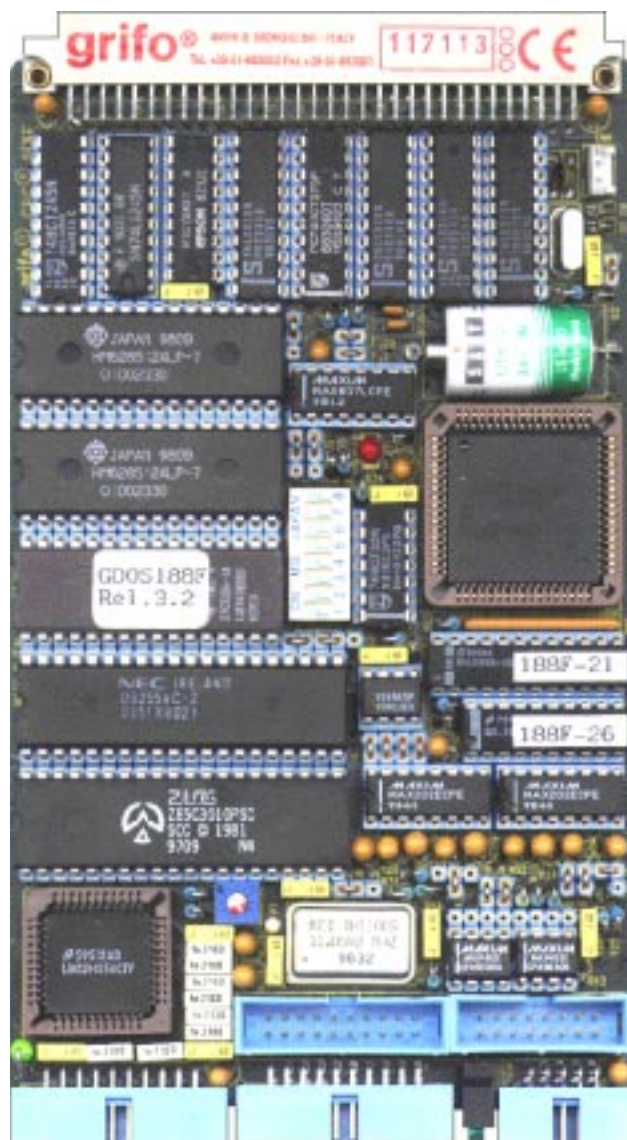


GPC[®] 188F

General Purpose Controller 80C188

MANUALE TECNICO



grifo[®]

ITALIAN TECHNOLOGY

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

E-mail: grifo@grifo.it

<http://www.grifo.it>

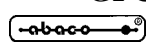
<http://www.grifo.com>

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



GPC[®] 188F

Edizione 3.10 Rel. 06 Dicembre 1999

, GPC[®], grifo[®], sono marchi registrati della ditta grifo[®]

GPC[®] 188F

General Purpose Controller 80C188

MANUALE TECNICO

Scheda in formato singola Europa da 100x160mm con interfaccia per il **BUS Industriale Abaco[®]**; **CPU CMOS 80C188** da **20 MHz**. Fino a **1MEPROM** o **fino a 512K FLASH EPROM**; fino a **1M RAM** di cui 512K sono anche proteggibili in scrittura e quindi gestite come **RAM disk**; **EEPROM** seriale fino a 8 K. **Dip Switch** da 8 vie leggibile da software; **2 LEDs** di attività, posizionati sul frontale, gestibili da software. **2 linee seriali in RS 232** di cui una settabile in **RS 422, RS 485** o **Current Loop** gestite dal potente **Z85C30** che supporta i protocolli **HDLC, SDLC**, ecc. con **baud rate** settabile da software, fino a **115 KBaud**. **24 linee di I/O TTL**, settabili da software, gestite dal **PPI 82C55**. **1 timer** a 16 bits e **2 timer counter** da 16 bits in grado di generare segnali **PWM**. **8 linee di A/D Converter** con **Sample & Hold**, 5,5 μ s, range 0÷2,048V con possibilità di lavorare in differenziale ($\pm 2,048V$), 12 bits+segno, gestite dal potente **LM 12H458**. Circuiteria di **power failure** in grado di generare interrupt. **Real Time Clock** in grado di gestire giorno, mese, anno, giorno della settimana, ore, minuti, secondi e di generare un **INT** con cadenze definibili da software. **Watch Dog** resettabile da software visualizzato tramite **LED**. Circuiteria di **back up** per RAM e RTC con batteria al **Litio** e connettore per eventuale batteria esterna. Lo stato di carica é acquisibile via software. Unica tensione di alimentazione a **5Vdc, 386mA**. Vasta disponibilità di software di base e di **ambienti di sviluppo** che consentono di poter utilizzare la scheda tramite un normale **PC**. Tra i pacchetti disponibili si possono citare: **GDOS 188; PASCAL 188; Monitor Debugger; GCTR 188; ROM-DOS; HI TECH C 86; GET 188; FLASH WRITER; DDS MICRO C 86**; ecc.

grifo[®]

ITALIAN TECHNOLOGY

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

E-mail: grifo@grifo.it

<http://www.grifo.it>

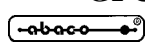
<http://www.grifo.com>

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



GPC[®] 188F

Edizione 3.10 Rel. 06 Dicembre 1999

, GPC[®], grifo[®], sono marchi registrati della ditta grifo[®]

Vincoli sulla documentazione **grifo**[®] Tutti i Diritti Riservati

Nessuna parte del presente manuale può essere riprodotta, trasmessa, trascritta, memorizzata in un archivio o tradotta in altre lingue, con qualunque forma o mezzo, sia esso elettronico, meccanico, magnetico ottico, chimico, manuale, senza il permesso scritto della **grifo**[®].

IMPORTANTE

Tutte le informazioni contenute sul presente manuale sono state accuratamente verificate, ciononostante **grifo**[®] non si assume nessuna responsabilità per danni, diretti o indiretti, a cose e/o persone derivanti da errori, omissioni o dall'uso del presente manuale, del software o dell' hardware ad esso associato.

grifo[®] altresì si riserva il diritto di modificare il contenuto e la veste di questo manuale senza alcun preavviso, con l' intento di offrire un prodotto sempre migliore, senza che questo rappresenti un obbligo per **grifo**[®].

Per le informazioni specifiche dei componenti utilizzati sui nostri prodotti, l'utente deve fare riferimento agli specifici Data Book delle case costruttrici o delle seconde sorgenti.

LEGENDA SIMBOLI

Nel presente manuale possono comparire i seguenti simboli:

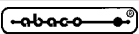


Attenzione: Pericolo generico



Attenzione: Pericolo di alta tensione

Marchi Registrati

 , GPC[®], **grifo**[®] : sono marchi registrati della **grifo**[®].

Altre marche o nomi di prodotti sono marchi registrati dei rispettivi proprietari.

INDICE GENERALE

| | |
|---|----|
| INTRODUZIONE | 1 |
| VERSIONE SCHEDA | 1 |
| INFORMAZIONI GENERALI | 2 |
| PROCESSORE DI BORDO | 3 |
| MEMORIE | 3 |
| CLOCK | 4 |
| COMUNICAZIONE SERIALE | 4 |
| BUS ABACO® | 4 |
| LINEE DI I/O DIGITALI | 6 |
| REAL TIME CLOCK | 6 |
| WATCH DOG | 6 |
| A/D CONVERTER | 6 |
| SPECIFICHE TECNICHE | 8 |
| CARATTERISTICHE GENERALI | 8 |
| CARATTERISTICHE FISICHE | 8 |
| CARATTERISTICHE ELETTRICHE | 9 |
| INSTALLAZIONE | 10 |
| CONNESSIONI CON IL MONDO ESTERNO | 10 |
| CN6 - CONNETTORE PER BATTERIA ESTERNA DI BACK UP | 10 |
| CN5 - CONNETTORE PER PORT B PPI 82C55 | 11 |
| CN2 - CONNETTORE PER PORT A E C DEL PPI 82C55 | 12 |
| CN3 - CONNETTORE PER INGRESSI A/D CONVERTER | 14 |
| CN1 - CONNETTORE PER LINEE SERIALI RS 232 E TIMER COUNTER..... | 16 |
| CN4 - CONNETTORE PER SERIALE IN RS 422, RS 485 E CURRENT LOOP | 18 |
| K1 - CONNETTORE PER BUS ABACO® | 22 |
| TRIMMER E TARATURE | 24 |
| INTERFACCIAMENTO DEGLI I/O CON IL CAMPO | 24 |
| INTERFACCIE PER I/O DIGITALI | 26 |
| SEGNALAZIONI VISIVE | 26 |
| JUMPERS | 27 |
| JUMPERS A 2 VIE | 28 |
| JUMPERS A 4 VIE | 28 |
| JUMPERS A 3 VIE | 30 |
| JUMPERS A 5 VIE | 30 |
| RESET E WATCH DOG | 31 |
| INTERRUPTS | 31 |
| COMUNICAZIONE SERIALE | 32 |
| BACK UP | 34 |
| SELEZIONE MEMORIE | 35 |
| POWER FAILURE | 36 |
| INGRESSI DI CONFIGURAZIONE | 36 |

| | |
|---|------|
| TRASFERIMENTI DMA | 36 |
| DESCRIZIONE SOFTWARE | 38 |
| MAPPAGGI ED INDIRIZZAMENTI..... | 40 |
| MAPPAGGIO DELLE RISORSE DI BORDO | 40 |
| MAPPAGGIO PERIFERICHE IN I/O | 41 |
| MAPPAGGIO BUS ABACO® | 43 |
| MAPPAGGIO MEMORIE | 43 |
| DESCRIZIONE SOFTWARE DELLE PERIFERICHE DI BORDO | 46 |
| WATCH DOG | 46 |
| EEPROM SERIALE | 46 |
| STATO BATTERIA | 47 |
| INGRESSI DI CONFIGURAZIONE | 47 |
| LEDS DI ATTIVITA' | 48 |
| PROTEZIONE IN SCRITTURA | 48 |
| DISATTIVAZIONE RICHIESTA DMA | 48 |
| REAL TIME CLOCK | 49 |
| MEMORY MANAGEMENT UNIT | 50 |
| A/D CONVERTER | 51 |
| SCC 85C30 | 51 |
| PPI 82C55 | 59 |
| PERIFERICHE DELLA CPU | 60 |
| SCHEDE ESTERNE | 61 |
| BIBLIOGRAFIA | 65 |
| APPENDICE A: SCHEMI ELETTRICI | A-1 |
| APPENDICE B: DESCRIZIONE COMPONENTI DI BORDO | B-1 |
| CPU 80C188 | B-1 |
| A/D CONVERTER LM12H458 | B-19 |
| APPENDICE C: INDICE ANALITICO | C-1 |

INDICE DELLE FIGURE

| | |
|--|-----|
| FIGURA 1: SCHEMA A BLOCCHI | 5 |
| FIGURA 2: FOTO SCHEDA | 7 |
| FIGURA 3: CN6 - CONNETTORE PER BATTERIA ESTERNA DI BACK UP | 10 |
| FIGURA 4: CN5 - CONNETTORE PER PORT B PPI 82C55 | 11 |
| FIGURA 5: CN2 - CONNETTORE PER PORT A E C DEL PPI 82C55 | 12 |
| FIGURA 6: SCHEMA DI COLLEGAMENTO LINEE DI I/O DEL PPI | 13 |
| FIGURA 7: CN3 - CONNETTORE PER INGRESSI A/D CONVERTER | 14 |
| FIGURA 8: SCHEMA D'INGRESSO A/D CONVERTER | 15 |
| FIGURA 9: CN1 - CONNETTORE PER LINEE SERIALI RS 232 E TIMER COUNTER | 16 |
| FIGURA 10: SCHEMA DI COLLEGAMENTO TIMER COUNTER | 17 |
| FIGURA 11: SCHEMA DI COMUNICAZIONE SERIALE | 17 |
| FIGURA 12: CN4 - CONNETTORE PER SERIALE IN RS 422, RS 485 E CURRENT LOOP | 18 |
| FIGURA 13: ESEMPIO COLLEGAMENTO PUNTO PUNTO IN RS 232 | 19 |
| FIGURA 14: ESEMPIO COLLEGAMENTO PUNTO PUNTO IN RS 422 | 19 |
| FIGURA 15: ESEMPIO COLLEGAMENTO PUNTO PUNTO IN RS 485 | 19 |
| FIGURA 16: ESEMPIO COLLEGAMENTO IN RETE IN RS 485 | 20 |
| FIGURA 17: ESEMPIO COLLEGAMENTO PUNTO PUNTO IN CURRENT LOOP A 4 FILI | 21 |
| FIGURA 18: ESEMPIO COLLEGAMENTO PUNTO PUNTO IN CURRENT LOOP A 2 FILI | 21 |
| FIGURA 19: K1 - CONNETTORE PER BUS ABACO® | 22 |
| FIGURA 20: DISPOSIZIONE CONNETTORI, TRIMMER, BATTERIA, MEMORIE, ECC. | 25 |
| FIGURA 21: TABELLA DELLE SEGNALAZIONI VISIVE | 26 |
| FIGURA 22: TABELLA RIASSUNTIVA JUMPERS | 27 |
| FIGURA 23: TABELLA JUMPERS A 2 VIE | 28 |
| FIGURA 24: TABELLA JUMPERS A 4 VIE | 28 |
| FIGURA 25: DISPOSIZIONE JUMPERS | 29 |
| FIGURA 26: TABELLA JUMPERS A 3 VIE | 30 |
| FIGURA 27: TABELLA JUMPERS A 5 VIE | 30 |
| FIGURA 28: DISPOSIZIONE DRIVER PER COMUNICAZIONE SERIALE | 33 |
| FIGURA 29: TABELLA DI SELEZIONE MEMORIE | 35 |
| FIGURA 30: PIANTA COMPONENTI | 37 |
| FIGURA 31: COLLEGAMENTO HARDWARE DEI DISPOSITIVI | 40 |
| FIGURA 32: TABELLA INDIRIZZAMENTO I/O - PARTE 1 | 41 |
| FIGURA 33: TABELLA INDIRIZZAMENTO I/O - PARTE 2 | 42 |
| FIGURA 34: MAPPAGGIO DELLE MEMORIE - PARTE 1 | 44 |
| FIGURA 35: MAPPAGGIO DELLE MEMORIE - PARTE 2 | 45 |
| FIGURA 36: SCHEMA DELLE POSSIBILI CONNESSIONI | 63 |
| FIGURA A1: SCHEMA ELETTRICO IAC 01 | A-1 |
| FIGURA A2: SCHEMA ELETTRICO KDX x24 | A-2 |
| FIGURA A3: SCHEMA ELETTRICO QTP 16P | A-3 |
| FIGURA A4: SCHEMA ELETTRICO QTP 24P - PARTE 1 | A-4 |
| FIGURA A5: SCHEMA ELETTRICO QTP 24P - PARTE 2 | A-5 |
| FIGURA A6: SCHEMA ELETTRICO SPA 01 | A-6 |



INTRODUZIONE

L'uso di questi dispositivi é rivolto - **IN VIA ESCLUSIVA** - a personale specializzato.

Scopo di questo manuale é la trasmissione delle informazioni necessarie all'uso competente e sicuro dei prodotti. Esse sono il frutto di un'elaborazione continua e sistematica di dati e prove tecniche registrate e validate dal Costruttore, in attuazione alle procedure interne di sicurezza e qualità dell'informazione.

I dati di seguito riportati sono destinati - **IN VIA ESCLUSIVA** - ad un utenza specializzata, in grado di interagire con i prodotti in condizioni di sicurezza per le persone, per la macchina e per l'ambiente, interpretando un'elementare diagnostica dei guasti e delle condizioni di funzionamento anomale e compiendo semplici operazioni di verifica funzionale, nel pieno rispetto delle norme di sicurezza e salute vigenti.

Le informazioni riguardanti installazione, montaggio, smontaggio, manutenzione, aggiustaggio, riparazione ed installazione di eventuali accessori, dispositivi ed attrezzature, sono destinate - e quindi eseguibili - sempre ed in via esclusiva da personale specializzato avvertito ed istruito, o direttamente dall'**ASSISTENZA TECNICA AUTORIZZATA**, nel pieno rispetto delle raccomandazioni trasmesse dal costruttore e delle norme di sicurezza e salute vigenti.

I dispositivi non possono essere utilizzati all'aperto. Si deve sempre provvedere ad inserire i moduli all'interno di un contenitore a norme di sicurezza che rispetti le vigenti normative. La protezione di questo contenitore non si deve limitare ai soli agenti atmosferici, bensì anche a quelli meccanici, elettrici, magnetici, ecc.

Per un corretto rapporto coi prodotti, é necessario garantire leggibilità e conservazione del manuale, anche per futuri riferimenti. In caso di deterioramento o più semplicemente per ragioni di approfondimento tecnico ed operativo, consultare direttamente l'Assistenza Tecnica autorizzata.

Al fine di non incontrare problemi nell'uso di tali dispositivi, é conveniente che l'utente - **PRIMA DI COMINCIARE AD OPERARE** - legga con attenzione tutte le informazioni contenute in questo manuale. In una seconda fase, per rintracciare più facilmente le informazioni necessarie, si può fare riferimento all'indice generale e all'indice analitico, posti rispettivamente all'inizio ed alla fine del manuale.

VERSIONE SCHEDA

Il presente manuale è riferito alla scheda **GPC® 188F** versione **121198** e successive. La validità delle informazioni riportate è quindi subordinata al numero di versione della scheda in uso e l'utente deve quindi sempre verificare la giusta corrispondenza tra le due indicazioni. Sulla scheda il numero di versione è riportato in più punti sia a livello di serigrafia che di stampato (ad esempio sul bordo esterno della scheda, a fianco della EPROM di IC18, sia sul lato componenti che sul lato stagnature).

INFORMAZIONI GENERALI

La scheda **GPC® 188F** é un potentissimo modulo di controllo e di gestione nel formato standard **Singola Europa** da 100x160 mm. Essa opera sul potente **BUS Industriale Abaco®**, di cui sfrutta la ricca serie di periferiche, intelligenti e non, disponibili su questo BUS. La **GPC® 188F** é basata sulla potente e nota **CPU 80C188 Intel**, codice compatibile con qualsiasi PC, ed ha a bordo scheda notevoli risorse hardware. Particolarmente interessante é la disponibilità di 8 linee di **A/D Converter** ad alta velocità da 13 bits. La estrema modularità e la notevole completezza di risorse hardware della scheda **GPC® 188F** le consentono di poter affrontare applicazioni anche di notevole complessità con estrema disinvoltura. E' inoltre il componente ideale in tutte le applicazioni che richiedono molta memoria, infatti a bordo scheda si può raggiungere una configurazione massima superiore ai **2M Bytes**. La notevole facilità di impiego é determinata anche dalla ricca serie di tools di sviluppo software con linguaggi ad alto livello che, in un ambiente molto amichevole ed efficiente, consentono di poter lavorare al meglio utilizzando unicamente un normale PC. Tra questi tools una particolare attenzione va rivolta ai **sistemi operativi**, ad un efficiente **compilatore romato PASCAL**, oppure ai potenti **compilatori C**. Grande attenzione é stata posta alla più complessa delle operazioni, quella del **debugger**, rendendo disponibili dei programmi che consentono di effettuare il **debugger simbolico remoto** direttamente sulla scheda.

- Singola Europa da 100x160mm con interfaccia per il **BUS Industriale Abaco®**
- **CPU CMOS 80C188** da **20 MHz**
- Fino a **1M EPROM** o **fino a 512K FLASH EPROM**
- Fino a **1M RAM** di cui 512K sono anche proteggibili in scrittura e quindi gestite come **RAM disk**.
- **EEPROM** seriale fino a 8K.
- **Dip Switch** da 8 vie leggibile da software.
- **2 LEDs** di attività, posizionati sul frontale, gestibili da software.
- **2 linee seriali** in **RS 232** di cui una settabile in **RS 422, RS 485** o **Current Loop** gestite dal potente **Z85C30** che supporta i protocolli **HDLC, SDLC**, ecc. con **baud rate** settabile da software, fino a **115 KBaud**.
- **24 linee di I/O TTL**, settabili da software, gestite dal **PPI 82C55**.
- **1 timer** a 16 bits e **2 timer counter** da 16 bits in grado di generare segnali **PWM**.
- **8 linee di A/D Converter** con **Sample & Hold**, 5,5 µs, range 0÷±2,048V con possibilità di lavorare in differenziale(±2,048V), 12 bits+segno, gestite dal potente **LM 12H458**. Sviluppa oltre **140.000** conversioni al secondo, dispone di un Sequencer interno, del trasferimento dati in **DMA**, funzione di **Self-Calibration** e programmazione del **Conversion-Rates**. Ha la possibilità di monitorare autonomamente un ingresso analogico generando un **INT** quando questo esce dai limiti impostati.
- Circuiteria di **power failure** in grado di generare interrupt.
- **Real Time Clock** in grado di gestire giorno, mese, anno, giorno della settimana, ore, minuti, secondi e di generare un **INT** con cadenze definibili da software.
- **Watch Dog** resettabile da software visualizzato tramite **LED**.
- Circuiteria di **back up** per RAM e RTC con batteria al **Litio** e connettore per eventuale batteria esterna. Lo stato di carica é acquisibile via software.
- Unica tensione di alimentazione a **5Vdc, 386mA**.
- Vasta disponibilità di software di base e di **ambienti di sviluppo** che consentono di poter utilizzare la scheda tramite un normale PC. Tra i pacchetti disponibili si possono citare: **GDOS 188; PASCAL 188; Monitor Debugger; GCTR 188; ROM-DOS; HI TECH C 86; GET 188; FLASH WRITER; DDS MICRO C 86**; ecc.

Viene di seguito riportata una descrizione dei blocchi funzionali della scheda, con indicate le operazioni effettuate da ciascuno di essi. Per una più facile individuazione di tali blocchi e per una verifica delle loro connessioni, fare riferimento alla figura 1.

PROCESSORE DI BORDO

La scheda **GPC® 188F** é predisposta per accettare il processore **80C188** prodotto dalla **INTEL**. Tale processore a 16 bit é codice compatibile con la diffusissima famiglia Intel 86 ed é quindi caratterizzato da: un esteso set di istruzioni, un'alta velocità di esecuzione, un'efficiente gestione degli interrupts, una vasta scelta di modalità d'indirizzamento, una efficacissima manipolazione dei dati, ecc. Di fondamentale importanza é la presenza delle seguenti periferiche interne al microprocessore:

- 2 Timer Counter ed 1 Timer a 16 bit, con possibilità di generare forme d'onda;
- 2 canali di DMA per trasferimenti dati ad alta velocità;
- sezione di controllo interrupt;
- sezione di gestione basso consumo;
- sezione di generazione segnali di controllo periferiche e memorie.

Per maggiori informazioni sul componente si faccia riferimento all'apposita documentazione della casa costruttrice, oppure all'appendice B di questo manuale.

MEMORIE

E' possibile dotare la scheda di un massimo di 2056K di memoria variamente suddivisi con un massimo di 1024K Byte di EPROM o 512K di FLASH EPROM, 1024K Byte di RAM statica ed infine 8K Byte di EEPROM seriale. Particolarmente interessante é la possibilità di proteggere fino a 512K di RAM in scrittura; tale protezione é selezionabile tramite un apposito jumper e consente di prevenire eventuali scritture accidentali all'interno della stessa RAM.

La scelta della configurazione delle memorie presenti sulla scheda può avvenire in relazione all'applicazione da risolvere e quindi alle esigenze dell'utente. Da questo punto di vista si ricorda che la scheda viene normalmente fornita con 128KByte SRAM di lavoro più 512 Byte di EEPROM seriale e che tutte le rimanenti configurazioni di memoria devono essere quindi opportunamente specificate in fase di ordine della scheda. Sfruttando la circuiteria di back up di bordo più la batteria tampone esterna, si ha la possibilità di mantenere i dati anche in assenza di alimentazione. Questa caratteristica fornisce alla scheda la possibilità di ricordare in ogni condizione, una serie di parametri come ad esempio la configurazione o lo stato del sistema, anche per lunghi periodi di inattività. Il mappaggio delle risorse di memoria avviene tramite una opportuna circuiteria di bordo, che provvede ad allocare i dispositivi all'interno dello spazio d'indirizzamento del microprocessore. Per maggiori informazioni fare riferimento al capitolo "MAPPAGGI ED INDIRIZZAMENTI" e "DESCRIZIONE SOFTWARE DELLE PERIFERICHE DI BORDO". Per una descrizione più approfondita sui dispositivi di memoria, sugli zoccoli da utilizzare e sullo strappaggio della scheda, fare riferimento al paragrafo "SELEZIONE MEMORIE".

CLOCK

Il segnale di clock della scheda **GPC® 188F** é generato dallo stesso processore che provvede a generarlo partendo da un quarzo e dimezzandone la frequenza. Si ricorda che la frequenza di clock per la CPU ha un valore standard di **20 MHz**, che viene utilizzata anche da altre sezioni della scheda (Timer, DMA, ecc.) e che si può intervenire via software sull'apposita sezione di power management per ridurne i valori e quindi i consumi.

A bordo scheda sono inoltre presenti altre due circuiterie di generazione segnali di clock di cui una da **11,0592 MHz** pilota direttamente il SCC, per la gestione delle due linee seriali, ed una da **8 MHz** per la sezione di A/D converter. La scelta di utilizzare circuiterie di clock diverse per le varie sezioni della scheda semplifica notevolmente l'uso della stessa, infatti ad esempio l'utente può variare la frequenza di lavoro della CPU senza dover intervenire sul firmware di gestione delle altre sezioni.

COMUNICAZIONE SERIALE

La comunicazione seriale é completamente settabile via software per quanto riguarda sia il protocollo fisico sia la velocità (da un minimo di 50 ad un massimo di 115200 Baud) ed in modo completamente autonomo per entrambe le linee di comunicazione. Di particolare importanza é la possibilità di gestire le due linee di comunicazione anche tramite protocolli logici SDLC ed HDLC avendo quindi la possibilità di utilizzare la scheda all'interno di una rete composta da più unità intelligenti. Tali settaggi avvengono tramite la programmazione degli appositi registri del **SCC 85C30**, quindi per ulteriori informazioni si faccia riferimento alla documentazione tecnica della casa costruttrice o al paragrafo "DESCRIZIONE SOFTWARE DELLE PERIFERICHE DI BORDO", di questo manuale.

Dal punto di vista hardware é invece possibile selezionare, tramite una serie di comodi jumpers, il protocollo elettrico di comunicazione. In particolare una linea é sempre bufferata in RS 232, mentre la rimanente può essere bufferata in RS 232, current loop, RS 485 ed RS 422; in quest'ultimo caso é definibile anche se la comunicazione avviene in Full Duplex o Half Duplex.

BUS ABACO®

Una delle caratteristiche di fondamentale importanza della **GPC® 188F** è quella di disporre dell'interfacciamento nei confronti del BUS industriale **ABACO®**: ovvero un connettore normalizzato con cui è possibile collegare la scheda ad una serie di moduli esterni intelligenti e non. Tra questi si trovano moduli per acquisizione di segnali analogici (A/D), per la generazione di segnali analogici (D/A), per gestione di linee di I/O logico, per counter, ecc. e ne possono essere realizzati anche su specifiche richieste dell'utente. Utilizzando mother board come l'**ABB 03** o l'**ABB 05** é inoltre possibile gestire anche le schede periferiche della serie 3 e 4 provviste di **ABACO® I/O BUS**. Tale caratteristica rende la scheda espandibile con un ottimo rapporto prezzo/prestazioni e quindi adatta a risolvere molti dei problemi dell'automazione industriale.

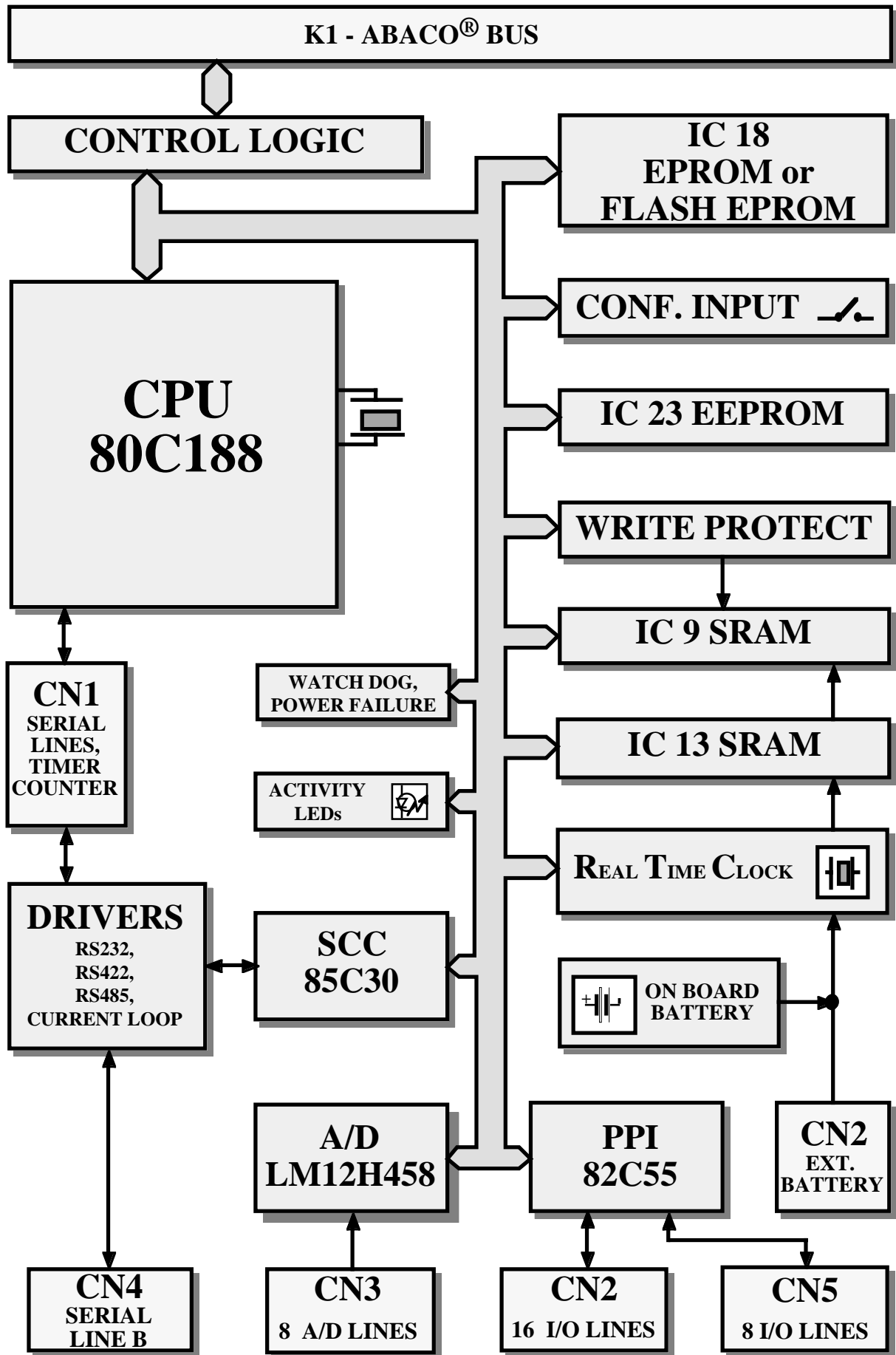


FIGURA 1: SCHEMA A BLOCCHI

LINEE DI I/O DIGITALI

Sulla scheda sono disponibili 24 linee di I/O digitale a livello TTL, raggruppati in tre port da 8 bit con direzionalità settabile a livello di byte, gestite dal Programmable Peripheral Interface **82C55**. Tali linee sono collegate direttamente a due connettori a 20 vie con pin out standardizzato **I/O ABACO®** ed hanno quindi la possibilità di essere direttamente collegate a numerose schede d'interfaccia, aprendo nuove possibilità d'impiego della **GPC® 188F**.

Via software é definibile la direzionalità e la funzionalità di queste linee, tramite una semplice programmazione di 4 registri situati nello spazio di I/O della CPU.

REAL TIME CLOCK

La **GPC® 188F** dispone di un completo Real Time Clock in grado di gestire ore, minuti, secondi, giorno del mese, mese, anno e giorno della settimana in modo completamente autonomo. Il componente é alimentato dalla circuiteria di back up in modo da garantire la validità dei dati in ogni condizione operativa ed é completamente gestito via software, tramite la programmazione di 16 registri situati nello spazio di I/O della CPU dall'apposita logica di controllo. La sezione di RTC può inoltre generare interrupt in corrispondenza di intervalli di tempo programmabili via software, in modo da poter periodicamente distogliere la CPU dalle normali operazioni oppure periodicamente risvegliarla dagli stati di basso consumo.

WATCH DOG

La scheda **GPC® 188F** é provvista di una circuiteria di watch dog che, se utilizzata, consente di uscire da stati di loop infinito o da condizioni anomale non previste dal programma applicativo. Tale circuiteria é composta da una sezione astabile con un tempo d'intervento di circa 1400 msec.

Tutta la gestione avviene via software (tramite l'accesso ad un opportuno registro situato nello spazio d'indirizzamento della CPU) e conferisce al sistema basato sulla scheda, una sicurezza estrema.

A/D CONVERTER

La sezione di A/D converter della **GPC® 188F** é basata sul potente **LM 12H458** in grado di acquisire 8 canali nel range 0÷2,048V oppure 4 canali differenziali nel range $\pm 2,048V$, con una risoluzione massima di 12 bits più segno. La sezione é provvista di **Sample & Hold**, di un A/D converter ad approssimazioni successive, con 5,5 μs di tempo di conversione, ed é in grado di sviluppare oltre **140.000** conversioni al secondo. Alcune caratteristiche come: un **sequenzer** interno, il trasferimento dati in **DMA**, la funzione di **self calibration**, la programmazione del **conversion rates**, il settaggio della **risoluzione**, una FIFO per le conversioni ed il controllo autonomo di **limiti** (monitorizza un ingresso analogico generando un interrupt quando questo esce dai valori impostati), facilitano notevolmente la sua gestione senza continuamente richiedere l'intervento della CPU.

Dal punto di vista software sono programmabili tutte le funzionalità del componente tramite 27 registri situati nello spazio di I/O.

SPECIFICHE TECNICHE

CARATTERISTICHE GENERALI

| | |
|------------------------------------|--|
| Risorse della scheda: | 24 input/output digitali TTL (PPI 82C55) 2 timer counter a 16 bit 1 timer a 16 bit 2 canali trasferimento dati (DMA) 1 linea seriale RS 232 (SCC A) 1 linea seriale RS 232, RS 422, RS 485, current loop (SCC B) 8 linee di A/D converter 1 watch dog hardware astabile 1 real time clock 4 jumper di configurazione 1 dip switch ad 8 vie acquisibile via software 1 interfaccia BUS ABACO® |
| Memoria indirizzabile: | IC 18: EPROM da 128K x 8 a 1024K x 8 FLASH EPROM da 128K x 8 a 512K x 8 IC 13: SRAM da 128K x 8 a 512K x 8 IC 9: SRAM da 128K x 8 a 512K x 8 IC 23: EEPROM seriale da 256 byte a 8K byte |
| CPU di bordo: | Intel 80C188 |
| Frequenza clock CPU: | 20 MHz |
| Frequenza clock SCC: | 11,0592 MHz |
| Frequenza clock A/D: | 8 MHz |
| Risoluzione A/D: | 12 bit + segno |
| Tempo conversione A/D: | 5,5 µsec |
| Tempo intervento watch dog: | da 940 msec a 2060 msec (tipico 1420 msec) |

CARATTERISTICHE FISICHE

| | |
|--------------------------------|---|
| Dimensioni (L x A x P): | formato EUROPA: 100 x 160 x 15 mm |
| Peso: | 220 g (versione base) |
| Connettori: | K1: 64 pin DIN 41612 corpo C CN1: 16 vie scatolino verticale M CN2: 20 vie scatolino 90 gradi M |

| | |
|------|------------------------------|
| CN3: | 20 vie scatolino 90 gradi M |
| CN4: | 10 vie scatolino 90 gradi M |
| CN5: | 20 vie scatolino verticale M |
| CN6: | 2 vie scatolino verticale M |

| | | |
|------------------------------|----------------------------|------------------|
| Range di temperatura: | da 0 a 70 gradi Centigradi | |
| Umidità relativa: | 20% fino a 90% | (senza condensa) |

CARATTERISTICHE ELETTRICHE

| | | |
|---|---|--------|
| Tensione di alimentazione: | +5 Vcc | |
| Corrente assorbita sui 5 Vdc: | 390 mA nella configurazione base 440 mA nella configurazione massima | |
| Batteria di bordo di back up: | 3,0 Vdc; 1/2 AA | |
| Batteria esterna di back up: | 3,6÷5 Vdc | |
| Corrente di back up: | 3,8 µA (batteria di bordo) 5,5 µA (batteria esterna da 3,6 Vdc) | |
| Ingressi analogici: | 0÷2,048 V; ±2,048 V | |
| Impedenza ingressi analogici: | < 4K Ω | |
| Rete terminazione RS 422-485: | Resistenza terminazione linea= | 120 Ω |
| | Resistenza di pull up sul positivo= | 3,3 KΩ |
| | Resistenza di pull down sul negativo= | 3,3 KΩ |
| Soglia d'intervento power failure: | 52 mV prima dell'intervento del reset | |

INSTALLAZIONE

In questo capitolo saranno illustrate tutte le operazioni da effettuare per il corretto utilizzo della scheda. A questo scopo viene riportata l'ubicazione e la funzione degli strip, dei connettori, dei trimmers, dei LEDs, ecc. presenti sulla **GPC® 188F**.

CONNESSIONI CON IL MONDO ESTERNO

Il modulo **GPC® 188F** è provvisto di 7 connettori con cui vengono effettuati tutti i collegamenti con il campo e con le altre schede del sistema di controllo da realizzare. Di seguito viene riportato il loro pin out ed il significato dei segnali collegati; per una facile individuazione di tali connettori, si faccia riferimento alla figura 20, mentre per ulteriori informazioni a riguardo del tipo di connessioni, fare riferimento alle figure successive che illustrano il tipo di collegamento effettuato a bordo scheda.

CN6 - CONNETTORE PER BATTERIA ESTERNA DI BACK UP

CN6 é un connettore a scatolino, verticale, maschio, con passo 2,54 mm a 2 vie.
Tramite CN6 può essere collegata una batteria esterna che provvede a mantenere i dati delle SRAM di bordo ed a garantire il funzionamento del real time clock, in assenza di tensione di alimentazione (per maggiori informazioni fare riferimento al paragrafo “BACK UP”).

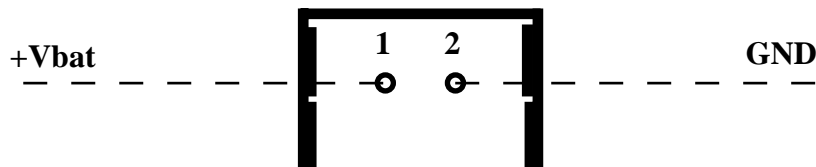


FIGURA 3: CN6 - CONNETTORE PER BATTERIA ESTERNA DI BACK UP

Legenda:

+Vbat = I - Positivo della batteria esterna di back up.
GND = - Negativo della batteria esterna di back up.

CN5 - CONNETTORE PER PORT B PPI 82C55

CN5 è un connettore a scatolino verticale con passo 2.54 mm a 20 piedini.

Tramite CN5 si effettua la connessione delle 8 linee di I/O digitale gestite dal port B del PPI 82C55, con il campo. Tutti i parametri che riguardano l'uso di questo componente (direzionalità, modo di gestione dati, ecc.) sono definibili via software tramite la programmazione della stesso. I segnali del PPI coincidono con segnali logici a livello TTL e seguono il pin out standardizzato I/O **ABACO®**.

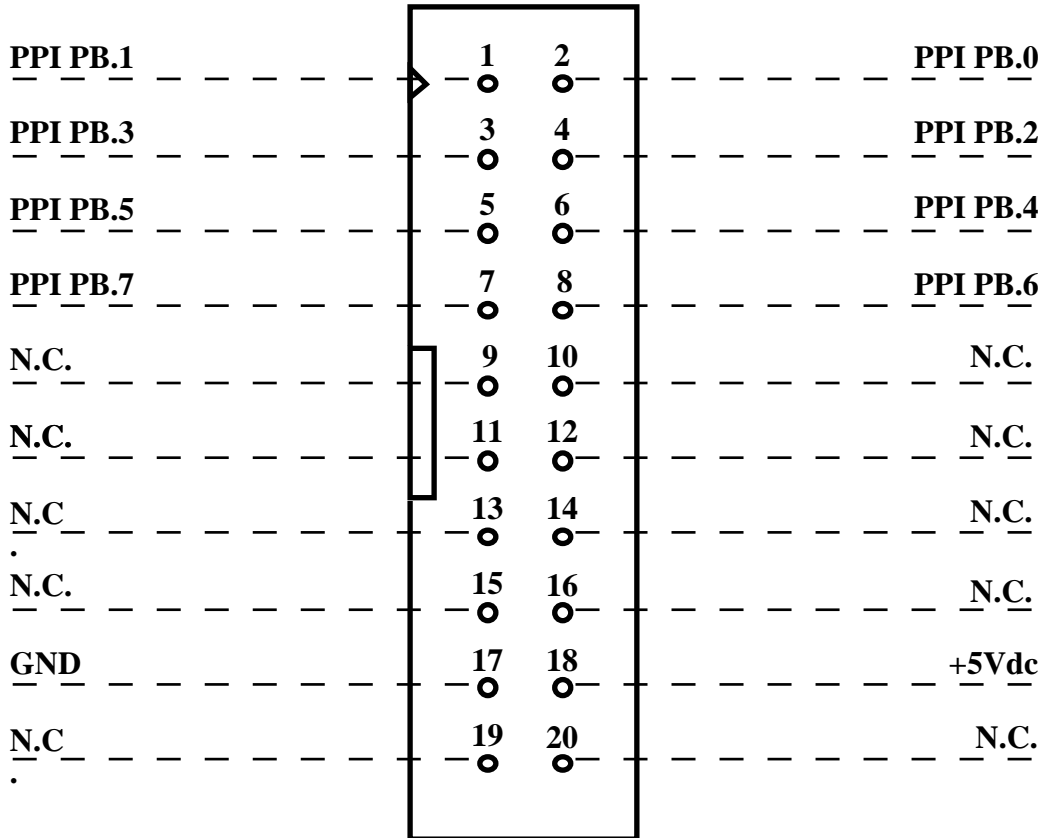


FIGURA 4: CN5 - CONNETTORE PER PORT B PPI 82C55

Legenda:

- PPI PB.n** = I/O - Linea digitale TTL n del port B del PPI 82C55
- +5 Vdc** = O - Linea di alimentazione a +5 Vcc
- GND** = - Linea di massa digitale
- N.C.** = - Non collegato

CN2 - CONNETTORE PER PORT A E C DEL PPI 82C55

CN2 é un connettore a scatolino a 90 gradi con passo 2.54 mm a 20 piedini.

Tramite CN2 si effettua la connessione delle 16 linee di I/O digitale gestite dai port A,C del PPI 82C55, con il campo. Tutti i parametri che riguardano l'uso di questo componente (direzionalità, modo di gestione dati, ecc.) sono definibili via software tramite la programmazione della stesso. I segnali del PPI coincidono con segnali logici a livello TTL e seguono il pin out standardizzato I/O **ABACO®**.

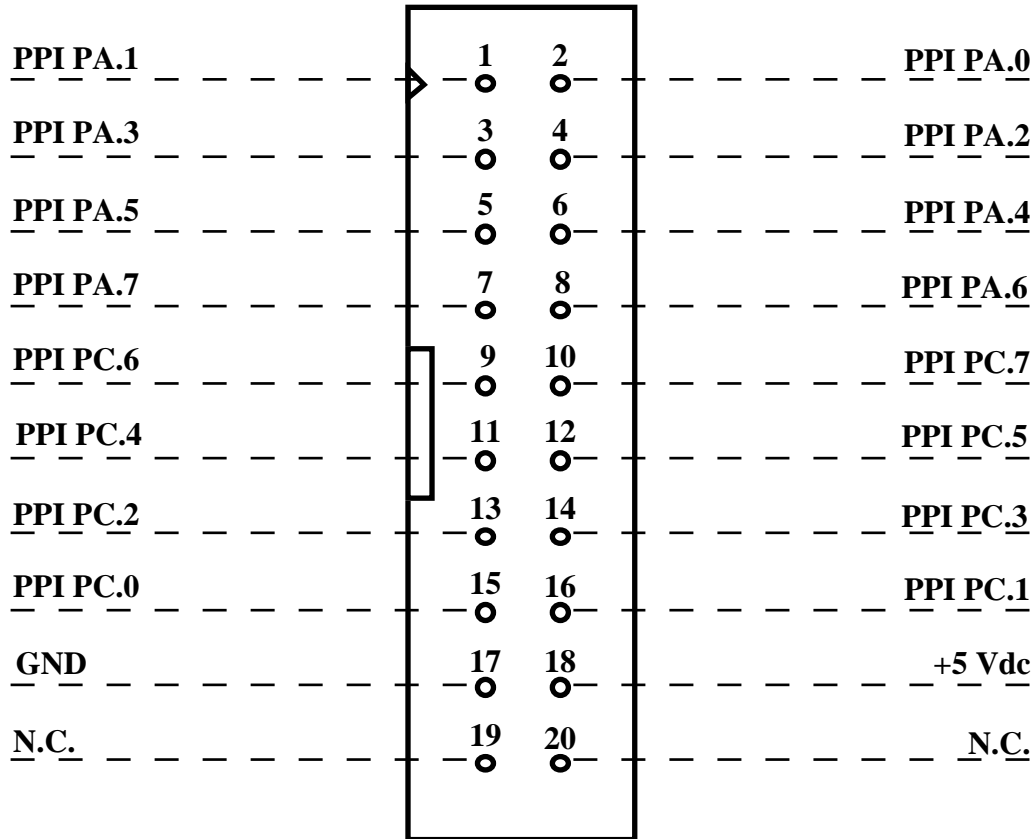


FIGURA 5: CN2 - CONNETTORE PER PORT A E C DEL PPI 82C55

Legenda:

| | |
|-----------------|---|
| PPI PA.n | = I/O - Linea digitale TTL n del port A del PPI 82C55 |
| PPI PC.n | = I/O - Linea digitale TTL n del port C del PPI 82C55 |
| +5 Vdc | = O - Linea di alimentazione a +5 Vcc |
| GND | = - Linea di massa digitale |
| N.C. | = - Non collegato |

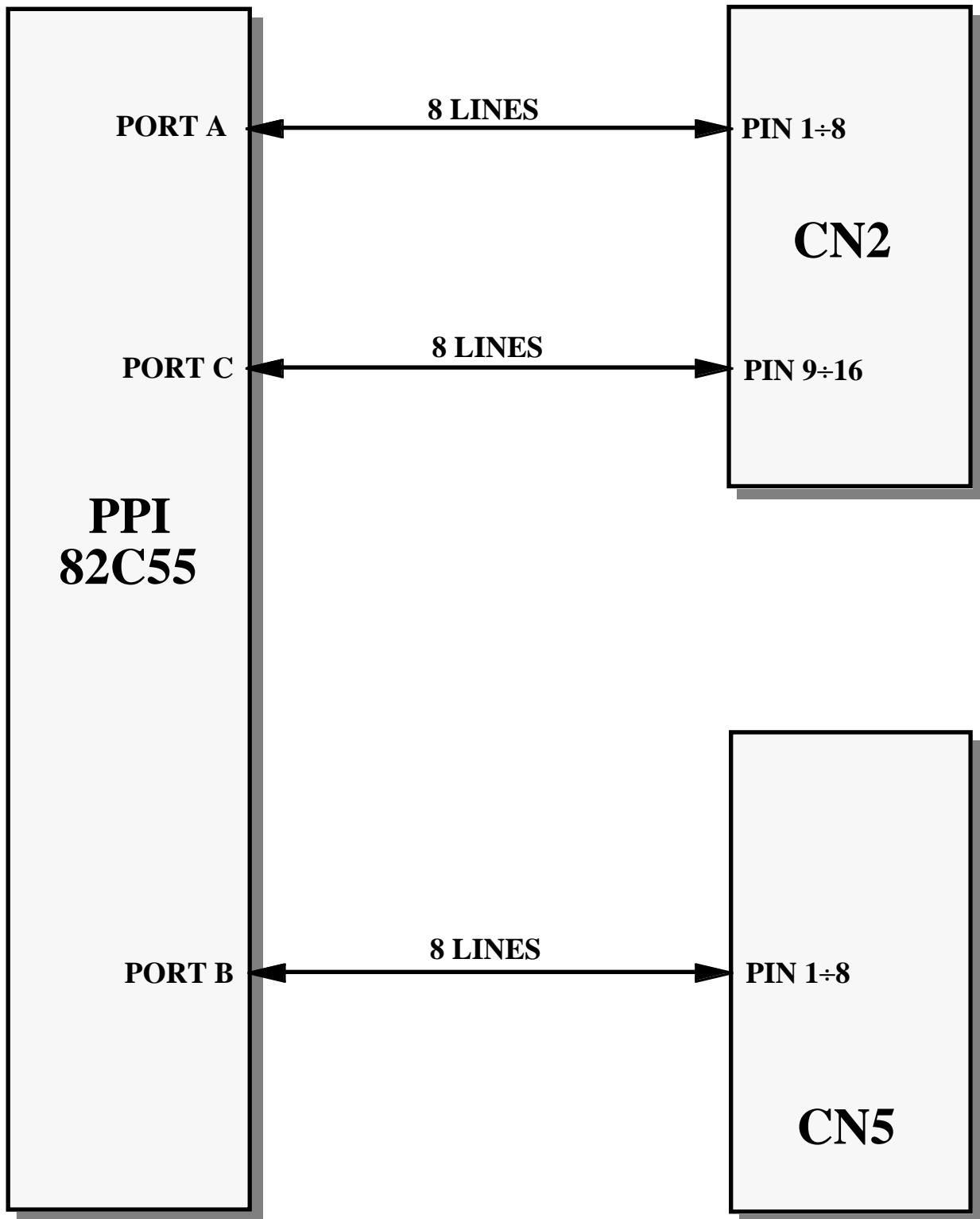


FIGURA 6: SCHEMA DI COLLEGAMENTO LINEE DI I/O DEL PPI

CN3 - CONNETTORE PER INGRESSI A/D CONVERTER

CN3 é un connettore a scatola a 90 gradi con passo 2.54 mm a 20 piedini.

Tramite CN3 possono essere collegate le 8 linee analogiche d'ingresso all'apposita circuiteria esterna. Tali linee sono direttamente collegate all'A/D di bordo della scheda, sono a bassa impedenza, sono provviste di un condensatore di filtro e possono variare nel range 0÷2,048 V nel caso di ingressi singoli o nel range $\pm 2,048$ V nel caso di ingressi differenziali. La gestione della conversione é completamente gestita via software tramite la programmazione dell'LM 12H458, mentre la disposizione dei segnali su questo connettore é studiata in modo da ridurre tutti i problemi di rumore ed interferenza, garantendo quindi un'ottima trasmissione del segnale.

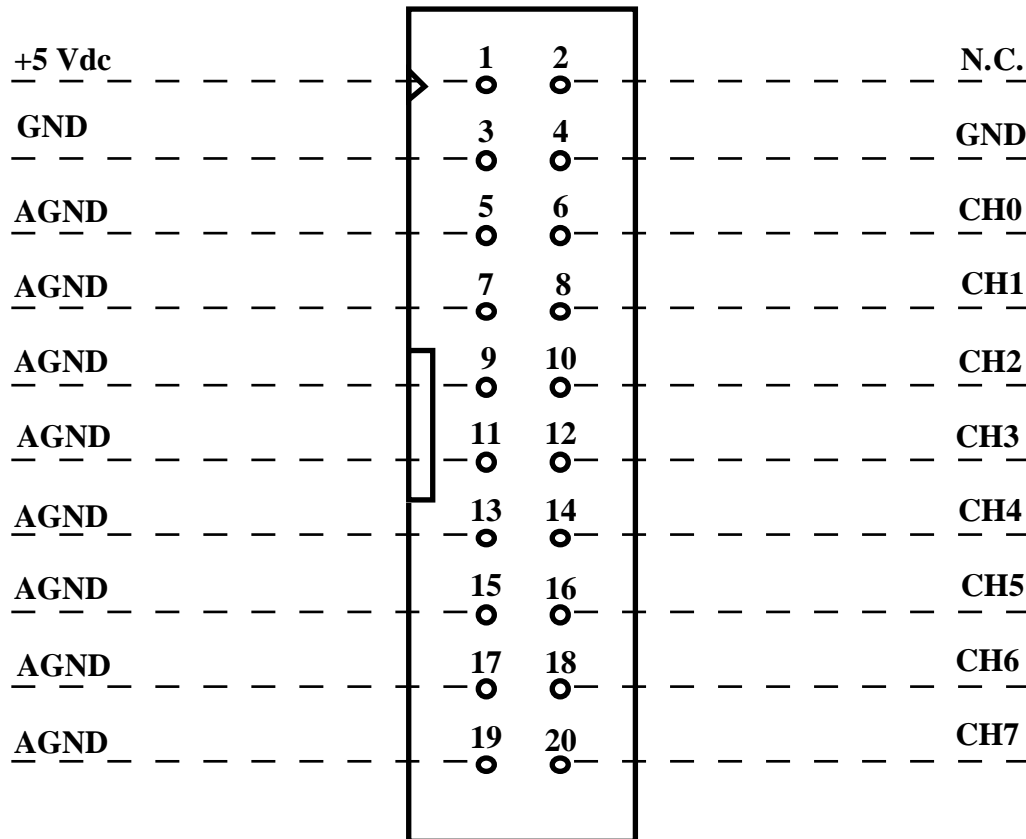


FIGURA 7: CN3 - CONNETTORE PER INGRESSI A/D CONVERTER

Legenda:

| | | |
|---------------|-----|---|
| CHn | = I | - Linea analogica d'ingresso collegata al canale n dell'A/D |
| AGND | = | - Linea di massa analogica |
| +5 Vdc | = O | - Linea di alimentazione a +5 Vcc |
| GND | = | - Linea di massa digitale |
| N.C. | = | - Non collegato |

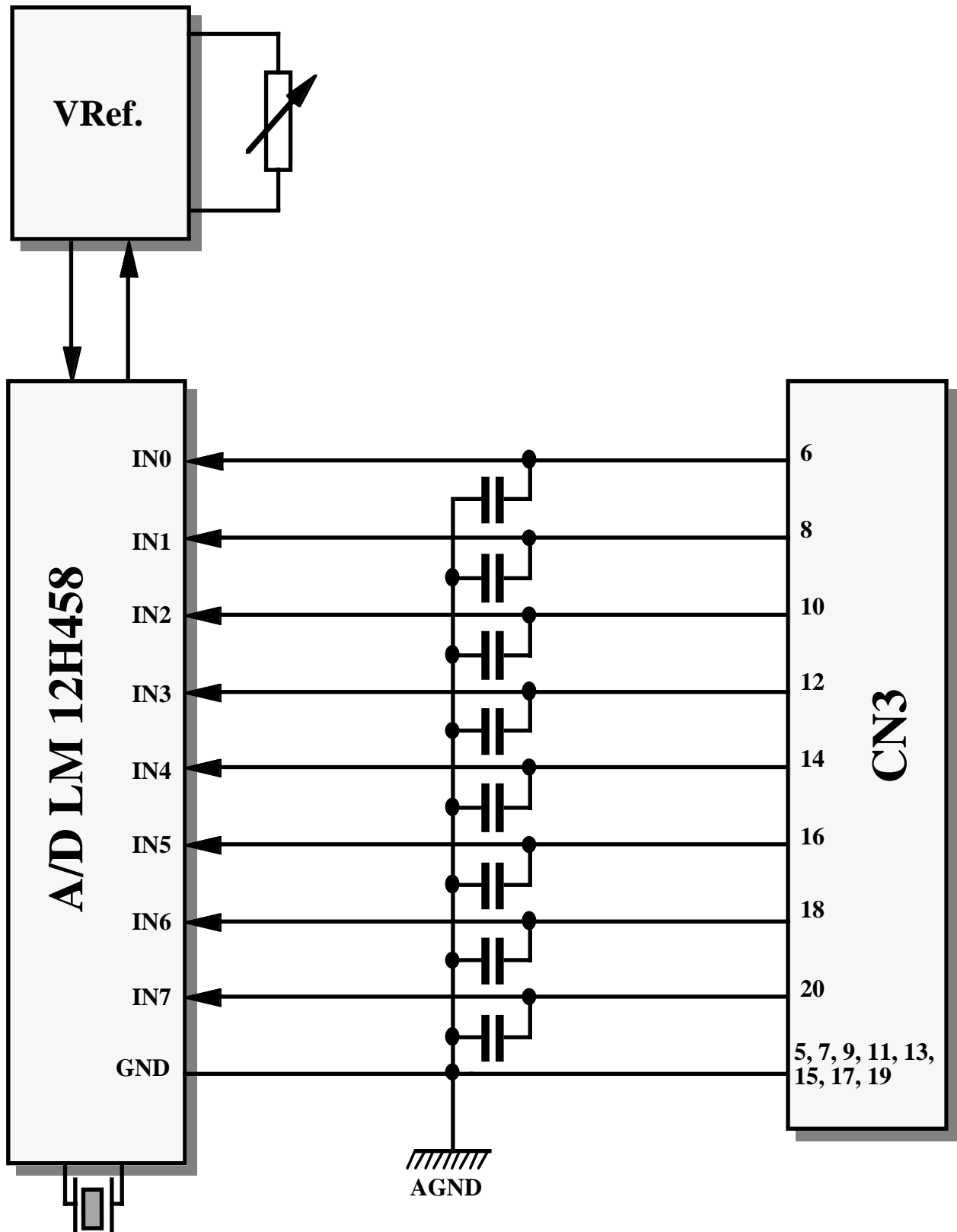


FIGURA 8: SCHEMA D'INGRESSO A/D CONVERTER

CN1 - CONNETTORE PER LINEE SERIALI RS 232 E TIMER COUNTER

CN1 è un connettore a scatola verticale con passo 2.54 mm a 16 piedini.

Tramite CN1 si effettua la connessione delle due linee seriali in RS 232 ed i due counter timer della scheda con l'ambiente esterno. Le due linee seriali sono gestibili via hardware e via software tramite lo strappaggio di appositi jumpers e la programmazione del SCC 85C30. I segnali presenti su questo connettore coincidono con segnali logici a livello TTL e segnali a livello RS 232, secondo le normative definite dal CCITT; la disposizione dei segnali, é invece stata studiata in modo da ridurre al minimo le interferenze ed in modo da facilitare la connessione con il campo

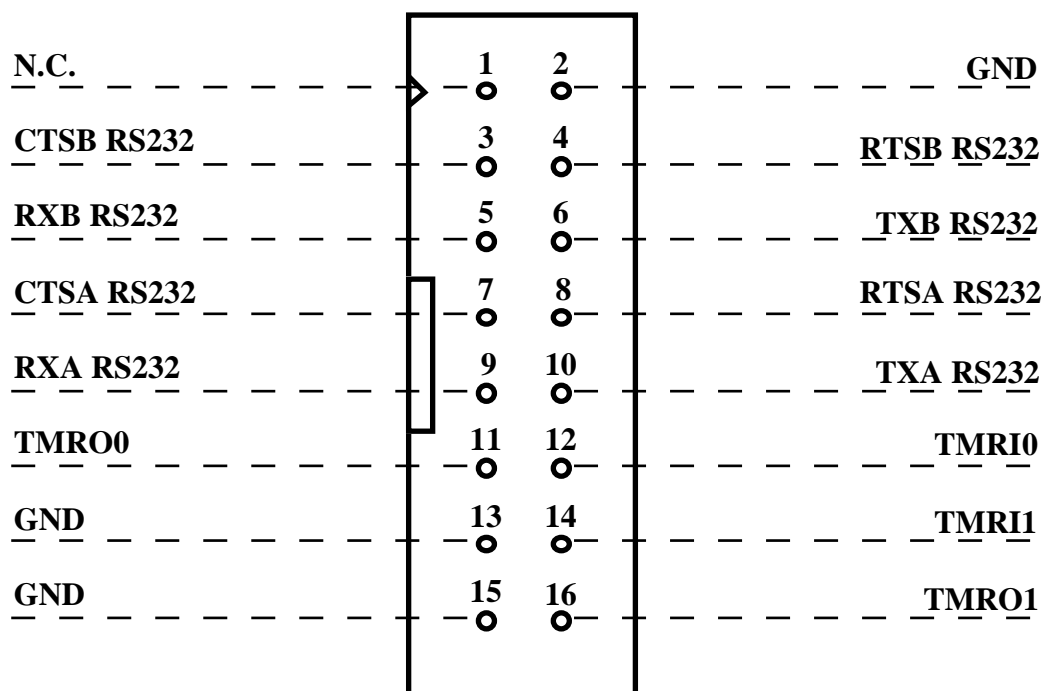


FIGURA 9: CN1 - CONNETTORE PER LINEE SERIALI RS 232 E TIMER COUNTER

Legenda:

- TMRI0,1 = I - Linee TTL d'ingresso timer counter 0,1
- TMRO0,1 = O - Linee TTL d'uscita timer counter 0,1
- CTSA,B RS232 = I - Clear To Send A,B: linee di abilitazione della trasmissione sulla linea seriale A,B in RS 232
- RTSA,B RS232 = O - Request To Send A,B: linee di richiesta di trasmissione sulla linea seriale A,B in RS 232
- RXA,B RS232 = I - Receive Data: linee di ricezione della linea seriale A,B in RS 232
- TXA,B RS232 = O - Transmit Data: linee di trasmissione della linea seriale A,B in RS 232
- GND = - Linea di massa digitale
- N.C. = - Non collegato

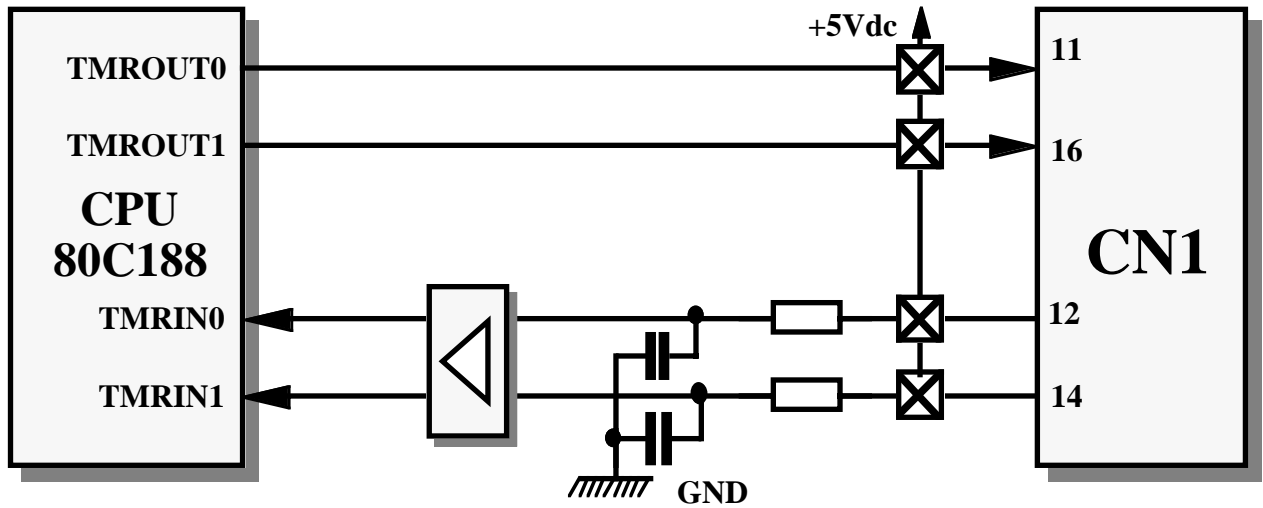


FIGURA 10: SCHEMA DI COLLEGAMENTO TIMER COUNTER

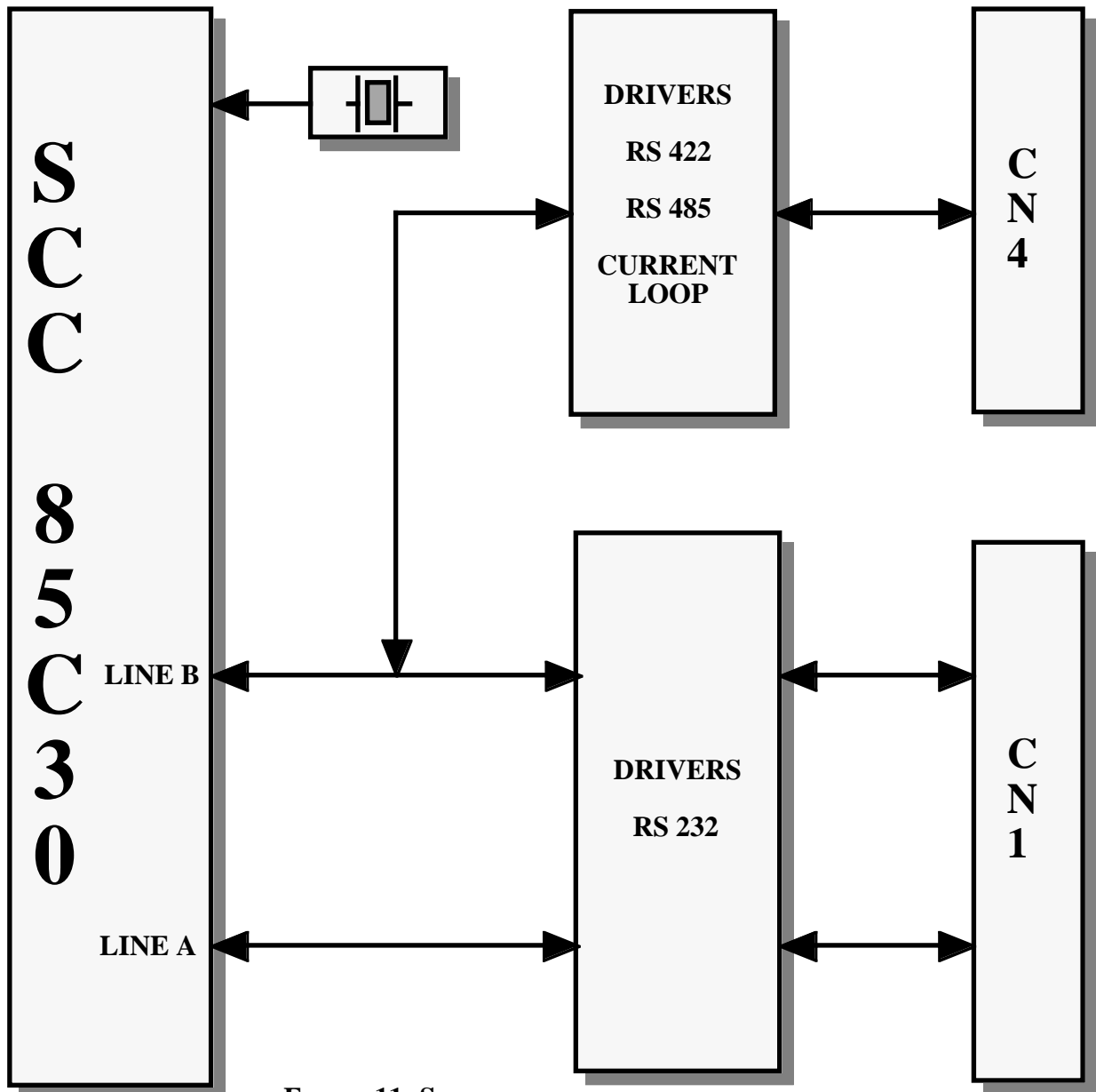


FIGURA 11: SCHEMA DI COMUNICAZIONE SERIALE

CN4 - CONNETTORE PER SERIALE IN RS 422, RS 485 E CURRENT LOOP

CN4 è un connettore a scatola a 90 gradi con passo 2.54 mm a 10 piedini.

Su questo connettore sono riportati tutti i segnali per la comunicazione in RS 422, RS 485 e current loop della linea seriale B. La disposizione di tali segnali è stata studiata in modo da ridurre al minimo le interferenze ed in modo da facilitare le connessioni con il campo, mentre i segnali rispettano le normative definite dal CCITT relative ad ognuno degli standard di comunicazione usati. Si ricorda inoltre che l'interfaccia current loop é di tipo passivo.

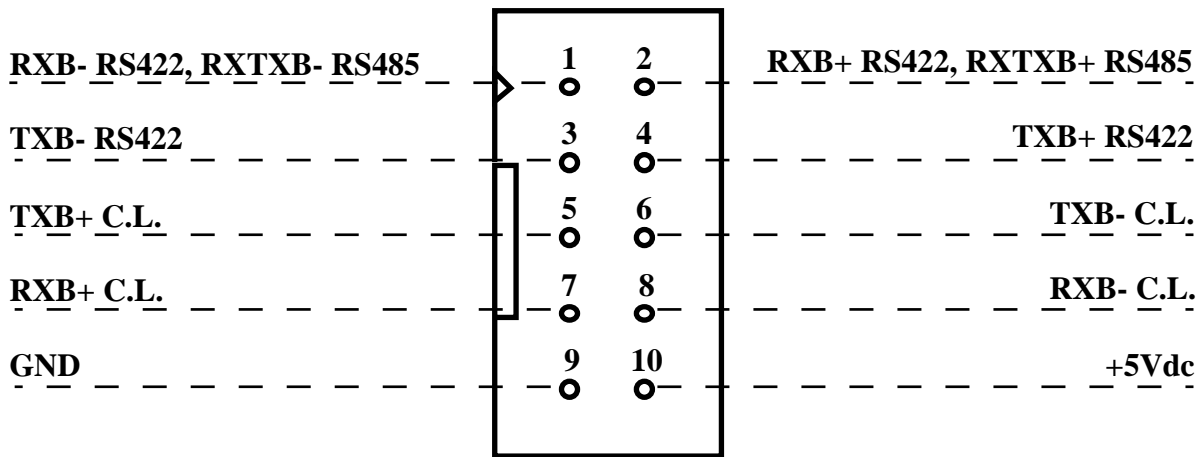


FIGURA 12: CN4 - CONNETTORE PER SERIALE IN RS 422, RS 485 E CURRENT LOOP

Legenda:

- RXB- RS422** = I - Receive Data Negative: linea bipolare negativa di ricezione differenziale in RS 422 della seriale B
- RXB+ RS422** = I - Receive Data Positive: linea bipolare positiva di ricezione differenziale in RS 422 della seriale B
- TXB- RS422** = O - Transmit Data Negative: linea bipolare negativa di trasmissione differenziale in RS 422 della seriale B
- TXB+ RS422** = O - Transmit Data Positive: linea bipolare positiva di trasmissione differenziale in RS 422 della seriale B
- RXTXB- RS485** = I/O - Receive Transmit Data Negative: linea bipolare negativa di ricezione e trasmissione differenziale in RS 485 della seriale B
- RXTXB+ RS485** = I/O - Receive Transmit Data Positive: linea bipolare positiva di ricezione e trasmissione differenziale in RS 485 della seriale B
- RXB- C.L.** = I - Receive Data Negative: linea bipolare negativa di ricezione in Current Loop della seriale B
- RXB+ C.L.** = I - Receive Data Positive: linea bipolare positiva di ricezione in Current Loop della seriale B
- TXB- C.L.** = O - Transmit Data Negative: linea bipolare negativa di trasmissione in Current Loop della seriale B
- TXB+ C.L.** = O - Transmit Data Positive: linea bipolare positiva di trasmissione in Current Loop della seriale B
- +5 Vdc** = I - Linea di alimentazione a +5 Vcc
- GND** = - Linea di massa digitale

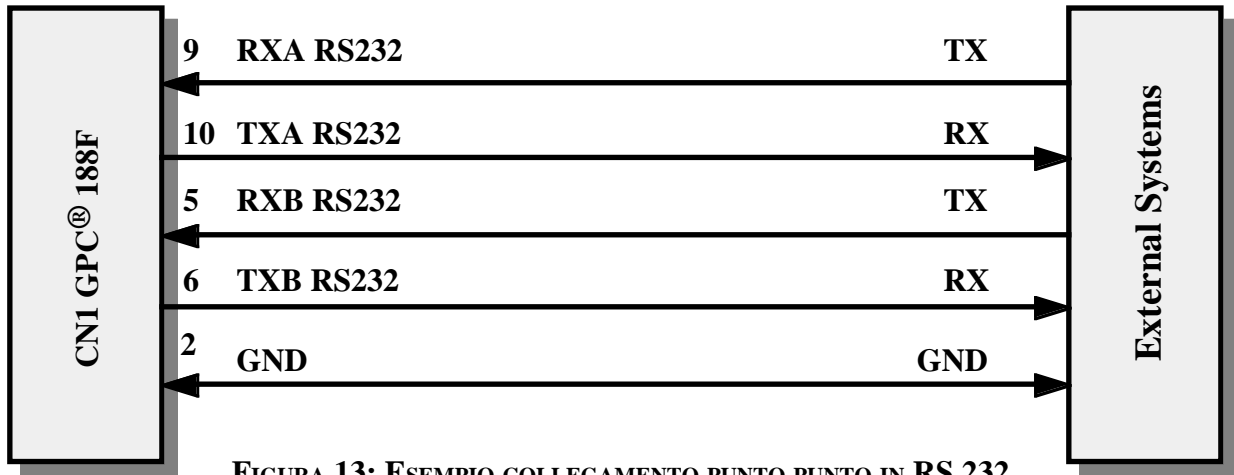


FIGURA 13: ESEMPIO COLLEGAMENTO PUNTO PUNTO IN RS 232

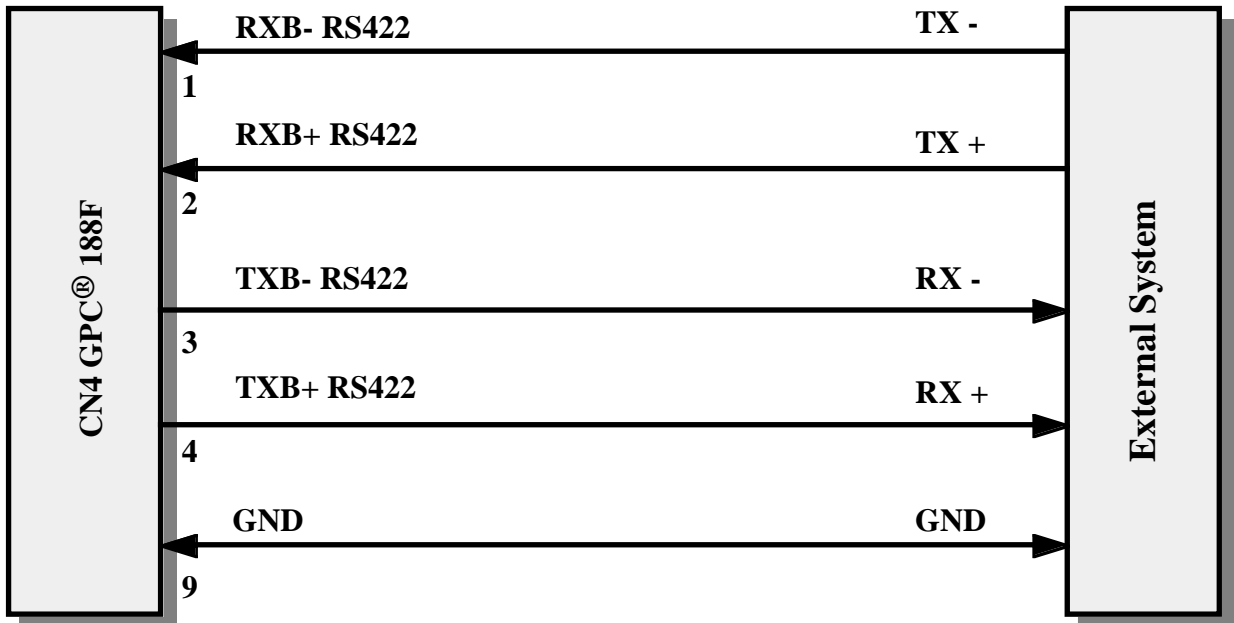


FIGURA 14: ESEMPIO COLLEGAMENTO PUNTO PUNTO IN RS 422

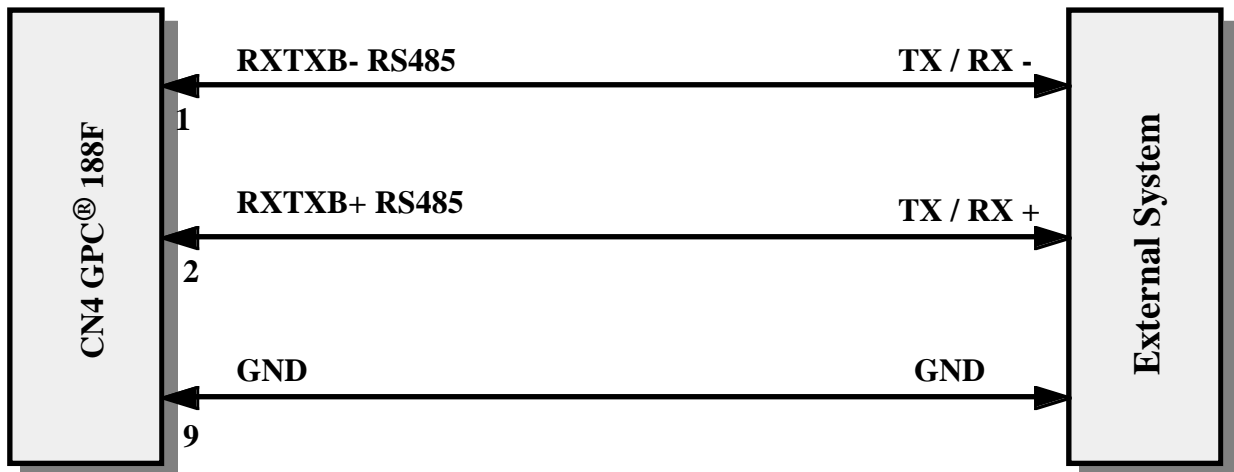


FIGURA 15: ESEMPIO COLLEGAMENTO PUNTO PUNTO IN RS 485

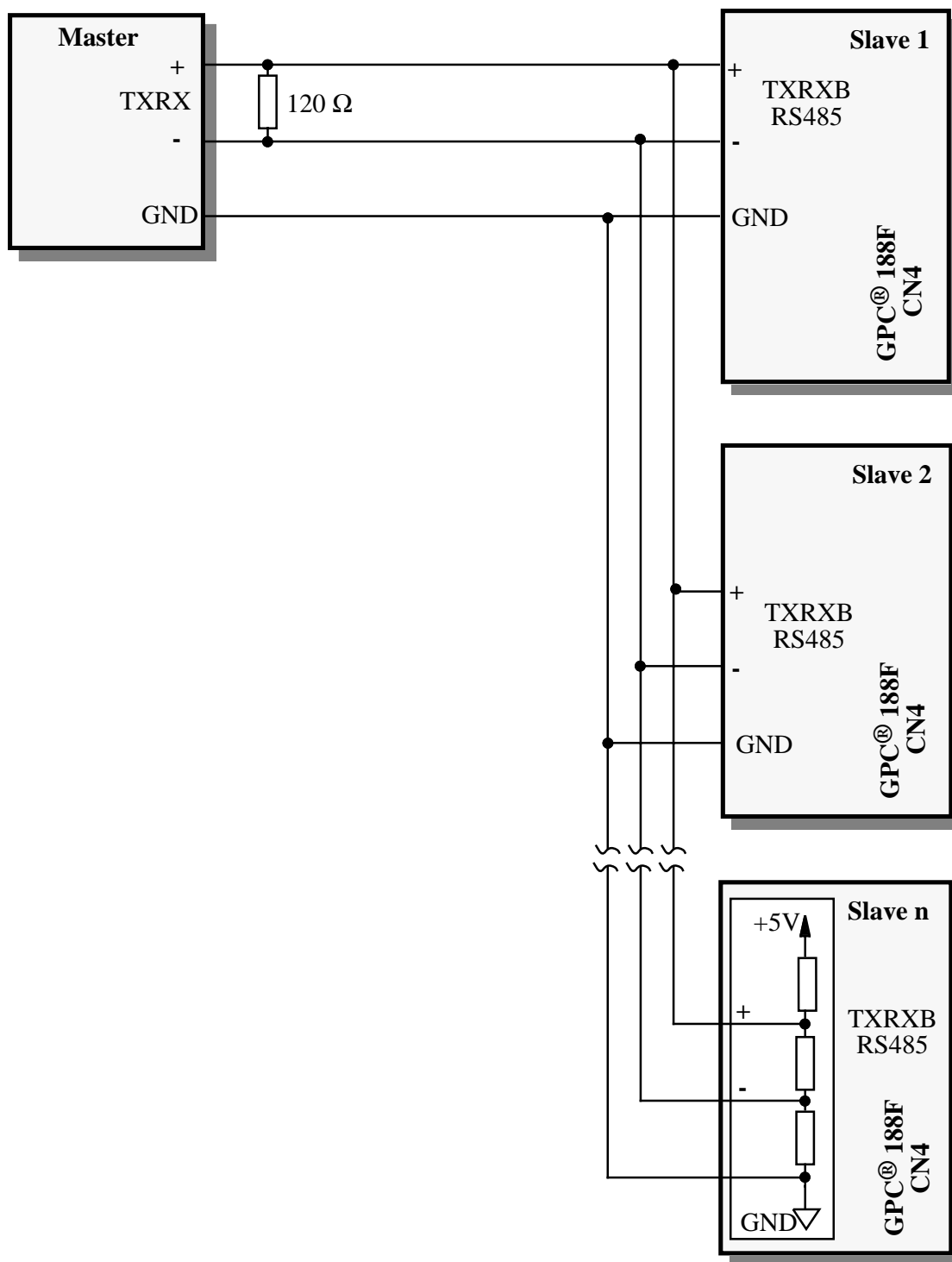


FIGURA 16: ESEMPIO COLLEGAMENTO IN RETE IN RS 485

Da notare che in una rete RS 485, devono essere presenti due resistenze di forzatura lungo la linea e due resistenze di terminazione (120Ω), alle estremità della stessa, rispettivamente vicino all'unità Master ed all'ultima unità Slave.

A bordo della **GPC® 188F** è presente la circuiteria di terminazione e forzatura, che può essere inserita o disinserita, tramite appositi jumpers, come illustrato in seguito.

In merito alla resistenza di terminazione dell'unità Master, provvedere a collegarla solo se questa non è già presente al suo interno (ad esempio molti convertitori RS232-RS485 ne sono già provvisti). Per maggiori informazioni consultare il Data-Book TEXAS INSTRUMENTS, "RS 422 and RS 485 Interface Cicuits", nella parte introduttiva riguardante le reti RS 422-485.

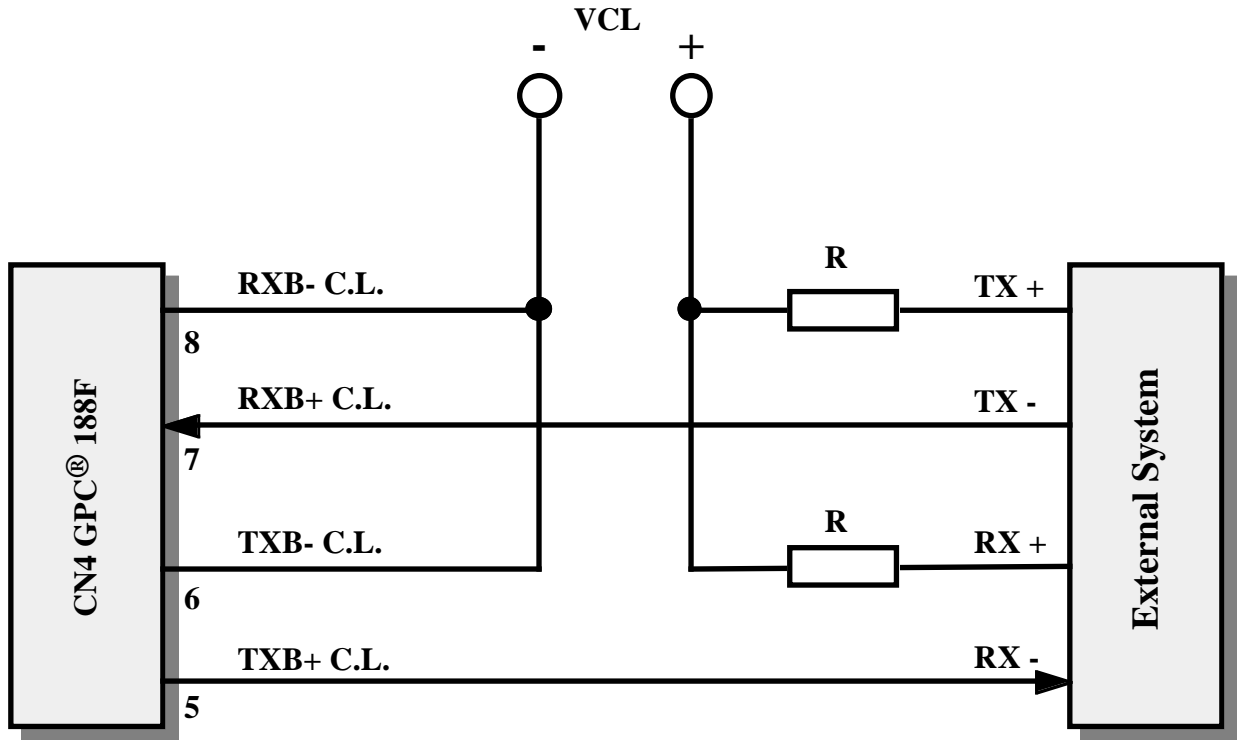


FIGURA 17: ESEMPIO COLLEGAMENTO PUNTO PUNTO IN CURRENT LOOP A 4 FILI

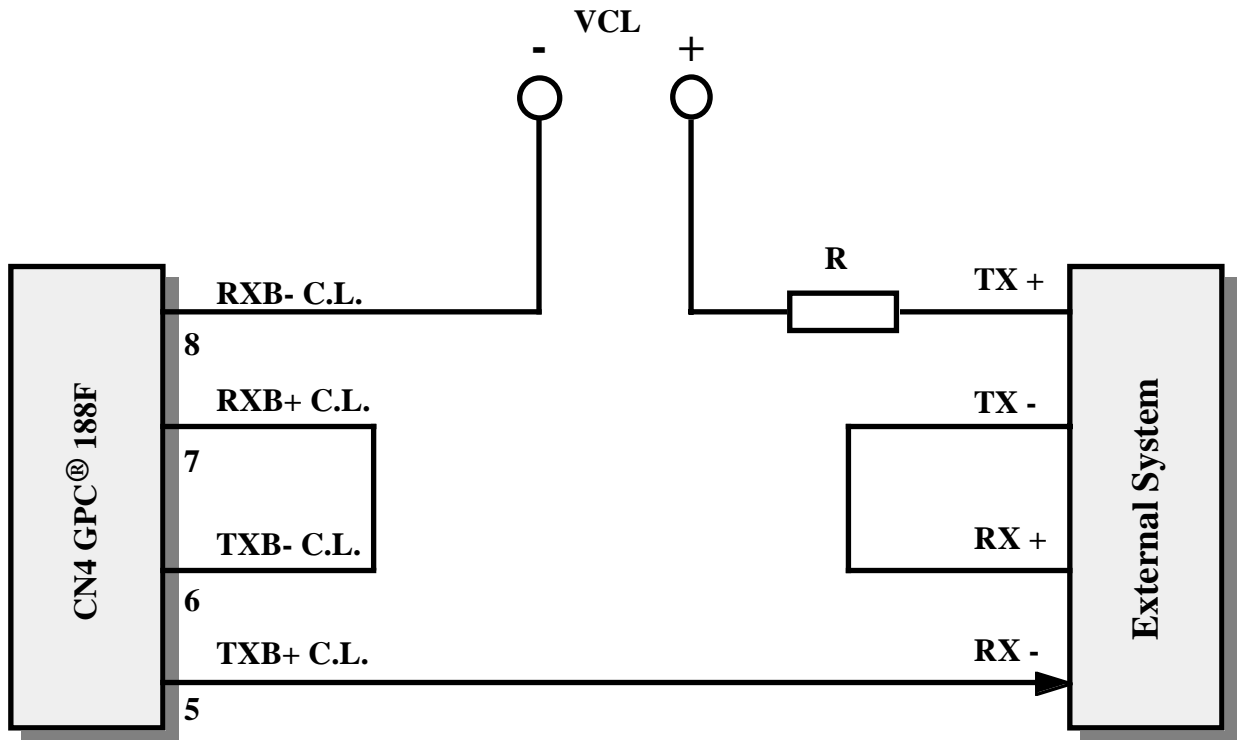


FIGURA 18: ESEMPIO COLLEGAMENTO PUNTO PUNTO IN CURRENT LOOP A 2 FILI

K1 - CONNETTORE PER BUS ABACO®

K1 è un connettore DIN 41612 corpo C a 90 gradi da 64 piedini.

Tramite K1 si effettua la connessione tra la scheda e la serie di moduli esterni di espansione, da utilizzare per l'interfacciamento diretto con il campo. Tale collegamento è effettuato tramite il BUS industriale **ABACO®** di cui questo connettore riporta i segnali a livello TTL. Nella figura seguente è riportato il pin out del BUS e quindi anche del relativo connettore, con le variazioni per l'utilizzo di CPU a 16 Bit rispetto a quelle a 8 Bit.

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

FIGURA 19: K1 - CONNETTORE PER BUS ABACO®

Legenda:

CPU a 8 bit

| | |
|----------------|---|
| A0-A15 | = O - Address BUS: BUS degli indirizzi. |
| D0-D7 | = I/O - Data BUS: BUS dei dati. |
| /INT | = I - Interrupt request: richiesta d'interrupt. |
| /NMI | = I - Non Mascherabile Interrupt: richiesta d'interrupt non mascherabile. |
| /HALT | = O - Halt state: stao di Halt. |
| /MREQ | = O - Memory Request: richiesta di operazione in memoria. |
| /IORQ | = O - Input Output Request: richiesta di operazione in Input Output. |
| /RD | = O - Read cycle status: richiesta di lettura. |
| /WR | = O - Write cycle status: richiesta di scrittura. |
| /BUSAK | = O - BUS Acknowledge: riconoscimento della richiesta di utilizzo del BUS. |
| /WAIT | = I - Wait: Attesa. |
| /BUSRQ | = I - BUS Request: richiesta di utilizzo del BUS. |
| /RESET | = O - Reset: azzeramento. |
| /M1 | = O - Machine cycle one: primo ciclo macchina. |
| /RFSH | = O - Refresh: rinfresco per memorie dinamiche. |
| /MEMDIS | = I - Memory Display: segnale emesso dal dispositivo periferico mappato in memoria. |
| VDUSEL | = O - VDU Selection: abilitazione per il dispositivo periferico ad essere mappato in memoria. |
| /IEI | = I - Interrupt Enable Input: abilitazione interrupt da BUS in catene di priorità. |
| CLK | = O - Clock: clock di sistema. |
| /R.T. | = I - Reset Tast: tasto di reset. |
| +5 Vdc | = I - Linea di alimentazione a +5 Vcc. |
| +12 Vdc | = O - Linea di alimentazione a +12 Vcc. |
| -12 Vdc | = O - Linea di alimentazione a -12 Vcc. |
| GND | = O - Linea di massa per tutti i segnali del BUS. |
| N.C. | = - Non Collegato |

CPU a 16 bit

| | |
|----------------|---|
| A0-A22 | = O - Address BUS: BUS degli indirizzi. |
| D0-D15 | = I/O - Data BUS: BUS dei dati. |
| /RD UDS | = O - Read Upper Data Strobe: lettura del byte superiore sul BUS dati. |
| /WR UDS | = O - Write Upper Data Strobe: scrittura del byte superiore sul BUS dati. |
| /IACK | = O - Interrupt Acknowledge: riconoscimento della richiesta d'interrupt da parte della CPU. |
| /RD LDS | = O - Read Lower Data Strobe: lettura del byte inferiore sul BUS dati. |
| /WR LDS | = O - Write Lower Data Strobe: scrittura del byte inferiore sul BUS dati. |

N.B.

Le indicazioni di direzionalità sopra riportate sono riferite ad una scheda di comando (CPU o GPC®) e sono state mantenute inalterate in modo da non avere ambiguità d'interpretazione nel caso di sistemi composti da più schede.

TRIMMER E TARATURE

Sulla **GPC® 188F** è presente il trimmer RV1 utilizzato per la taratura della scheda; tale componente permette di fissare il valore della tensione di riferimento su cui si basa la sezione di A/D converter. La scheda viene sottoposta ad un accurato test di collaudo che provvede a verificare la funzionalità della stessa ed allo stesso tempo a tararla in tutte le sue parti. La taratura viene effettuata in laboratorio a temperatura costante di +20 gradi centigradi, seguendo la procedura di seguito descritta:

- Si effettua la taratura di precisione della V_{ref} della sezione A/D ad un valore di 2,0480 V regolando il trimmer RV1, tramite un multimetro galvanicamente isolato a 5 cifre, sul test point TP1 .
- Si verifica la corrispondenza tra segnale analogico fornito in ingresso e combinazione letta dalla sezione A/D converter. La verifica viene effettuata fornendo un segnale di verifica con un calibratore campione e controllando che la differenza tra la combinazione determinata dalla scheda e quella determinata in modo teorico, non superi la somma degli errori della sezione A/D.
- Si blocca il trimmer della scheda, opportunamente tarato, tramite vernice.

Le sezioni d'interfaccia analogica utilizzano componenti di alta precisione che vengono addirittura scelti in fase di montaggio, proprio per evitare lunghe e complicate procedure di taratura. Per questo una volta completato il test di collaudo e quindi la taratura, il trimmer RV1 viene bloccato, in modo da garantire una immunità della taratura anche ad eventuali sollecitazioni meccaniche (vibrazioni, spostamenti, ecc.).

La circuiteria di generazione della tensione di riferimento definisce anche il fondo scala per tutti gli 8 canali di ingresso analogico; via software è possibile definire la modalità di acquisizione dei segnali tra "single ended" (8 ingressi riferiti ad AGND nel range $0 \div 2,048$ V) e "fully differential" (4 ingressi differenziali nel range $\pm 2,048$ V), come descritto nell'appendice B di questo manuale. La scelta di questo valore di fondo scala deve essere specificata in fase d'ordine della scheda, infatti implica il montaggio di diversi componenti ed una diversa procedura di taratura. In assenza di indicazioni, la scheda viene fornita nella versione standard con fondo scala a 2,048 V.

L'utente di norma non deve intervenire sulla taratura della scheda, ma se lo dovesse fare (a causa di derive termiche, derive del tempo, ecc.) deve rigorosamente seguire la procedura sopra illustrata. Per una facile individuazione di RV1 e TP1 a bordo scheda, si faccia riferimento alla figura 20.

INTERFACCIAMENTO DEGLI I/O CON IL CAMPO

Al fine di evitare eventuali problemi di collegamento della scheda con tutta l'elettronica del campo a cui la **GPC® 188F** si deve interfacciare, si devono seguire le informazioni riportate nei precedenti paragrafi e le relative figure che illustrano le modalità interne di connessione.

- Per i segnali che riguardano la comunicazione seriale con i protocolli RS 232, RS 422, RS 485 e current loop fare riferimento alle specifiche standard di ognuno di questi protocolli.
- Tutti i segnali a livello TTL possono essere collegati a linee dello stesso tipo riferite alla massa digitale della scheda. Il livello 0V corrisponde allo stato logico 0, mentre il livello 5V corrisponde allo stato logico 1.
- I segnali d'ingresso alla sezione A/D devono essere collegati a segnali analogici a bassa impedenza che rispettino il range di variazione ammesso che può essere $0 \div 2,048$ V o $\pm 2,048$ V a seconda della configurazione. Da notare che gli 8 ingressi analogici presenti su CN3 sono dotati di condensatori di filtro che garantiscono una maggiore stabilità sul segnale acquisito, ma che allo stesso tempo abbassano la frequenza di taglio.

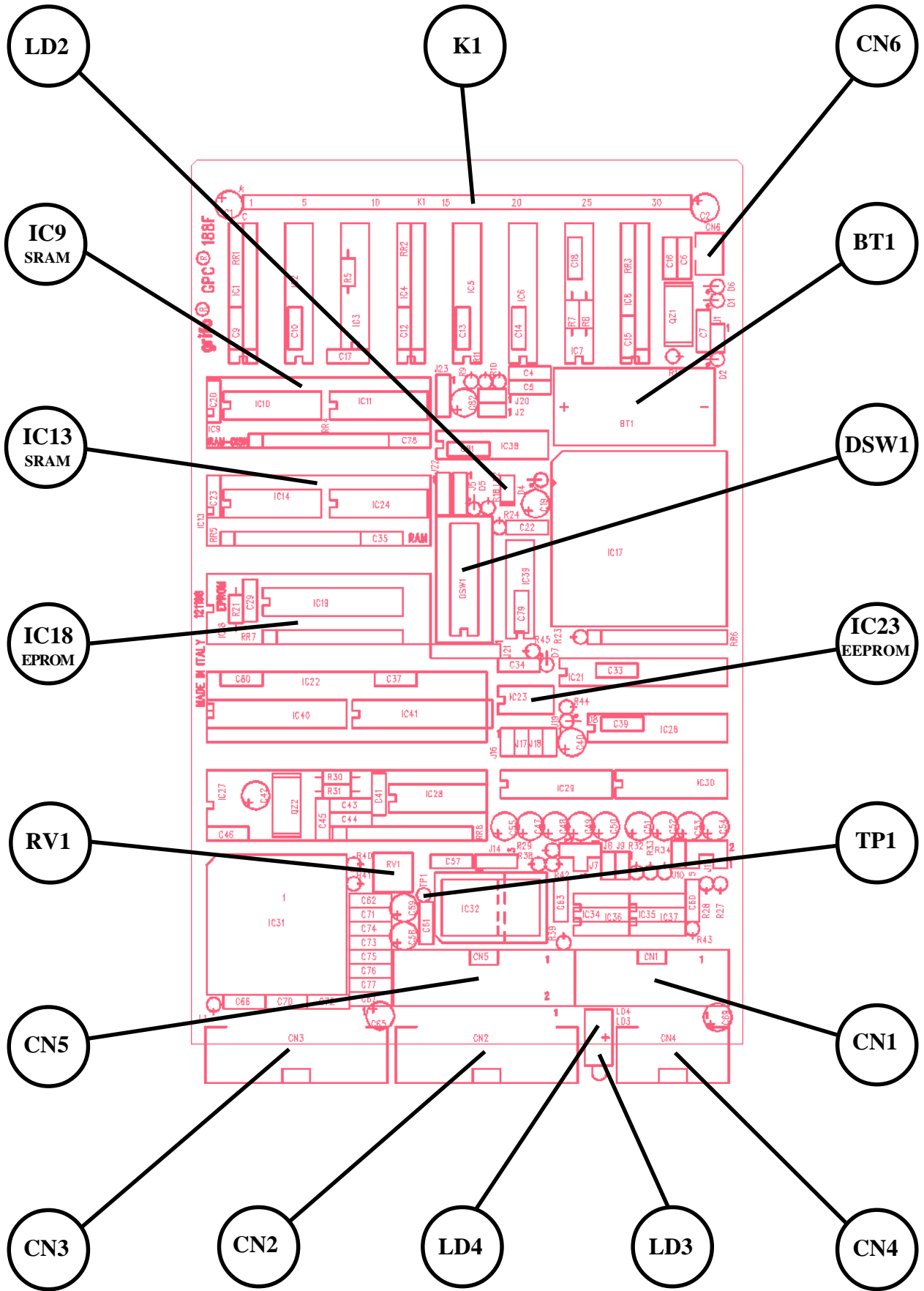


FIGURA 20: DISPOSIZIONE CONNETTORI, TRIMMER, BATTERIA, MEMORIE, ECC.

INTERFACCIE PER I/O DIGITALI

Tramite CN2 (connettore compatibile con standard di I/O ABACO®) si può collegare la GPC® 188F ai numerosi moduli del carteggio grifo® che riportano lo stesso pin out. Dal punto di vista dell'installazione, queste interfacce richiedono solo un flat cable da 20 vie (FLT.20+20) con cui è possibile portare anche le alimentazioni, mentre dal punto di vista software la gestione è altrettanto semplice ed immediata. Di particolare interesse è la possibilità di collegare direttamente una serie di moduli come:

- **QTP 16P, QTP 24P, KDL x24, KDF 224, DEB 01**, ecc. con cui risolvere tutti i problemi di interfacciamento operatore locale. Questi moduli sono già dotati delle risorse necessarie per gestire un buon livello di colloquio uomo-macchina (includono infatti display alfanumerici, tastiera a matrice e LEDs di visualizzazione) ad una breve distanza dalla GPC® 188F. Dal punto di vista software i driver disponibili rendono utilizzabili le risorse dell'interfaccia operatore direttamente con le istruzioni ad alto livello per la gestione della console.
- **MCI 64** con cui risolvere tutti i problemi di salvataggio di grosse quantità di dati. Questo modulo è dotato di un connettore per memory card PCMCIA su cui possono essere inserite vari tipi di memory card (RAM, FLASH, ROM, ecc) nei vari size disponibili. Dal punto di vista software i driver disponibili coincidono con un completo file system e rendono utilizzabili le memory card direttamente con le istruzioni ad alto livello per la gestione dei files, oppure con procedure che consentono di leggere e scrivere dati ad indirizzi specifici della memory card.
- **IAC 01, DEB 01** con cui gestire una stampante con interfaccia parallela CENTRONICS. Quest'ultima può essere collegata direttamente all'interfaccia, con un cavo standard, e quindi gestita con le istruzioni relative alla stampante del linguaggio di programmazione utilizzato.
- **RBO xx, TBO xx, XBI xx, OBI xx** con cui bufferare i segnali di I/O TTL nei confronti del campo. Con questi moduli i segnali di input vengono convertiti in ingressi optoisolati di tipo NPN o PNP, mentre i segnali di output vengono convertiti in uscite galvanicamente isolate a transistor o relé. Alcune di queste interfacce possono essere collegate direttamente anche al CN5.

Per maggiori informazioni relative alle interfacce per I/O digitali si veda il capitolo "SCHEDE ESTERNE" e la documentazione del software utilizzato.

SEGNALAZIONI VISIVE

La scheda GPC® 188F è dotata di 3 LEDs con cui segnala alcune condizioni di stato, come descritto nella seguente tabella:

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

FIGURA 21: TABELLA DELLE SEGNALAZIONI VISIVE

La funzione principale di questi LEDs è quella di fornire un'indicazione visiva dello stato della scheda, facilitando quindi le operazioni di debug e di verifica di funzionamento di tutto il sistema. Per una più facile individuazione di tali segnalazioni visive, si faccia riferimento alla figura 20.

JUMPERS

Esistono a bordo della **GPC® 188F** 17 jumpers a cavaliere, con cui é possibile effettuare alcune selezioni che riguardano il modo di funzionamento della stessa. Di seguito ne é riportato l'elenco, l'ubicazione e la loro funzione nelle varie modalit  di connessione.

| JUMPERS | N. VIE | UTILIZZO |
|---------|--------|---|
| J1 | 2 | Collega batteria al Litio di bordo alla circuiteria di back up. |
| J2 | 2 | Abilita circuiteria di write protect per la RAM di IC9. |
| J5 | 3 | Collega circuiteria di watch dog. |
| J7 | 4 | Seleziona tipo di comunicazione seriale per linea seriale B (RS 232, RS 422, RS 485, current loop). |
| J8, J9 | 2 | Collega circuiteria di terminazione RS 422, RS 485. |
| J10 | 2 | Forza stato della linea di handshake CTSB. |
| J11 | 5 | Seleziona direzionalit  e modalit  di attivazione della linea seriale B in RS 422, RS 485. |
| J14 | 3 | Seleziona sorgente trasferimento DMA canale 0. |
| J16 | 2 | Seleziona stato della linea di handshake DCDA. |
| J17 | 2 | Seleziona stato della linea di handshake DCDB. |
| J18 | 2 | Seleziona stato della linea di handshake SYNCA. |
| J19 | 2 | Seleziona stato della linea di handshake SYNCB. |
| J20 | 2 | Collega la circuiteria di Power Failure. |
| J21 | 5 | Seleziona dimensioni e tipo memoria su IC18. |
| J22 | 3 | Seleziona dimensioni SRAM su IC13. |
| J23 | 3 | Seleziona dimensioni SRAM su IC9. |

FIGURA 22: TABELLA RIASSUNTIVA JUMPERS

Nelle successive tabelle é riportata una descrizione tabellare delle possibili connessioni dei 17 jumpers con la loro relativa funzione. Per riconoscere tali connessioni sulla scheda si faccia riferimento alla serigrafia della stessa o alla figura 30 di questo manuale, dove viene riportata la numerazione dei pin dei jumpers, che coincide con quella utilizzata nella seguente descrizione. Per l'individuazione dei jumpers a bordo della scheda, si utilizzi invece la figura 25. In tutte le seguenti tabelle l'* indica la connessione di default, ovvero quella impostata in fase di collaudo, con cui la scheda viene fornita.

JUMPERS A 2 VIE

| JUMPERS | CONNESSIONE | UTILIZZO | DEF. |
|---------|--------------|---|------|
| J1 | non connesso | Non collega batteria BT1 a circuiteria di back up. | * |
| | connesso | Collega batteria BT1 a circuiteria di back up. | |
| J2 | non connesso | Disattiva circuiteria di write protect. | * |
| | connesso | Attiva circuiteria di write protect. | |
| J8, J9 | non connessi | Non collega circuiteria di terminazione e forzatura alla linea seriale RS 485 o alla linea di ricezione RS 422. | * |
| | connessi | Non collega circuiteria di terminazione e forzatura alla linea seriale RS 485 o alla linea di ricezione RS 422. | |
| J10 | non connesso | Non forza lo stato dell'handshake CTSB. | * |
| | connesso | Forza attivo (GND) lo stato dell'handshake CTSB. | |
| J16 | non connesso | Setta stato logico 1 sulla linea DCDA. | * |
| | connesso | Setta stato logico 0 sulla linea DCDA. | |
| J17 | non connesso | Setta stato logico 1 sulla linea DCDB. | * |
| | connesso | Setta stato logico 0 sulla linea DCDB. | |
| J18 | non connesso | Setta stato logico 1 sulla linea SYNCA. | * |
| | connesso | Setta stato logico 0 sulla linea SYNCA. | |
| J19 | non connesso | Setta stato logico 1 sulla linea SYNCB. | * |
| | connesso | Setta stato logico 0 sulla linea SYNCB. | |
| J20 | non connesso | Non collega la circuiteria di power failure al segnale NMI. | * |
| | connesso | Collega la circuiteria di power failure al segnale NMI. | |

FIGURA 23: TABELLA JUMPERS A 2 VIE
JUMPERS A 4 VIE

| JUMPERS | CONNESSIONE | UTILIZZO | DEF. |
|---------|---------------|---|------|
| J7 | posizione 1-2 | Predisporre linea seriale B in current loop. | * |
| | posizione 2-3 | Predisporre linea seriale B in RS 232. | |
| | posizione 2-4 | Predisporre linea seriale B in RS 422 o RS 485. | |

FIGURA 24: TABELLA JUMPERS A 4 VIE

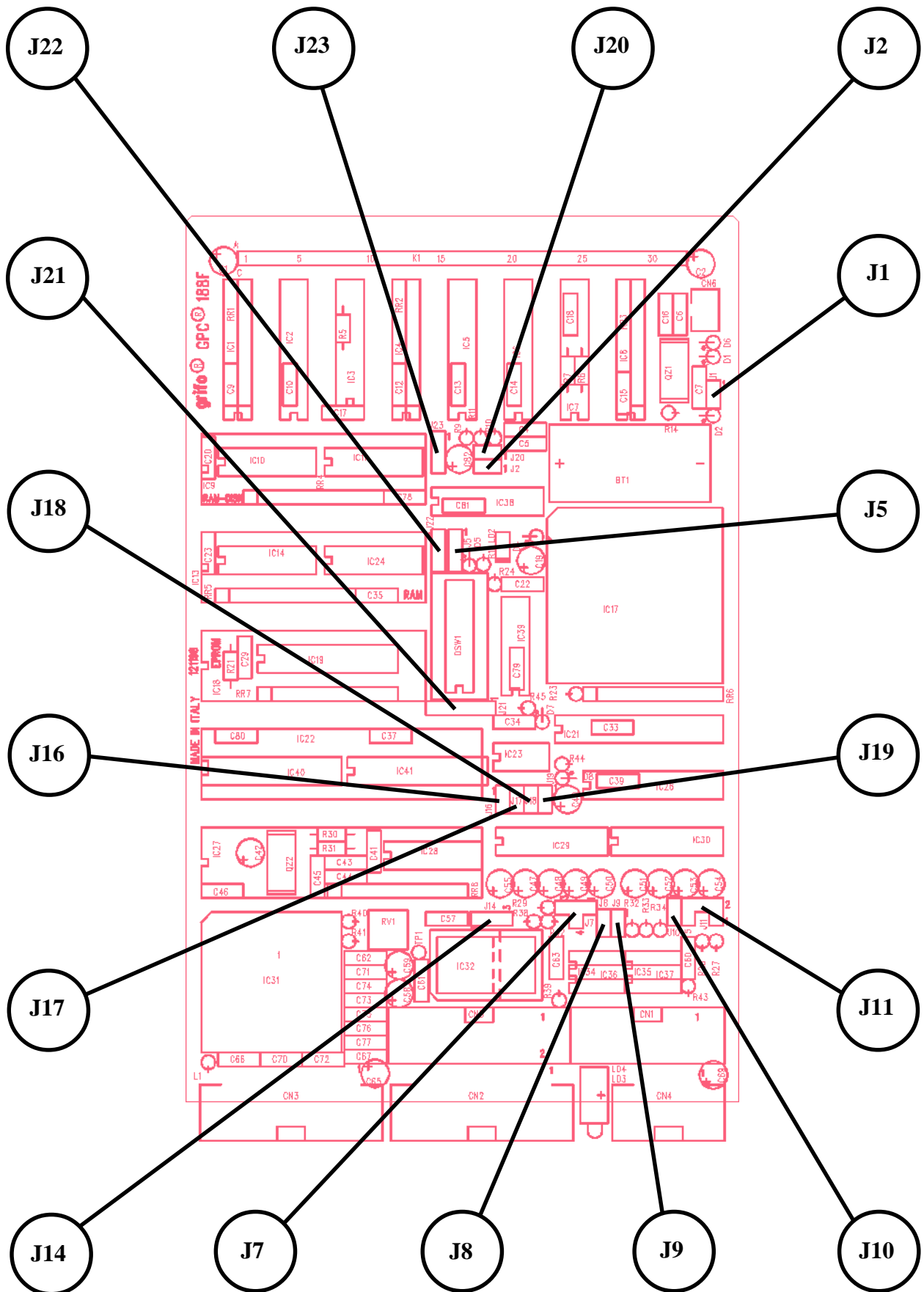


FIGURA 25: DISPOSIZIONE JUMPERS

JUMPERS A 3 VIE

| JUMPERS | CONNESSIONE | UTILIZZO | DEF. |
|---------|---------------|--|------|
| J5 | posizione 1-2 | Non collega circuiteria di watch dog alla circuiteria di reset. | * |
| | posizione 2-3 | Collega circuiteria di watch dog alla circuiteria di reset. | |
| J14 | non connesso | Nessuna sorgente collegata al canale DMA 0 del microprocessore. | * |
| | posizione 1-2 | Collega richiesta del SCC 8530 al canale DMA 0 del microprocessore. | |
| | posizione 2-3 | Collega richiesta dell'A/D LM 12458 al canale DMA 0 del microprocessore. | |
| J22 | posizione 1-2 | Configura zoccolo IC13 per SRAM da 128K bytes. | * |
| | posizione 2-3 | Configura zoccolo IC13 per SRAM da 512K bytes. | |
| J23 | posizione 1-2 | Configura zoccolo IC9 per SRAM da 128K bytes. | * |
| | posizione 2-3 | Configura zoccolo IC9 per SRAM da 512K bytes. | |

FIGURA 26: TABELLA JUMPERS A 3 VIE
JUMPERS A 5 VIE

| JUMPERS | CONNESSIONE | UTILIZZO | DEF. |
|---------|--------------------|---|------|
| J11 | posizione 1-2, 3-4 | Selezione comunicazione su linea seriale B in RS 485 (half duplex a 2 fili). | * |
| | posizione 2-3, 4-5 | Selezione comunicazione su linea seriale B in RS 422 in full duplex o half duplex a 4 fili. | |
| J21 | posizione 3-4 | Configura zoccolo IC18 per EPROM da 1024K bytes. | * |
| | posizione 1-2, 3-4 | Configura zoccolo IC18 per EPROM fino a 512K bytes. | |
| | posizione 2-3, 4-5 | Configura zoccolo IC18 per FLASH EPROM fino a 512K bytes. | |

FIGURA 27: TABELLA JUMPERS A 5 VIE

RESET E WATCH DOG

La scheda **GPC® 188F** è dotata di una circuiteria di watch dog molto efficiente e di facile gestione software. In particolare le caratteristiche di questa circuiteria sono le seguenti:

- funzionamento astabile;
- tempo d'intervento di circa 1420 msec;
- attivazione via hardware;
- retrigger via software;

Si ricorda che nel funzionamento astabile una volta scaduto il tempo d'intervento la circuiteria si attiva, rimane attiva per il tempo di reset (della durata di 185 msec) e quindi si disattiva nuovamente. Con il jumper J5 si seleziona se collegare la circuiteria di watch dog alla circuiteria di reset, ovvero si attiva, via hardware, la sua gestione.

In corrispondenza dell'attivazione e successiva disattivazione del segnale di /RESET la scheda riprende l'esecuzione del programma salvato su IC13 (all'indirizzo FFFF0H), partendo da una condizione di azzeramento generale.

Si ricorda inoltre che il segnale di /RESET generato dalla scheda é riportato anche sul connettore K1 (pin 16C) e che tra le sorgenti di reset della **GPC® 188F**, oltre all'eventuale circuiteria di watch dog, sono sempre presenti il contatto di reset R.T. (pin 29C di K1) e la circuiteria di power good.

Per quanto riguarda l'operazione di retrigger della circuiteria di watch dog esterna, si faccia riferimento al paragrafo "WATCH DOG" del capitolo "DESCRIZIONE SOFTWARE DELLE PERIFERICHE DI BORDO".

INTERRUPTS

Una caratteristica peculiare della **GPC® 188F** è la notevole potenza nella gestione delle interruzioni. Di seguito viene riportata una breve descrizione di quali sono i dispositivi che possono generare interrupts e con quale modalità; per quanto riguarda la gestione di tali interrupts si faccia riferimento ai data sheets del microprocessore oppure all'appendice B di questo manuale.

- BUS **ABACO®** -> Genera un NMI sulla CPU, tramite la linea /NMI di K1.
Genera un INT0 sulla CPU, tramite la linea /INT di K1.
- Power failure -> Genera un NMI sulla CPU, a seconda del collegamento di J20.
- Real Time Clock -> Genera un INT2 sulla CPU.
- A/D converter -> Genera un INT3 sulla CPU.
- Linee seriali SCC -> Genera un INT1 sulla CPU.
- Periferiche della CPU -> Generano un interrupt interno. In particolare le possibili sorgenti d'interrupt interno sono le sezioni: TC 0, TC 1, TC 2, DMA 0, DMA 1, interrupt software.

Sulla scheda é presente un gestore d'interrupt che consente di attivare, disattivare, mascherare le sorgenti d'interrupt e che regola l'attivazione contemporanea di più interrupts. In questo modo l'utente ha sempre la possibilità di rispondere in maniera efficace e veloce a qualsiasi evento esterno, stabilendo anche la priorità delle varie sorgenti.

COMUNICAZIONE SERIALE

La linea di comunicazione seriale A della scheda **GPC® 188F** può essere bufferata solo in RS 232, mentre la linea seriale B può essere bufferata in RS 232, RS 422, RS 485 o current loop. La selezione del tipo d'interfacciamento avviene via hardware e viene effettuata tramite un opportuno strippaggio dei jumpers di bordo, come può essere desunto dalla lettura delle precedenti tabelle. Dal punto di vista software sono invece definibili tutti i parametri del protocollo fisico di comunicazione tramite la programmazione dei registri interni del SCC 85C30.

Alcuni componenti necessari per le configurazioni RS 422, RS 485 e current loop non sono montati e collaudati sulla scheda in configurazione di default; per questo la prima configurazione della seriale B non in RS 232 deve essere sempre effettuata dai tecnici **grifo®**. A questo punto l'utente può cambiare autonomamente la configurazione seguendo le informazioni sotto riportate:

- LINEA SERIALE B SETTATA IN RS 232 (configurazione default)

| | | | | | |
|--------|---|---------------|------|---|----------------|
| J7 | = | posizione 2-3 | IC29 | = | driver MAX 202 |
| J8, J9 | = | indifferenti | IC34 | = | indifferente |
| J10 | = | non connesso | IC35 | = | indifferente |
| J11 | = | indifferente | IC36 | = | indifferente |
| | | | IC37 | = | indifferente |

- LINEA SERIALE B SETTATA IN CURRENT LOOP (opzione .CLOOP)

| | | | | | |
|--------|---|---------------|------|---|-------------------|
| J7 | = | posizione 1-2 | IC29 | = | indifferente |
| J8, J9 | = | indifferenti | IC34 | = | driver HP 4200 |
| J10 | = | (*1) | IC35 | = | driver HP 4100 |
| J11 | = | indifferente | IC36 | = | nessun componente |
| | | | IC37 | = | nessun componente |

Da ricordare che l'interfaccia seriale in current loop é di tipo passivo e si deve quindi collegare una linea current loop attiva, ovvero provvista di un proprio alimentatore. L'interfaccia current loop può essere utilizzata per realizzare sia connessioni punto punto che multipunto con un collegamento a 4 o 2 fili.

- LINEA SERIALE B SETTATA IN RS 422 (opzione .RS 422)

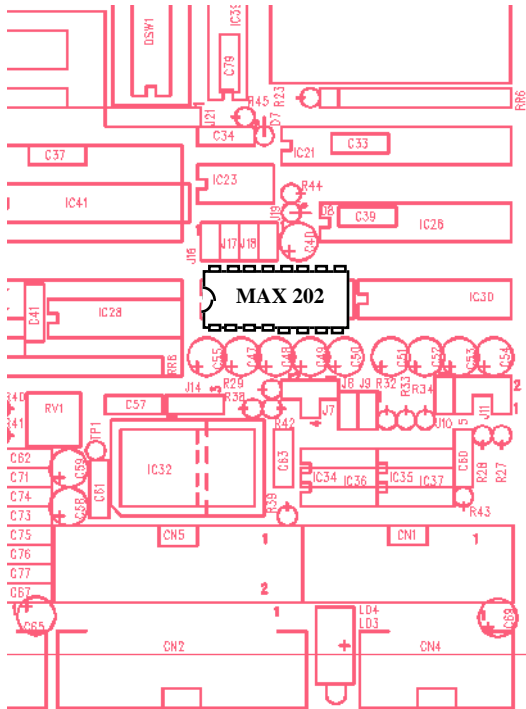
| | | | | | |
|--------|---|--------------------|------|---|-------------------|
| J7 | = | posizione 2-4 | IC29 | = | indifferente |
| J8, J9 | = | (*2) | IC34 | = | nessun componente |
| J10 | = | (*1) | IC35 | = | nessun componente |
| J11 | = | posizione 2-3, 4-5 | IC36 | = | driver SN 75176 |
| | | | IC37 | = | driver SN 75176 |

Lo stato del segnale /RTSB, gestito via software, consente di abilitare o disabilitare il trasmettitore come segue:

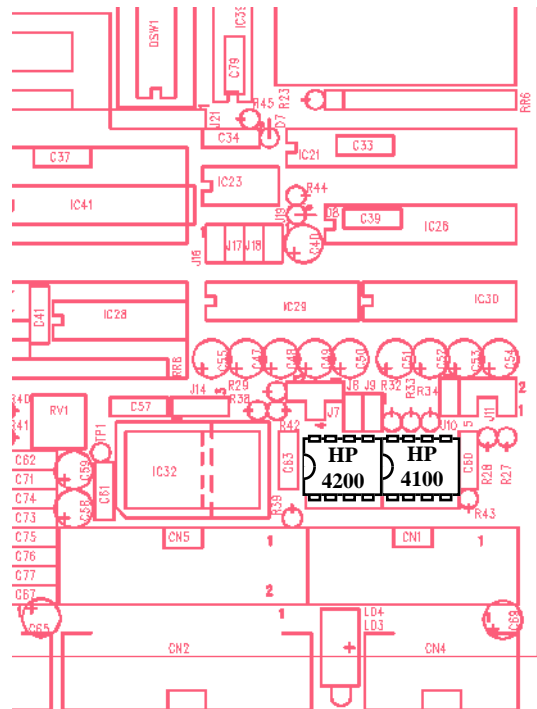
/RTSB = livello basso = stato logico 0 -> trasmettitore attivo

/RTSB = livello alto = stato logico 1 -> trasmettitore disattivo

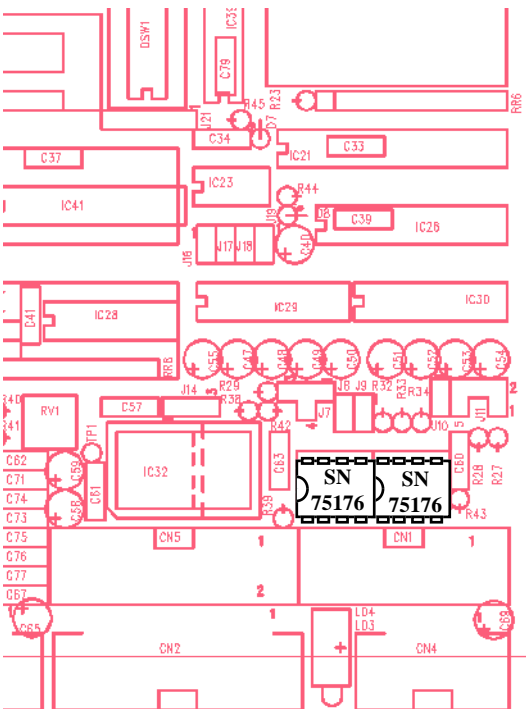
Per sistemi punto punto, la linea /RTSB può essere mantenuta sempre bassa (trasmettitore sempre attivo), mentre per sistemi multipunto si deve attivare il trasmettitore solo in corrispondenza della trasmissione.



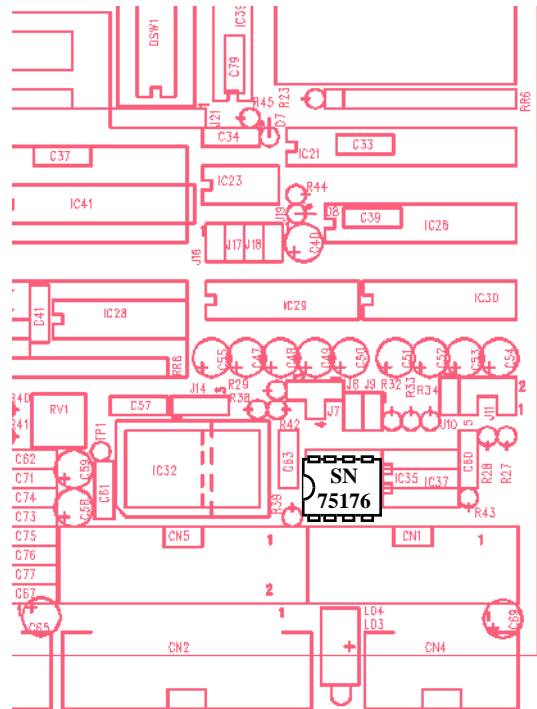
Seriale B in RS 232



Seriale B in current loop



Seriale B in RS 422



Seriale B in RS 485

FIGURA 28: DISPOSIZIONE DRIVER PER COMUNICAZIONE SERIALE

- LINEA SERIALE B SETTATA IN RS 485 (opzione .RS 485)

| | | | | | |
|--------|---|--------------------|------|---|-------------------|
| J7 | = | posizione 2-4 | IC29 | = | indifferente |
| J8, J9 | = | (*2) | IC34 | = | nessun componente |
| J10 | = | (*1) | IC35 | = | nessun componente |
| J11 | = | posizione 1-2, 3-4 | IC36 | = | driver SN 75176 |
| | | | IC37 | = | nessun componente |

In questa modalità le linee da utilizzare sono i pin 1 e 2 di CN4, che quindi diventano le linee di trasmissione o ricezione a seconda dello stato del segnale /RTSB, gestito via software, come segue:

/RTSB = livello basso = stato logico 0 -> linea in trasmissione

/RTSB = livello alto = stato logico 1 -> linea in ricezione

Questa comunicazione la si utilizza sia per connessioni punto punto che multipunto con un collegamento a 2 fili. Sempre in questa modalità è possibile ricevere quanto trasmesso, in modo da fornire al sistema la possibilità di verificare autonomamente la riuscita della trasmissione; infatti in caso di conflitti sulla linea, quanto trasmesso non viene ricevuto correttamente e viceversa.

(*2) Nel caso si utilizzi la linea seriale in RS 422 o RS 485, con i jumpers J8 e J9 è possibile connettere la circuiteria di terminazione e forzatura sulla linea. Tale circuiteria deve essere sempre presente nel caso di sistemi punto punto, mentre nel caso di sistemi multipunto, deve essere collegata solo sulle schede che risultano essere alla maggior distanza, ovvero ai capi della linea di comunicazione.

(*1) La funzione del jumper J10 è quella di mantenere attivo l'handshake d'ingresso /CTSB quando la linea seriale B non è configurata in RS 232. In particolare se l'utente attiva la modalità autoenable (modalità in cui la ricezione e la trasmissione è autonomamente condizionata dallo stato degli handshake hardware) questo jumper dovrà essere connesso al fine di mantenere abilitata la trasmissione seriale.

In fase di reset o power on, il segnale /RTSB è mantenuto a livello logico alto di conseguenza in seguito ad una di queste fasi il driver RS 485 è in ricezione o il driver di trasmissione RS 422 è disattivo, in modo da eliminare eventuali conflittualità sulla linea di comunicazione.

Per ulteriori informazioni relative alla comunicazione seriale fare riferimento agli esempi di collegamento delle figure 13÷18 ed al paragrafo SCC 85C30.

BACK UP

La **GPC® 188F** è provvista di una batteria al litio BT1 che provvede a tamponare le SRAM ed il RTC di bordo anche in assenza della tensione di alimentazione. Il jumper J1 provvede a collegare o meno questa batteria in modo da salvaguardarne la durata prima dell'installazione o in tutti i casi in cui il back up non è necessario. Una seconda batteria esterna può essere collegata alla circuiteria di back up tramite il connettore CN6: quest'ultima non è interessata dalla configurazione del jumper J1 e sostituisce a tutti gli effetti la BT1.

Per la scelta della batteria esterna di back up seguire le indicazioni del paragrafo "CARATTERISTICHE ELETTRICHE", mentre per la sua individuazione si veda la figura 20.

SELEZIONE MEMORIE

La **GPC® 188F** può montare fino ad un massimo di 2056K bytes di memoria variamente suddivisa. In particolare per la configurazione seguire le informazioni riportate nella seguente tabella:

| IC | DISPOSITIVO | DIMENSIONE | STRIPPAGGIO |
|----|-------------|-------------|----------------------------|
| 18 | EPROM | 128K Byte | J21 in posizione 3-4 |
| | EPROM | 256K Byte | J21 in posizione 3-4 |
| | EPROM | 512K Byte | J21 in posizione 3-4 |
| | EPROM | 1024K Byte | J21 in posizione 1-2 e 3-4 |
| | FLASH EPROM | 128K Byte | J21 in posizione 2-3 e 4-5 |
| | FLASH EPROM | 512K Byte | J21 in posizione 2-3 e 4-5 |
| 9 | SRAM | 128K Byte | J23 in posizione 1-2 |
| | SRAM | 512K Byte | J23 in posizione 2-3 |
| 13 | SRAM | 128K Byte | J22 in posizione 1-2 |
| | SRAM | 512K Byte | J22 in posizione 2-3 |
| 23 | EEPROM | 256÷8K Byte | - |

FIGURA 29: TABELLA DI SELEZIONE MEMORIE

Tutti i dispositivi sopra descritti devono essere con pin out di tipo JEDEC a parte l'EEPROM seriale di IC 23 che deve essere richiesta alla **grifo®** in fase di ordine della scheda. Per quanto riguarda le sigle dei vari dispositivi che possono essere montati, fare riferimento alla documentazione della casa costruttrice.

Normalmente la **GPC® 188F** é fornita nella sua configurazione di default con solamente 128K SRAM su IC13 e 512 bytes di EEPROM su IC23; ogni configurazione diversa può essere autonomamente montata dall'utente oppure richiesta nella fase di ordine. Sotto sono riportate i codici delle opzioni di memoria disponibili:

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

Per ulteriori informazioni e costi delle opzioni, contattare direttamente la **grifo®**, mentre per una facile individuazione dei dispositivi di memoria fare riferimento alla figura 20.

POWER FAILURE

In abbinamento alla circuiteria di power management gestita dalla CPU della **GPC® 188F** é inoltre disponibile un'interessante circuiteria di power failure. Quest'ultima, con il jumper J20, può essere collegata all'interrupt NMI del microprocessore.

La circuiteria si preoccupa di controllare la tensione di alimentazione e quando questa scende al valore di soglia (**52 mV** prima dell'intervento del reset), provvede ad attivare l'uscita richiedendo l'attenzione della CPU nel caso che J20 sia collegato.

Da notare che il tempo che intercorre tra l'attivazione del power failure e quello del reset, varia in funzione del tipo di alimentazione della scheda; questo normalmente é nell'ordine dei 100 µsec, sufficienti solo per eseguire procedure di risposta veloci (ad esempio il salvataggio di un flag nella memoria tamponata).

L'uso classico della circuiteria di power failure é quello di informare la scheda dell'imminente caduta della tensione di alimentazione, in modo da salvare le necessarie condizioni di stato, grazie alla carica residua della stessa sezione alimentatrice.

INGRESSI DI CONFIGURAZIONE

La scheda **GPC® 188F** é provvista di un dip switch ad 8 vie (DSW1) e di 4 jumpers (J16+J19) tipicamente utilizzabili per la configurazione del sistema, i cui valori sono acquisibili via software. Le applicazioni più immediate possono essere quelle destinate al settaggio delle condizioni di lavoro od alla selezione di parametri relativi al firmware di bordo, come ad esempio: selezione della lingua di rappresentazione, identificazione del sistema all'interno di una rete di comunicazione seriale, selezione della modalità di test o di configurazione, ecc.

Le modalità di acquisizione degli ingressi di configurazione sono riportate nel capitolo "DESCRIZIONE SOFTWARE DELLE PERIFERICHE DI BORDO", mentre per una facile individuazione della loro posizione si vedano le figure 20 e 25.

TRASFERIMENTI DMA

Alcune delle periferiche esterne della scheda supportano un trasferimento dati ad alta velocità (DMA) nei confronti della memoria, come di seguito riportato:

- SCC 85C30 -> I dati da trasmettere sulla linea seriale B possono essere trasferiti in DMA, tramite il canale 1 della CPU.
I dati ricevuti dalla linea seriale B possono essere trasferiti in DMA tramite il canale 0 della CPU, quando J14 é collegato in posizione 1-2.
- A/D LM 12H458 -> I dati determinati dall'A/D converter possono essere trasferiti in DMA tramite il canale 0 della CPU, quando J14 é collegato in posizione 2-3.

Per maggiori informazioni sulla gestione del DMA sulla fare riferimento all'appendice B ed al paragrafo "Disattivazione richiesta DMA".

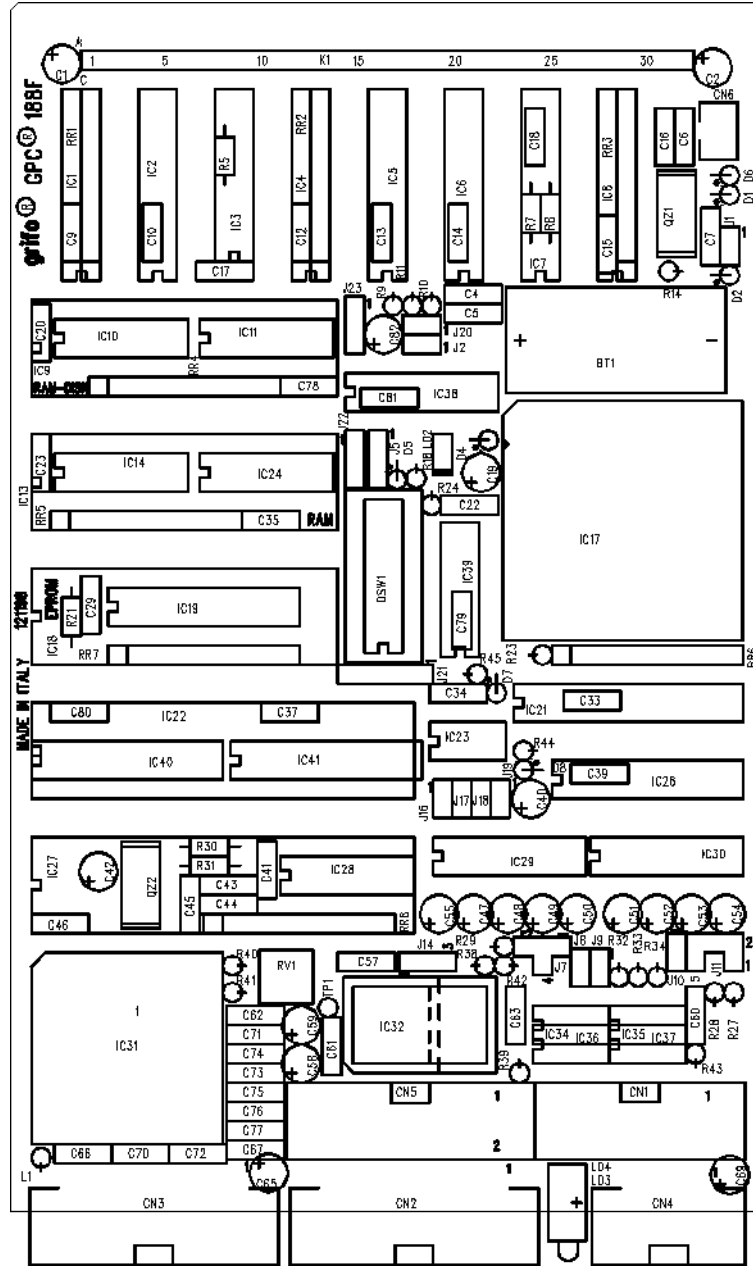


FIGURA 30: PIANTA COMPONENTI

DESCRIZIONE SOFTWARE

Questa scheda ha la possibilità di usufruire di una ricca serie di strutture software che consentono di utilizzarne al meglio le caratteristiche. In generale la scheda può utilizzare tutte le risorse software disponibili per il processore montato, ovvero i numerosi pacchetti ideati per l'8086 e derivati. Si ricorda inoltre che anche tutto il software disponibili sui P.C. potrebbe essere eseguito sulla **GPC® 188F** ma che realmente ciò non è possibile a causa della sua incompatibilità hardware con i P.C. Tra questi ricordiamo:

GET188

Completo programma di EDITOR, Comunicazione e gestione delle memorie di massa per le schede della famiglia 188. Questo programma, sviluppato dalla **grifo®**, consente di operare in condizioni ottimali, tutte le volte che si deve usare il GDOS o la versione per FLASH EPROM FGDOS. Viene fornito in abbinamento all'acquisto di uno dei pacchetti citati e viene personalizzato con il nome ed i dati dell'acquirente. Una serie di comodi menù a tendina facilita l'uso del programma, il quale può funzionare anche in abbinamento ad un mouse. Il programma, oltre che girare in ambiente MS-DOS, gira tranquillamente anche sulle macchine MACINTOSH in abbinamento al programma SOFT-PC. Viene fornito su dischetti MS-DOS da 3"1/2 con relativa documentazione sul manuale GDOS 188.

GDOS 188

Ambiente di sviluppo completo per le schede della famiglia Intel 86. Viene fornito in abbinamento al programma GET188, per consentire un immediato e pieno utilizzo di questo potente strumento di sviluppo. Il GDOS può essere concettualmente diviso in due distinte strutture. Una struttura lavora essenzialmente su PC, mantenendo il collegamento con la seconda tramite la linea seriale. La seconda risiede in EPROM ed opera a bordo scheda. La parte a bordo scheda è essenzialmente un potente sistema operativo che si preoccupa di eseguire tutte quelle funzioni a più basso livello e nello stesso tempo consente di poter operare con alcuni linguaggi ad alto livello direttamente a bordo scheda. L'abbinamento delle due strutture fa sì che la scheda ed il PC si comportino come un'unica macchina. Infatti la scheda usa, come se fossero le proprie, le risorse del PC come la memoria di massa quali i floppy disk, l'hard disk; la stampante ecc. Il tutto avviene in modo completamente trasparente per l'utente il quale usa questo tipo di macchina virtuale esattamente come è abituato ad adoperare il suo PC.

Il GDOS, oltre ai tipici drivers del PC, gestisce come RAM disk una porzione della memoria RAM di bordo. Questo significa che il dispositivo di SRAM a bordo scheda, tamponato tramite batteria, può essere gestito in modo diretto dai linguaggi ad alto livello, trattando comodamente come files, le informazioni da archiviare o ricercare.

Nel GDOS 188 è inoltre presente un potente monitor debugger che è in grado di caricare, eseguire e debuggare programmi applicativi generati da un generico cross assembler compiler esterno.

Viene fornito in EPROM, in abbinamento al disco in formato MS-DOS del GET188, alcuni esempi di uso e la relativa manualistica sul sistema operativo.

FGDOS 188

Caratteristiche analoghe al GDOS, con la differenza che è in grado di programmare e cancellare le FLASH-EPROM a bordo scheda, con i programmi generati dall'utente. In questo modo non è necessario un programmatore di EPROM esterno per congelare il programma. E' inoltre possibile, tramite un PC Portatile, intervenire direttamente a bordo macchina per cambiare il programma di gestione.

Viene fornito in FLASH EPROM, in abbinamento al disco in formato MS-DOS del GET188, alcuni esempi di uso e la relativa manualistica sul sistema operativo.

PASCAL 188

Completo e molto efficiente compilatore PASCAL in grado di generare un codice romato per la famiglia 86 di CPU. Ha delle caratteristiche operative analoghe a quelle del Turbo PASCAL Ver.3 della Borland, a cui si fa riferimento per quanto riguarda sia le caratteristiche che la manualistica. Il PASCAL 86 lavora in abbinamento ad una delle varie versioni di sistema operativo GDOS188. Le modalità di emulazione terminale offerta dal programma GET188, supportano pienamente il tipico editor a pieno schermo del PASCAL, compresa la gestione degli attributi. Sfruttando la possibilità di gestione di RAM disk offerta dal GDOS, si possono sfruttare appieno le istruzioni di file system offerte dal linguaggio. Genera un codice molto compatto, il quale risiede e viene eseguito in EPROM o FLASH EPROM e che richiede pochissimo spazio di RAM.

Il programma viene fornito in EPROM o FLASH EPROM, assieme al sistema operativo GDOS, e su dischetto MS-DOS in abbinamento alle note tecniche e ad una serie di esempi.

GCTR 188F

Completo pacchetto di sviluppo, per la scheda **GPC® 188F**, che consente di sviluppare dei programmi applicativi adoperando un compilatore C della Borland, in abbinamento ad un normale PC. Un potentissimo Debugger remoto, residente sulla scheda, consente di operare in modalità simbolica e sorgente, con delle prestazioni e delle comodità di uso analoghe a quelle di un emulatore hardware, grazie ad un semplice collegamento seriale. Alla fine delle operazioni é possibile congelare il programma in EPROM o FLASH EPROM, con un uso molto limitato delle risorse RAM. Viene fornito in EPROM o FLASH EPROM, con una serie di floppy disk , la relativa manualistica tecnica e degli esempi d'uso.

HCT 86

Cross compilatore C professionale della Hi-Tech Software. Questo compilatore é estremamente veloce e genera pochissimo codice. Questo risultato é ottenuto grazie a delle avanzate tecniche di ottimizzazione del codice generato, basato su tecniche di intelligenza artificiale che gli consentono di ottenere un codice compatto ed estremamente veloce. Il pacchetto comprende IDE, compilatore, ottimizzatore del codice, assembler, linker, debugger remoto, ecc. Questo tools é completamente compatibile allo standard ANSI/ISO ed é provvisto di tutti i sorgenti delle librerie. Una volta fatto il porting del modulo del debugger, consente di caricare e provare il codice generato direttamente sull'hardware in sperimentazione. Questo tipo di specializzazione del debugger remoto é già disponibile, e viene fornito, per tutte le schede di CPU della **grifo®**. Il pacchetto software viene fornito su dischetti da 3"1/2 nel formato MS-DOS, completo di un esauriente manuale.

Questa versione supporta le CPU 8088, 8086, 80188, 80C188, 80186, 80C186, 80286, V20, V30, V25, ecc.

DDS MICRO C 86

E' un comodo pacchetto software, a basso costo, che tramite un completo I.D.E. permette di utilizzare un editor, un compilatore "C" (integer), un assembler, un linker a livello sorgente e un monitor in grado di caricare ed eseguire il codice generato. Sono inclusi i sorgenti delle librerie, una serie di utility ed una ricca documentazione su dischetto da 3"1/2 nel formato MS-DOS.

MAPPAGGI ED INDIRIZZAMENTI

In questo capitolo ci occuperemo di fornire tutte le informazioni relative all'utilizzo della scheda, dal punto di vista della programmazione via software. Tra queste si trovano le informazioni riguardanti il mappaggio delle memorie, delle periferiche e di tutte le altre sezioni componenti.

MAPPAGGIO DELLE RISORSE DI BORDO

La gestione delle risorse della scheda è interamente affidata ad una logica di controllo che si occupa del mappaggio delle zone di RAM, EPROM, FLASH EPROM e di tutte le periferiche di bordo con una semplice gestione software.

Tale logica è in grado di gestire separatamente il mappaggio delle memorie di bordo ed il mappaggio delle periferiche viste in Input/Output. Complessivamente la CPU 80C188 indirizza direttamente 64K indirizzi di I/O ed 1M Bytes di memoria, quindi alla logica di controllo è assegnato il compito di allocare lo spazio fisico delle memorie in questo spazio logico d'indirizzamento tramite un'adeguata impaginazione.

La logica di controllo della **GPC® 188F** include due sezioni distinte: una di generazione segnali di controllo periferiche e memorie (/CS), interna al microprocessore, ed una memory management unit (MMU), esterna alla CPU. L'utente può quindi definire le dimensioni, gli indirizzi e le modalità di mappaggio di tutti i dispositivi via software, tramite la programmazione di entrambe le sezioni.

Visto che il mappaggio della scheda è completamente relazionato alla programmazione della CPU e della MMU, di seguito si riporta la descrizione dei collegamenti effettuati tra i vari dispositivi di bordo ed i segnali della CPU, in modo che con il semplice aiuto del manuale tecnico dell'80C188 l'utente è in grado di configurare correttamente la scheda.

Riassumendo l'elenco dei dispositivi mappati a bordo scheda è il seguente:

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

FIGURA 31: COLLEGAMENTO HARDWARE DEI DISPOSITIVI

Normalmente i pacchetti software sviluppati per la scheda si preoccupano già di effettuare una programmazione della logica di controllo con modalità descritte nella loro documentazione d'uso.

MAPPAGGIO PERIFERICHE IN I/O

Per il mappaggio dei dispositivi nello spazio di I/O la logica di controllo usa i 7 segnali di /CS (/PCS0÷/PCS6) generati dall'80C188, in corrispondenza di determinati indirizzi fissati dall'utente in fase di programmazione dello stesso processore. La gestione delle periferiche di bordo invece avviene tramite opportuni registri allocati nelle suddette locazioni di I/O.

Viene di seguito riportato l'indirizzamento di tali registri il loro nome, il tipo di accesso ed una breve descrizione del loro significato. Per quanto riguarda la descrizione dettagliata del loro significato, si faccia riferimento al capitolo successivo "DESCRIZIONE SOFTWARE DELLE PERIFERICHE DI BORDO".

| DISP. | REG. | INDIRIZZO | R/W | SIGNIFICATO |
|------------------------|-------------|-------------------------------|----------------------------|--|
| W.DOG | RWD | INDPCS0+00H | R/W | Retrigger watch dog |
| EEPROM | RE2 | INDPCS0+00H | R/W | Accesso seriale ad EEPROM |
| MMU | MMU | INDPCS0+00H | R/W | Impaginazione memorie con MMU |
| LD3,4 | LED | INDPCS0+00H | R/W | Registro gestione LEDs di attività |
| BT1 | BAT | INDPCS0+00H | R | Registro acquisizione stato batteria |
| SCC 85C30 | RSB | INDPCS1+00H | R/W | Registro stato linea seriale B |
| | RDB | INDPCS1+01H | R/W | Registro dati linea seriale B |
| | RSA | INDPCS1+02H | R/W | Registro stato linea seriale A |
| | RDA | INDPCS1+03H | R/W | Registro dati linea seriale A |
| DMA | DMA | INDPCS2+00H | R/W | Registro disattivazione richiesta DMA |
| A/D LM12458 | IRL0÷7 | INDPCS3+ 00H÷0EH (pari) | R/W | Registro istruzioni low 0÷7 del sequencer |
| | IRH0÷7 | INDPCS3+ 01H÷0FH (dispari) | R/W | Registro istruzioni high 0÷7 del sequencer |
| | CNTL | INDPCS3+10H | R/W | Registro di configurazione low |
| | CNTH | INDPCS3+11H | R/W | Registro di configurazione high |
| | INTENL | INDPCS3+12H | R/W | Registro abilitazione interrupt low |
| | INTENH | INDPCS3+13H | R/W | Registro abilitazione interrupt high |
| | INTSTL | INDPCS3+14H | R | Registro di stato interrupt low |
| | INTSTH | INDPCS3+15H | R | Registro di stato interrupt high |
| | TMRL | INDPCS3+16H | R/W | Registro per timer low |
| | TMRH | INDPCS3+17H | R/W | Registro per timer high |
| | FIFOL | INDPCS3+18H | R | Registro per conversioni in FIFO low |
| | FIFOH | INDPCS3+19H | R | Registro per conversioni in FIFO high |
| | LIMSTL | INDPCS3+1AH | R | Registro stato limiti low |
| LIMSTH | INDPCS3+1BH | R | Registro stato limiti high | |

FIGURA 32: TABELLA INDIRIZZAMENTO I/O - PARTE 1

| DISP. | REG. | INDIRIZZO | R/W | SIGNIFICATO |
|--------------------------|------|-------------|-----|-------------------------------------|
| PPI 82C55 | PA | INDPCS4+00H | R/W | Registro dati del port A |
| | PB | INDPCS4+01H | R/W | Registro dati del port B |
| | PC | INDPCS4+02H | R/W | Registro dati del port C |
| | RC | INDPCS4+03H | R/W | Registro di controllo e comando |
| RTC 62421 | S1 | INDPCS5+00H | R/W | Registro unità secondi |
| | S10 | INDPCS5+01H | R/W | Registro decine secondi |
| | MI1 | INDPCS5+02H | R/W | Registro unità minuti |
| | MI10 | INDPCS5+03H | R/W | Registro decine minuti |
| | H1 | INDPCS5+04H | R/W | Registro unità ore |
| | H10 | INDPCS5+05H | R/W | Registro decine ore; AM/PM |
| | D1 | INDPCS5+06H | R/W | Registro unità giorno |
| | D10 | INDPCS5+07H | R/W | Registro decine giorno |
| | MO1 | INDPCS5+08H | R/W | Registro unità mese |
| | MO10 | INDPCS5+09H | R/W | Registro decine mese |
| | Y1 | INDPCS5+0AH | R/W | Registro unità anno |
| | Y10 | INDPCS5+0BH | R/W | Registro decine anno |
| | W | INDPCS5+0CH | R/W | Registro giorno della settimana |
| | REGD | INDPCS5+0DH | R/W | Registro di stato e controllo D |
| | REGE | INDPCS5+0EH | R/W | Registro di stato e controllo E |
| | REGF | INDPCS5+0FH | R/W | Registro di stato e controllo F |
| WRITE PROTECT | WRP | INDPCS6+00H | W | Registro sprotezione scrittura SRAM |
| DIP SWITCH | DSW1 | INDPCS6+00H | R | Registro acquisizione dip switch |

FIGURA 33: TABELLA INDIRIZZAMENTO I/O - PARTE 2

Si ricorda che la precedente tabella riporta la descrizione dei soli registri delle periferiche esterne al microprocessore e che per la descrizione di quelli interni si può fare riferimento all'apposita documentazione della casa costruttrice, oppure all'appendice B di questo manuale.

Il valore degli indirizzi riportati in tabella (INDPCS_n) è definito via software tramite la programmazione della sezione di generazione segnali di controllo e può quindi essere stabilita dall'utente a seconda delle proprie esigenze. In caso di utilizzo dei pacchetti software già configurati per la **GPC® 188F**, sulla documentazione d'uso degli stessi si trovano le due precedenti tabelle con i valori assoluti di tutti gli indirizzi.

MAPPAGGIO BUS ABACO®

La logica di controllo della **GPC® 188F** provvede anche alla gestione del **BUS ABACO®**, definendo gli indirizzi in cui tale BUS viene allocato. In particolare tutti gli indirizzi di I/O nel range **0000H÷EFFFH** sono dedicati al BUS, meno gli indirizzi occupati dai 7 segnali di /CS (/PCS0÷/PCS6). Un accesso in I/O in un qualsiasi indirizzo compreso in questo range abilita il segnale /IORQ e tutti gli altri segnali di controllo di K1.

Gli ultimi 4K (F000H÷FFFFH) dei 64K complessivi della CPU sono stati riservati per i registri interni della CPU = control block. Al fine di evitare conflitti l'utente dovrà opportunamente programmare il registro interno "relocation" in modo che tali registri interni siano mappati in questo campo (ad esempio si può mantenere il settaggio di default di questo registro = 20FFH che alloca gli indirizzi interni a partire da FF00H). Per ulteriori chiarimenti fare riferimento all'appendice B di questo manuale.

MAPPAGGIO MEMORIE

Sulla scheda i 2056K Bytes di memoria che possono essere montati sono così allocati:

- fino a 1024K Bytes di EPROM o 512K Bytes di FLASH EPROM allocati nello spazio di memoria;
- fino a 512K Bytes di SRAM su IC9 allocati nello spazio di memoria;
- fino a 512K Bytes di SRAM su IC13 allocati nello spazio di memoria;
- fino a 8K Byte di EEPROM seriale allocati indirettamente nello spazio di I/O.

La **GPC® 188F** può indirizzare direttamente un massimo di 1M Bytes di memoria che coincide con lo spazio d'indirizzamento fisico del microprocessore. Questa capacità di memoria, sulla scheda, può essere suddivisa in diversi segmenti distinti ognuno dei quali può avere indirizzi d'inizio e dimensioni programmabili via software. In particolare la circuiteria di MMU della scheda ha il compito di impaginare i 2048K Bytes di memoria in pagine di 768K Bytes massimi direttamente indirizzabili dalla CPU, tramite i 6 segnali /LCS, /MCS0÷/MCS3, /UCS; tali segnali sono a loro volta gestiti dalla circuiteria di generazione segnali di controllo interna al microprocessore, che si occupa appunto di dividere lo spazio direttamente indirizzato dalla CPU in vari segmenti e di allocarli in memoria nello spazio dei dispositivi fisici. Programmando le due circuiterie descritte tramite gli appositi registri, è quindi possibile indirizzare direttamente la quantità di memoria effettivamente montata su IC9, IC13 ed IC18, con la facoltà di definire anche le modalità e le tempistiche d'accesso.

Vengono di seguito riportate alcune figure che illustrano le possibili configurazioni ed allocazioni dei dispositivi di memoria; per maggiori informazioni sulle modalità di gestione della circuiteria di generazione segnali di controllo fare riferimento all'appendice B, mentre per maggiori informazioni sulla circuiteria di MMU fare riferimento all'omonimo paragrafo del capitolo successivo.

All'atto del power on o del reset la logica di controllo é programmata in modo da allocare un segmento di 1K Bytes alla fine dello spazio fisico d'indirizzamento della CPU, quindi la scheda parte con l'esecuzione del codice posto all'indirizzo fisico FFFF0H che corrisponde sempre con l'ultimo segmento da 16 Bytes della EPROM o FLASH EPROM di IC18.

Alcuni pacchetti software, come il GDOS ed il GCTR, si occupano autonomamente della gestione delle circuiterie di gestione memoria senza interessare direttamente l'utente.

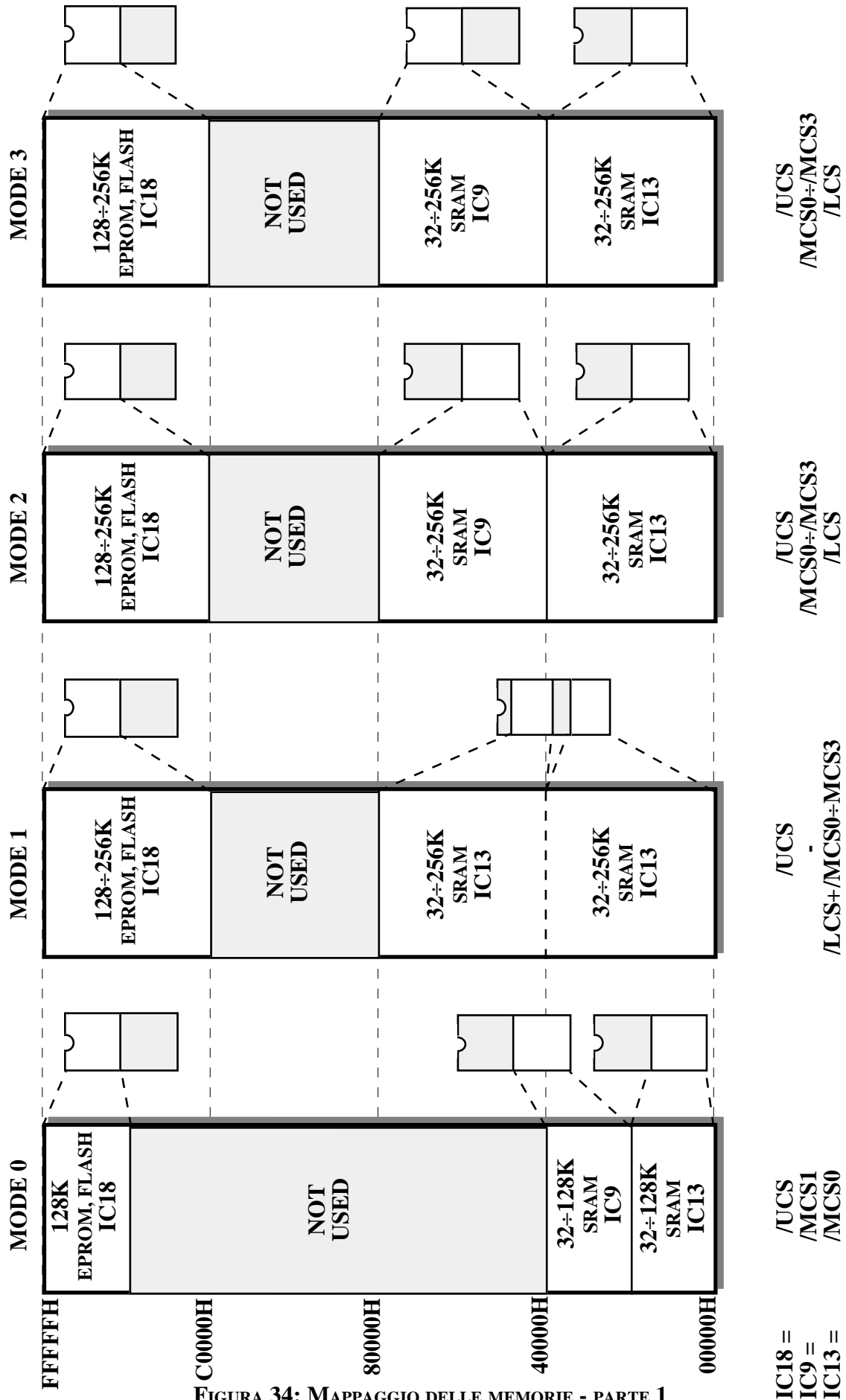


FIGURA 34: MAPPAGGIO DELLE MEMORIE - PARTE 1

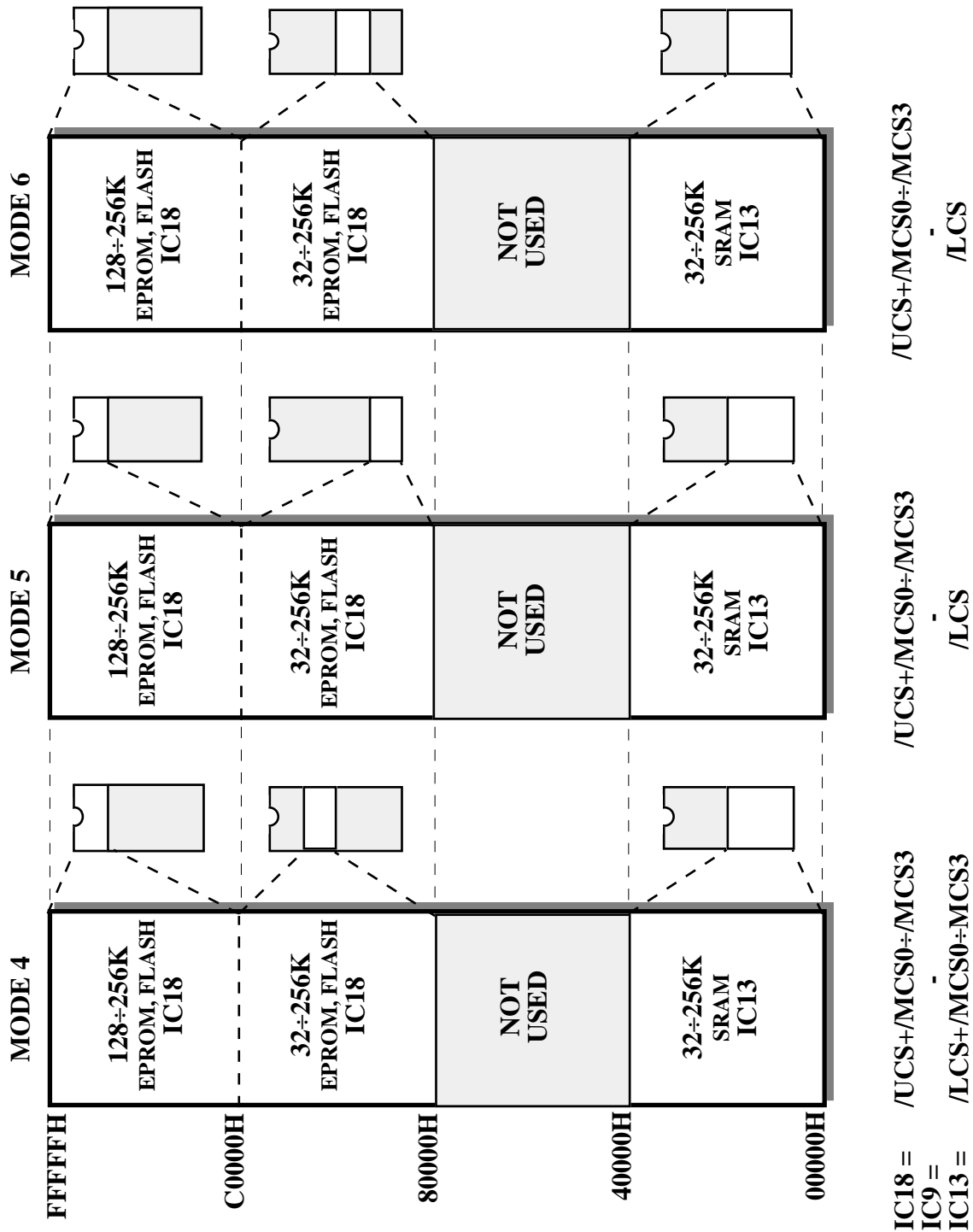


FIGURA 35: MAPPAGGIO DELLE MEMORIE - PARTE 2

DESCRIZIONE SOFTWARE DELLE PERIFERICHE DI BORDO

Nel paragrafo precedente sono stati riportati gli indirizzi di allocazione di tutte le periferiche e di seguito viene riportata una descrizione dettagliata della funzione e del significato dei relativi registri (al fine di comprendere le successive informazioni, fare sempre riferimento alle tabelle di indirizzamento I/O). Qualora la documentazione riportata fosse insufficiente fare riferimento direttamente alla documentazione tecnica della casa costruttrice del componente. In questo paragrafo inoltre non vengono descritte le sezioni che fanno parte del microprocessore; per quanto riguarda la programmazione di quest'ultime si faccia riferimento all'appendice B di questo manuale. Nei paragrafi successivi si usano le indicazioni **D0÷D7** e **.0÷7** per fare riferimento ai bits della combinazione utilizzata nelle operazioni di I/O ad 8 bits.

WATCH DOG

Il retrigger della circuiteria di watch dog presente sulla **GPC® 188F**, avviene tramite una semplice operazione di input e/o output al registro RWD. Affinché la circuiteria di watch dog non intervenga, è indispensabile retriggerarla ad intervalli regolari di durata inferiore al tempo d'intervento. Se ciò non avviene e tramite il jumper J5 la circuiteria è attivata, una volta scaduto il tempo d'intervento la scheda viene resettata. Il tempo d'intervento è di circa 1,4 sec e nella condizione di default, la circuiteria è disabilitata. Da ricordare che il dato letto o scritto durante le operazioni di retrigger è completamente ininfluenza per la circuiteria di watch dog, ma riguarda invece le altre circuiterie i cui registri sono allocati allo stesso indirizzo di I/O.

EEPROM SERIALE

Per quanto riguarda la gestione del modulo di EEPROM seriale (IC 23), si faccia riferimento alla documentazione specifica del componente. In questo manuale tecnico non viene riportata alcuna informazione software in quanto la modalità di gestione è articolata e prevede una conoscenza approfondita del componente e comunque l'utente può usare le apposite procedure ad alto livello fornite nel pacchetto di programmazione. Si ricorda solo che i primi 32 bytes (0÷31) sono riservati e perciò si deve evitare la modifica dei medesimi. La logica di controllo consente la gestione software della EEPROM tramite alcuni bits del registro RE2, con le seguenti corrispondenze:

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

Data l'implementazione hardware della circuiteria di gestione del modulo di EEPROM seriale, si ricorda che di tale dispositivo i segnali **A0,A1,A2** dello slave address sono tutti posti a **0** logico. Lo stato logico 0 dei bit corrisponde allo stato logico basso (=0 V) del relativo segnale, mentre lo stato logico 1 dei bit corrisponde allo stato logico alto (=5 V) del segnale.

I rimanenti sei bit del registro RE2 devono essere gestiti tenendo conto della programmazione fornita alle circuiterie di MMU, stato batteria LED di attività infatti i registri di queste sezioni sono allocati allo stesso indirizzo di I/O.

Il registro RE2 è azzerato (tutti i bits a 0) in fase di reset o power on, di conseguenza in seguito ad una di queste fasi i due segnali dell'EEPROM sono allo stato logico basso.

Lo stato dei segnali di gestione EEPROM può essere acquisito via software effettuando un'operazione di input sullo stesso registro ed esaminando i bits descritti.

STATO BATTERIA

Lo stato della batteria BT1 presente sulla **GPC® 188F** può essere acquisito via software, effettuando una semplice operazione di input all'indirizzo di allocazione del registro BAT ed esaminando il bit D4, che ha la seguente corrispondenza:

| | | |
|------------------|----|-----------------------------|
| BAT.4 = 0 | -> | batteria scarica (<2,265 V) |
| BAT.4 = 1 | -> | batteria carica (> 2,265 V) |

Per ulteriori informazioni sulla batteria di bordo e della relativa circuiteria di back up fare riferimento ai precedenti appositi paragrafi.

INGRESSI DI CONFIGURAZIONE

La **GPC® 188F** dispone di 12 ingressi di configurazione settabili dall'utente ed acquisibili via software, con le modalità di seguito riportate.

Il dip switch DSW1 può essere acquisito effettuando una semplice operazione di input all'indirizzo di allocazione del registro DSW1. La corrispondenza tra i bit del registro e le linee del dip switch è la seguente:

| | | |
|----|----|---------------|
| D7 | -> | DSW1.8 |
| D6 | -> | DSW1.7 |
| D5 | -> | DSW1.6 |
| D4 | -> | DSW1.5 |
| D3 | -> | DSW1.4 |
| D2 | -> | DSW1.3 |
| D1 | -> | DSW1.2 |
| D0 | -> | DSW1.1 |

La combinazione è in logica negata, ovvero il dip in **ON** fornisce lo stato logico **0** al corrispondente bit, mentre il dip in **OFF** fornisce lo stato logico **1**.

I jumpers di configurazione J16÷J19 sono collegati a 4 linee di handshake hardware del SCC 85C30 e possono essere acquisiti via software, effettuando una semplice operazione di input sui registri di stato delle due sezioni (RSA e RSB), con la seguente corrispondenza:

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

L'acquisizione è in logica diretta, ovvero il jumper **non connesso** fornisce lo stato logico **0** al corrispondente bit, mentre il jumper **connesso** fornisce lo stato logico **1**. Per quanto riguarda le modalità di acquisizione dei registri di stato del SCC 85C30, fare riferimento al paragrafo successivo. Il jumper J18 svolge la funzione di selettore delle modalità RUN (non connesso) o DEBUG (connesso), caratteristica di alcuni pacchetti software della **grifo®**.

LEDS DI ATTIVITA'

La logica di controllo consente la gestione software dei due LEDs di attività LD3 ed LD4, tramite altrettanti bits del registro LED, con le seguenti corrispondenze:

| | | |
|--------------|----|-----|
| LED.2 | -> | LD4 |
| LED.3 | -> | LD3 |

L'**attivazione** avviene effettuando una operazione di output all'indirizzo di allocazione del registro LED con il relativo bit resettato a **0**. Logicamente la **disattivazione** avviene tramite un'analogica operazione di output ma con il bit settato a **1**.

I rimanenti sei bits del registro LED devono essere settati tenendo conto della programmazione fornita alla circuiteria di MMU, stato batteria ed EEPROM infatti i registri di queste sezioni sono allocati allo stesso indirizzo di I/O.

Il registro LED è azzerato (tutti i bits a 0) in fase di reset o power on, di conseguenza in seguito ad una di queste fasi i due LEDs sono attivi.

Lo stato dei LEDs di attività può essere acquisito via software effettuando un'operazione di input sullo stesso registro ed esaminando i bits descritti.

PROTEZIONE IN SCRITTURA

La SRAM montata su IC9 può essere gestita anche con protezione in scrittura. In questo caso ogni scrittura su questa SRAM può avvenire solo effettuando due operazioni sequenziali:

- effettuare operazione di output all'indirizzo di allocazione del registro WRP, in modo da disabilitare la protezione;
- effettuare la scrittura in memoria.

Il dato da scrivere nel registro WRP è privo di significato, infatti è la sola operazione di output che disarmo la protezione. In corrispondenza di ogni successiva scrittura in memoria od operazione di output la protezione in scrittura viene riabilitata in modo da avere la massima garanzia contro tutte le scritture accidentali. Da ricordare che il jumper J2 abilita o disabilita la circuiteria di protezione, quindi in caso di J2 non connesso la SRAM di IC9 non è protetta e può essere gestita con le modalità tradizionali.

Alcuni pacchetti software, come il GDOS, si occupano autonomamente del riconoscimento e della eventuale gestione della circuiteria di protezione in scrittura, senza interessare direttamente l'utente. Dopo un reset o power on la circuiteria di protezione è abilitata.

DISATTIVAZIONE RICHIESTA DMA

I dati da trasmettere sulla linea seriale B possono essere trasferiti in DMA tramite il canale 1 della CPU, in modo da non rubare tempo al programma applicativo in esecuzione. Effettuando una operazione di input od output all'indirizzo di allocazione del registro DMA si disattiva la richiesta di trasferimento dati del SCC 85C30. In questo modo via software è possibile disabilitare il trasferimento dati da memoria a seriale B, quando non ci sono dati da trasferire, senza dover disabilitare il canale 1 di DMA.

Dopo un reset o power on la richiesta trasferimento DMA descritta non è disattivata.

REAL TIME CLOCK

Questa periferica è vista in 16 locazioni di I/O consecutive di cui 3 di stato e le rimanenti 13 per i dati. I registri dati sono utilizzati sia per operazioni di input (acquisizione dell'orario attuale) che di output (per l'inizializzazione dell'orologio) così come i registri di stato i quali sono utilizzati in scrittura (per la programmazione del modo di funzionamento dell'orologio) ed in lettura (per determinare lo stato dell'orologio). Per quanto riguarda il significato dei registri dati vale la corrispondenza:

| | | | |
|--|--------------------------|-----------------------------|---------------|
| S1 | - Unità dei secondi | - 4 bit meno significativi: | S1.3÷S1.0 |
| S10 | - Decine dei secondi | - 3 bit meno significativi: | S10.2÷S10.0 |
| MI1 | - Unità dei minuti | - 4 bit meno significativi: | MI1.3÷MI1.0 |
| MI10 | - Decine dei minuti | - 3 bit meno significativi: | MI10.2÷MI10.0 |
| H1 | - Unità delle ore | - 4 bit meno significativi: | H1.3÷H1.0 |
| H10 | - Decine delle ore | - 2 bit meno significativi: | H10.1÷H10.0 |
| Il terzo bit di tale registro, H10.2, indica l'AM/PM | | | |
| D1 | - Unità del giorno | - 4 bit meno significativi: | D1.3÷D1.0 |
| D10 | - Decine del giorno | - 2 bit meno significativi: | D10.1÷D10.0 |
| MO1 | - Unità del mese | - 4 bit meno significativi: | MO1.3÷MO1.0 |
| MO10 | - Decine del mese | - 1 bit meno significativo: | MO10.0 |
| Y1 | - Unità dell'anno | - 4 bit meno significativi: | Y1.3÷Y1.0 |
| Y10 | - Decine dell'anno | - 4 bit meno significativi: | Y10.3÷Y10.0 |
| W | - Giorno della settimana | - 3 bit meno significativi: | W.2÷W.0 |

Per quest'ultimo registro vale la corrispondenza:

| W.2 | W.1 | W.0 | Giorno della settimana |
|-----|-----|-----|------------------------|
| 0 | 0 | 0 | Domenica |
| 0 | 0 | 1 | Lunedì |
| 0 | 1 | 0 | Martedì |
| 0 | 1 | 1 | Mercoledì |
| 1 | 0 | 0 | Giovedì |
| 1 | 0 | 1 | Venerdì |
| 1 | 1 | 0 | Sabato |

I tre registri di controllo sono invece utilizzati come segue:

D7 D6 D5 D4 D3 D2 D1 D0
REGD = NU NU NU NU 30S IF B H

dove:

NU = Non usato

30S = Se attivo (1) permette di effettuare una correzione di 30 secondi dell'orario. Una volta settato i secondi del RTC vengono azzerati ed i minuti incrementati se il precedente valore dei secondi era superiore o uguale a 30.

IF = Gestisce lo stato d'interrupt del RTC. In lettura riporta lo stato attuale d'interrupt (1=attivo e viceversa), mentre se resettato con una scrittura determina la fine interrupt, quando il RTC lavora in interrupt mode.

B = Indica se possono essere effettuate operazioni di lettura/scrittura dei registri:
 1 -> operazioni impossibili e viceversa.

H = Se attivo (1) effettua la memorizzazione dell'orario fissato.

D7 D6 D5 D4 D3 D2 D1 D0

REGE = NU NU NU NU T1 T0 I M

dove:

NU = Non usato.

T1 T0 = Determinano la durata del periodo di interrupt

0 0 -> 1/64 secondo

0 1 -> 1 secondo

1 0 -> 1 minuto

1 1 -> 1 ora

I = Determina modalità di gestione interrupt: se settato seleziona l'interrupt mode in cui l'interrupt si attiva allo scadere del periodo programmato e si disattiva con un reset del bit IF del registro D; se resettato seleziona lo standard mode in cui l'interrupt si attiva allo scadere del periodo programmato e si disattiva autonomamente dopo 7,8 msec.

M = Se settato disabilita il pin di interrupt del RTC e viceversa.

D7 D6 D5 D4 D3 D2 D1 D0

REGF = NU NU NU NU T 24/12 S R

dove:

NU = Non usato.

T = Stabilisce da quale contatore interno prelevare il segnale di conteggio:

1 -> contatore principale (conteggio veloce per test);

0 -> 15° contatore (conteggio normale).

24/12 = Stabilisce il modo di conteggio delle ore:

1 -> 0÷23;

0 -> 0÷11 con AM/PM.

S = Se settato provoca l'arresto dell'avanzamento dell'orologio fino alla successiva abilitazione.

R = Se settato (1) provoca il reset di tutti i contatori interni.

Dopo un reset o power on il real time clock non viene inizializzato in modo da garantire il corretto mantenimento dei suoi dati anche dopo uno spegnimento od un azzeramento, assicurato dall'eventuale circuiteria di back up.

MEMORY MANAGEMENT UNIT

L'allocazione dello spazio d'indirizzamento fisico delle memorie che possono essere montate sulla **GPC® 188F** all'interno dello spazio d'indirizzamento logico del microprocessore, è affidato ad una efficiente circuiteria di MMU. Il compito della MMU é quello di impaginare la memoria in modo da allocare diverse aree dei dispositivi fisici nella stessa area logica della CPU, aumentandone così l'indirizzamento complessivo. Le modalità di impaginazione sono state scelte dalla **grifo®** tenendo conto delle esigenze dei pacchetti software disponibili per la scheda e non possono essere variati, a meno di specifiche esigenze dell'utente.

La sezione viene programmata tramite l'apposito registro MMU allocato nello spazio di I/O, con la seguente corrispondenza:

| | | | |
|--------------|----|--------------------------------|-----------|
| MMU.5 | -> | prima linea di impaginazione | M0 |
| MMU.6 | -> | seconda linea di impaginazione | M1 |
| MMU.7 | -> | terza linea di impaginazione | M2 |

dove:

| M2 | M1 | M0 | = Selezionano modo di impaginazione memorie. |
|----|----|----|--|
| 0 | 0 | 0 | -> Modo 0 |
| 0 | 0 | 1 | -> Modo 1 |
| 0 | 1 | 0 | -> Modo 2 |
| 0 | 1 | 1 | -> Modo 3 |
| 1 | 0 | 0 | -> Modo 4 |
| 1 | 0 | 1 | -> Modo 5 |
| 1 | 1 | 0 | -> Modo 6 |
| 1 | 1 | 1 | -> Non usato (disponibile per future espansioni) |

Fare riferimento alle figure 34 e 35, per avere tutte le informazioni relative ai 7 modi di configurazione memoria, che la sezione MMU può gestire.

All'atto del power on o del reset il registro MMU é azzerato (tutti i bits a 0); questo equivale ad una programmazione della sezione di MMU in modo 0 in cui i primi 32÷128K indirizzati dalla CPU coincidono con i primi 32÷128K della SRAM di IC13, i secondi 32÷128K indirizzati dalla CPU coincidono con i primi 32÷128K della SRAM di IC9 e gli ultimi 128K indirizzati dalla CPU coincidono con gli ultimi 128K della EPROM o FLASH EPROM di IC18.

Lo stato delle linee di impaginazione può essere anche acquisito via software effettuando un'operazione di input sullo stesso registro ed esaminando i bits descritti.

A/D CONVERTER

Fare riferimento all'appendice B di questo manuale in cui é riportata la descrizione software dell'A/D converter LM 12H458. Qualora queste informazioni fossero ancora insufficienti, fare riferimento alla documentazione tecnica della casa costruttrice.

SCC 85C30

Questa periferica si occupa di gestire la comunicazione seriale sulla **GPC® 188F** in modo indipendente per le due linee A e B, che può avvenire in uno dei tre modi possibili:

- Funzionamento asincrono
- Funzionamento sincrono
- Funzionamento SDLC/HDLC

La periferica é gestita tramite 4 registri allocati nello spazio di I/O. Di questi i registri RSA, RSB sono utilizzati per gestire e determinare lo stato della periferica (uno per ogni linea seriale) mentre i registri RDA, RDB sono utilizzati per trasferire dati nei confronti della stessa. Sia i registri di stato che quelli per i dati possono essere utilizzati sia in operazioni di input (acquisizione dello stato della periferica o dei dati ricevuti) che di output (per la programmazione della periferica o per i dati da trasmettere). Il SCC 85C30 usa una tecnica di accesso indiretto, con cui si tramite un opportuno settaggio del registro di stato si può accedere ad una serie di altri registri interni; in particolare sono disponibili 16 registri di scrittura ed 9 registri di lettura, che vengono descritti nelle seguenti pagine riportando il significato di tutti i bits che li compongono.

Registro di scrittura 0

Tale registro esegue operazioni di comando, di azzeramento di alcuni stati della periferica e permette di puntare ad altri registri:

D7 D6 D5 D4 D3 D2 D1 D0

WR0 = CRC1 CRC0 CD2 CD1 CD0 P2 P1 P0

dove:

CRC1 CRC0 = Selezionano uno dei seguenti comandi di reset

- | | | |
|---|---|---|
| 0 | 0 | -> Codice nullo |
| 0 | 1 | -> Reset del controllore CRC del ricevitore |
| 1 | 0 | -> Reset del controllore CRC del trasmettitore |
| 1 | 1 | -> Reset memorizzazione mancanza dati in trasm. |

CD2 CD1 CD0 = Seleziona uno dei seguenti comandi base

- | | | | |
|---|---|---|---|
| 0 | 0 | 0 | -> Comando nullo |
| 0 | 0 | 1 | -> Puntatore ai registri WR8÷WR15, RD8÷RD15 |
| 0 | 1 | 0 | -> Reset interruzioni di stato esterno |
| 0 | 1 | 1 | -> Invio di Abort in modo SDLC |
| 1 | 0 | 0 | -> Abilitazione interrupt sul successivo chr in ricezione |
| 1 | 0 | 1 | -> Reset interrupt di trasmissione in corso |
| 1 | 1 | 0 | -> Reset errori |
| 1 | 1 | 1 | -> Reset IUS |

P2 P1 P0 = Determinano quale registro di stato deve essere interessato dalla prossima operazione di lettura/scrittura del registro di stato (puntatore)

- | | | | |
|---|---|---|-------------------------|
| 0 | 0 | 0 | -> WR0, RDO, WR8, RD8 |
| 0 | 0 | 1 | -> WR1, RD1, WR9 |
| 0 | 1 | 0 | -> WR2, RD2, WR10, RD10 |
| 0 | 1 | 1 | -> WR3, RD3, WR11 |
| 1 | 0 | 0 | -> WR4, WR12, RD12 |
| 1 | 0 | 1 | -> WR5, WR13, RD13 |
| 1 | 1 | 0 | -> WR6, WR14 |
| 1 | 1 | 1 | -> WR7, WR15, RD15 |

Registro di scrittura 1

Tale registro contiene i bit di controllo per i vari modi di interruzione ed i modi di Wait/Ready:

D7 D6 D5 D4 D3 D2 D1 D0

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

dove:

- | | |
|---------|---|
| AWR | = Abilitazione Wait/Ready: AWR=0 -> disabilitato |
| W/R | = Funzione /Wait o Ready: W/R=0 -> /Wait |
| R/T | = Wait/Ready su ricezione o trasmissione: R/T=0 -> trasmissione |
| IM1 IM0 | = Selezionano tipo di interrupt in ricezione |
| 0 0 | -> Interrupt in ricezione disabilitate |
| 0 1 | -> Interrupt in ricezione solo su primo carattere |
| 1 0 | -> Interrupt su tutti i caratteri in ricezione |
| 1 1 | -> Interrupt su condizioni particolari di ricezione |
| P | -> Se attivo la parità costituisce una condizione particolare |
| AIT | -> Abilitazione interrupt di trasmissione: AIT=1 -> abilitata |
| AIE | -> Abilitazione interrupt da variazione esterna: AIE=1 -> abilitata |

Registro di scrittura 2

Tale registro é utilizzato per definire il vettore d'interrupt per la periferica (tale vettore é condiviso da entrambi i canali):

D7 D6 D5 D4 D3 D2 D1 D0

WR2 = V7 V6 V5 V4 V3 V2 V1 V0

dove:

Vn = Bit n del vettore d'interrupt

Registro di scrittura 3

Tale registro contiene i bit di controllo della logica del ricevitore ed altri parametri:

D7 D6 D5 D4 D3 D2 D1 D0

WR3 = R1 R0 AA IF AR RI CS A

dove:

R1 R0 = Fissano il numero di bit per carattere in ricezione

0 0 -> 5 bit

0 1 -> 6 bit

1 0 -> 7 bit

1 1 -> 8 bit

AA = Autoabilitazione tramite handshake: AA=1 -> autoab.

IF = Introduce fase di ricerca: IF=1 -> fase introdotta

AR = Abilitazione CRC del ricevitore: AR=1 -> abilitato

RI = Modo ricerca indirizzi SDLC: RI=1 -> abilitato

CS = Inibizione caricamento carattere di sincronizzazione: CS=1 -> inibizione attiva

A = Abilitazione ricevitore: A=1 -> abilitato

Registro di scrittura 4

Tale registro contiene i bit di controllo che influenzano sia il ricevitore che il trasmettitore:

D7 D6 D5 D4 D3 D2 D1 D0

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

dove:

VC1 VC0 = Selezionano la frequenza di comunicazione dati

0 0 -> Frequenza dati= frequenza clock (BITRATE=1)

0 1 -> Frequenza dati= 1/16 frequenza clock (BITRATE=16)

1 0 -> Frequenza dati= 1/32 frequenza clock (BITRATE=32)

1 1 -> Frequenza dati= 1/64 frequenza clock (BITRATE=64)

MS1 MS0 = Selezionano tipo di sincronizzazione

0 0 -> Sincronismo programmato a 8 bit

0 1 -> Sincronismo programmato a 16 bit

1 0 -> Modo SDLC (sequenza di flag 01111110)

1 1 -> Modo sincronismo esterno

BS1 BS0 = Selezionano i bit di stop per comunicazioni asincrone

0 0 -> Modi sincroni

0 1 -> 1 bit di stop per carattere

1 0 -> 1+1/2 bit di stop per carattere

1 1 -> 2 bit di stop per carattere

P/D = Parità pari o dispari: P/D=1 -> parità pari

P = Abilitazione controllo di parità: P=1 -> abilitato

Registro di scrittura 5

Tale registro contiene i bit di controllo che influenzano le operazioni del trasmettitore (eccetto C/S che condiziona anche il ricevitore):

D7 D6 D5 D4 D3 D2 D1 D0

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

dove:

| | | |
|---------|----|---|
| DTR | = | Abilitazione pin /DTR: DTR=1 -> /DTR attivo (basso) |
| BC1 BC0 | = | Selezionano il numero di bits per carattere in trasmissione |
| 0 0 | -> | 5 bit o meno |
| 0 1 | -> | 7 bit |
| 1 0 | -> | 6 bit |
| 1 1 | -> | 8 bit |
| IB | = | Invio di break su linea di trasmissione: IB=1 -> invio |
| AT | = | Abilitazione trasmettitore: AT=1 -> abilitato |
| C/S | = | Seleziona polinomio CRC: C/S=1 -> polinomio CRC 16 C/S=0 -> polinomio SDLC |
| RTS | = | Abilitazione pin /RTS: RTS=1 -> /RTS attivo (basso) |
| A | = | Abilitazione CRC di trasmissione: A=1 -> abilitato |

Registro di scrittura 6

Tale registro contiene parte delle informazioni necessarie per la sincronizzazione dei dati da trasmettere/ricevere:

D7 D6 D5 D4 D3 D2 D1 D0

WR6 = S7 S6 S5 S4 S3 S2 S1 S0

dove:

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

Registro di scrittura 7

Tale registro contiene la seconda parte di informazioni necessarie per la sincronizzazione dei dati da trasmettere/ricevere

D7 D6 D5 D4 D3 D2 D1 D0

WR7 = S15 S14 S13 S12 S11 S10 S9 S8

dove:

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

Registro di scrittura 8

Tale registro contiene il dato da trasmettere, infatti riporta il contenuto del buffer di trasmissione:

D7 D6 D5 D4 D3 D2 D1 D0

WR8 = TX7 TX6 TX5 TX4 TX3 TX2 TX1 TX0

dove:

TXn = Bit n del dato da trasmettere

Registro di scrittura 9

Tale registro consente di definire i bit di controllo dell'interrupt e di resettare i canali dell'UART:

D7 D6 D5 D4 D3 D2 D1 D0

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

dove:

R1 R0 = Definiscono il reset dei canali dell'UART

0 0 -> Nessun canale resettato

0 1 -> Reset del canale B

1 0 -> Reset del canale A

1 1 -> Reset hardware di entrambi i canali

0 = Non usato (deve essere a zero)

SH/L = Modifica del vettore d'interrupt alto, basso: SH/L=1 -> modifica V6÷V4
SH/L=0 -> modifica V3÷V1

MIE = Abilitazione globale interrupts: MIE=1 -> interrupts abilitati

DLC = Disattiva interrupts in catena con priorità inferiore: DLC=1 -> disattiva

NV = Disattiva restituzione del vettore d'interrupt: NV=1 -> disattiva

VIS = Attiva variazione vettore d'interrupt da stato: VIS=1 -> vettore variabile

Registro di scrittura 10

Tale registro contiene i bit che determinano il modo di funzionamento del canale dell'UART in uso:

D7 D6 D5 D4 D3 D2 D1 D0

WR10 = CRC FM1 FM0 GP MFI AFU LM S

dove:

CRC = Definisce inizializzazione CRC: CRC=1 -> CRC inizializzato a 1
CRC=0 -> CRC inizializzato a 0

FM1 FM0 = Definiscono il modo di codifica dei dati

0 0 -> Modo NRZ

0 1 -> Modo NRZI

1 0 -> Modo FM1 (transizione alta)

1 1 -> Modo FM0 (transizione bassa)

GP = Attivazione su POLL: GP=1 -> attivazione

MFI = Definisce stato della linea inattiva: MFI=1 -> stato alto
MFI=0 -> stato basso

AFU = Attiva trasmissione abort in caso di mancanza dati: AFU=1 attiva

LM = Attiva modalità di loop: LM=1 -> attiva modo loop

S = Seleziona lunghezza bits di sincronismo: S=1 -> 6 bits
S=0 -> 8 bits

Registro di scrittura 11

Tale registro consente di definire i segnali da cui prelevare il segnale di clock per la ricezione e la trasmissione:

D7 D6 D5 D4 D3 D2 D1 D0

WR11 = XT RC1 RC0 TC1 TC0 TR TR1 TR0

dove:

| | | |
|---------|--|---|
| XT | | = Se attivo (1) porta sul pin /RTxC il segnale XTAL |
| RC1 RC0 | | = Selezionano la sorgente per il clock di ricezione |
| 0 0 | | -> Clock = segnale presente sul pin /RTxC |
| 0 1 | | -> Clock = segnale presente sul pin /TRxC |
| 1 0 | | -> Clock = segnale in uscita dal baud rate generator interno |
| 1 1 | | -> Clock = segnale in uscita dal DPLL |
| TC1 TC0 | | = Selezionano la sorgente per il clock di trasmissione |
| 0 0 | | -> Clock = segnale presente sul pin /RTxC |
| 0 1 | | -> Clock = segnale presente sul pin /TRxC |
| 1 0 | | -> Clock = segnale in uscita dal baud rate generator interno |
| 1 1 | | -> Clock = segnale in uscita dal DPLL |
| TR | | = Definisce direzionalità per il pin /TRxC: TR=0 -> input TR=1 -> output |
| TR1 TR0 | | = Selezionano quale segnale portare sul piedino /TRxC |
| 0 0 | | -> Segnale d'uscita XTAL |
| 0 1 | | -> Segnale di clock del trasmettitore |
| 1 0 | | -> Segnale in uscita dal baud rate generator |
| 1 1 | | -> Segnale in uscita dal DPLL |

Registri di scrittura 12 e 13

Tali registri contengono la costante di tempo attraverso la quale è definibile la velocità di comunicazione della linea seriale in questione:

D7 D6 D5 D4 D3 D2 D1 D0

WR12 = TC7 TC6 TC5 TC4 TC3 TC2 TC1 TC0

WR13 = TC15 TC14 TC13 TC12 TC11 TC10 TC9 TC8

Da notare che dal baud rate di comunicazione si può calcolare la costante di tempo, con la seguente formula:

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

dove BITRATE coincide con il valore selezionato e programmato nel registro WR4, BAUDRATE coincide con la velocità desiderata espressa in bit per secondo e TC coincide con la costante a 16 bit da introdurre nei registri WR12 e WR13.

Registro di scrittura 14

Tale registro contiene una serie di bits con cui è possibile selezionare la sorgente per la temporizzazione della periferica e il modo di funzionamento della stessa:

D7 D6 D5 D4 D3 D2 D1 D0

WR14 = C2 C1 C0 LL AE DTR BRS BRE

dove:

| | |
|----------|---|
| C2 C1 C0 | = Definiscono i modi di funzionamento della periferica |
| 0 0 0 | -> Comando nullo |
| 0 0 1 | -> Seleziona modo ricerca |
| 0 1 0 | -> Azzeramento segnali di clock persi |
| 0 1 1 | -> Disabilita DPLL |
| 1 0 0 | -> Seleziona temporizzazione determinata dal baud rate generator interno |
| 1 0 1 | -> Seleziona temporizzazione determinata dal segnale applicato al piedino /RTxC |
| 1 1 0 | -> Seleziona modo FM |
| 1 1 1 | -> Seleziona modo NRZI |
| LL | = Attiva loop back locale: LL=1 -> loopback attivo |
| AE | = Seleziona funzionamento in auto echo: AE=1 -> auto echo selezionato |
| DTR | = Abilita funzionamento del /DTR: DTR=1 -> abilitato |
| BRS | = Seleziona sorgente per il baud rate: BRS=1 -> sorgente da pin PCLK BRS=0 -> sorgente da /RTxC o XTAL |
| BRE | = Abilita baud rate generator interno: BRE=1 -> abilita |

Registro di scrittura 15

Tale registro é composto da una serie di 8 bits con cui si definiscono particolari condizioni di interrupt della periferica:

D7 D6 D5 D4 D3 D2 D1 D0

WR15 = BA TU CTS SU DCD 0 ZC 0

dove:

| | |
|-----|--|
| BA | = Genera interrupt in corrispondenza di break/abort |
| TU | = Genera interrupt in corrispondenza di un carattere non trasmesso |
| CTS | = Genera interrupt in corrispondenza della variazione del /CTS |
| SH | = Genera interrupt in corrispondenza di variazioni del SYNC o hunt |
| DCD | = Genera interrupt in corrispondenza di variazioni del /DCD |
| 0 | = Non usato (deve essere a zero) |
| ZC | = Genera interrupt in corrispondenza di zero count |

Tutte queste indicazioni sono riferite ad uno stato logico alto del corrispondente pin.

Registro di lettura 0

Tale registro contiene una serie di bits con cui si possono avere informazioni a riguardo dello stato dell'UART:

D7 D6 D5 D4 D3 D2 D1 D0

RD0 = BA TU CTS SU DCD TBE ZC RCA

dove:

| | |
|-----|---|
| BA | = Break/abort avvenuto: BA=1 -> avvenuto |
| TU | = Carattere in trasmissione perso: TU=1 -> perso |
| CTS | = Stato del pin /CTS della periferica |
| SH | = Stato del pin /SYNC della periferica o stato di hunt |
| DCD | = Stato del pin /DCD della periferica |
| TBE | = Stato del buffer di trasmissione: TBE=1 -> vuoto |
| ZC | = Indica azzeramento del contatore per baud rate generator: ZC=1 -> azzeramento |
| RCA | = Carattere ricevuto: RCA=1 -> ricevuto |

Registro di lettura 1

Tale registro contiene indicazioni a riguardo dello stato di ricezione e trasmissione della periferica:

D7 D6 D5 D4 D3 D2 D1 D0

RD1 = EOF CRC ROE PE RC0 RC1 RC2 AS

dove:

EOF = Indica ricezione della fine del frame in modo SDLC

CRC = Indica presenza di un errore di CRC o di frame

ROE = Indica errore di overrun in ricezione

PE = Indica errore di parità in ricezione

RC0 = Indica presenza codice residuo 0

RC1 = Indica presenza codice residuo 1

RC2 = Indica presenza codice residuo 2

AS = Indica completamento delle trasmissioni in corso

Tutti questi stati descritti sono da associare allo stato logico 1 dei corrispondenti bits.

Registro di lettura 2

Tale registro contiene il dato programmato nel vettore d'interrupt della periferica:

D7 D6 D5 D4 D3 D2 D1 D0

RD2 = V7 V6 V5 V4 V3 V2 V1 V0

dove:

Vn = Bit n del vettore d'interrupt

Registro di lettura 3

Tale registro contiene lo stato degli interrupts pendenti. E' disponibile solo sulla seriale A ma riguarda entrambi i canali della periferica:

D7 D6 D5 D4 D3 D2 D1 D0

RD3 = 0 0 CAR CAT CAE CBR CBT CBE

dove:

0 = Non usato (restituito a zero)

CAR = Interrupt pendente in ricezione su canale A

CAT = Interrupt pendente in trasmissione su canale A

CAE = Interrupt pendente da variazione esterna su canale A

CBR = Interrupt pendente in ricezione su canale B

CBT = Interrupt pendente in trasmissione su canale B

CBE = Interrupt pendente da variazione esterna su canale B

Tutti gli stati riportati sono corrispondenti ad uno stato logico 1 del corrispondente bit.

Registro di lettura 8

Tale registro contiene il dato ricevuto, infatti riporta il contenuto del registro dati:

D7 D6 D5 D4 D3 D2 D1 D0

RD8 = RX7 RX6 RX5 RX4 RX3 RX2 RX1 RX0

dove:

RXn = Bit n del dato ricevuto

Registro di lettura 10

Tale registro contiene alcune informazioni generali sullo stato attuale della periferica:

D7 D6 D5 D4 D3 D2 D1 D0

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

dove:

1CM = Indica la perdita di un ciclo di clock

2CM = Indica la perdita di due cicli di clock

0 = Non usato (restituito a zero)

LS = Periferica in loop di trasmissione

OL = Periferica in lavoro sul loop

Tutte le indicazioni riportate sono relative ad uno stato logico 1 del corrispondente bit.

Registri di lettura 12 e 13

Tali registri riportano il valore della costante di tempo descritta in corrispondenza dei registri di scrittura WR12 e WR13 di cui costituiscono una semplice copia che può essere letta.

D7 D6 D5 D4 D3 D2 D1 D0

RD12 = TC7 TC6 TC5 TC4 TC3 TC2 TC1 TC0

RD13 = TC15 TC14 TC13 TC12 TC11 TC10 TC9 TC8

Registro di lettura 15

Tale registro riporta le informazioni definite con il registro di scrittura WR15, ovvero le informazioni riguardanti la programmazione in interrupt della periferica. Quindi per conoscere il significato dei bits che compongono questo registro si faccia riferimento alla descrizione del registro WR15.

D7 D6 D5 D4 D3 D2 D1 D0

RD15 = BA TU CTS SU DCD 0 ZC 0

Dopo una fase di reset o di power on il SCC 85C30 entrambe le linee seriali sono disattive, così come i relativi segnali hardware.

PPI 82C55

Questa periferica è vista in 4 registri: uno di stato (RC) e tre dei dati (PA, PB, PC) con cui si effettua la programmazione ed il comando della stessa. I registri dati sono utilizzati sia per operazioni di input (acquisizione linee dei port) che per quelle di output (settaggio linee dei port) ed ognuno di tali registri riporta i dati di I/O del corrispondente port. La periferica può operare in tre modi diversi:

MODO 0 = Prevede due port bidirezionali da 8 bit (A,B) e due port bidirezionali da 4 bit (C LOW, C HIGH); gli ingressi non sono latched, mentre le uscite lo sono; nessun segnale di handshaking.

MODO 1 = Prevede due port da 12 bit (A+C LOW, B+C HIGH) dove gli 8 bit dei port A e B costituiscono le linee di I/O, mentre i 4 bit del port C costituiscono le linee di handshaking. Gli ingressi e le uscite sono latched.

MODO 2 = Prevede un port da 13 bit (A+C3-7) dove gli 8 bit del port A costituiscono le linee di I/O, mentre i rimanenti 5 bit del port C costituiscono le linee di controllo. Un port da 11 bit (B+ C0-

2) dove gli 8 bit del port B costituiscono le linee di I/O ed i rimanenti 3 bit del port C costituiscono le linee di controllo. Sia gli ingressi che le uscite sono latchate.

La programmazione della periferica avviene scrivendo un byte nel registro di controllo RC, settando gli 8 bits del dato scritto con la seguente corrispondenza:

| | | | | | | | | |
|-------------|-----------|-----------|-----------|----------|-----------|-----------|----------|-----------|
| | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
| RC = | SF | M1 | M2 | A | CH | M3 | B | CL |

dove:

- SF = Se attivo (1) abilita il comando della periferica
- M1 M2 = Selezionano il modo di funzionamento
 - 0 0 = Selezione del modo 0
 - 0 1 = Selezione del modo 1
 - 1 X = Selezione del modo 2
- A = Se attivo (1) setta il port A in input e viceversa
- CH = Se attivo setta il nibble più significativo del port C in input e viceversa
- M3 = Se attivo (1) seleziona modo 1, viceversa seleziona modo 0
- B = Se attivo setta il port B in input e viceversa
- CL = Se attivo setta il nibble meno significativo del port C in input e viceversa.

Dopo una fase di reset o di power on il PPI 82C55 viene settato in modo 0 con tutti i port settati in input.

PERIFERICHE DELLA CPU

La descrizione dei registri e del relativo significato di tutte le periferiche interne della CPU (Timer Counter, DMA, controllore interrupts, generazione segnali di controllo, ecc) é disponibile nell'appendice B. Qualora queste informazioni fossero ancora insufficienti, fare riferimento alla documentazione tecnica della casa costruttrice.

SCHEDE ESTERNE

La scheda **GPC® 188F** si interfaccia a buona parte dei moduli della serie BLOCK e di interfaccia utente. Le risorse di bordo possono essere facilmente aumentate collegando la **GPC® 188F** alle numerose schede periferiche del carteggio **grifo®** tramite il **BUS ABACO®**. Anche schede in formato block con **ABACO® I/O BUS** possono essere collegate, sfruttando gli appositi mother boards. A titolo di esempio ne riportiamo un elenco con una breve descrizione delle caratteristiche di massima, per maggiori informazioni, richiedere la documentazione specifica:

KDL X24 - KDF 224

Keyboard Display LCD 2,4 righe 24 tasti - Keyboard Display Fluorescent 2 righe 24 tasti
Interfaccia tra 16 I/O TTL su connettore normalizzato I/O **ABACO®** a 20 vie e tastiera a matrice esterna da 24 tasti; display alfanumerico fluorescente 20x 2 o LCD 20x2, 20x4 retroilluminato a LEDs. Predisposizione per collegamento a tastiera telefonica.

QTP 24P

Quick Terminal Panel 24 tasti con interfaccia Parallela

Interfaccia operatore provvista di display alfanumerico fluorescente 20x 2 o LCD 20x2, 20x4 retroilluminato a LEDs; tastiera a membrana da 24 tasti di cui 12 configurabili dall'utente; 16 LEDs di stato; alimentatore a bordo scheda in grado di pilotare anche carichi esterni; interdaccia parallela basata su 16 I/O TTL di un connettore normalizzato I/O **ABACO®** a 20 vie. Tasti ed etichette personalizzabili tramite serigrafie da inserire in apposite tasche; opzione di contenitore metallico.

QTP G28

Quick Terminal Panel 28 tasti con LCD grafico

Interfaccia operatore provvista di display LCD grafico da 240x128 pixel retroilluminato con lampada a catodo freddo; tastiera a membrana da 28 tasti di cui 6 configurabili dall'utente; 16 LEDs di stato; alimentatore a bordo scheda; interdaccia seriale in RS 232, RS 422-485 o current loop; linea seriale ausiliaria in RS 232. Tasti ed etichette personalizzabili dall'utente tramite serigrafie da inserire in apposite tasche; contenitore metallico e plastico; EEPROM di set up; 256K EPROM o FLASH; Real Time Clock; 128K RAM; buzzer. Firmware di gestione che svolge funzione di terminale con primitive grafiche.

MB8 01

Mother Board 8 slots **ABACO®**

Mother Board con 8 slots del BUS industriale **ABACO®**; passo 5 TE; connettori normalizzati di alimentazione e di servizio; tasto di reset; 3 LEDs per le alimentazioni; foratura per aggancio ai rack.

SPB 04

Switch Power Bus mother board 4 slots **ABACO®**

Mother Board con 4 slots del BUS industriale **ABACO®**; 1 slot per alimentatore; passo 5 TE; connettori normalizzati di alimentazione; tasto di reset; foratura per aggancio ai rack.

ABB 05

Abaco® Block BUS 5 slots

Mother board **ABACO®** da 5 slots; passo 4 TE; guidaschede; connettori normalizzati di alimentazione; tasto di reset; LEDs per alimentazioni; interfaccia **ABACO® I/O BUS**; sezione alimentatrice per +5 Vdc; sezione alimentatrice per +V Opto; sezioni alimentatrici galvanicamente isolate; tre tipi di alimentazione: da rete, bassa tensione o stabilizzata. Attacco rapido per guide Ω .

IAC 01

Interface Adapter Centronics

Interfaccia tra 16 I/O TTL su connettore normalizzato I/O **ABACO**[®] a 20 vie e connettore a vaschetta D 25 vie femmina con pin out standard Centronics per la gestione di una stampante parallela.

OBI N8 - OBI P8

Opto BLOCK Input NPN-PNP

Interfaccia per 8 input optoisolati e visualizzati tipo NPN, PNP, connettore a morsettiera, connettore normalizzato I/O **ABACO**[®] a 20 vie; sezione alimentatrice; attacco rapido per guide DIN 46277-1 e 3.

TBO 01 - TBO 08

Transistor BLOCK Output

Interfaccia per 16 connettore normalizzato I/O **ABACO**[®] a 20 vie; 16 o 8 output a transistor in Open Collector da 45 Vcc 3 A su connettore a morsettiera. Uscite optoisolate e visualizzate; attacco rapido per guide DIN 6277-1 e 3.

RBO 08 - RBO 16

Relé BLOCK Output

Interfaccia per connettore normalizzato I/O **ABACO**[®] a 20 vie; 8 o 16 output visualizzati con relé da 3 A con MOV; connettore a morsettiera; attacco rapido per guide DIN 46277-1 e 3.

FBC 20 - FBC 120

Flat Block Contact 20 vie

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

DEB 01

Didactis Experimental Board

Scheda di supporto per l'utilizzo di 16 linee di I/O TTL. Comprende: 16 tasti; 16 LED; 4 digits; tastiera a matrice da 16 tasti; interfaccia per stampante Centronics, display LCD, display Fluorescente, connettore I/O **GPC**[®] 68; collegamento con il campo.

XBI 01

miXed BLOCK Input Output

Interfaccia tra 8 input + 8 output TTL (connettore normalizzato I/O **ABACO**[®] a 20 vie), con 8 output a transistor in Open Collector da 45 Vcc 3 A + 8 input con filtro a Pi-Greco (connettore a morsettiera). I/O optoisolati e visualizzati; attacco rapido per guide DIN 46277-1 e 3.

MCI 64

Memory Cards Interfaces 64 MBytes

Interfaccia per la gestione di Memory cards PCMCIA a 68 pins tramite un connettore normalizzato I/O **ABACO**[®]; sono disponibili driver per linguaggi ad alto livello.

DAC 16

Digital to Analog Converter 16 bits

2 D/A converter da 16 bit galvanicamente isolati; visualizzazione dati programmati; uscita ± 10 Vcc; taratura offset e guadagno. BUS a 8 bit; indirizzamento normale.

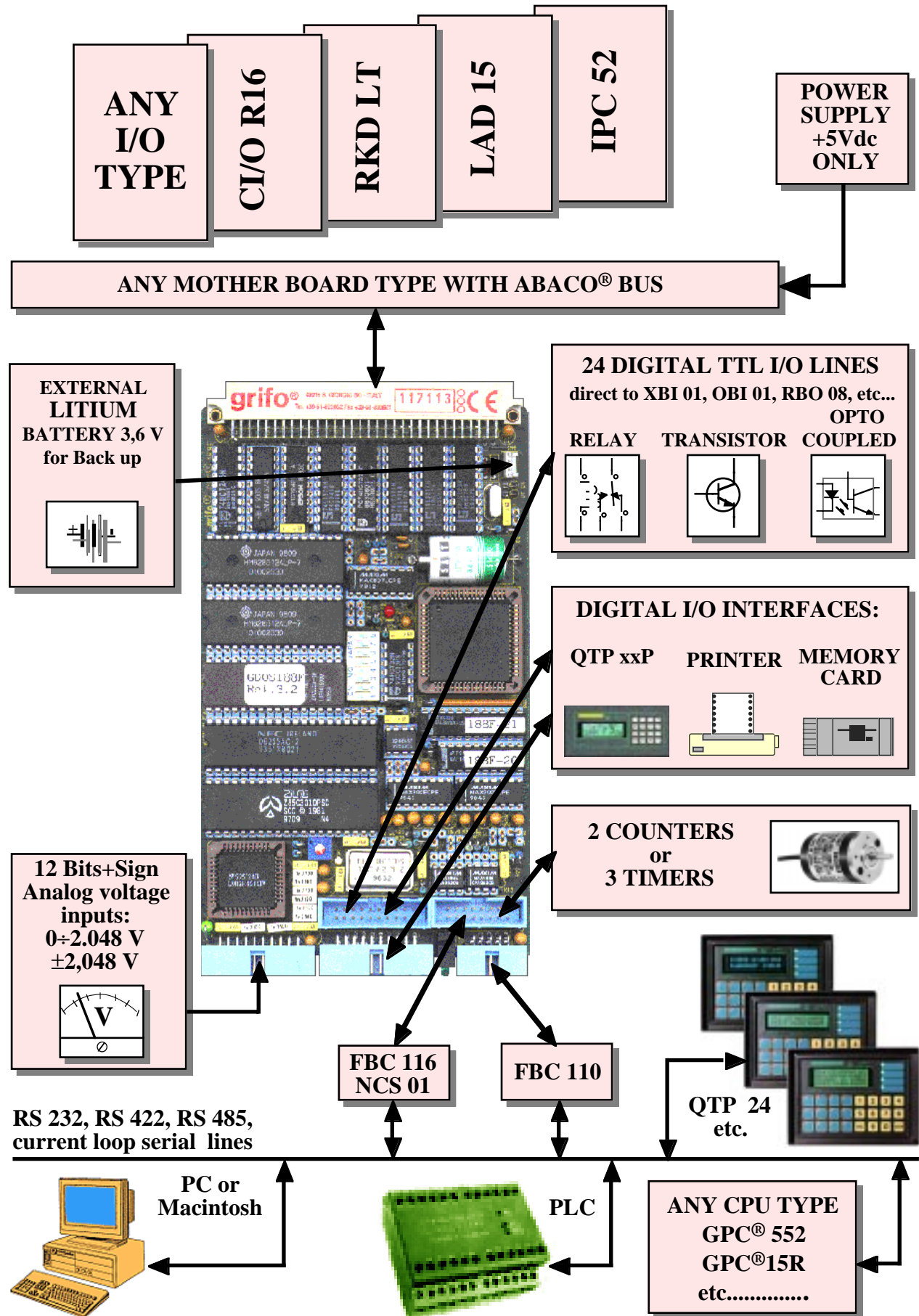


FIGURA 36: SCHEMA DELLE POSSIBILI CONNESSIONI

UCC A2

UART Communication Card

2 indipendenti linee seriali in RS 232, RS 422, RS 485 o current loop. Per ogni linea: buffer di 3 caratteri; comunicazione gestita dall'UART SCC 85C30; baud rate (da 50 a 115K baud), parità, stop bit e lunghezza dato programmabili via software; 4 dip switch. BUS a 8 bit; indirizzamento normale.

CI/O R16

16 Coupled Input Output Relé

16 ingressi optoisolati con filtro a Pi-Greco; tensione nominale di ingresso 24 Vcc. 16 output a microrelé da 1 A con soppressori di disturbi tipo MOV da 24 Vca. I/O visualizzati tramite LED; BUS a 8 bit; indirizzamento normale.

PCI 01

Peripheral Coupled Input

32 ingressi optoisolati con filtro a Pi-Greco; tensione nominale di ingresso 24 Vcc; ingressi visualizzati tramite LEDs; BUS a 8 o 16 bits; indirizzamento normale.

PCO 01

Peripheral Coupled Output

32 uscite a transistor in Open Collector da 45 Vcc, 500 mA, su connettore standardizzato. Uscite optoisolate e visualizzate tramite LEDs; unica tensione di alimentazione; BUS a 8 o 16 bits; indirizzamento normale.

IPC 52

Intelligent Peripheral Controller

Scheda periferica intelligente in grado di acquisire 24 segnali analogici generati da trasduttori da campo; 8 ingressi per PT 100, PT 1000; 8 ingressi per termocoppie J,K,S,T; 8 ingressi per segnali in tensione ± 2 V o corrente 0-20 mA; interrogazione tramite BUS **ABACO**[®] o tramite linea seriale in RS 232, RS 422-485 o current loop; 16 linee di I/O TTL; risoluzione di 16 bit più segno; 0,1 °C di precisione; 5 acquisizioni al secondo; funzionamento come data logger.

RKD LT

Remote Keyboard Display LCD Toshiba e Fluorescent FUTABA

Terminale intelligente con interfacciamento seriale (RS 232, RS 422-485, current loop) o parallelo (BUS **ABACO**[®]). Gestisce tastiera a matrice da 56 tasti; display fluorescenti FUTABA e/o LCD TOSHIBA; buzzer; 8 LEDs di segnalazione; EEPROM di configurazione.

JMS 34

Jumbo Multifunction Support per controllo assi

Scheda periferica per il controllo assi. 3 ingressi optoisolati per l'acquisizione di encoder incrementali bidirezionali; gestione tacca di zero. 4 canali di D/A converter da 12 bits; range di uscita ± 10 V. 8 ingressi optoisolati NPN. 8 uscite a transistor in Open Collector da 45 Vcc, 500 mA. Tutte le linee di I/O visualizzate tramite LEDs; BUS a 8 bit; indirizzamento esteso.

SBP 01

Switch BLOCK Power

Alimentatore switching in grado di generare tensioni da -12 a +40 Vdc e correnti fino a 4 A; ingresso da 12 a 26 Vac; ingresso per batteria di back up; uscita di power good; connettori a morsetti a rapida estrazione; montaggio su guide ad Ω .

BIBLIOGRAFIA

E' riportato di seguito, un elenco di manuali e note tecniche, a cui l'utente può fare riferimento per avere maggiori chiarimenti, sui vari componenti montati a bordo della scheda **GPC® 188F**.

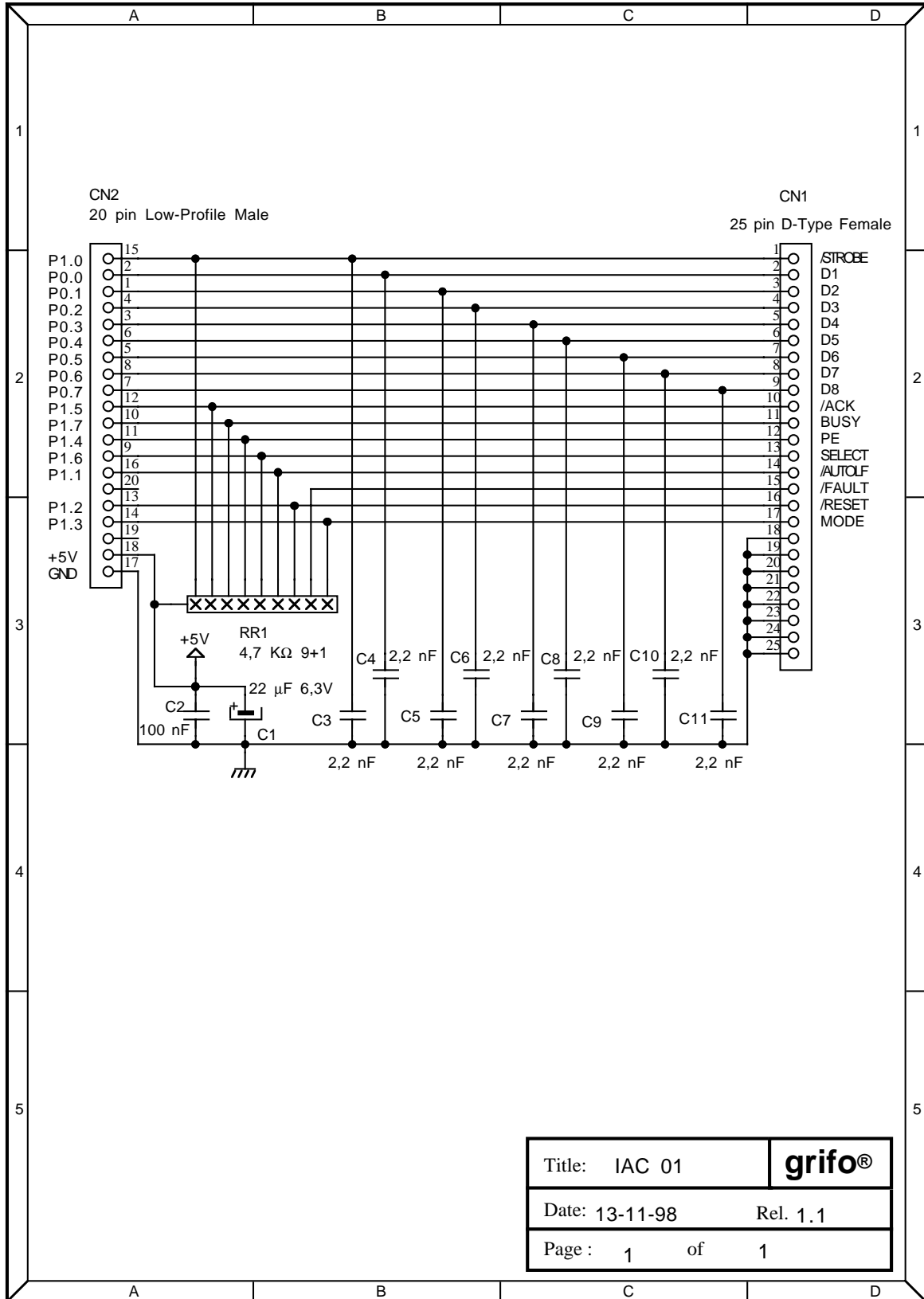
| | |
|---------------------------------|--|
| Manuale TEXAS INSTRUMENTS: | <i>The TTL Data Book - SN54/74 Families</i> |
| Manuale TEXAS INSTRUMENTS: | <i>RS-422 and RS-485 Interface Circuits</i> |
| Manuale HEWLETT PACKARD: | <i>Optoelectronics Designer's Catalog</i> |
| Manuale NEC: | <i>Microprocessors and Peripherals - Volume 3</i> |
| Manuale NEC: | <i>Memory Products</i> |
| Manuale AMD | <i>Flash Memory Products</i> |
| Manuale SGS-THOMSON: | <i>Programmable Logic Manual GAL Products</i> |
| Manuale MAXIM: | <i>New Releases Data Book - Volume IV</i> |
| Manuale MAXIM: | <i>New Releases Data Book - Volume V</i> |
| Manuale XICOR: | <i>Data Book</i> |
| Manuale NATIONAL SEMICONDUCTOR: | <i>LM12458 12-Bit + Sign Data Acquisition System</i> |
| Manuale ZILOG: | <i>Z8530 SCC Serial Communication Controller</i> |
| Documentazione SEIKO EPSON: | <i>RTC-62421 Real Time Clock module</i> |
| Manuale INTEL: | <i>16 Bit Embedded Controller Handbook</i> |

Per avere tutti gli aggiornamenti di tali manuali e di tutti i data-sheet fare riferimento anche ai siti INTERNET delle case madri costruttrici.



APPENDICE A: SCHEMI ELETTRICI

In questa appendice sono disponibili gli schemi elettrici delle interfacce per la GPC® 188F più frequentemente utilizzate. Tutte queste interfacce possono essere prodotte autonomamente dall'utente mentre solo alcune di esse sono schede grifo® standard e possono quindi essere ordinate.



| | |
|----------------|----------|
| Title: IAC 01 | grifo® |
| Date: 13-11-98 | Rel. 1.1 |
| Page : 1 | of 1 |

FIGURA A1: SCHEMA ELETTRICO IAC 01



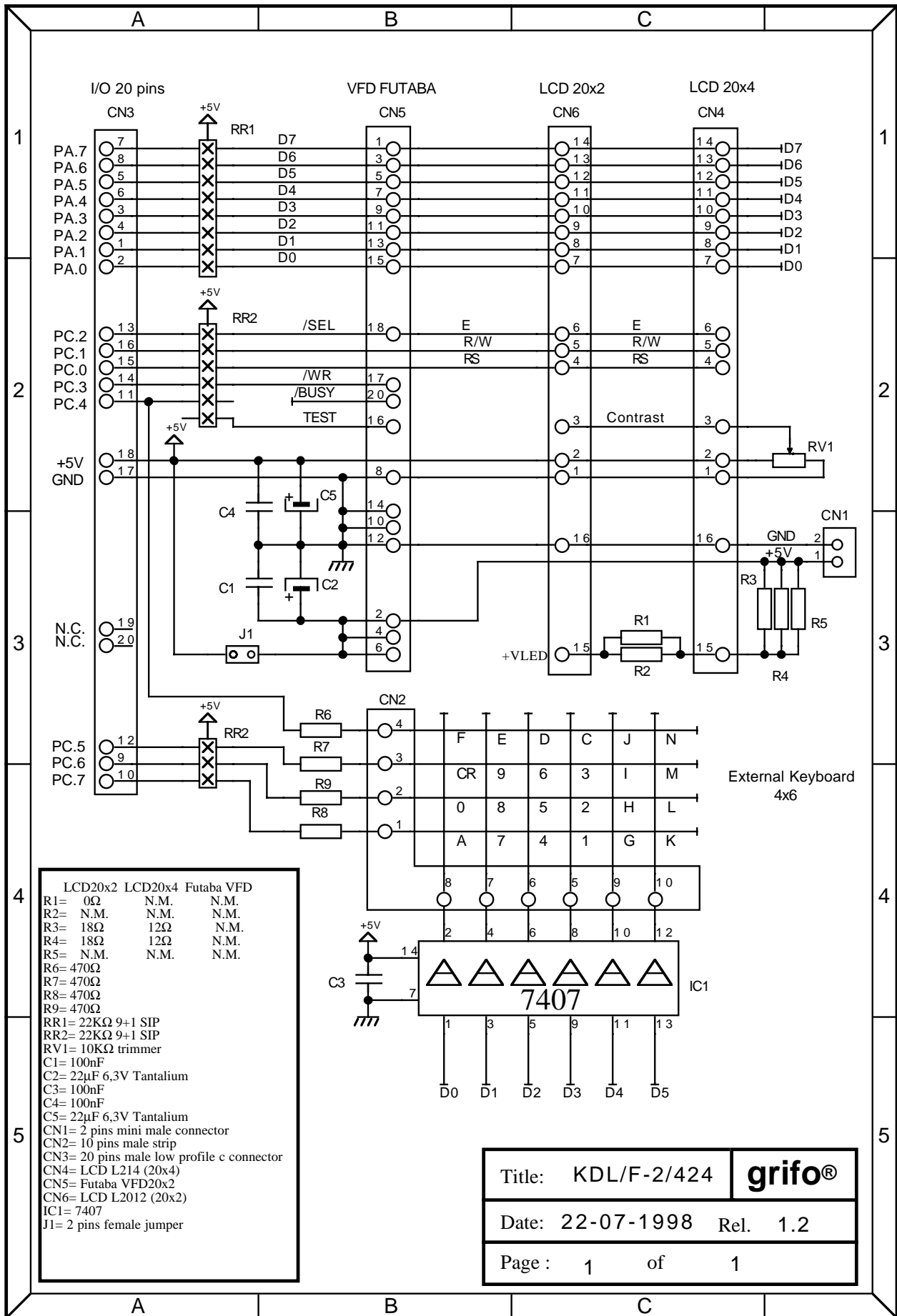
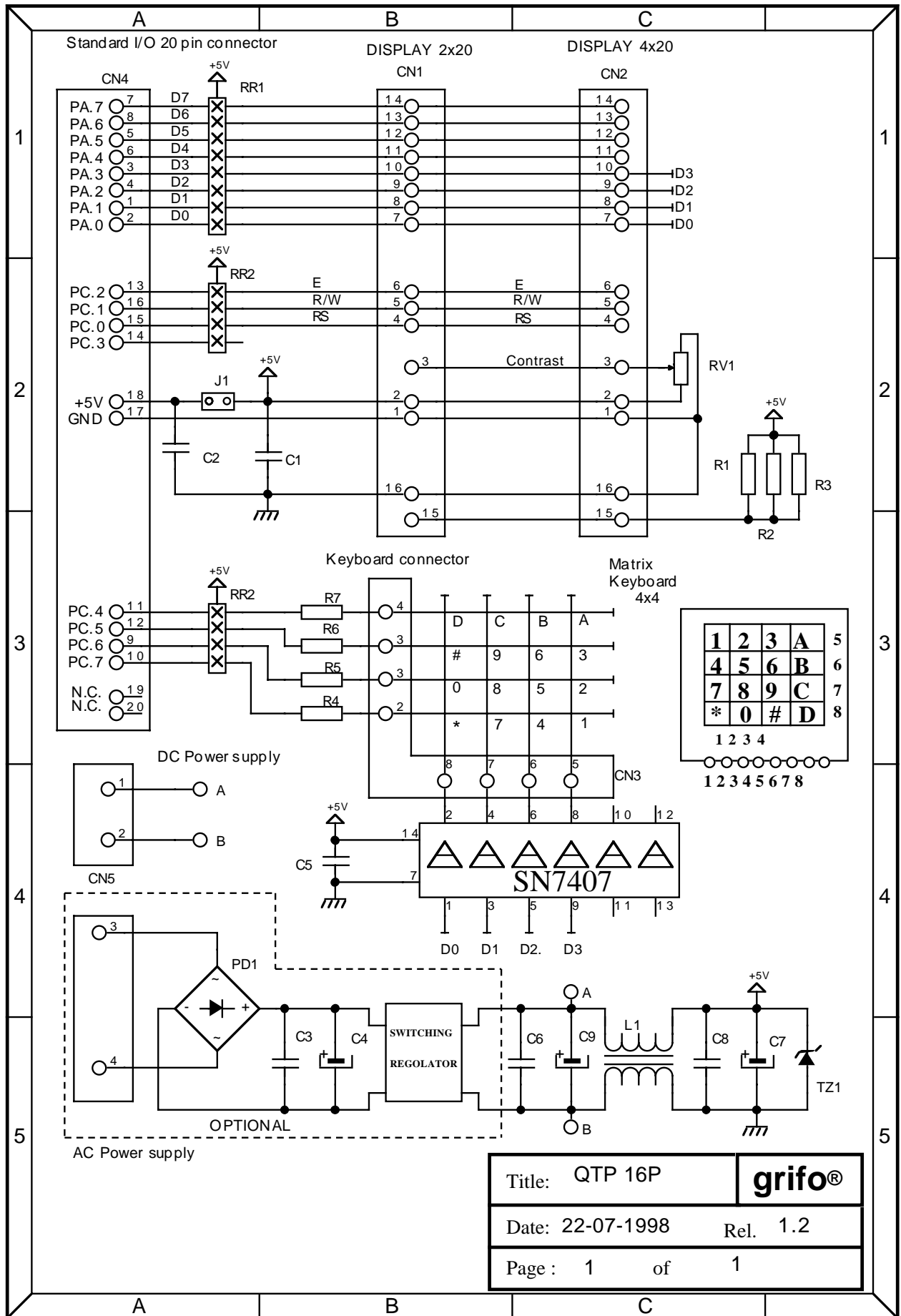


FIGURA A2: SCHEMA ELETTRICO KDX x24





| | |
|------------------|---------------|
| Title: QTP 16P | grifo® |
| Date: 22-07-1998 | Rel. 1.2 |
| Page : 1 | of 1 |

FIGURA A3: SCHEMA ELETTRICO QTP 16P



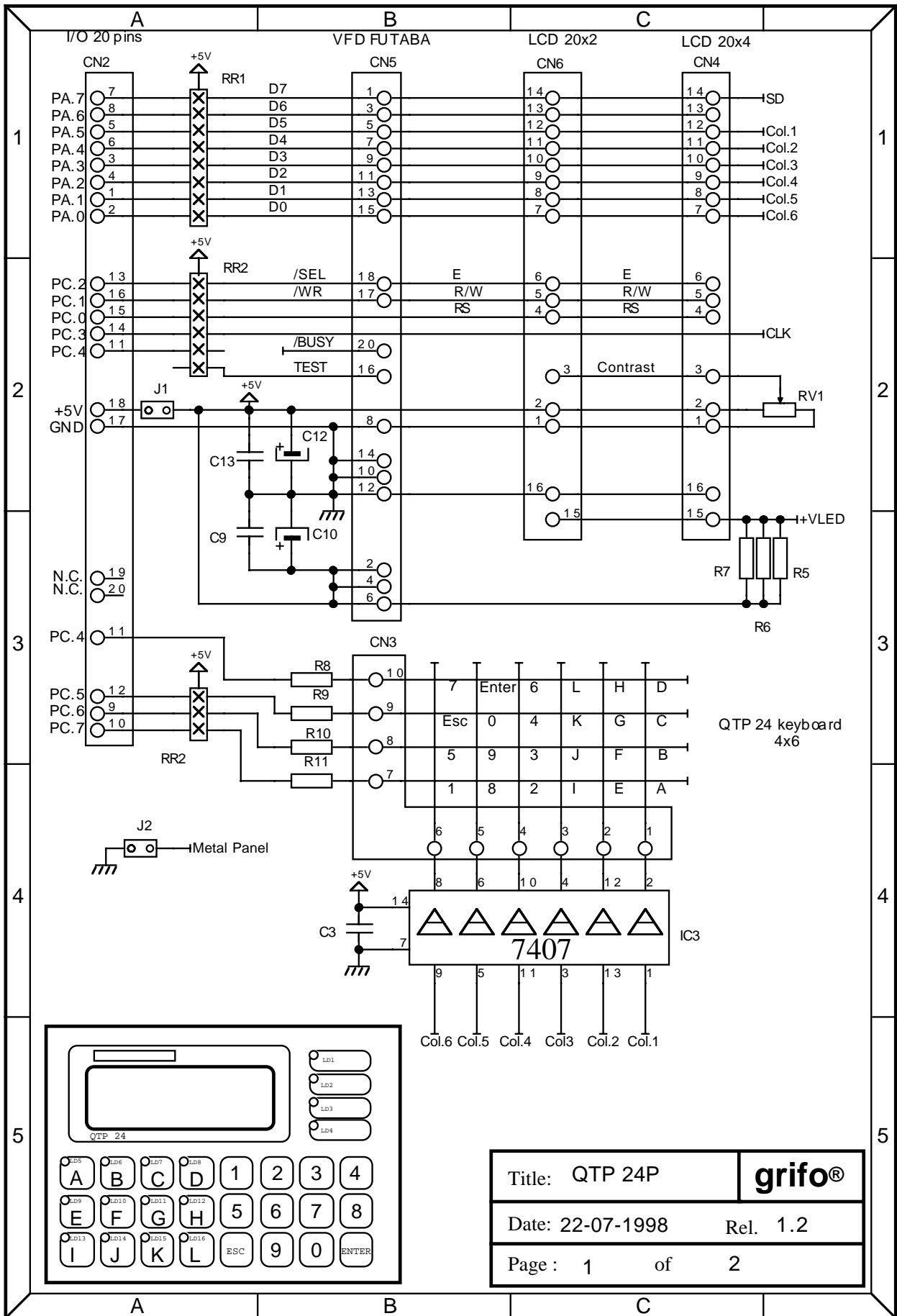
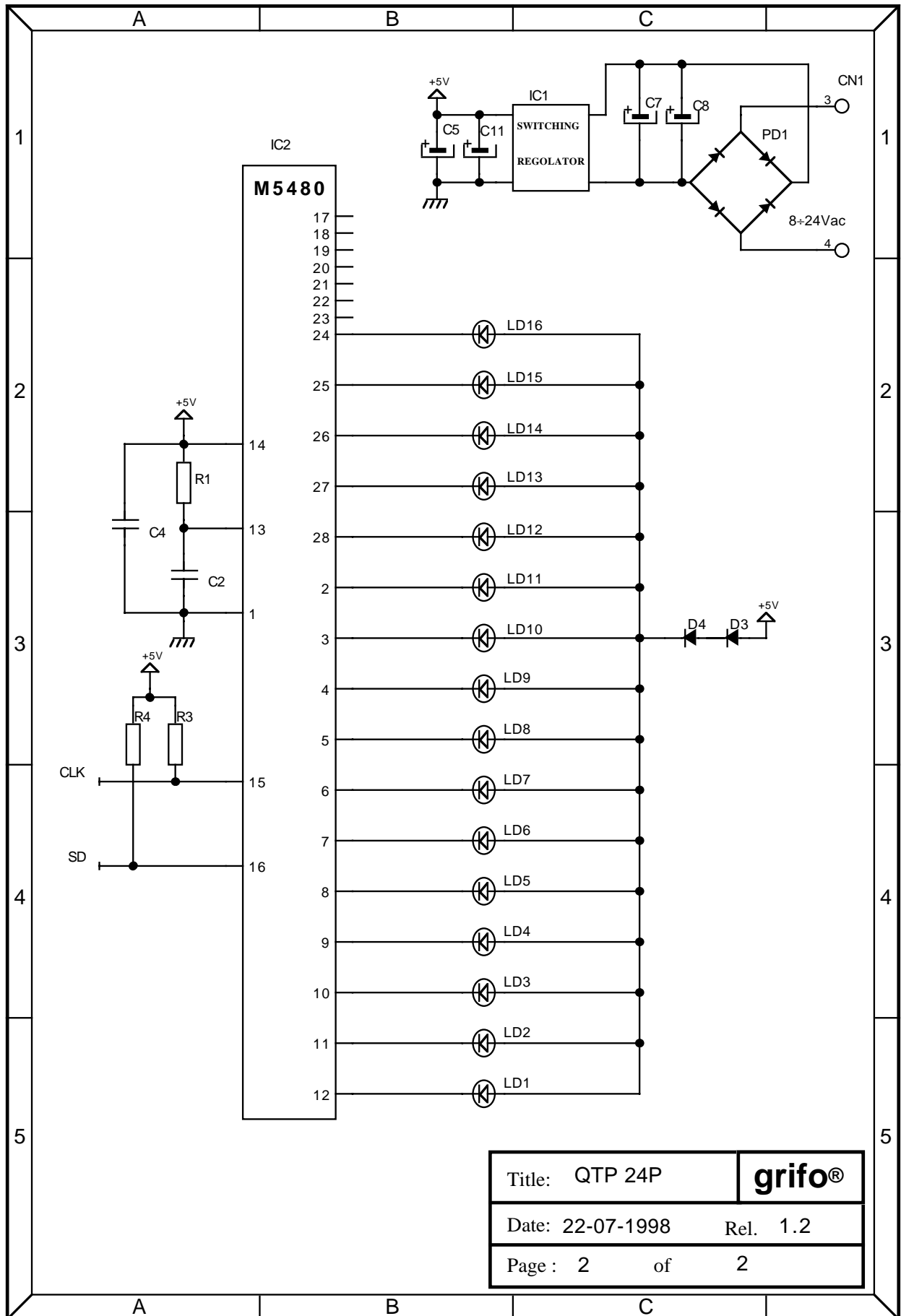


FIGURA A4: SCHEMA ELETTRICO QTP 24P - PARTE 1



| | |
|------------------|---------------|
| Title: QTP 24P | grifo® |
| Date: 22-07-1998 | Rel. 1.2 |
| Page : 1 | of 2 |



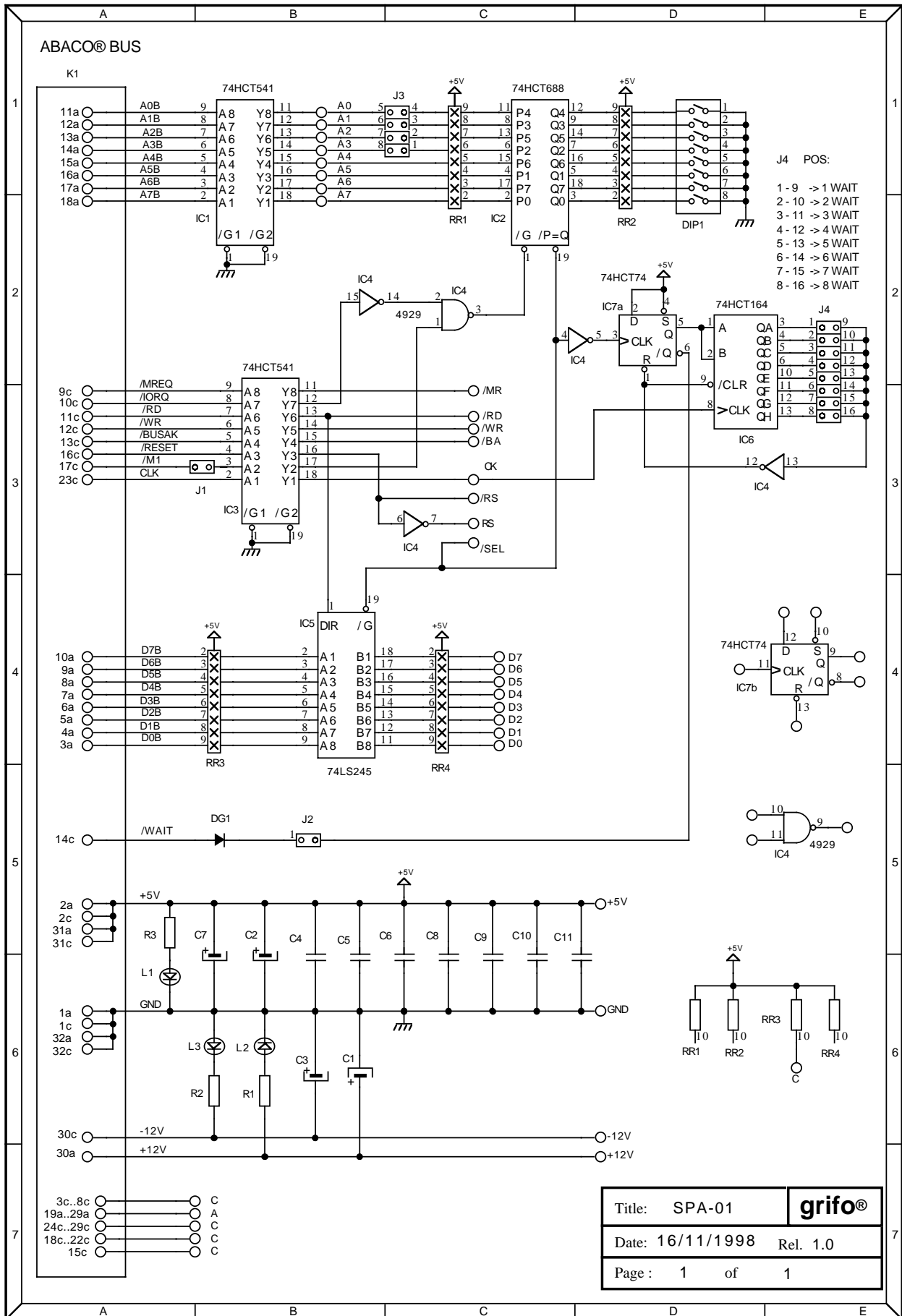


FIGURA A6: SCHEMA ELETTRICO SPA 01



APPENDICE B: DESCRIZIONE COMPONENTI DI BORDO

CPU 80C188



M80C186
CHMOS HIGH INTEGRATION 16-BIT MICROPROCESSOR

Military

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

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

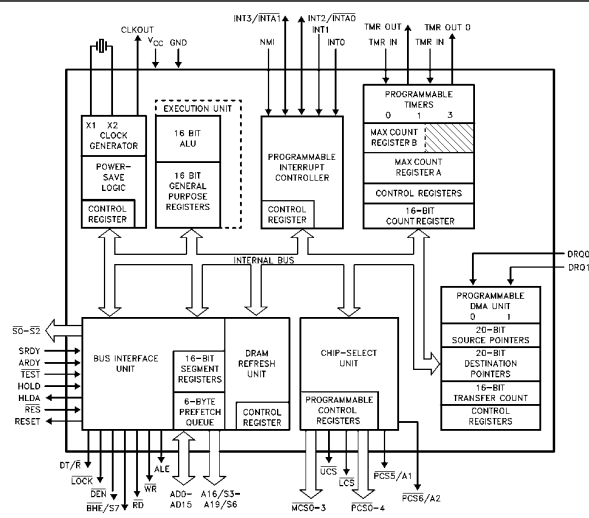


Figure 1. M80C186 Block Diagram

270500-1



FUNCTIONAL DESCRIPTION

Introduction

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

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

M80C186 BASE ARCHITECTURE

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

Register Set

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

General Registers

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

M80C186

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

Segment Registers

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

Base and Index Registers

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

Status and Control Registers

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

Status Word Description

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



M80C186

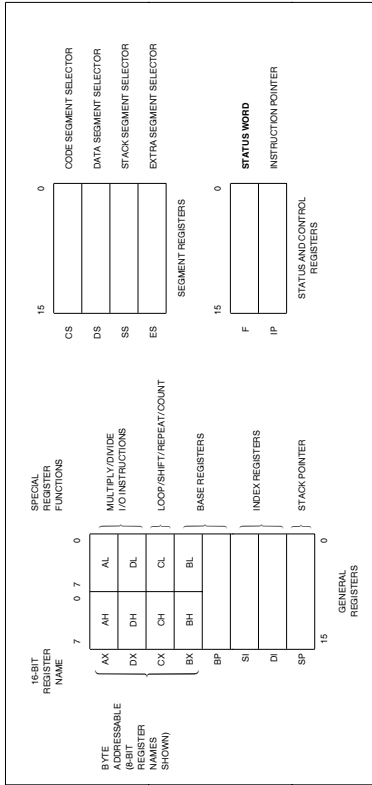


Figure 3a. M80C186 Register Set

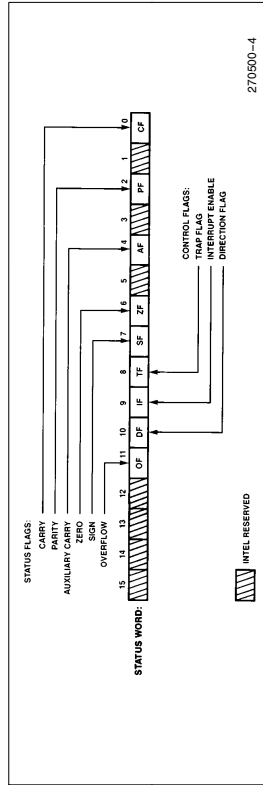


Figure 3b. Status Word Format

Table 2. Status Word Bit Functions

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

Instruction Set

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

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

Memory Organization

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

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

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

| MOVES | GENERAL PURPOSE | LOGICALS |
|-------|--|----------|
| MOV | Move byte or word | NOT |
| PUSH | Push word onto stack | AND |
| POP | Pop word off stack | OR |
| PUSHA | Push all registers on stack | XOR |
| POPA | Pop all registers from stack | TEST |
| XCHG | Exchange byte or word | SHL/SAL |
| XLAT | Translate byte | SHR |
| IN | Input byte or word | SAR |
| OUT | Output byte or word | ROL |
| LEA | Load effective address | ROR |
| LDS | Load pointer using DS | RCL |
| LES | Load pointer using ES | RCR |
| LAHF | Load AH register from flags | STC |
| SAHF | Store AH register in flags | CLC |
| PUSHF | Push flags onto stack | CMC |
| POPF | Pop flags off stack | STD |
| ADD | Add byte or word | CLD |
| ADC | Add byte or word with carry | STI |
| INC | Increment byte or word by 1 | CLI |
| AAA | ASCII adjust for addition | HLT |
| DAA | Decimal adjust for addition | WAIT |
| SUB | Subtract byte or word | ESC |
| SBB | Subtract byte or word with borrow | LOCK |
| DEC | Decrement byte or word by 1 | NOP |
| NEG | Negate byte or word | ENTER |
| CMP | Compare byte or word | LEAVE |
| AAS | ASCII adjust for subtraction | BOUND |
| DAS | Decimal adjust for subtraction | CWD |
| MUL | Multiply byte or word unsigned | |
| IMUL | Integer multiply byte or word | |
| AAM | ASCII adjust for multiply | |
| DIV | Divide byte or word unsigned | |
| IDIV | Integer divide byte or word | |
| AAD | ASCII adjust for division | |
| CBW | Convert byte to word | |
| CWD | Convert word to doubleword | |
| | ADDRESS OBJECT | |
| | Load effective address | |
| | Load pointer using DS | |
| | Load pointer using ES | |
| | FLAG TRANSFER | |
| | Load AH register from flags | |
| | Store AH register in flags | |
| | Push flags onto stack | |
| | Pop flags off stack | |
| | ADDITION | |
| | Add byte or word | |
| | Add byte or word with carry | |
| | Increment byte or word by 1 | |
| | ASCII adjust for addition | |
| | Decimal adjust for addition | |
| | SUBTRACTION | |
| | Subtract byte or word | |
| | Subtract byte or word with borrow | |
| | Decrement byte or word by 1 | |
| | Negate byte or word | |
| | Compare byte or word | |
| | ASCII adjust for subtraction | |
| | Decimal adjust for subtraction | |
| | MULTIPLICATION | |
| | Multiply byte or word unsigned | |
| | Integer multiply byte or word | |
| | ASCII adjust for multiply | |
| | DIVISION | |
| | Divide byte or word unsigned | |
| | Integer divide byte or word | |
| | ASCII adjust for division | |
| | Convert byte to word | |
| | Convert word to doubleword | |
| | NO OPERATION | |
| | No operation | |
| | HIGH LEVEL INSTRUCTIONS | |
| | Format stack for procedure entry | |
| | Restore stack for procedure exit | |
| | Detects values outside prescribed range | |
| | SHIFTS | |
| | Shift logical/arithmetic left byte or word | |
| | Shift logical right byte or word | |
| | Shift arithmetic right byte or word | |
| | ROTATES | |
| | Rotate left byte or word | |
| | Rotate right byte or word | |
| | Rotate through carry left byte or word | |
| | Rotate through carry right byte or word | |
| | FLAG OPERATIONS | |
| | Set carry flag | |
| | Clear carry flag | |
| | Complement carry flag | |
| | Set direction flag | |
| | Clear direction flag | |
| | Set interrupt enable flag | |
| | Clear interrupt enable flag | |
| | EXTERNAL SYNCHRONIZATION | |
| | Halt until interrupt or reset | |
| | Wait for TEST pin active | |
| | Escape to extension processor | |
| | Lock bus during next instruction | |

Figure 4. M80C186 Instruction Set



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

Figure 4. M80C186 Instruction Set (Continued)

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

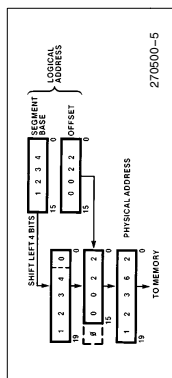
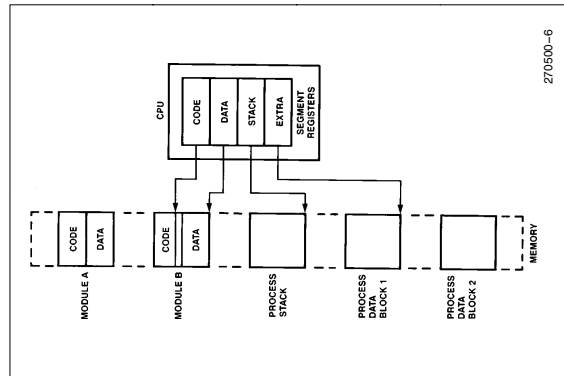


Figure 5. Two Component Address

Table 3. Segment Register Selection Rules

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

Figure 6. Segmented Memory Helps Structure Software



Addressing Modes

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

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

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

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

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

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

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

Data Types

The M80C186 directly supports the following data types:

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

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

I/O Space

The I/O space consists of 64K 8-bit or 32K 16-bit ports. Separate instructions address the I/O space with either an 8-bit port address, specified in the instruction, or a 16-bit port address in the DX register. 8-bit port addresses are zero extended such that A15-A8 are LOW. I/O port addresses 00F8(H) through 00FF(H) are reserved.

Interrupts

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



M80C186

Table 4. M80C186 Interrupt Vectors

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

ARRAY BOUNDS EXCEPTION (TYPE 5)

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

UNUSED OP CODE EXCEPTION (TYPE 6)

Generated if execution is attempted on undefined opcodes.

ESCAPE OP CODE EXCEPTION (TYPE 7)

Generated if execution is attempted of ESC opcodes (D8H–DFH). In compatible mode operation, ESC opcodes will always generate this exception. In enhanced mode operation, the exception will be generated only if a bit in the relocation register is set. The return address of this exception will point to the ESC instruction causing the exception. If a segment override prefix preceded the ESC instruction, the return address will point to the segment override prefix.

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

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

Non-Maskable Interrupt Request (NMI)

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

M80C186

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

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

Interrupt Sources

The M80C186 can service interrupts generated by software or hardware. The software interrupts are generated by specific instructions (INT, ESC, unused OP, etc.) or the results of conditions specified by instructions (array bounds check, INT0, DIV, IDIV, etc.). All interrupt sources are serviced by an indirect call through an element of a vector table. This vector table is indexed by using the interrupt vector type (Table 4), multiplied by four. All hardware-generated interrupts are sampled at the end of each instruction. Thus, the software interrupts will begin service first. Once the service routine is entered and interrupts are enabled, any hardware source of sufficient priority can interrupt the service routine in progress.

The software generated M80C186 interrupts are described below.

DIVIDE ERROR EXCEPTION (TYPE 0)

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

SINGLE-STEP INTERRUPT (TYPE 1)

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

NON-MASKABLE INTERRUPT—NMI (TYPE 2)

An external interrupt source which cannot be masked.

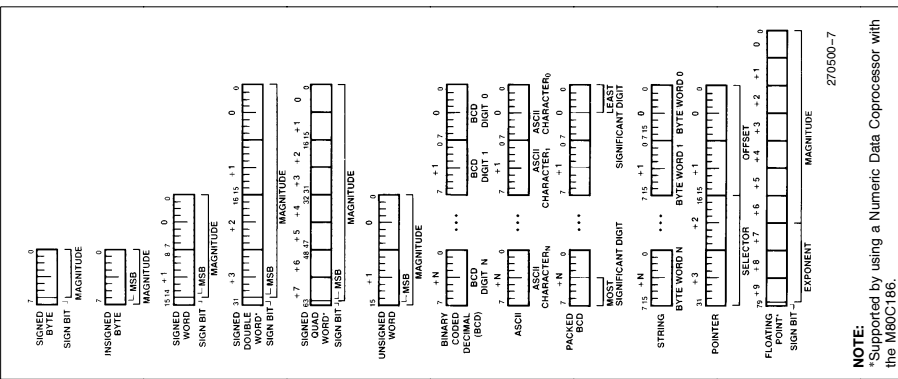


Figure 7. M80C186 Supported Data Types

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



Single-Step Interrupt

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

Initialization and Processor Reset

Processor initialization or startup is accomplished by driving the RES input pin. LOW RES forces the M80C186 to terminate all execution and local bus activity. No instruction or bus activity will occur as long as RES is active. After RES becomes inactive and an internal processing interval elapses, the M80C186 begins execution with the instruction at physical location FFFF0(H). RES also sets some registers to predefined values as shown in Table 5.

Table 5. M80C186 Initial Register State after RESET

| | |
|-----------------------|---------|
| Status Word | F002(H) |
| Instruction Pointer | 0000(H) |
| Code Segment | FFFF(H) |
| Data Segment | 0000(H) |
| Extra Segment | 0000(H) |
| Stack Segment | 0000(H) |
| Reallocation Register | 20FF(H) |
| UMCS | FFFF(H) |

M80C 186 CLOCK GENERATOR

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

Oscillator

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

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

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

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

The following parameters may be used for choosing a crystal:

- Temperature Range: -55°C to +125°C
- ESR (Equivalent Series Resistance): 40Ω max
- C₀ (Shunt Capacitance of Crystal): 7.0 pF max
- C₁ (Load Capacitance): 20 pF ± 2 pF
- Drive Level: 1 mW max

Clock Generator

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

READY Synchronization

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

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

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

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

RESET Logic

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

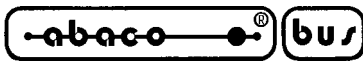
Multiple M80C186 processors may be synchronized through the RES input pin, since this input resets both the processor and divide-by-two internal counter in the clock generator. In order to insure that the divide-by-two counters 'all begin counting at the same time, the active going edge of RES must satisfy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

LOCAL BUS CONTROLLER

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

Memory/Peripheral Control

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



Transceiver Control

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

Table 6. Transceiver Control Signals Description

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

Local Bus Arbitration

The M80C186 uses a HOLD/HLDA system of local bus exchange. This provides an asynchronous bus exchange mechanism. This means multiple masters utilizing the same bus can operate at separate clock frequencies. The M80C186 provides a single HOLD/HLDA pair through which all other bus masters may gain control of the local bus. This requires external circuitry to arbitrate which external device will gain control of the bus from the M80C186 when there is more than one alternate local bus master. When the M80C186 relinquishes control of the local bus, it floats DEN, RD, WR, S0-S2, LOOK, AD0-AD15, A16-A19, BHE, and DT/R to allow another master to drive these lines directly.

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

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

Local Bus Controller and Reset

Upon receipt of a RESET pulse from the RES input, the local bus controller will perform the following actions:

- Drive DEN, RD, and WR HIGH for one clock cycle, then float

NOTE:

RD is also provided with an internal pull-up device to prevent the processor from inadvertently entering Queue Status mode during reset.

- Drive S0-S2 to the passive state (all HIGH) and then float.
- Drive LOCK HIGH and then float.
- Float A0-15, A16-19, BHE, DT/R.
- Drive ALE LOW (ALE is never floated).
- Drive HILDA LOW.

INTERNAL PERIPHERAL INTERFACE

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

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

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

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

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

CHIP-SELECT/READY GENERATION LOGIC

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

Memory Chip Selects

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

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

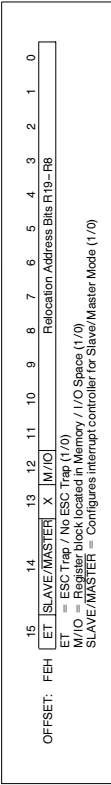


Figure 9. Relocation Register

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

Table 7. UMCS Programming Values

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

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

Any internally generated 20-bit address whose upper 16 bits are greater than or equal to UMCS (with bits 0-5 "0") will cause UCS to be activated. UMCS bits R2-R0 are used to specify READY mode for the area of memory defined by this chip-select register, as explained below.

Lower Memory CS

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

Upper Memory CS

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

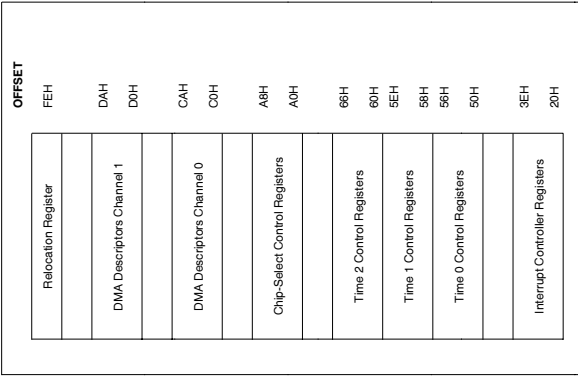


Figure 10. Internal Register Map

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



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

Table 8. LMCS Programming Values

| Upper Address | Memory Block Size | LMCS Value (Assuming R0 = R1 = R2 = 0) |
|---------------|-------------------|--|
| 003FFH | 1K | 0038H |
| 007FFH | 2K | 0078H |
| 00FFFH | 4K | 00F8H |
| 01FFFH | 8K | 01F8H |
| 03FFFH | 16K | 03F8H |
| 07FFFH | 32K | 07F8H |
| 0FFFFH | 64K | 0FF8H |
| 1FFFFH | 128K | 1FF8H |
| 3FFFFH | 256K | 3FF8H |

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

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

Mid-Range Memory CS

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

dress and size of this memory block are programmable.

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

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

Table 9. MPCS Programming Values

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

The base address of the mid-range memory block is defined by bits 15-9 of the MMCS register (see Figure 14). This register is at offset A6H in the internal control block. These bits correspond to bits A19-A13 of the 20-bit memory address. Bits A12-A0 of the base address are always 0. The base address may be set at any integer multiple of the size of the total memory block selected. For example, if the mid-range block size is 32K (for the size of the block for which each MCS line is active is 8K), the block could be located at 10000H or 18000H, but not at 14000H, since the first few integer multiples of a 32K memory block are 0H, 8000H, 10000H, 18000H, etc. After reset, the contents of both of these registers is undefined. However, none of the MCS lines will be active until both the MMCS and MPCS registers are accessed.



Figure 11. UMCS Register



Figure 12. LMCS Register

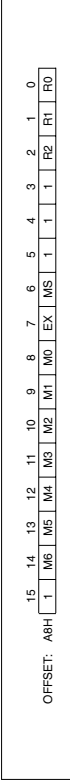


Figure 13. MPCS Register

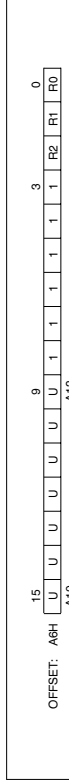


Figure 14. MMCS Register

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

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

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

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

Peripheral Chip Selects

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

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



Figure 15. PACS Register



DMA CHANNELS

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

DMA Operation

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

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

Table 10. PCS Address Ranges

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

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

Table 11. MS, EX Programming Values

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

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

READY Generation Logic

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

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

Table 12. READY Bits Programming

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

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

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

Chip Select/Ready Logic and Reset

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

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



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

Table 13. DMA Control Block Format

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

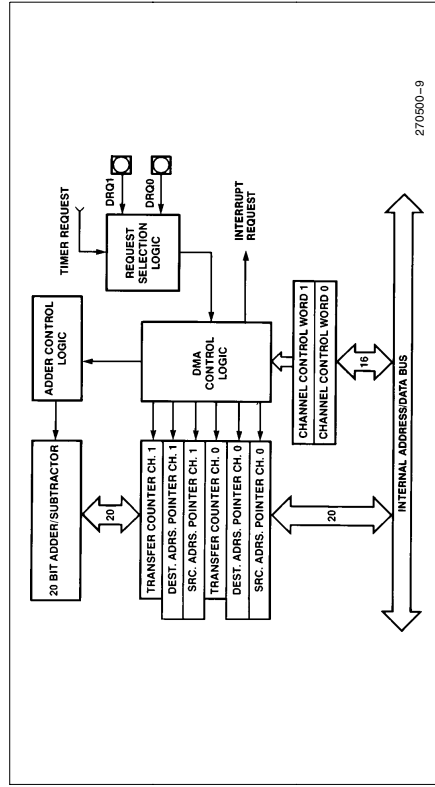


Figure 16. DMA Unit Block Diagram

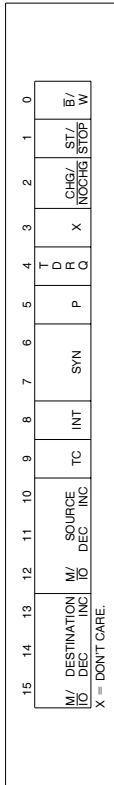


Figure 17. DMA Control Register

DMA Channel Control Word Register

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

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

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

DMA Control Word Bit Descriptions

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

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

DMA Destination and Source Pointer Registers

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

DMA Transfer Count Register

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

DMA Requests

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

Table 14. Maximum DMA Transfer Rates

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

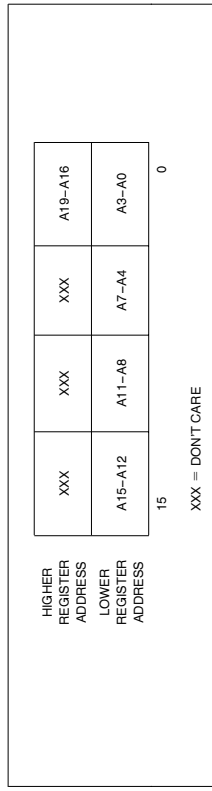


Figure 18. DMA Memory Pointer Register Format



DMA Acknowledge

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

DMA Priority

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

DMA Programming

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



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

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

DMA Channels and Reset

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

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

TIMERS

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

Timer Operation

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

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

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

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

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

Timer Mode/Control Register

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

Table 15. Timer Control Block Format

| Register Name | Register Offset | |
|-------------------|-----------------|--------|
| | Tmr. 0 | Tmr. 1 |
| Mode/Control Word | 56H | 5EH |
| Max Count B | 54H | 5CH |
| Max Count A | 52H | 5AH |
| Count Register | 50H | 58H |
| | | 60H |

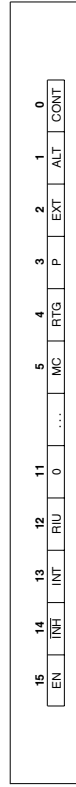


Figure 20. Timer Mode/Control Register

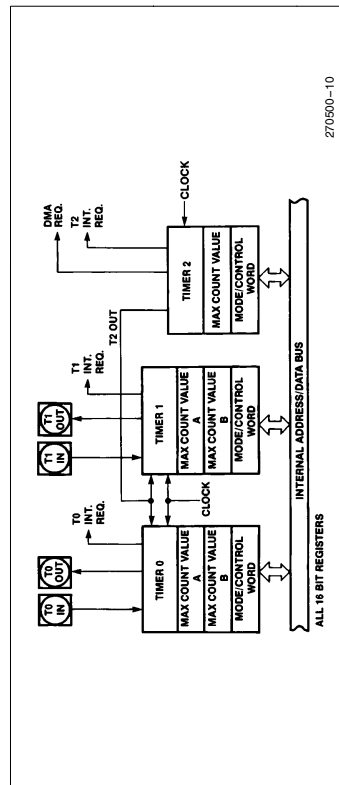


Figure 19. Timer Block Diagram

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

Programmer intervention is required to clear this bit.

RIU:

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

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

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

Count Registers

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

Max Count Registers

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

Timers and Reset

Upon RESET, the Timers will perform the following actions:

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

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

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

EN:

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

INH:

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

INT:

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

MC:

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

ALT:

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

CONT:

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

EXT:

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

P:

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

RTG:

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

INTERRUPT CONTROLLER

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

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

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

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

MASTER MODE OPERATION

Interrupt Controller External Interface

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

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

Interrupt Controller Modes of Operation

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

Fully Nested Mode

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

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

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

Cascade Mode

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

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

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

Master Mode Features

Programmable Priority

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

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

End-of-Interrupt Command

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

Trigger Mode

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

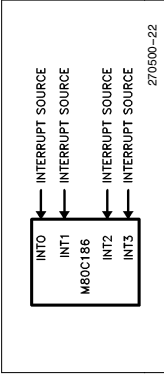


Figure 22. Fully Nested (Direct) Mode Interrupt Controller Connections

Special Fully Nested Mode

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

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

Operation in a Polled Environment

The controller may be used in a polled mode if interrupts are undesirable. When polling, the processor disables interrupts and then polls the interrupt controller whenever it is convenient. Polling the interrupt controller is accomplished by reading the Poll Word (Figure 32). Bit 15 in the poll word indicates to the processor that an interrupt of high enough priority is requesting service. Bits 0-4 indicate to the processor the type vector of the highest-priority source requesting service. Reading the Poll Word causes the In-Service bit of the highest priority source to be set. It is desirable to be able to read the Poll Word information without guaranteeing service of any pending

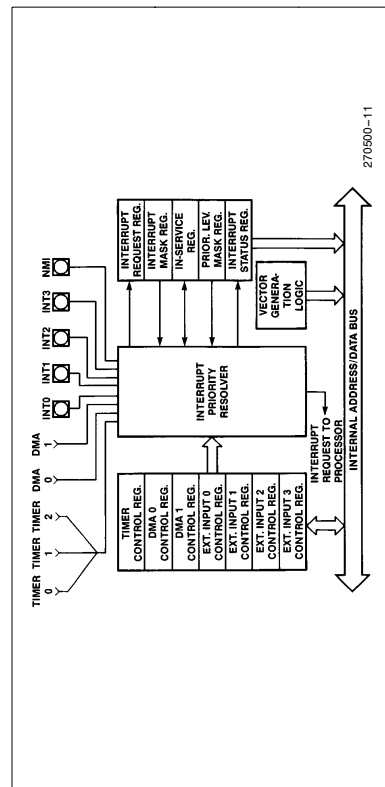


Figure 21. Interrupt Controller Block Diagram

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

Interrupt Vectoring

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

Interrupt Controller Registers

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

In-Service Register

This register can be read from or written into. The format is shown in Figure 25. It contains the In-Service bit for each of the interrupt sources. The In-Service bit is set to indicate that a source's service routine is in progress. When an In-Service bit is set, the CPU when it receives interrupt requests from devices with a lower programmed priority level. The TMR bit is the In-Service bit for all three timers; the D0 and D1 bits are the In-Service bits for the two DMA channels; the I0-I3 are the In-Service bits for the external interrupt pins. The IS bit is set when the

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

Interrupt Request Register

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

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

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

Mask Register

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

Priority Mask Register

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

Interrupt Status Register

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

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

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

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

Figure 24. Interrupt Controller Registers (Master Mode)

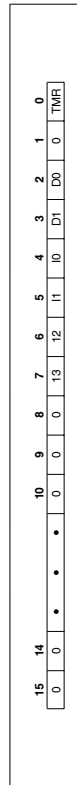


Figure 25. In-Service, Interrupt Request, and Mask Register Formats



Figure 26. Priority Mask Register Format

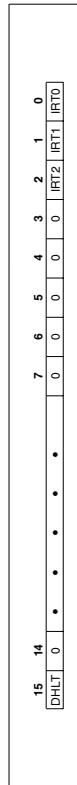


Figure 27. Interrupt Status Register Format (Master Mode)

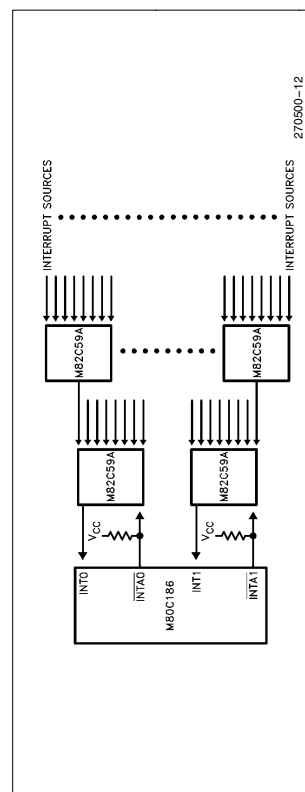


Figure 23. Cascade and Special Fully Nested Mode Interrupt Controller Connections



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

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

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

Slave Mode External Interface

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

Poll and Poll Status Registers

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

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

S_x: Encoded information that indicates the vector type of the highest priority interrupting source. Valid only when INTREQ = 1.

INTREQ: This bit determines if an interrupt request is present. Interrupt Request = 1; no Interrupt Request = 0.

SLAVE MODE OPERATION

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

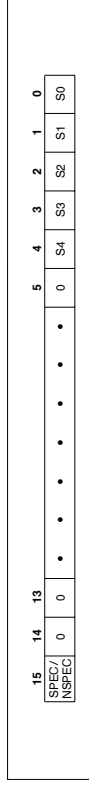


Figure 31. EOJ Register Format

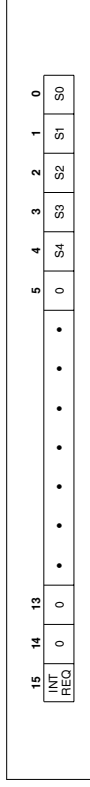


Figure 32. Poll and Poll Status Register Format

Timer, DMA 0, 1; Control Register

These registers are the control words for all the internal interrupt sources. The format for these registers is shown in Figure 28. The three bit positions PR0, PR1, and PR2 represent the programmable priority level of the interrupt source. The MSK bit inhibits interrupt requests from the interrupt source. The MSK bits in the individual control registers are the exact same bits as are in the Mask Register; modifying them in the individual control registers will also modify them in the Mask Register, and vice versa.

INT0-INT3 Control Registers

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

The bits in the various control registers are encoded as follows:

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

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

Mask bit, 1 = mask; 0 = non-mask.

Cascade mode bit, 1 = cascade; 0 = direct

SFNM: Special fully nested mode bit, 1 = SFNM

EOJ Register

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

The bits in the EOJ register are encoded as follows:

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

NOTE:

To reset the single In-Service bit for any of the three timers, the vector type for timer 0 (8) should be written in this register.

NSPEC/: A bit that determines the type of EOJ command. Nonspecific = 1, Specific = 0.

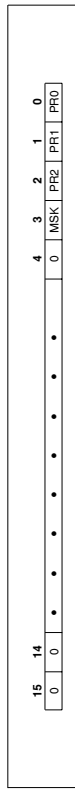


Figure 28. Timer/DMA Control Registers Formats

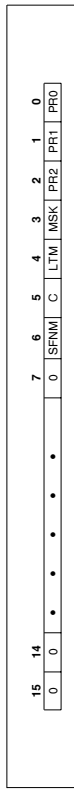


Figure 29. INTO/INT1 Control Register Formats



Figure 30. INT2/INT3 Control Register Formats





M80C186

Interrupt Vector Register

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

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

Priority-Level Mask Register

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

The encoding of the bits in this register is:
 mx: 3-bit encoded field indicating priority-level value. All levels of lower priority will be masked.

Interrupt Status Register

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

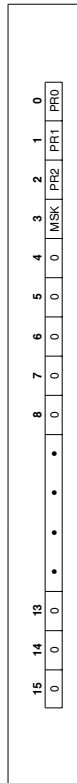


Figure 37. Control Word Format

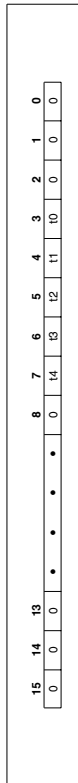


Figure 38. Interrupt Vector Register Format

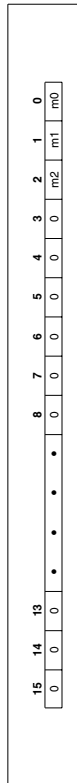


Figure 39. Priority Level Mask Register



Interrupt Controller and Reset

Upon RESET, the interrupt controller will perform the following actions:

- All SFNM bits reset to 0, implying Fully Nested Mode.
- All PR bits in the various control registers set to 1, 111).
- This places all sources at lowest priority (level 111).
- All LTM bits reset to 0, resulting in edge-sense mode.
- All Interrupt Service bits reset to 0.
- All Interrupt Request bits reset to 0.
- All MSK (Interrupt Mask) bits set to 1 (mask).
- All C (Cascade) bits reset to 0 (non-cascade).
- All PRM (Priority Mask) bits set to 1, implying no levels masked.
- Initialized to master mode.

M80C186

Enhanced Mode Operation

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

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

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

Entering Enhanced Mode

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

Queue-Status Mode

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

M80C186

DRAM Refresh Control Unit Description

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

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

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

DRAM Refresh Addresses

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

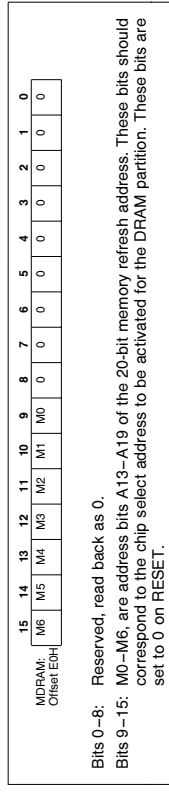


Figure 40. Memory Partition Register

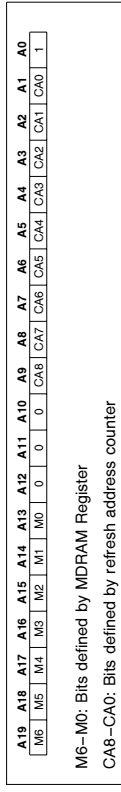


Figure 41. Addresses Generated by RCU



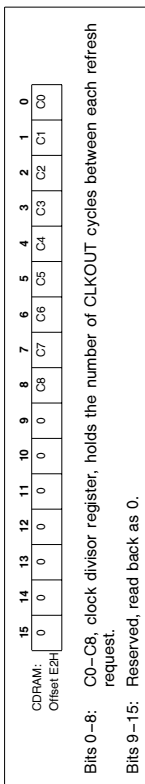


Figure 42. Clock Pre-Scaler Register

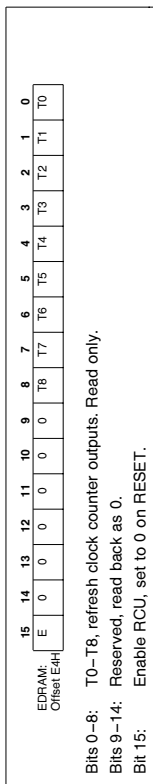


Figure 43. Enable RCU Register

Refresh Control Unit Programming and Operation

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

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

POWER-SAVE CONTROL

Power Save Operation

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

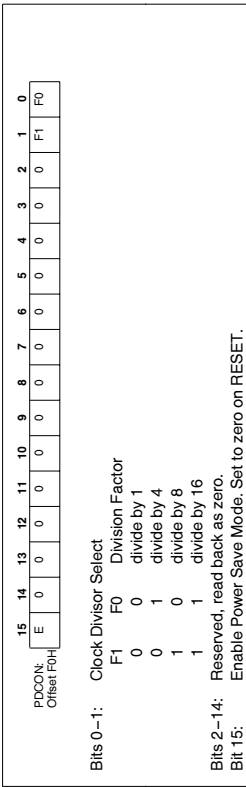


Figure 44. Power-Save Control Register

Numeric Coprocessor (NPX) Extension

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

Table 16. MCS Assignments

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

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

Table 17. Numerics Coprocessor I/O Port Assignments

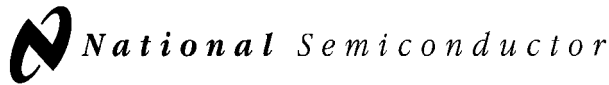
| I/O Address | Read Definition | Write Definition |
|-------------|---------------------|----------------------|
| 00F8H | Status/Control Data | Opcode Data |
| 00FAH | reserved | CS:IP_DS:EA reserved |
| 00FCH | Opcode Status | |
| 00FEH | | |

"ONCE" Test Mode

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

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



A/D CONVERTER LM12H458

July 1999

LM12454/LM12458/LM12H458 12-Bit + Sign Data Acquisition System with Self-Calibration

General Description

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

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

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

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

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

Key Specifications

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

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

Features

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

Applications

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

**LM12454/LM12458/LM12H458
12-Bit + Sign Data Acquisition System with Self-Calibration**

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



Application Information

1.0 Functional Description

The LM12454 and LM12(H)458 are multi-functional Data Acquisition Systems that include a fully differential 12-bit-plus-sign self-calibrating analog-to-digital converter (ADC) with a two's-complement output format, an 8-channel (LM12(H)458) or a 4-channel (LM12454) analog multiplexer, an internal 2.5V reference, a first-in-first-out (FIFO) register that can store 32 conversion results, and an instruction RAM that can store as many as eight instructions to be sequentially executed. The LM12454 also has a differential multiplexer output and a differential S/H input. All of this circuitry operates on only a single +5V power supply.

The LM12(H)454/8 have three modes of operation:
 12-bit + sign with correction
 8-bit + sign without correction
 8-bit + sign comparison mode ("watchdog" mode)

The fully differential 12-bit-plus-sign ADC uses a charge redistribution topology that includes calibration capabilities. Charge re-distribution ADCs use a capacitor ladder in place of a resistor ladder to form an internal DAC. The DAC is used by a successive approximation register to generate intermediate voltages between the voltages applied to V_{REF+} and V_{REF-} . These intermediate voltages are compared against the sampled analog input voltage as each bit is generated. The number of intermediate voltages and comparisons equals the ADC's resolution. The correction of each bit's accuracy is accomplished by calibrating the capacitor ladder used in the ADC.

Two different calibration modes are available; one compensates for offset voltage, or zero error, while the other corrects both offset error and the ADC's linearity error. When correcting offset only, the offset error is measured once and a correction coefficient is created. During the full calibration, the offset error is measured eight times, averaged, and a correction coefficient is created. After completion of either calibration mode, the offset correction coefficient is stored in an internal offset correction register.

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

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

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

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

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

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

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

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

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

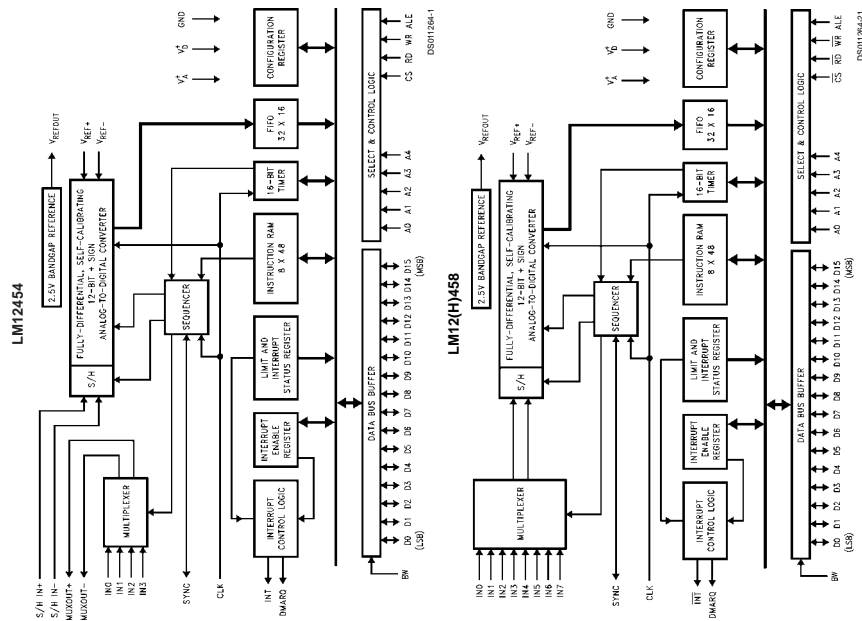
A diagnostic mode is available that allows verification of the LM12(H)458's operation. The diagnostic mode is disabled in the LM12454. This mode internally connects the voltages present at the V_{REFOUT} , V_{REF+} , V_{REF-} , and GND pins to the internal V_{IN+} and V_{IN-} S/H inputs. This mode is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

2.0 Internal User-Programmable Registers

INSTRUCTION RAM

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

Functional Diagrams



2.0 Internal User-Programmable Registers (Continued)

Bits 12-15 are used to store the user-programmable acquisition mode for a fixed number of clock cycles (nine clock cycles, for 12-bit + sign conversions and two clock cycles for 8-bit + sign conversions or "watchdog" comparisons) plus a variable number of clock cycles equal to twice the value stored in Bits 12-15. Thus, the SH's acquisition time is (9 + 2D) clock cycles for 12-bit + sign conversions and (2 + 2D) clock cycles for 8-bit + sign conversions or "watchdog" comparisons, where D is the value stored in Bits 12-15. The minimum acquisition time compensates for the typical internal multiplexer series resistance of 2 kΩ, and any additional delay created by Bits 12-15 compensates for source resistances greater than 600 Ω (1000). (For this acquisition time discussion, numbers in parentheses shown for the LM12(H)454/8 operating at 5 MHz.) The necessary acquisition time is determined by the source impedance at the multiplexer input. If the source resistance (R_S) < 600 Ω (1000) and the clock frequency is 8 MHz, the value stored in bits 12-15 (D) can be 0000. If R_S > 600 Ω (1000), the following equations determine the value that should be stored in bits 12-15.

$$D = 0.45 \times R_S \times f_{CLK}$$

$$D = 0.36 \times R_S \times f_{CLK}$$

for 8-bits + sign and "watchdog"
 for 8-bits + sign and "watchdog"

R_S is kΩ and f_{CLK} is in MHz. Round the result to the next higher integer value. If D is greater than 15, it is advisable to lower the signal impedance by using an analog buffer between the signal source and the LM12(H)454/8 multiplexer inputs. The value of D can also be used to compensate for the settling or response time of external processing circuits connected between the LM12454's MUXOUT and SH IN pins.

2.0 Internal User-Programmable Registers (Continued)

| A4 | A3 | A2 | A1 | A0 | Purpose | Type | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
|----|----|----|----|----|------------------------------|------|---|------------------------|-------------------------|-----------------------|-----------------------|---|-------|-------|
| 0 | 0 | 0 | 0 | 0 | Instruction RAM Pointer = 00 | R/W | (MUXOUT-) | V _{INT} | (MUXOUT-) | (Note 22) | Watchdog | 8/12 | Pause | Loop |
| 0 | 0 | 0 | 0 | 0 | | | | | | | | | | |
| 0 | 0 | 0 | 0 | 1 | Instruction RAM Pointer = 01 | R/W | Acquisition Time | Comparison Limit #1 | Don't Care | Comparison Limit #2 | Sign | >/< | Sign | |
| 0 | 0 | 0 | 0 | 1 | | | | | | | | | | |
| 0 | 0 | 0 | 0 | 1 | Instruction RAM Pointer = 10 | R/W | Don't Care | Don't Care | Don't Care | Don't Care | Sign | >/< | Sign | |
| 0 | 0 | 0 | 0 | 1 | | | | | | | | | | |
| 1 | 0 | 0 | 0 | 0 | Configuration Register | R/W | IO Sel | Auto Zero | Chan Mask | Standby | Full Cal | Auto-Zero Test = 0 | Reset | Start |
| 1 | 0 | 0 | 0 | 1 | | | | | | | | | | |
| 1 | 0 | 0 | 1 | 0 | Interrupt Enable Register | R/W | INT7 | INT6 | INT5 | INT4 | INT3 | INT2 | INT1 | INT0 |
| 1 | 0 | 0 | 1 | 1 | | | | | | | | | | |
| 1 | 0 | 0 | 1 | 1 | Interrupt Status Register | R | INST7 | INST6 | INST5 | INST4 | INST3 | INST2 | INST1 | INST0 |
| 1 | 0 | 1 | 0 | 0 | | | | | | | | | | |
| 1 | 0 | 1 | 0 | 1 | Timer Register | R/W | Actual Number of Conversions in Conversion FIFO | Timer Preset: Low Byte | Timer Preset: High Byte | Conversion Data: LSBs | Conversion Data: MSBs | Address of Sequencer Instruction being Executed | | |
| 1 | 0 | 1 | 1 | 0 | | | | | | | | | | |
| 1 | 0 | 1 | 0 | 0 | Conversion FIFO | R | Address or Sign | Limit #1 Status | Limit #2 Status | | | | | |
| 1 | 1 | 0 | 0 | 1 | | | | | | | | | | |
| 1 | 1 | 0 | 1 | 1 | Limit Status Register | R | | | | | | | | |

FIGURE 14. LM12(H)454/8 Memory Map for 8-Bit Wide Databus (BW = "1" and Test Bit = "0")

Note 22: LM12454 (Refer to Table 2).
 Note 23: LM12(H)458 only. Must be set to "0" for the LM12454.

Bit 9's state determines the limit condition that generates a "watchdog" interrupt. A "1" causes a voltage greater than limit #2 to generate an interrupt, while a "0" causes a voltage less than limit #2 to generate an interrupt.
 Bits 10-15 are not used.

2.2 CONFIGURATION REGISTER

The Configuration register, 1000 (A4-A1, BW = 0) or 1000x (A4-A0, BW = 1) is a 16-bit control register with read/write capability. It acts as the LM12454's and LM12(H)458's "control panel" holding global information as well as start/stop, reset, self-calibration, and stand-by commands.

Bit 0 is the START/STOP bit. Reading Bit 0 returns an indication of the Sequencer's status. A "0" indicates that the Sequencer is stopped and waiting to execute the next instruction. A "1" shows that the Sequencer is running. Writing a "0" halts the Sequencer when the current instruction has finished execution. The next instruction to be executed is pointed to by the instruction pointer found in the status register. A "1" restarts the Sequencer with the instruction currently pointed to by the instruction pointer. (See Bits 8-10 in the Interrupt Status register.)

Bit 1 is the LM12(H)454/8's system RESET bit. Writing a "1" to Bit 1 stops the Sequencer (resetting the Configuration register's START/STOP bit), resets the instruction pointer to "0000" (found in the Interrupt Status register), clears the Conversion FIFO, and resets all interrupt flags. The RESET bit will return to "0" after five clock cycles unless it is forced high by writing a "1" into the Configuration register's Standby bit. A reset signal is internally generated when power is first applied to the part. No operation should be started until the RESET bit is "0".

Writing a "1" to Bit 2 initiates an auto-zero offset voltage calibration. Unlike the eight-sample auto-zero calibration performed during the full calibration procedure, Bit 2 initiates a "short" auto-zero by sampling the offset once and creating a correction coefficient (full calibration averages eight samples of the converter offset voltage when creating a correction coefficient). If the Sequencer is running when Bit 2 is set to "1", an auto-zero starts immediately after the conclusion of the currently running instruction. Bit 2 is reset automatically to a "0" and an interrupt flag (Bit 3, in the Interrupt Status register) is set at the end of the auto-zero (76 clock cycles). After completion of an auto-zero calibration, the Sequencer fetches the next instruction as pointed to by the Instruction RAM's pointer and resumes execution. If the Sequencer is stopped, an auto-zero is performed immediately at the time requested.

Writing a "1" to Bit 3 initiates a complete calibration process that includes a "long" auto-zero offset voltage correction (this calibration averages eight samples of the comparator offset voltage when creating a correction coefficient) followed by an ADC linearity calibration. This complete calibration is started after the currently running instruction is completed if the Sequencer is running when Bit 3 is set to "1". Bit 3 is reset automatically to a "0" and an interrupt flag (Bit 4, in the Interrupt Status register) will be generated at the end of the calibration procedure (4944 clock cycles). After completion of a full auto-zero and linearity calibration, the Sequencer fetches the next instruction as pointed to by the Instruction RAM's pointer and resumes execution. If the Sequencer is stopped, a full calibration is performed immediately at the time requested.

Bit 4 is the Standby bit. Writing a "1" to Bit 4 immediately places the LM12(H)454/8 in Standby mode. Normal operation returns when Bit 4 is reset to a "0". The Standby comparison



2.0 Internal User-Programmable Registers (Continued)

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

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

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

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

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

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

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

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

2.3 INTERRUPTS

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

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

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

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

TABLE 2. LM12454 Input Multiplexer Channel Configuration

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

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

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

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

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

2.0 Internal User-Programmable Registers (Continued)

The completion of a full auto-zero and linearity self-calibration generates **Interrupt 4**.

Interrupt 5 is generated when the Sequencer encounters an instruction that has its Pause bit (Bit 1 in Instruction RAM "00") set to "1".

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

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

2.4 INTERRUPT ENABLE REGISTER

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

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

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

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

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

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

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

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

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

Bits 8-10 form the storage location of the user-programmable value against which the Sequencer's address is compared. When the Sequencer reaches an address that is equal to the value stored in Bits 8-10, an internal interrupt is generated and appears in Bit 1 of the Interrupt Status register. If Bit 1 of the Interrupt Enable register is set to "1", an external interrupt will appear at pin 31 (INT). The value stored in bits 8-10 ranges from 000 to 111, representing 0 to 7 instructions stored in the Instruction RAM. After the Instruction RAM has been programmed and the RESET bit is set to "1", the Sequencer is started by placing a "1" in the Configuration register's START bit. Setting the INT 1 trigger value to 000 does not generate an INT 1 the first time the Sequencer retrieves and decodes instruction 000. The Sequencer generates INT 1 (by placing a "1" in the Interrupt Status register's Bit 1) the second time and after the Sequencer encounters instruction 000. It is important to re-

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

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

2.5 INTERRUPT STATUS REGISTER

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

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

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

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

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

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

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

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

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

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

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

2.6 LIMIT STATUS REGISTER

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

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



4.0 Sequencer

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

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

- State 0:** The current instruction's first 16 bits are read from the Instruction RAM "00". This state is one clock cycle long.
- State 1:** Checks the state of the Calibration and Start bits. This is the "rest" state whenever the Sequencer is stopped using the reset, a Pause command, or the Start bit is reset low ("0"). When the Start bit is set to a "1", this state is one clock cycle long.
- State 2:** Perform calibration. If bit 2 or bit 6 of the Configuration register is set to a "1", state 2 is 76 clock cycles long. If the Configuration register's bit 3 is set to a "1", state 2 is 4944 clock cycles long.
- State 3:** Run the internal 16-bit Timer. The number of clock cycles for this state varies according to the value stored in the Timer register. The number of clock cycles is found by using the expression below

$$\text{where } 0 \leq T \leq 2^{16} - 1.$$

State 4: Read Limit #2. This state is 1 clock cycle long. This state takes 44 clock cycles when using the 12-bit + sign mode or 21 clock cycles when using the 8-bit + sign mode. The "watchdog" mode takes 5 clock cycles.

$$9 + 2D$$

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

$$2 + 2D$$

where D is the user-programmable 4-bit value stored in bits 12-15 of Instruction RAM "00" and is limited to $0 \leq D \leq 15$. **State 6:** Perform first comparison. This state is 5 clock cycles long.

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

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

The FIFO status should be read in the Interrupt Status register (Bits 11-15) to determine the number of conversion results that are held in the FIFO before retrieving them. This will help prevent conversion data corruption that may take place if the number of reads are greater than the number of conversion results contained in the FIFO. Trying to read the FIFO when it is empty may corrupt new data being written into the FIFO. Writing more than 32 conversion data into the FIFO by the ADC results in loss of the first conversion data. Therefore, to prevent data loss, it is recommended that the LM12(H)4548's interrupt capability be used to inform the system controller that the FIFO is full.

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

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

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

Using the FIFO's full depth is achieved as follows. Set the value of the Interrupt Enable register's Bits 11-15 to 11111 and the Interrupt Enable register's Bit 2 to a "1". This generates an external interrupt when the 31st conversion is stored in the FIFO. This gives the host processor a chance to send a "0" to the LM12(H)4548's Start bit (Configuration register) and halt the ADC before it completes the 32nd conversion. The Sequencer halts after the current (32) conversion is completed. The conversion data is then transferred to the FIFO and occupies the 32nd location. FIFO overflow is avoided if the Sequencer is halted before the start of the 32nd conversion by placing a "0" in the Start bit (Configuration register). It is important to remember that the Sequencer continues to operate even if a FIFO interrupt (INT 2) is internally or externally generated. The only mechanisms that stop the Sequencer are an instruction with the PAUSE bit set to "1" (halts before instruction execution), placing a "0" in the Configuration register's START bit, or placing a "1" in the Configuration register's RESET bit.

2.0 Internal User-Programmable Registers (Continued)

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

2.7 TIMER

The LM12(H)4548 have an on-board 16-bit timer that includes a 5-bit pre-scaler. It uses the clock signal applied to pin 23 as its input. It can generate time intervals of 0 through 2^8 clock cycles in steps of 2⁵. This time interval can be used to delay the execution of instructions. It can also be used to slow the conversion rate when converting slowly changing signals. This can reduce the amount of redundant data stored in the FIFO and retrieved by the controller.

The user-defined timing value used by the Timer is stored in the 16-bit READ/WRITE Timer register at location 1011 (A4-A1, BW = 0) or 101X (A4-A0, BW = 1) and is pre-loaded automatically. Bits 0-7 hold the preset value's low byte and Bits 8-15 hold the high byte. The Timer is activated by the Sequencer only if the current instruction's Bit 9 is set ("1"). If the equivalent decimal value "N" ($0 \leq N \leq 2^{16} - 1$) is written inside the 16-bit Timer register and the Timer is enabled by setting an instruction's bit 9 to a "1", the Sequencer will delay the same instruction's execution by halting at state 3 (S3), as shown in Figure 15, for $32 \times N + 2$ clock cycles.

2.8 DMA

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

DMA operation is optimized through the use of the 16-bit databus connection (a logic "0" applied to the BW pin). Using this bus width allows DMA controllers that have single address Read/Write capability to easily unload the FIFO. Using DMA on an 8-bit databus is more difficult. Two read operations (low byte, high byte) are needed to retrieve each conversion result from the FIFO. Therefore, the DMA controller must be able to repeatedly access two constant addresses when transferring data from the LM12(H)4548 to the host system.

3.0 FIFO

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



5.0 Analog Considerations

5.1 REFERENCE VOLTAGE

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

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

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

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

5.2 INPUT RANGE

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

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

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

Round up to the next integer value between -4096 to 4095 for 12-bit resolution and between -256 to 255 for 8-bit resolution if the result of the above equation is not a whole number. As an example, $V_{REF+} = 2.5V$, $V_{REF-} = 1V$, $V_{IN+} = 1.5V$ and $V_{IN-} = GND$. The 12-bit + sign output code is positive full-scale, or 0,1111,1111,1111. If $V_{REF+} = 5V$, $V_{REF-} = 1V$, $V_{IN+} = 3V$, and $V_{IN-} = GND$, the 12-bit + sign output code is 0,1100,0000,0000.

5.3 INPUT CURRENT

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

5.4 INPUT SOURCE RESISTANCE

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

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

5.5 INPUT BYPASS CAPACITANCE

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

5.6 NOISE

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

5.7 POWER SUPPLIES

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

5.8 GROUNDING

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

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

5.9 CLOCK SIGNAL LINE ISOLATION

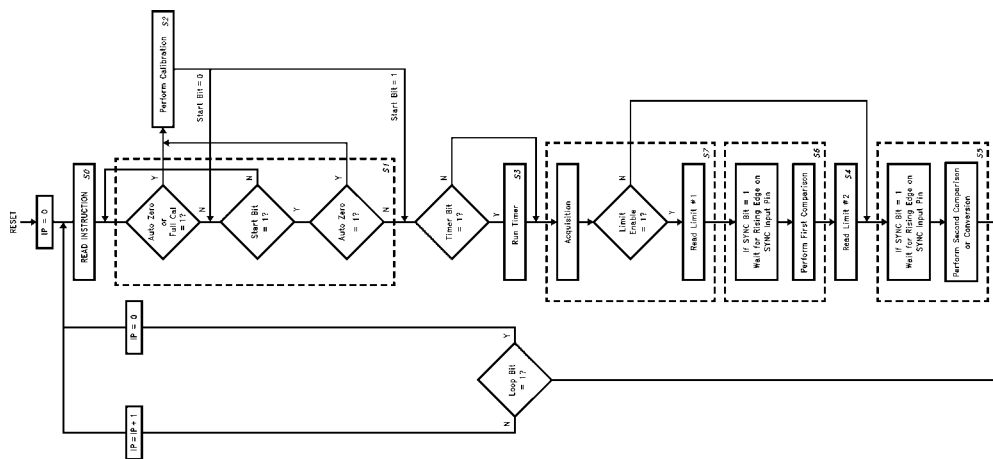
The LM12(H)454/8's performance is optimized by routing the analog input/output and reference signal conductors (pins 34-44) as far as possible from the conductor that carries the clock signal to pin 23. Ground traces parallel to the clock signal trace can be used on printed circuit boards to reduce clock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

6.0 Application Circuits

PC EVALUATION/INTERFACE BOARD

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

4.0 Sequencer (Continued)



6.0 Application Circuits (Continued)

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

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

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

All digital interface control signals (IOR, IOW, and AEN), data lines (DB0-DB7), address lines (A0-A9), and IRQ (interrupt request) lines (IRQ2, IRQ3, and IRQ5) connections are made through the motherboard slot connector. All analog signals applied to, or received by, the input multiplexer (IN0-IN7 for the LM12(H)454/8 and IN0-IN3, MUXOUT+, MUXOUT-, S/H IN+ and S/H IN- for the LM12(H)454), V_{REF+} , V_{REF-} , V_{REFOUT} , and the SYNC signal input/output are applied through a DB-37 connector on the rear side of the board. Figure 16 shows that there are numerous analog ground connections available on the DB-37 connector.

The voltage applied to V_{REF-} and V_{REF+} is selected using two jumpers, JP1 and JP2. JP1 selects between the voltage applied to the DB-37's pin 24 or GND and applies it to the LM12(H)454/8's V_{REF-} input. JP2 selects between the LM12(H)454/8's internal reference output, V_{REFOUT} , and the voltage applied to the DB-37's pin 22 and applies it to the LM12(H)454/8's V_{REF+} input.

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

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

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

Switches 1-4, along with address lines A5-A9 are used as inputs to GAL16V8 Programmable Gate Array (U2). This device forms the interface between the computer's control and address lines and generates the control signals used by the LM12(H)454/8 for CS, WR, and RD. It also generates the signal that controls the data buffers. Several address ranges within the computer's I/O memory map are available. Refer to Table III for the switch settings that gives the desired I/O memory address range. Selection of an address range must be done so that there are no conflicts between the evaluation board and any other boards attached to the computer's motherboard. The GAL equations are shown in Figure 18. The GAL functional block diagram is shown in Figure 19. Figures 20, 21, 22, 23 show the layout of each layer in the 3-layer evaluation/interface board plus the silk-screen layout showing parts placement. Figure 21 is the top or component side, Figure 22 is the middle or ground plane layer, Figure 23 is the circuit side, and Figure 20 is the parts layout.

APPENDICE C: INDICE ANALITICO

A

A/D converter 6, 8, 14, 15, 24, 31, 51, B-19
Alimentazione 9, 23, 36
Assistenza 1

B

Back up 3, 6, 9, 10, 28, 34
Batteria 9, 10, 25, 34, 47
BUS ABACO® 4, 8, 22, 31, A-6

C

Caratteristiche elettriche 9
Caratteristiche fisiche 8
Caratteristiche generali 8
Circuiteria terminazione 9, 20, 34
Clock 4, 8
Conessioni 63
Connettori 8, 10, 25
 CN1 16
 CN2 12
 CN3 14
 CN4 18, 34
 CN5 11
 CN6 10
 K1 22
Consumo 9
Contenitore 1
Counter 8, 16, 17
CPU 3, 8, 16, B-1
Current loop 4, 18, 21, 24, 28, 32

D

Data 6, 49
Dimensioni 8
Dip switch 8, 25, 36, 42, 47
DMA 8, 30, 36, 41, 48
Drivers seriali 33

E

EEPROM 3, 8, 25, 41, 43, 46
EPROM 3, 8, 25, 43
Espansione 26, 61

F

FLASH EPROM 3, 8, 43

Foto 7

G

Garanzia 1

H

Handshake 34, 48

I

I/O digitale 6, 8, 11, 12, 26, 59, 62

IAC 01 A-1

Indirizzamenti 40

Indirizzi memorie 44

Informazioni generali 2

Ingressi analogici 9, 14, 24

Ingressi configurazione 28, 36, 47

Installazione 1, 10

Interfacciamento 24

Interfacce 26, A-1

Interrupts 31

J

Jumpers 27

2 vie 28

3 vie 30

4 vie 28

5 vie 30

K

KDx x24 A-2

L

LEDs 25, 26

LEDs attività 26, 41, 48

LM 12H458 6, 14, 36, 41, 51, B-19

M

Mappaggi 40

BUS ABACO® 43

I/O 41

memorie 43



Memorie 3, 8, 25, 30, 35, 43, 50
Memorie di massa 26
MMU 40, 41, 44, 50
Mother board 61

O

Opzioni 32, 34, 35
Ora 6, 49

P

Pannelli operatore 26, 61, A-2
Periferiche interne 31, 60
Peso 8
Pianta componenti 37
Port A 12, 59
Port B 11, 59
Port C 12, 59
Power failure 9, 28, 31, 36
PPI 82C55 6, 11, 12, 42, 59
Protezione scrittura 28, 42, 48

Q

QTP 16P A-3
QTP 24P A-4
Quarzo 4

R

Real Time Clock 6, 8, 31, 34, 42, 49
Registri 41
Reset 31, 43
Rete seriale 20
Risoluzione 8
Risorse 8
RS 232 4, 16, 19, 24, 28, 32
RS 422 4, 9, 18, 19, 24, 28, 30, 32
RS 485 4, 9, 18, 19, 20, 24, 28, 30, 34

S

SCC 85C30 4, 16, 31, 32, 36, 41, 51
Schema a blocchi 5
Schemi elettrici A-1
Segnalazioni visive 26
Seriali 4, 8, 16, 17, 31, 32, 51
Software 38
SPA 01 A-6

Specifiche tecniche 8
SRAM 3, 8, 25, 34, 43
Stampante 26, A-1
Stato batteria 41, 47

T

Tarature 24
Temperatura 9
Tensione riferimento 24
Test point 24, 25
Timer 8, 16, 17
Trimmer 24, 25
TTL 24

U

Umidità 9

V

Versione 1

W

Watch dog 6, 8, 26, 30, 31, 41, 46