tr>
<td>MOV direct,A</td>
<td>Move Accumulator to direct byte</td>
<td>2</td>
<td>12</td>
</tr>
<tr>
<td>MOV direct,Rn</td>
<td>Move register to direct byte</td>
<td>2</td>
<td>24</td>
</tr>
<tr>
<td>MOV direct,Rdirect</td>
<td>Move direct byte to direct</td>
<td>3</td>
<td>24</td>
</tr>
<tr>
<td>MOV direct,@Ri</td>
<td>Move indirect RAM to direct byte</td>
<td>2</td>
<td>24</td>
</tr>
<tr>
<td>MOV direct,#data</td>
<td>Move immediate data to direct byte</td>
<td>3</td>
<td>24</td>
</tr>
<tr>
<td>MOV @Ri,A</td>
<td>Move Accumulator to indirect RAM</td>
<td>1</td>
<td>12</td>
</tr>
<tr>
<td>MOV @Rdirect</td>
<td>Move direct byte to indirect RAM</td>
<td>2</td>
<td>24</td>
</tr>
<tr>
<td>MOV @Rdirect,#data</td>
<td>Move immediate data to indirect RAM</td>
<td>2</td>
<td>12</td>
</tr>
<tr>
<td>MOV DPTR,#16data</td>
<td>Load Data Pointer with a 16-bit constant</td>
<td>3</td>
<td>24</td>
</tr>
<tr>
<td>MOVX A,@DPTR</td>
<td>Move Code byte relative to DPTR to A</td>
<td>1</td>
<td>24</td>
</tr>
<tr>
<td>MOVX A,@PC</td>
<td>Move Code byte relative to PC to A</td>
<td>1</td>
<td>24</td>
</tr>
<tr>
<td>MOVX A,@Ri</td>
<td>Move external RAM (8-bit addr) to A</td>
<td>1</td>
<td>24</td>
</tr>
<tr>
<td>MOVX A,@DPTR</td>
<td>Move external RAM (16-bit addr) to A</td>
<td>1</td>
<td>24</td>
</tr>
<tr>
<td>MOVX A,#data16</td>
<td>Load Data Pointer with a 16-bit constant</td>
<td>3</td>
<td>24</td>
</tr>
<tr>
<td>MOVX A,@Ri,#data</td>
<td>Move immediate data to indirect RAM</td>
<td>2</td>
<td>24</td>
</tr>
<tr>
<td>PUSH direct</td>
<td>Push direct byte onto stack</td>
<td>2</td>
<td>24</td>
</tr>
<tr>
<td>POP direct</td>
<td>Pop direct byte from stack</td>
<td>2</td>
<td>24</td>
</tr>
<tr>
<td>XCH A,Rn</td>
<td>Exchange register with Accumulator</td>
<td>1</td>
<td>12</td>
</tr>
<tr>
<td>XCH A direct</td>
<td>Exchange direct byte with Accumulator</td>
<td>2</td>
<td>12</td>
</tr>
<tr>
<td>XCH A,@Ri</td>
<td>Exchange indirect RAM with Accumulator</td>
<td>1</td>
<td>12</td>
</tr>
<tr>
<td>XCHD A,@Ri</td>
<td>Exchange low order digit indirect RAM with A</td>
<td>1</td>
<td>12</td>
</tr>
</tbody>
</table>

**BOOLEAN VARIABLE MANIPULATION**

<table>
<thead>
<tr>
<th>MNEMONIC</th>
<th>DESCRIPTION</th>
<th>BYTE</th>
<th>OSCILLATOR PERIOD</th>
</tr>
</thead>
<tbody>
<tr>
<td>CLR C</td>
<td>Clear carry</td>
<td>1</td>
<td>12</td>
</tr>
<tr>
<td>CLR bit</td>
<td>Clear direct bit</td>
<td>2</td>
<td>12</td>
</tr>
<tr>
<td>SETB C</td>
<td>Set carry</td>
<td>1</td>
<td>12</td>
</tr>
<tr>
<td>SETB bit</td>
<td>Set direct bit</td>
<td>2</td>
<td>12</td>
</tr>
<tr>
<td>CPL C</td>
<td>Complement carry</td>
<td>1</td>
<td>12</td>
</tr>
<tr>
<td>CPL bit</td>
<td>Complement direct bit</td>
<td>2</td>
<td>12</td>
</tr>
<tr>
<td>ANL C,bit</td>
<td>AND direct bit to carry</td>
<td>2</td>
<td>24</td>
</tr>
<tr>
<td>ANL C,bit</td>
<td>AND complement of direct bit to carry</td>
<td>2</td>
<td>24</td>
</tr>
<tr>
<td>ORL C,bit</td>
<td>OR direct bit to carry</td>
<td>2</td>
<td>24</td>
</tr>
<tr>
<td>ORL C,bit</td>
<td>OR complement of direct bit to carry</td>
<td>2</td>
<td>24</td>
</tr>
<tr>
<td>MOV C,bit</td>
<td>Move direct bit to carry</td>
<td>2</td>
<td>12</td>
</tr>
<tr>
<td>MOV direct,C</td>
<td>Move carry to direct bit</td>
<td>2</td>
<td>24</td>
</tr>
<tr>
<td>JC rel</td>
<td>Jump if carry is set</td>
<td>2</td>
<td>24</td>
</tr>
<tr>
<td>INC rel</td>
<td>Jump if carry is not set</td>
<td>2</td>
<td>24</td>
</tr>
</tbody>
</table>

All mnemonics copyrighted © Intel Corporation 1980
In this appendix are available some electric diagrams of the most frequently used GPC® 324 interfaces. All these interface can be yourself produced and some of them are standard grifo® cards and, if required, they can be directly ordered.

**APPENDIX C: ELECTRIC DIAGRAMS**

![Electric Diagram](image)

**FIGURE C1: PPI EXPANSION ELECTRIC DIAGRAM**
Abaco I/O Bus
26 pin connector

**Figure C2: SPA 03 electrical diagram**
FIGURE C3: QTP 16P ELECTRIC DIAGRAM
FIGURE C4: QTP 24P ELECTRIC DIAGRAM (1 OF 2)
**FIGURE C5: QTP 24P ELECTRIC DIAGRAM (2 OF 2)**
Figure C6: ABACO® I/O BUS input output electric diagram
FIGURE C7: BUS INTERFACE ELECTRIC DIAGRAM
FIGURE C8: IAC 01 ELECTRIC DIAGRAM

Title: IAC 01
Date: 13-11-98
Page: 1 of 1
APPENDIX D: ALPHABETICAL INDEX

Symbol

µP 80C32  3, 32, B-1
µP 80C320 3, 32, B-7
µP 51 family  B-14
82c55  C-1

A

ABACO® I/O BUS  4, 11, 29, 40, 44, C-7
Addresses  36
Assistance  1

B

Back up  7, 9, 10, 26, 33
Battery  10, 33
Block diagram  5
Bibliography  48

C

Card version  1
Clock  7, 8
Communication  4, 15, 30
Components maps  23
Configuration input  6, 26, 41
Configuration jumper  6
Connections  45
Connectors  8, 10
  CN1  11
  CN2  10
  CN3A  12, 27
  CN3B  14, 27
  CN5  20
Consumption  9
Container  1, A-1
Control logic  4
CPU  3, 8, B-1
Current  9
Current loop  4, 12, 18, 30

D

Debug mode  6, 26, 41
Default configuration  24, 28, 30
Dimension  A-1
DIN Ω rails  A-1
DIR signal  30, 41
Drivers  31

E
EEPROM  7, 28, 36
EEPROM serial  6, 42
Electric diagrams  C-1
Electric features  9
EPROM  7, 28, 36
Expansion  C-6
External cards  44

F
Features  2
  electric  9
  general  8
  physical  8
FLASH EPROM  7, 28, 33, 36

G
General features  8
GET 51  35

H
Humidity  8

I
I/O addresses  40
I/O connection  22
I/O lines  C-6
Idle  7
In system programming  20, 27, 33
Installation  10
Interfaces  22, C-3
Internal ROM  27
Interrupt  11, 20, 26, 29, 33, 43

J
Jumpers  24
  2 pins  26
  3 pins  27
  5 pins  27
  location  25
M
Maintenance  1
Maps  36
   Mode 0  37
   Mode 1  38
   Mode 3  39
Memory  7, 8, 13, 27, 28, 36
Memory addresses  36
Microprocessor  3, 8, B-1
Mounting  A-1

N
Network  17, 19

O
Operator interface  22, C-3, C-4
Options  28, 30, 33, A-1

P
Peripheral devices  6, 8, 41
Photo  9
Physical features  8
Piggy back  A-1
Port I/O  20
Power failure  7, 9, 29, 33
Power supply  7, 9, 33
Printer  22, C-8
Programming voltage  20, 33

R
RAM  7, 28, 33, 36
Real Time Clock  6, 29, 33, 40, 42
Registers  40, B-3, B-13, B-15
Reset  29, 41
Reset contact  4, 13
RS 232  4, 12, 14, 16, 30, 32
RS 422  4, 9, 12, 16, 30, 41
RS 485  4, 9, 12, 16, 17, 32, 41
Run mode  6, 26, 41

S
Serial line A  12, 30
Serial line B  14, 32
Size  8
Software  34
Solder jumpers 28

T
Temperature 8
Timer/counter 43
TransZorb™ 7

U
User interfaces 22, 44, C-3

V
Version 1

W
Watch dog 6, 8, 29, 40, 41
Weight 8