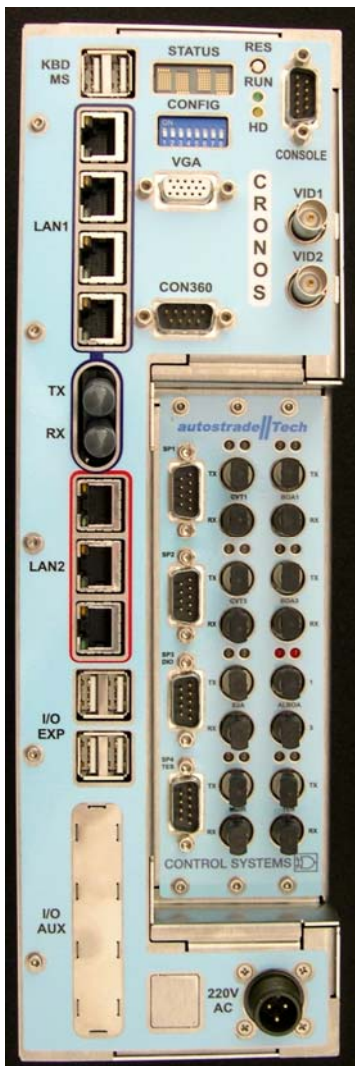


CRONOS

Manuale d'uso



Identificazione del prodotto

codice	U0933
Versione	V2.0
nome	CRONOS

Identificazione del documento

Document identification

codice	
Revisione	
data	
emesso da	
approvato da	

Sommario

	<i>pag.</i>
1 Organizzazione del manuale	4
2 Caratteristiche tecniche generali	5
2.1 Identificazione dell'unità	5
2.2 Alimentazione	6
2.3 Caratteristiche Ambientali	6
2.4 Caratteristiche Meccaniche	6
2.5 Blocchi funzionali	8
2.5.1 Main CPU Core	10
2.5.2 Blocchi funzionali specifici dell'applicazione	11
2.5.3 Pannello frontale	12
2.5.4 Pin-function dei connettori di I/O	16
2.5.5 Pin-function del connettore di alimentazione	17
3 Caratteristiche dettagliate dei blocchi funzionali	20
3.1 Main-CPU Core	20
3.2 Frame Grabber	20
3.3 Real-Time I/O Subsystem	21
3.3.1 Mappatura delle porte di I/O	23
3.4 Dispositivi Ausiliari	25
3.4.1 Status Display	25
3.4.2 DIP-Switch	25
3.4.3 Sensore di Temperatura	25
3.4.4 Watch-Dog	25
3.4.5 Flash Memory	25
3.4.6 Silicon Serial Number	25
4 Programmazione dei Dispositivi Ausiliari	26
4.1 Struttura dei registri dei Dispositivi Ausiliari	26
4.1.1 CSR - Control/Status Register (0x00)	27
4.1.2 DDS - Display Digit Select Register (0x01)	28
4.1.3 DDD - Display Digit Data Register (0x02)	28
4.1.4 SWP - DIP-switch Register (0x04)	30
4.1.5 TMP - Temperature Alarm Register (0x06)	30
4.1.6 WDP - Watch-Dog Counter Preset Register (0x09)	31
4.1.7 FLL - Flash Memory Low-byte Address Register (0x10)	32
4.1.8 FLM - Flash Memory Mid-byte Address Register (0x11)	32
4.1.9 FLH - Flash Memory High-byte Address Register (0x12)	32
4.1.10 FLD - Flash Memory Data Register (0x13)	33
4.1.11 FLC - Flash Memory Status/Control Register (0x14)	33
4.1.12 SND - "Silicon Serial Number" Data Register (0x20)	36
4.1.13 SNC - "Silicon Serial Number" Status/Control Register (0x21)	36
5 Test dell'unità CRONOS	38
5.1 Console Seriale	38
5.2 Dispositivi di Boot	38
5.3 Procedura di test funzionale	39
5.3.1 Materiale necessario	39
5.3.2 Sistema di test	39
5.3.3 Boot del Sistema	39
5.3.4 Memory Test	40
5.3.5 csLinux login	41
5.3.6 Test funzionali	42

Figure

	<i>pag.</i>
Figura 2-1: Vista 3D dell'unità CRONOS	7
Figura 2-2: Schema a blocchi dell'unità CRONOS	9
Figura 2-3: Pannello frontale dell'unità CRONOS – configurazione con modulo PIM-O	14
Figura 2-4: Pannello frontale dell'unità CRONOS – configurazione con modulo PIM-E	15
Figura 2-5: Posizionamento dei jumper per setting porta tipo RS232/422 o DIO	18
Figura 2-6: Circuito elettrico Digital Input	19
Figura 2-7: Circuito elettrico Digital Output	19
Figura 3-1: Blocco funzionale “Frame Grabber”	20
Figura 3-2: Schema a Blocchi del Real-Time I/O Subsystem	22
Figura 5-1: Pagina di visualizzazione del Memory Test	40
Figura 5-2: Pagina Welcome di csLinux	41
Figura 5-3: Pagina Lista dei PCI Devices	42
Figura 5-4: Pagina Test Porte Ethernet	43
Figura 5-5: Pagina Test Hard-Disk	44
Figura 5-6: Pagina Test Porte USB	45
Figura 5-7: Pagina Test Porte Seriali I/O	48

1 Organizzazione del manuale

Questo manuale descrive le caratteristiche tecniche e le modalità d'uso dell'unità CRONOS.

Qui di seguito è riportato un sommario dei contenuti di ciascun capitolo.

Capitolo 2: Caratteristiche tecniche generali

Questo capitolo descrive le caratteristiche generali dell'unità CRONOS. Vengono specificati il *codice di identificazione* dell'unità e le caratteristiche tecniche principali. Viene mostrato lo schema a blocchi dell'unità ed una breve descrizione di ogni singolo blocco funzionale. Vengono anche descritte le pin-function di tutti i connettori di I/O e le specifiche elettriche dei relativi segnali.

Capitolo 3: Caratteristiche dettagliate dei blocchi funzionali

Questo capitolo descrive le caratteristiche hardware dettagliate dell'unità CRONOS. Vengono elencate le caratteristiche tecniche di dettaglio di ogni blocco funzionale e/o i riferimenti alla documentazione del costruttore dei componenti principali.

Capitolo 4: Programmazione dei Dispositivi Ausiliari

Questo capitolo fornisce la descrizione dettagliata, ai fini della programmazione, dei dispositivi ausiliari: *Status Display, DIP-switch, Sensore di temperatura, Watch-Dog, Flash Memory, Silicon Serial Number*.

Capitolo 5: Test dell'unità CRONOS

In questo capitolo sono descritte le operazioni di test dell'unità CRONOS.

2 Caratteristiche tecniche generali

L'unità CRONOS è costituita da un *single-board* computer installato in una *enclosure fanless* appositamente dimensionata per essere alloggiata nel "Quadro Elettrico di Pista".

L'unità costituisce un *sistema embedded*, orientato verso applicazioni che richiedono elevate prestazioni di tipo computazionale e real-time, unitamente alla disponibilità di consistenti risorse di connettività in rete, comunicazione seriale e di I/O in genere.

Le caratteristiche principali dell'unità sono:

- CPU Intel Atom N270 @ 1.6GHz
- 512Kbyte L2 Cache
- Front Bus Speed 533MHz
- Chipset Intel 945GSE (GMCH) & Intel 82801DB (ICH7-M)
- Memoria DDR SDRAM da 1Gbyte (espandibile a 2 Gbyte)
- 2 porte Ethernet Controllers 10/100Mbps
- 2 Ethernet Switch : uno per connessione alla *LAN di Stazione*, uno per connessione alla *subLAN di Pista*
- Hard Disk interno di formato 2.5", di tipo IDE o SATA
- Real-Time I/O Subsystem costituito da 2 microprocessori Motorola MC68360 in configurazione Master-Slave
- Video Frame Grabber (1 Video Decoder Bt878A)
- 4 porte USB 2.0 /1.1 per espansione I/O
- 1 SmartCard Reader Controller
- 1 Slot di espansione PCI @33MHz per modulo opzionale standard Mezzanine PMC IEEE P1386.1
- Embedded BIOS (Phoenix)
- Alimentazione +220VAC @ 50W

2.1 Identificazione dell'unità

L'unità viene identificata dai seguenti codici:

codice prodotto : U0933
nome prodotto : CRONOS
numero di serie : Uaamm-ppp

dove:

aa : indica l'anno di produzione,
mm : indica il mese di produzione
ppp : indica il numero progressivo

Questi codici sono visibili sull'etichetta metallica posta all'esterno dell'unità stessa.

2.2 Alimentazione

L'unità viene alimentata mediante il connettore 220VAC posto sul pannello frontale.

La disposizione dei pin di alimentazione è riportata nel Capitolo 2.5.5 .

<i>Tensione di alimentazione:</i>	220 VAC/50 Hz (range universale 90÷264 VAC / 47÷400 Hz)
<i>Assorbimento medio:</i>	33W
<i>Assorbimento di picco:</i>	50W



ATTENZIONE Gli assorbimenti indicati si riferiscono ad una unità configurata con

- modulo di memoria DDR2 da 1 Gbyte@533MHz,
- Hard Disk 2.5" di tipo magnetico (capacità di 80 Gbyte),
- modulo di interfaccia di I/O tipo ottico (PIM-O),
- modulo PMC di espansione NON installato.

2.3 Caratteristiche Ambientali

L'unità CRONOS è dotata di un sistema di dissipazione termica capace di garantirne il funzionamento, nel range di temperatura specificato, in regime di *convezione naturale* (quindi, l'unità non è dotata di alcun sistema di raffreddamento mediante ventilazione forzata).

<i>Temperatura di operazione:</i>	-10 ÷ 60 °C
<i>Temperatura di stoccaggio:</i>	-40 ÷ +85 °C
<i>Umidità relativa:</i>	max 95 % non-condensing
<i>Caratteristiche EMI/EMC:</i>	conformi alle normative EN 61000-6-4 e EN 61000-6-2 applicabili

2.4 Caratteristiche Meccaniche

<i>Fattore di forma:</i>	Custom
<i>Materiale:</i>	Alluminio cromato ed Acciaio inox AISI 304
<i>Dimensioni:</i>	H= 325, L= 100, P= 200 mm
<i>Peso:</i>	3,9 Kg



Figura 2-1: Vista 3D dell'unità CRONOS

2.5 Blocchi funzionali

L'unità CRONOS è costituita dai seguenti blocchi funzionali principali:

- Main-CPU Core,
- Dispositivi Ausiliari,
- Interfacce di rete Ethernet,
- Blocchi funzionali specifici dell'applicazione:
 - Real-Time I/O Subsystem,
 - Frame-Grabber,
 - Porte di espansione USB,
 - Modulo di espansione PMC (opzionale).

In Figura 2-2 è rappresentato lo schema a blocchi completo dell'unità CRONOS.

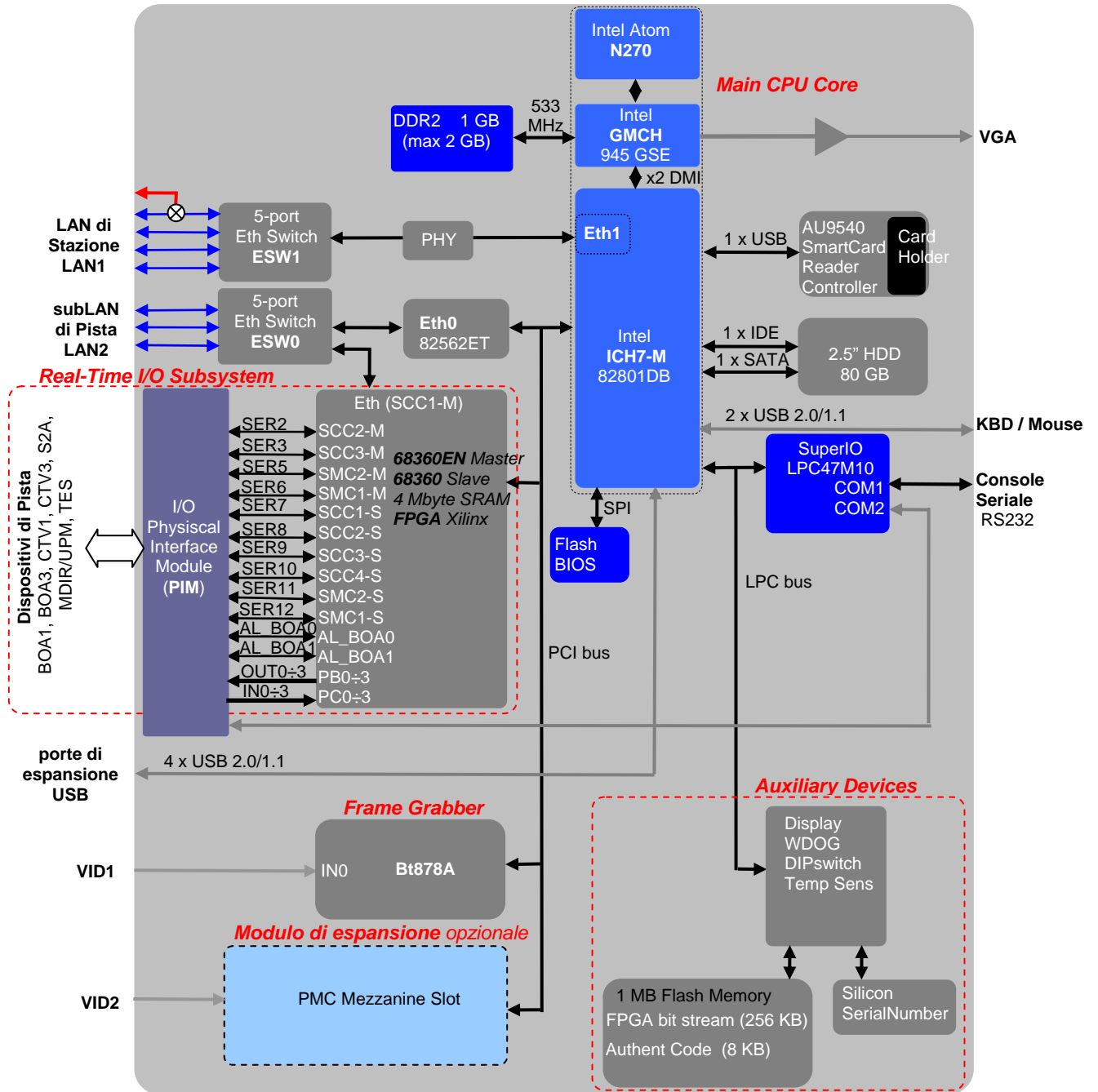


Figura 2-2: Schema a blocchi dell'unità CRONOS

2.5.1 *Main CPU Core*

Questo blocco comprende le risorse principali di elaborazione e di connessione in rete dell'unità.

Esso si basa su una piattaforma hardware *Intel Atom N270* (CPU clock da 1.6 GHz, 512 Kbyte di L2-cache, velocità del Front-Bus di 533 MHz) e Chipset *Intel 945GSE* (GMCH) e *82801DB* (ICH7-M).

L'unità dispone di una slot per ospitare un modulo DDR2 SODIMM da max 2 Gbyte (nella configurazione base viene installato un modulo da 1 Gbyte).

L'unità ospita internamente un dispositivo Hard Disk di formato 2.5" standard. Può essere installato un Hard Disk con interfaccia IDE o SATA, di tipo magnetico o flash-disk (nella configurazione base dell'unità, viene installato un Hard Disk magnetico IDE o SATA da 80 Gbyte, sul quale è pre-installato il Sistema Operativo Linux di Test ed il software di diagnostica dell'unità stessa).

L'unità è dotata sia di una interfaccia di Console Grafica standard PC (KBD, Mouse, VGA), sia di una porta di Console Seriale RS232 (COM1).

Il BIOS *Phoenix* installato dispone delle seguenti funzionalità particolari:

- o Console di Sistema direzionabile (durante il Boot) sulla porta seriale COM1 (default) o sull'interfaccia grafica standard (KBD, Mouse, Monitor VGA);
- o Boot da USB.

2.5.1.1 **Real Time Clock**

La memoria del RTC è dotata di alimentazione di backup di tipo *supercapacitor* capace di fornire una autonomia di circa 4 giorni. Il tempo di ricarica è praticamente istantaneo.

2.5.1.2 **Dispositivi Ausiliari**

Questo blocco realizza alcune funzioni ausiliarie del blocco Main CPU disponibili per l'utilizzo del modulo nell'ambito dell'applicazione "Pista Autostradale". Queste funzioni sono: *Status Display*, *DIP-switch*, *Sensore di temperatura*, *Watch-Dog*, *Flash Memory*, *Silicon Serial Number*.

Per quanto riguarda la programmazione di tali dispositivi fare riferimento al Cap.4.

2.5.1.3 **Interfacce di rete Ethernet**

Per la connettività in rete, sono disponibili due Ethernet Controller indipendenti (Eth0, Eth1).

L'Ethernet Controller Eth1 (integrato nel ICH7-M) è connesso internamente ad un Ethernet Switch (ESW1) a 5 porte; questo blocco costituisce l'interfaccia di rete per la connessione alla *LAN di Stazione*.

Questa interfaccia di rete rende disponibili:

- 3 porte Ethernet con interfaccia elettrica RJ45 (10/100 BaseTX)
- 1 porta Ethernet con interfaccia elettrica RJ45 (10/100 BaseTX), oppure con teste ottiche ST (10/100 BaseFX). Il tipo di interfaccia elettrica/optica è selezionata dallo stato ON/OFF del DIP-Switch 1 (vedi par. 3.4.2)

L'Ethernet Controller Eth0 (esterno al ICH7-M) è connesso internamente ad un Ethernet Switch (ESW0) a 5 porte; questo blocco costituisce l'interfaccia di rete per la connessione alla *SubLAN di Pista*.

Questa interfaccia di rete rende disponibili :

- 3 porte Ethernet con connessione elettrica RJ45 (10/100 BaseTX) .

Una porta Ethernet dello Switch ESW0 è internamente connessa al microcontroller MC68360EN (vedi Cap.3.3 Real-Time I/O Subsystem).

2.5.2 *Blocchi funzionali specifici dell'applicazione*

2.5.2.1 Rel-Time I/O-Subsystem

Questo blocco realizza le *funzioni di I/O specifiche per le varie tipologie di impianto "Pista Autostradale"* ed è realizzato dall'insieme di due microcontroller interconnessi in configurazione Master-Slave (un Motorola MC68EN360 Master ed un Motorola MC68360 Slave) e da funzioni logiche specifiche dell'applicazione (*Application Specific Programmable Logic - ASPL*) realizzate mediante un FPGA Xilinx. La connessione di questo sottosistema al Main Processor Intel N270 è costituita da una interfaccia PCI Master-Target realizzata all'interno dello stesso FPGA.

2.5.2.2 Interfaccia per il "Terminale Esattore"

Questo blocco realizza l'interconnessione con il Terminale Esattore ed è costituito dalla linea seriale asincrona COM2 della Main CPU (Intel Atom N270). Questa porta viene internamente diramata (all'interno del modulo PIM) su due porte fisiche:

- una porta ottica (TES)
- una porta RS232/422 (TES)

2.5.2.3 I/O Physical Interface Module (PIM)

Questo blocco contiene i circuiti per la "personalizzazione" delle interfacce di I/O:

- circuiti *transceiver* per la traslazione TTL ↔ RS232/422,
- circuiti *driver* e le *teste ottiche* per la conversione TTL ↔ ottico.

Il blocco è fisicamente realizzato da un modulo hardware rimovibile, in modo da poter generare set di interfacce di I/O diversificate in funzione delle versioni di impianto in cui l'unità viene impiegata.

Sono disponibili due versioni di PIM:

- **PIM-O** – modulo PIM per Pista con cablaggio in Fibra Ottica,
- **PIM-E** – modulo PIM per Pista con cablaggio in cavo elettrico.

2.5.2.4 Frame Grabber

Questo blocco realizza la funzione di *Video Frame Grabbing* ed è basato sul Video Decoder Bt878A. Tale funzione rende disponibile un ingresso video PAL standard.

2.5.2.5 SmartCard Reader

Questo blocco realizza la funzione di lettore di SmartCard ed è basato sul Controller AU9540 (ALCOR MICRO), collegato ad una porta USB della Main CPU (Intel Atom N270).

Il Controller AU9540 supporta i vari standard internazionali (ISO7816, PC/SC 2.0, WHQL, USB-IF CCID).

Per maggiori dettagli, fare riferimento al documento AU9540 Technical Reference Manual Rev. 1.60.

2.5.2.6 Porte di espansione USB

Per il collegamento di eventuali ulteriori dispositivi esterni, l'unità dispone di 4 porte USB 2.0/1.1 direttamente gestite dal blocco Main CPU (Intel Atom N270).

2.5.2.7 Slot PCI di espansione

L'unità dispone inoltre di una slot di espansione PCI (32-bit) capace di ospitare opzionalmente un modulo di formato standard *Mezzanine PMC IEEE P1386.1*.



ATTENZIONE L'interfaccia PCI del modulo PMC opzionale deve essere "5V tolerant".

2.5.3 *Pannello frontale*

Sul pannello frontale sono disponibili tutte le interfacce funzionali dell'unità:

Il pannello frontale si compone di due parti:

- ❖ una parte, appartenente all'enclosure base dell'unità, sulla quale si trovano:

DISPOSITIVI E CONNETTORI DI INTERFACCIA PER MANUTENZIONE/CONFIGURAZIONE

- **RES** pulsante di RESET
- **RUN** LED indicatore di RUN della CPU
- **HD** LED indicatore di ATTIVITÀ dell'Hard-Disk
- **STATUS** Status Display (4 caratteri alfanumerici)
- **CONFIG** DIP-Switch di configurazione (8-switch)
- Interfaccia di Console Grafica standard PC:
 - **KBD** porta USB 2.0 / 1.1 per Keyboard
 - **MS** porta USB 2.0 / 1.1 per Mouse
 - **VGA** porta VGA per Monitor
- Interfaccia di Console Seriale :
 - **CONSOLE** porta seriale RS232 full-modem per Console Seriale
- Interfaccia di Console per il Real-Time I/O Subsystem :
 - **CON360** porta seriale RS232 per Console 68360

CONNETTORI DI RETE ETHERNET

- **LAN1** connessione alla *LAN di Stazione*
 - 1 porta Ethernet 10/100, sia con interfaccia elettrica RJ45 (10/100 BaseTX), sia con interfaccia ottica (10/100 BaseFX).
 - 3 porte Ethernet con connessione elettrica RJ45 (10/100 BaseTX)
- **LAN2** connessione alla *subLAN di Pista*
 - 3 porte Ethernet con connessione elettrica RJ45 (10/100 BaseTX)

INTERFACCE DI ESPANSIONE

- **I/O EXP** 4 porte USB 2.0 / 1.1 per eventuali moduli esterni di Espansione I/O

INTERFACCE DI I/O AUSILIARIO

- **I/O AUX** slot PCI per modulo PMC opzionale
(i cui connettori eventuali si affacciano sempre sul pannello frontale)

- ❖ una parte, appartenente al modulo rimovibile PIM (*I/O Physical Interface Module*), sulla quale si trovano i connettori di interfaccia verso i Dispositivi di Pista (non dotati di porta ethernet).

Sono disponibili due versioni del modulo hardware *Physical Interface Module* (PIM):

PIM-O – modulo PIM per Pista con cablaggio in Fibra Ottica, sul cui pannello frontale si trovano i seguenti connettori:

BOA1	porta ottica TX-RX per dispositivo BOA1
BOA3	porta ottica TX-RX per dispositivo BOA3
CTV1	porta ottica TX-RX per dispositivo CTV1
CTV3	porta ottica TX-RX per dispositivo CTV3
S2A	porta ottica TX-RX per dispositivo S2A
ALBOA1	porta ottica per Ingresso ALLARME BOA1
ALBOA3	porta ottica per Ingresso ALLARME BOA3
MDIR	porta ottica TX-RX per dispositivo MDIR o UPM
TES	porta ottica TX-RX + per Terminale Esattore
SP4 TES	porta RS232/422 TX-RX per Terminale Esattore
SP1	porta RS232/422 TX-RX <i>spare</i>
SP2	porta RS232/422 TX-RX <i>spare</i>
SP3 DIO	porta RS232/422 TX-RX <i>spare</i> o (2 IN + 2 OUT) Digitali <i>spare</i> (configurabile via jumpers all'interno del modulo PIM)

PIM-E – modulo PIM per Pista con cablaggio in cavo elettrico, sul cui pannello frontale si trovano i seguenti connettori:

BOA1	porta RS232/422 TX-RX + Ingresso ALLARME BOA1 per dispositivo BOA1
BOA3	porta RS232/422TX-RX + Ingresso ALLARME BOA2 per dispositivo BOA3
CTV1	porta RS232/422TX-RX per dispositivo CTV1
CTV3	porta RS232/422TX-RX per dispositivo CTV3
S2A	porta RS232/422TX-RX per dispositivo S2A
MDIR	porta RS232/422ottica TX-RX per dispositivo MDIR o UPM
SP4 TES	porta RS232/422 TX-RX per Terminale Esattore
DIO	(2 IN + 2 OUT) Digitali <i>spare</i>
SP1	porta RS232/422 TX-RX <i>spare</i>
SP2	porta RS232/422 TX-RX <i>spare</i>
SP3 DIO	porta RS232/422 TX-RX <i>spare</i> o (2 IN + 2 OUT) Digitali <i>spare</i> (configurabile via jumpers all'interno del modulo PIM)

In Figura 2-3 e Figura 2-4 viene illustrato il pannello frontale dell'unità CRONOS nelle due configurazioni descritte sopra.

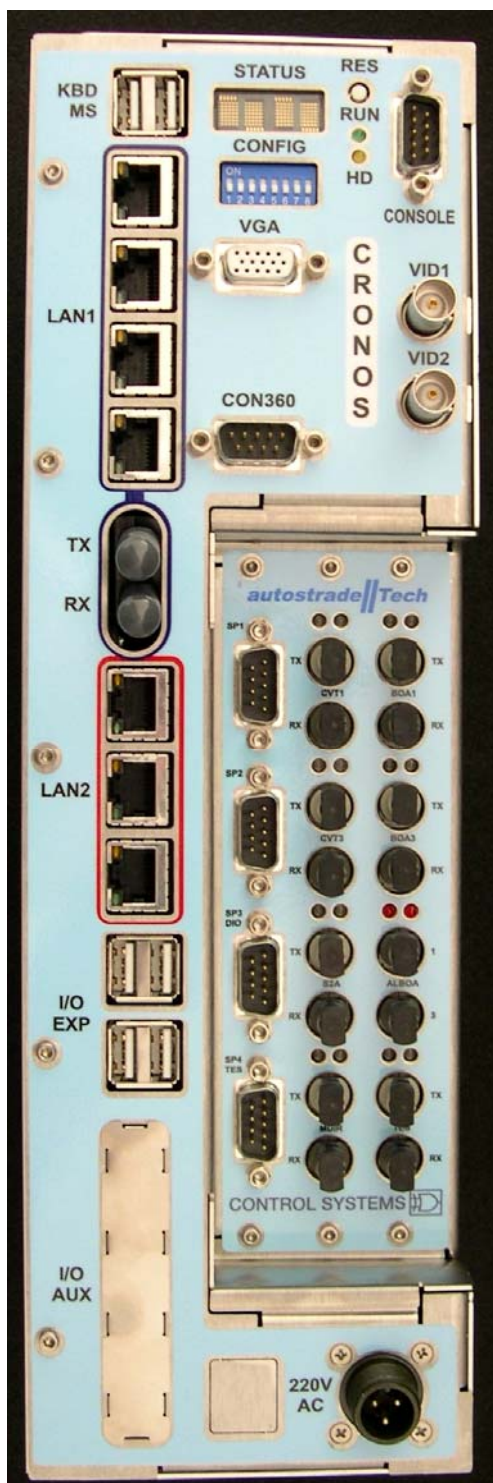


Figura 2-3: Pannello frontale dell'unità CRONOS – configurazione con modulo PIM-O



Figura 2-4: Pannello frontale dell'unità CRONOS – configurazione con modulo PIM-E

2.5.4 *Pin-function dei connettori di I/O*

Qui di seguito sono descritte le pin-function dei vari connettori di I/O disponibili sul pannello frontale dell'unità CRONOS.

CONSOLE connettore DB9P - Porta Seriale Asincrona RS232

<i>pin</i>	<i>segnale</i>	<i>funzione</i>	<i>circuito di interfaccia</i>
1	DCD	Data Carrier Detect	RS232
2	RXD	Receive Data	RS232
3	TXD	Transmit Data	RS232
4	DTR	Data Terminal Ready	RS232
5	GND	Signal Ground	
6	DSR	Data Set Ready	RS232
7	RTS	Request to Sensd	RS232
8	CTS	Clear to Send	RS232
9	RI	Ring Indicator	RS232

CON360 connettore DB9P - Porta Seriale Asincrona RS232

<i>pin</i>	<i>segnale</i>	<i>funzione</i>	<i>circuito di interfaccia</i>
2	RXD	Receive Data	RS232
3	TXD	Transmit Data	RS232
5	GND	Signal Ground	

MDIR
SP1
SP2
SP4 TES

connettore DB9P - Porta Seriale Asincrona RS232/422

<i>pin</i>	<i>segnale</i>	<i>funzione</i>	<i>circuito di interfaccia</i>
2	RXD	Receive Data	RS232
3	TXD	Transmit Data	RS232
5	GND	Signal Ground	
6	TX+	Transmit Data +	RS422
7	TX -	Transmit Data -	RS422
8	RX+	Receive Data +	RS422
9	RX -	Receive Data -	RS422

SP3 DIO connettore DB9P - configurata come Porta Seriale Asincrona RS232/422 (vedi Figura 2-5)

<i>pin</i>	<i>segnale</i>	<i>funzione</i>	<i>circuito di interfaccia</i>
2	RXD	Receive Data	RS232
3	TXD	Transmit Data	RS232
5	GND	Signal Ground	
6	TX+	Transmit Data +	RS422
7	TX -	Transmit Data -	RS422
8	RX+	Receive Data +	RS422
9	RX -	Receive Data -	RS422

connettore DB9P - configurata come Porta Digital I/O (vedi Figura 2-5)

<i>pin</i>	<i>segnale</i>	<i>funzione</i>	<i>circuito di interfaccia</i>
1	OUT1-	Digital Output 1-	optoisolatore
2	OUT1+	Digital Output 1+	vedi Figura 2-7
3	OUT2+	Digital Output 2+	optoisolatore
4	OUT2-	Digital Output 2-	vedi Figura 2-7
6	IN1+	Digital Input 1 +	optoisolatore
7	IN1-	Digital Input 1 -	vedi Figura 2-6
8	IN2+	Digital Input 2 +	optoisolatore
9	IN2 -	Digital Input 2 -	vedi Figura 2-6

DIO

connettore DB9P - Porta Digital I/O

<i>pin</i>	<i>segnale</i>	<i>funzione</i>	<i>circuito di interfaccia</i>
1	OUT1-	Digital Output 1-	optoisolatore
2	OUT1+	Digital Output 1+	vedi Figura 2-7
3	OUT2-	Digital Output 2+	optoisolatore
4	OUT2+	Digital Output 2-	vedi Figura 2-7
6	IN1+	Digital Input 1 +	optoisolatore
7	IN1-	Digital Input 1 -	vedi Figura 2-6
8	IN2+	Digital Input 2 +	optoisolatore
9	IN2 -	Digital Input 2 -	vedi Figura 2-6

BOA1
BOA3

connettore DB9P - Porta Seriale Sincrona RS232/422

<i>pin</i>	<i>segnale</i>	<i>funzione</i>	<i>circuito di interfaccia</i>
1	AL_BOA+	BOA Alarm +	RS422
2	RXD	Receive Data	RS232
3	TXD	Transmit Data	RS232
4	AL_BOA-	BOA Alarm -	RS422
5	GND	Signal Ground	
6	TX+	Transmit Data +	RS422
7	TX -	Transmit Data -	RS422
8	RX+	Receive Data +	RS422
9	RX -	Receive Data -	RS422

CTV1
CTV3
S2A

connettore DB9P - Porta Seriale Sincrona RS232/422

<i>pin</i>	<i>segnale</i>	<i>funzione</i>	<i>circuito di interfaccia</i>
2	RXD	Receive Data	RS232
3	TXD	Transmit Data	RS232
5	GND	Signal Ground	
6	TX+	Transmit Data +	RS422
7	TX -	Transmit Data -	RS422
8	RX+	Receive Data +	RS422
9	RX -	Receive Data -	RS422

2.5.5 *Pin-function del connettorei di alimentazione*

220VAC Connettore circolare

<i>pin</i>	<i>segnale</i>	<i>funzione</i>	<i>circuito di interfaccia</i>
A	L	220 VAC fase	ingresso L AC-DC
B	PE	TERRA	collegato internamente a DGND e CHASSIS
C	N	220 VAC neutro	ingresso N AC-DC

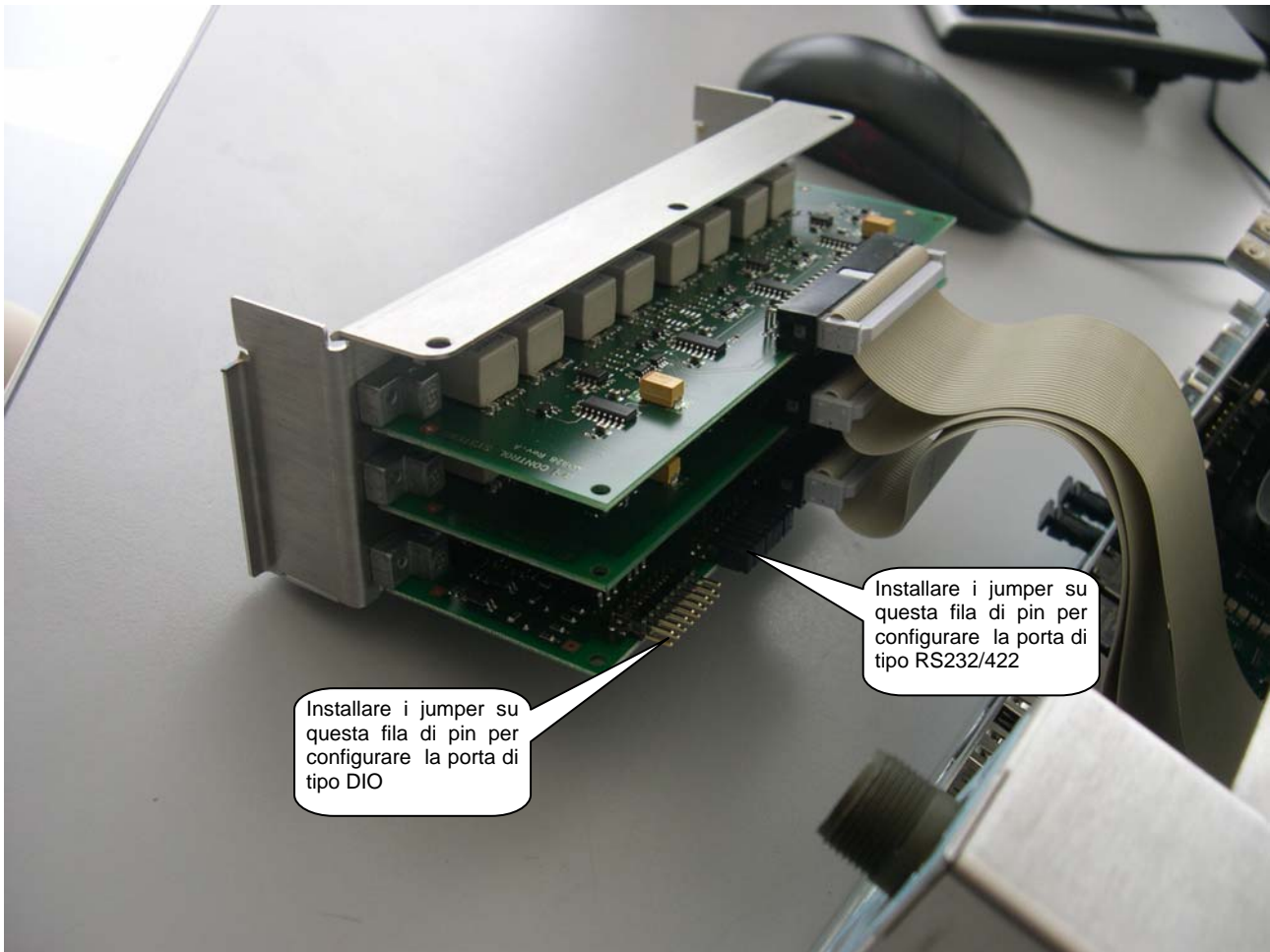


Figura 2-5: Posizionamento dei jumper per setting porta tipo RS232/422 o DIO

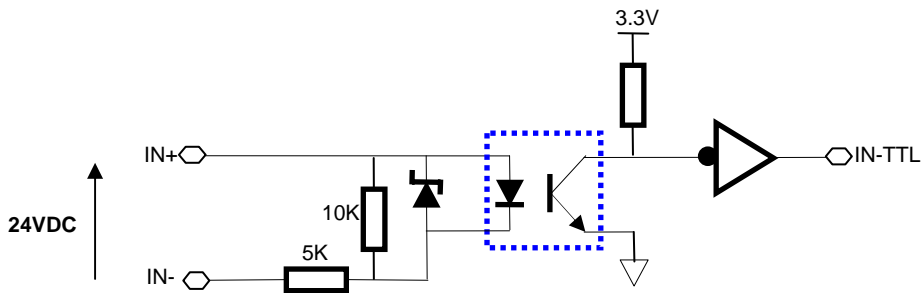


Figura 2-6: Circuito elettrico Digital Input

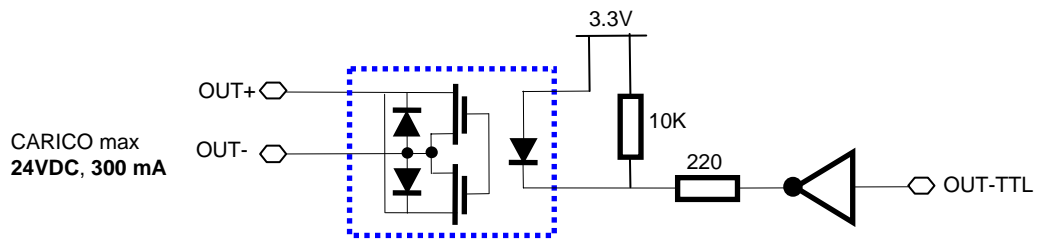


Figura 2-7: Circuito elettrico Digital Output

3 Caratteristiche dettagliate dei blocchi funzionali

In questo capitolo sono elencate le caratteristiche hardware dei vari blocchi funzionali. Vengono anche descritte le varie interfacce di I/O (tipologia e caratteristiche elettriche/ottiche) dell'unità CRONOS.

3.1 Main-CPU Core

Il blocco funzionale Main CPU d'unità CRONOS è basata su un processore Intel Atom N270 da 1.6GHz dotato di 512 Kbyte di Cache L2 e di un Front-bus da 533 MHz.

La memoria è costituita da un unico modulo DDR2 SODIMM da 533MHz (nella configurazione base è installato un modulo da 1 Gbyte).

Il Chipset è costituito dai chip Intel 945GSE (GMCH) e Intel 82801DB (ICH7-M).

L'Ethernet Controller Eth1 è costituito dal chip Intel 82562ET.

Per ogni dettaglio relativo ai componenti del Main CPU Core, si fa riferimento ai seguenti documenti Intel :

- Mobile Intel Atom Processor N270 Single Core datasheet (Intel doc: 320032-001)
- Intel Atom Processor N270 Series Specification Update (Intel doc: 320047-003)
- Mobile Intel 945 Express Chipset Family datasheet (Intel doc: 309219-006)
- Intel I/O Controller Hub 7 (ICH7) Family datasheet (Intel doc: 307013-003)
- Intel 82562ET 10/100 Mbps Platform LAN Connect (PLC) datasheet

3.2 Frame Grabber

Il blocco funzionale Frame Grabber è costituito da un Video Decoder *Bt878A* (agente del bus PCI),

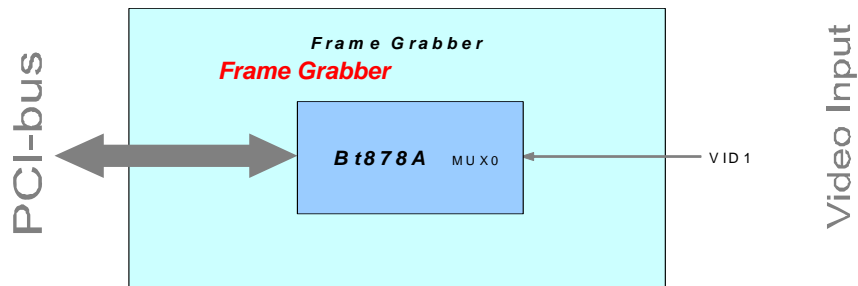


Figura 3-1: Blocco funzionale "Frame Grabber"

Per ogni dettaglio relativo al Video Decoder Bt878A, si fa riferimento ai seguenti documenti Conexant:

- Bt878A Datasheet (Conexant doc: DSH-200115-001)
- Bt878A Errata (Conexant doc: ERA-200113-001)

3.3 Real-Time I/O Subsystem

Questo sottosistema realizza tipicamente le funzioni di I/O (seriale e parallelo) che richiedono processing con requisiti real-time.

Esso consiste essenzialmente delle seguenti risorse:

- un blocco di processing costituito dall'insieme di due microcontrollers:
 - o un Motorola MC68EN360 in configurazione Master (CPU attiva),
 - o un MC68360 in configurazione Slave (CPU spenta);

Il processore Master gestisce, attraverso il proprio bus locale, tutte le risorse di I/O del processore Slave.

Il processore Master dispone di un banco di memoria SRAM da 4MByte (espandibile in fabbrica a 8 Mbyte) dedicata sia al codice eseguibile, che ai dati di sistema ed applicativi.

Il Master MC68EN360 dispone inoltre di una porta Ethernet (10 Mbps) collegata on-board allo Swtch Ethernet ESW0 della subLAN di Pista.

- un dispositivo FPGA (*Xilinx XC2S200*) che realizza :
 - le funzioni di interfaccia PCI Master/Target con la Main CPU (Intel Atom N270). Il meccanismo di interfaccia fa uso di una *Dual-Port RAM* (da 16 Kbyte) configurata all'interno del chip FPGA stesso.
 - altre funzioni di processing specifiche dell'applicazione (*Application Specific Programmable Logic - ASPL*), quali:
 - decodifica/codifica Manchester dei segnali delle porte seriali sincrone;
 - trattamento dei frames di comunicazione con le BOE RF, ai fini della funzionalità *bi-standard* (Telepass/P278);
 - decodifica dei segnali *Allarme BOA*.

Per ogni dettaglio funzionale di questo FPGA, si fa riferimento alla documentazione di progetto di Autostrade Tech.

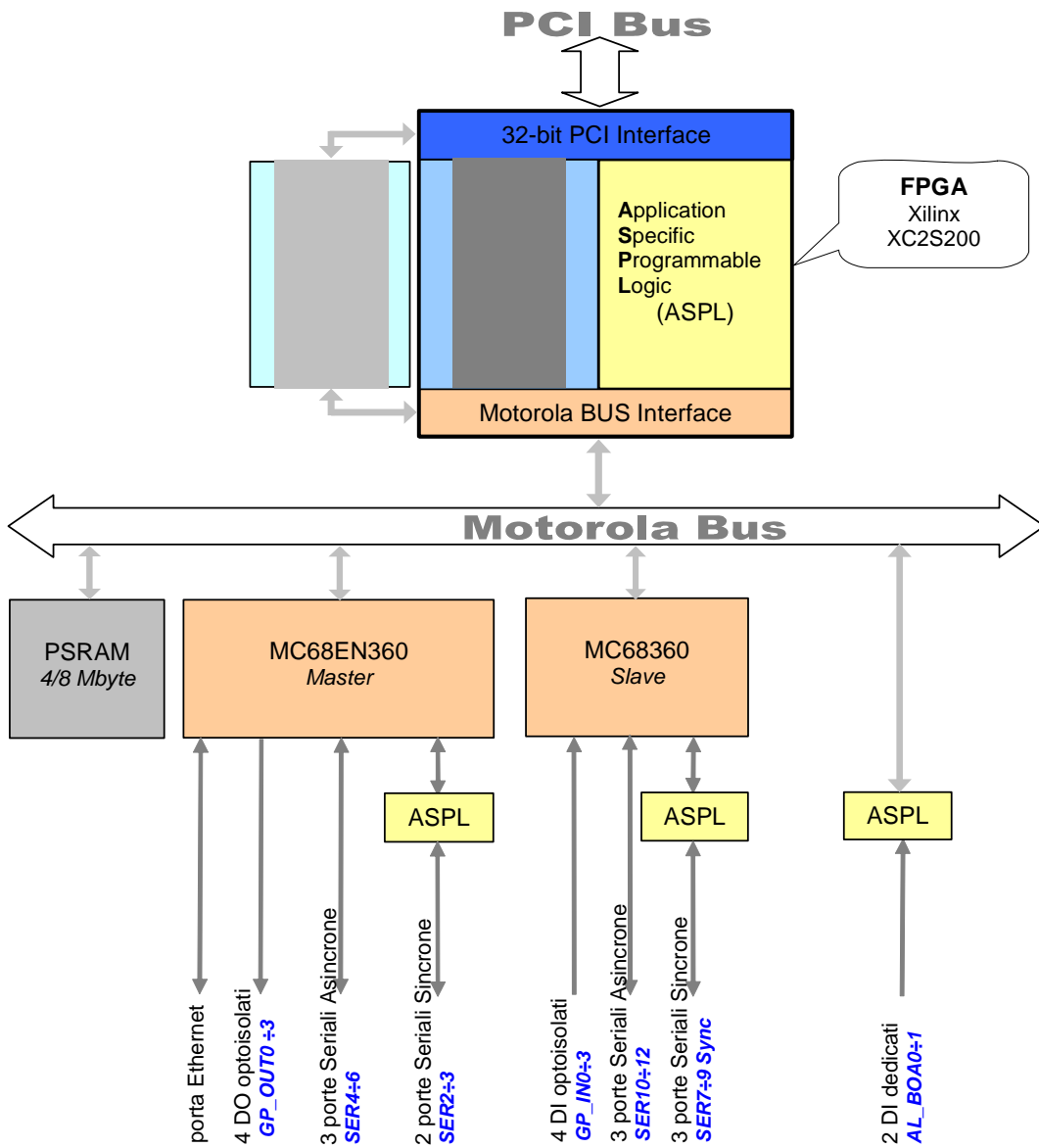


Figura 3-2: Schema a Blocchi del Real-Time I/O Subsystem

Per ogni dettaglio relativo al processore MC68360/ MC68EN360, si fa riferimento al seguente documento Motorola:

- MC68360UM Quad Integrated Communication Controller User's Manual

Per ogni dettaglio relativo alle funzionalità ASPL e firmware del Real-Time I/O Subsystem si fa riferimento alla documentazione specifica di Autostrade Tech.

3.3.1 Mappatura delle porte di I/O

Porta CONSOLE MC68360EN

<i>pin name MC68360</i>	<i>funzione di I/O del MC68360</i>	<i>segnale di I/O</i>	<i>Connettore di I/O</i>
SMRXD1	SMC1 Asynchronous Receive Data	SER6_RX	CON360 DB9P RS232
SMTXD1	SMC1 Asynchronous Transmit Data	SER6_TX	

Porte di I/O del MC68EN360 Master

<i>pin name 68EN360</i>	<i>funzione di I/O del MC68EN360</i>	<i>segnale di I/O</i>	<i>Connettore di I/O</i>	
			<i>pannello PIM-O</i>	<i>pannello PIM-E</i>
PB3	General Purpose Output	GP_OUT3		DIO DB9P <i>optoisolato</i>
PB2	General Purpose Output	GP_OUT2		DIO DB9P <i>optoisolato</i>
PB1	General Purpose Output	GP_OUT1	SP3 DIO DB9P <i>optoisolato</i>	SP3 DIO DB9P <i>optoisolato</i>
PB0	General Purpose Output	GP_OUT0	SP3 DIO DB9P <i>optoisolato</i>	SP3 DIO DB9P <i>optoisolato</i>
SMRXD2	SMC2 Asynchronous Receive Data	SER5_RX	MDIR RX testa ST <i>ottico</i>	MDIR DB9P RS232/422
SMTXD2	SMC2 Asynchronous Transmit Data	SER5_TX	MDIR TX testa ST <i>ottico</i>	
SCCRXD4	SCC4 Asynchronous Receive Data	SER4_RX		
SCCTXD4	SCC4 Asynchronous Transmit Data	SER4_TX		
SCCRXD3	SCC3 Synchronous Receive Data and Clock	SER3_RX	BOA3 RX testa ST <i>ottico</i>	BOA3 DB9P RS232/422
CLK5				
SCCTXD3	SCC3 Synchronous Transmit Data and Clock	SER3_TX (*)	BOA3 TX testa ST <i>ottico</i>	
CLK6				
SCCRXD2	SCC2 Synchronous Receive Data and Clock	SER2_RX (*)	BOA1 RX testa ST <i>ottico</i>	BOA1 DB9P RS232/422
CLK3				
SCCTXD2	SCC2 Synchronous Transmit Data and Clock	SER2_TX (*)	BOA1 TX testa ST <i>ottico</i>	
CLK4				

(*) : elaborati da logica ASPL

Porte di I/O dello MC68360 Slave

<i>pin name</i> <i>MC68360</i>	<i>funzione di I/O del MC68360</i>	<i>segnale di I/O</i>	<i>Connettore di I/O, Tipo connettore</i> <i>Tipo segnale</i>	
			<i>pannello PIM-O</i>	<i>pannello PIM-E</i>
PC3	General Purpose Input Output	GP_IN3		DIO DB9P <i>optoisolato</i>
PC2	General Purpose Input Output	GP_IN2		DIO DB9P <i>optoisolato</i>
PC1	General Purpose Input Output	GP_IN1	SP3 DIO DB9P <i>optoisolato</i>	SP3 DIO DB9P <i>optoisolato</i>
PC0	General Purpose Input Output	GP_IN0	SP3 DIO DB9P <i>optoisolato</i>	SP3 DIO DB9P <i>optoisolato</i>
SMRXD1	SMC1 Asynchronous Receive Data	SER12_RX	SP1 DB9P <i>RS232/422</i>	SP1 DB9P <i>RS232/422</i>
SMTXD1	SMC1 Asynchronous Transmit Data	SER12_TX		
SMRXD2	SMC2 Asynchronous Receive Data	SER11_RX	SP2 DB9P <i>RS232/422</i>	SP2 DB9P <i>RS232/422</i>
SMTXD2	SMC2 Asynchronous Transmit Data	SER11_TX		
SCCRXD4	SCC4 Asynchronous Receive Data	SER10_RX	SP3 DB9P <i>RS232/422</i>	SP3 DB9P <i>RS232/422</i>
SCCTXD4	SCC4 Asynchronous Transmit Data	SER10_TX		
SCCRXD3	SCC3 Synchronous Receive Data and Clock	SER9_RX (*)	S2A RX testa ST <i>ottico</i>	S2A DB9P <i>RS232/422</i>
CLK5				
SCCTXD3	SCC3 Synchronous Transmit Data and Clock	SER9_TX (*)	S2A TX testa ST <i>ottico</i>	S2A DB9P <i>RS232/422</i>
CLK6				
SCCRXD2	SCC2 Synchronous Receive Data and Clock	SER8_RX (*)	CTV3 RX testa ST <i>ottico</i>	CTV3 DB9P <i>RS232/422</i>
CLK3				
SCCTXD2	SCC2 Synchronous Transmit Data and Clock	SER8_TX (*)	CTV3 TX testa ST <i>ottico</i>	CTV3 DB9P <i>RS232/422</i>
CLK4				
SCCRXD1	SCC1 Synchronous Receive Data and Clock	SER7_RX (*)	CTV1 RX testa ST <i>ottico</i>	CTV1 DB9P <i>RS232/422</i>
CLK1				
SCCTXD1	SCC1 Synchronous Transmit Data and Clock	SER7_TX (*)	CTV1 TX testa ST <i>ottico</i>	CTV1 DB9P <i>RS232/422</i>
CLK2				

(*) : elaborati da logica ASPL

ALLARMI BOA (ingressi digitali del FPGA)

<i>pin name</i> <i>FPGA</i>	<i>funzione di I/O del FPGA</i>	<i>Connettore di I/O</i>	
		<i>pannello PIM-O</i>	<i>pannello PIM-E</i>
AL_BOA0	Ingresso Allarme BOA (*)	ALBOA1 testa ST <i>ottico</i>	BOA1 DB9P <i>RS422</i>
AL_BOA1	Ingresso Allarme BOA (*)	ALBOA3 testa ST <i>ottico</i>	BOA3 DB9P <i>RS422</i>

(*) : elaborati da logica ASPL

Porta COM2 del Main CPU Core

<i>pin name</i> <i>COM1</i>	<i>funzione di I/O del Chip SuperIO</i>	<i>segnale di I/O</i>	<i>Connettore di I/O</i>	
			<i>pannello PIM-O</i>	<i>pannello PIM-E</i>
RXD	Asynchronous Receive Data	COM1_RX	TES testa ST <i>ottico</i> e SP4 TES DB9P <i>RS232/422</i>	TES DB9P <i>RS232/422</i>
TXD	Asynchronous Transmit Data	COM1_TX		

3.4 Dispositivi Ausiliari

La board CRONOS dispone di alcuni dispositivi ausiliari connessi al bus LPC del ICH7-M, quindi accessibili dalla Main CPU Intel Atom N270.

Questi sono:

- Status Display
- DIP-Switch
- Watch Dog
- Temperatura Sensor
- Flash Memory
- Silicon Serial Number Interface

3.4.1 Status Display

Lo Status Display (posto sul pannello frontale) è costituito da una stringa di quattro caratteri alfanumerici e realizza due funzioni:

- *Visualizzazione porta 80* - mostra il contenuto esadecimale della porta 80. Questo modo viene impostato automaticamente dopo il RESET hardware dell'unità e può essere ripristinato in qualsiasi momento via software.
- *Display Alfanumerico* - permette di visualizzare una stringa di 4 caratteri alfanumerici, in funzione di particolari esigenze applicative. Questo modo può essere impostato solamente via software.

3.4.2 DIP-Switch

La board dispone di un DIP-Switch (8 switch). Gli switch SW2÷SW8 sono a disposizione per utilizzi applicativi. Lo switch SW1 è utilizzato per selezionare (al power-on/reset) il tipo di interfaccia della porta di rete verso la LAN di Stazione (vedi par. 3.4.2)

3.4.3 Sensore di Temperatura

La board dispone di un sensore di temperatura interna a due soglie (montato sul PCB). Le due soglie di temperatura sono fissate ai valori seguenti:

- temperatura di pre-allarme 56°C
- temperatura di allarme 68°C

3.4.4 Watch-Dog

La board dispone di un Watch-Dog hardware, costituito essenzialmente da un timer (presettabile via software) di tipo count-down che, se lasciato scadere, genera un RESET dell'unità stessa.

Il Watch-Dog può essere abilitato/disabilitato/presettato via software.

3.4.5 Flash Memory

La board dispone di un banco di memoria flash da 1 Mbyte così utilizzato:

- 256 Kbyte dedicati alla memorizzazione del il *bitstream* del chip FPGA ;
- 8 Kbyte dedicati a memorizzare l'*Authentication Code* ;
- il resto disponibile per altri impieghi.

3.4.6 Silicon Serial Number

La board dispone di un dispositivo "Silicon Serial Number" (DS2401 – Maxim).

4 Programmazione dei Dispositivi Ausiliari

Questo capitolo descrive in dettaglio la struttura e la modalità d'uso dei registri dei vari dispositivi ausiliari descritti nel capitolo precedente.

4.1 Struttura dei registri dei Dispositivi Ausiliari

Tutti i Dispositivi Ausiliari sono accessibili mediante due soli registri:

REG0 - Register Select registro a 8-bit il cui contenuto specifica il Registro cui si vuole accedere.

SEL7	SEL6	SEL5	SEL4	SEL3	SEL2	SEL1	SEL0	registro selezionato (che viene mappato su REG1)
0	0	0	0	0	0	0	0	CSR Control/Status Register
0	0	0	0	0	0	0	1	DDS Display Digit Select Register
0	0	0	0	0	0	1	0	DDD Display Digit Data Register
0	0	0	0	0	1	0	0	SWP Front panel DIP-switch Register
0	0	0	0	0	1	1	0	TMP Temperature Alarm Register
0	0	0	0	1	0	0	1	WDP Watch-Dog Counter Preset Register
0	0	0	0	1	0	1	0	FLL Flash Memory Low -byte Address Reg.
0	0	0	0	1	0	1	1	FLM Flash Memory Mid-byte Address Reg.
0	0	0	0	1	1	0	0	FLH Flash Memory High-byte Address Reg.
0	0	0	0	1	1	0	1	FLD Flash Memory Data Register
0	0	0	0	1	1	1	0	FLC Flash Memory Control Register
0	0	0	1	0	1	0	0	SND Silicon Serial Number Data Register
0	0	0	1	0	1	0	1	SNC Silicon Serial Number Control Register

Questo registro è di tipo *Write-only* ed è accessibile, con modalità “a byte”, all’indirizzo **0x0300** nello spazio di I/O della Main CPU.

Contenuto del registro dopo il Reset: 0x00

REG1- Device Register: registro a 8-bit su cui viene mappato il Registro specificato da REG0.

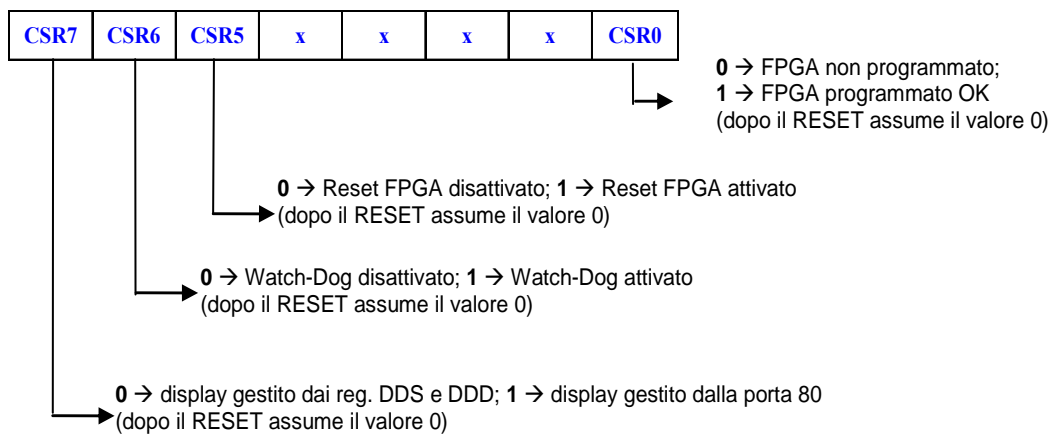
D7	D6	D5	D4	D3	D2	D1	D0
----	----	----	----	----	----	----	----

Questo registro è di tipo *Read/Write* ed è accessibile, con modalità “a byte”, all’indirizzo **0x0301** nello spazio di I/O della Main CPU.

Contenuto del registro dopo il Reset: 0x00

4.1.1 CSR - Control/Status Register (0x00)

Questo registro è di tipo *Read/Write* e viene utilizzato per il controllo dell'operazione dei Dispositivi Ausiliari, come specificato qui sotto.



Contenuto del registro dopo il Reset: 0b100000??

Per accedere a questo registro, procedere come segue:

- selezionare il registro, scrivendo il valore 0x00 in REG0,
- accedere al registro REG1, con l'operazione Read/Write (modalità "a byte") desiderata.

NOTA: Il bit CSR0 è *Read Only* ; CSR5/6/7 sono *Read/Write*.

4.1.2 **DDS - Display Digit Select Register (0x01)**

Questo registro è di tipo *Write Only* e viene utilizzato per selezionare il digit del Display Alfanumerico, su cui si vuole operare, come specificato qui di seguito.

x	x	x	x	x	x	DS1	DS0			
						0	0	seleziona il digit		□□□■
						0	1	seleziona il digit		□□■□
						1	0	seleziona il digit		□■□□
						1	1	seleziona il digit		■□□□

Contenuto del registro dopo il Reset: 0x00

Per accedere a questo registro, procedere come segue:

- selezionare il registro, scrivendo il valore 0x01 in REG0,
- accedere al registro REG1, con l'operazione Write (modalità "a byte") desiderata.

4.1.3 **DDD - Display Digit Data Register (0x02)**

Questo registro è di tipo *Write-Only* e viene utilizzato per modificare il digit selezionato del Display Alfanumerico, come specificato qui di seguito.

x	D6	D5	D4	D3	D2	D1	D0
---	----	----	----	----	----	----	----

Contenuto del registro dopo il Reset: 0x00

Per accedere a questo registro, procedere come segue:

- selezionare il registro, scrivendo il valore 0x02 in REG0,
- accedere al registro REG1, con l'operazione Write (modalità "a byte") desiderata.

La Tabella 4-1 fornisce la corrispondenza fra i valori (codici ASCII) da scrivere nel registro DDD ed i rispettivi caratteri che verranno visualizzati.

ASCII CODE				D0	0	1	0	1	0	1	0	1	0	1	0	1	0	1		
				D1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
				D2	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
				D3	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
D6	D5	D4	Hex	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
0	0	0	0	!	@	#	\$	%	&	'	()	*	+	,	-	.	:	;	
0	0	1	1	<	=	>	?	[\]	^	_	`	{		}	~			
0	1	0	2																	
0	1	1	3	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?	
1	0	0	4	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o		
1	0	1	5	p	q	r	s	t	u	v	w	x	y	z	[\]	^	_	
1	1	0	6		a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	
1	1	1	7	p	q	r	s	t	u	v	w	x	y	z	[]	^	_		

Tabella 4-1 Codifica caratteri ASCII del 4-digit Status Display

4.1.4 SWP - DIP-switch Register (0x04)

Questo registro è di tipo *Read Only* contiene lo stato del DIP-switch .

SW8	SW7	SW6	SW5	SW4	SW3	SW2	SW1
DS7	DS6	DS5	DS4	DS3	DS2	DS1	DS0

Contenuto del registro dopo il Reset: 0b???????

Per accedere a questo registro, procedere come segue:

- selezionare il registro, scrivendo il valore 0x04 in REG0,
- accedere al registro REG1, con l'operazione Read (modalità "a byte") desiderata.

NOTA: Lo switch SW1 è utilizzato per selezionare (al power-on/reset) il tipo di interfaccia della porta di rete verso la LAN di Stazione, come segue:

SW1 = OFF → Interfaccia elettrica - connettore attivo: RJ45
 SW1 = ON → Interfaccia ottica - connettore attivo: teste ottiche TX/RX ST

I LED indicatori di stato della connessione Ethernet integrati nel connettore RJ45 sono attivi indipendentemente dall'operazione effettuata.

4.1.5 TMP - Temperature Alarm Register (0x06)

Questo registro è di tipo *Read-Only* e viene utilizzato per leggere lo stato del sensore di temperatura posto internamente all'unità (sul PCB), come specificato qui di seguito.

x	x	x	x	x	x	TMP1	TMP0
---	---	---	---	---	---	------	------

0	0	temperatura interna Ti < Sp
0	1	temperatura interna Sp < Ti < Sa
1	0	<i>stato non possibile</i>
1	1	temperatura interna Ti > Sa

Sp = 56 °C : Temperatura di pre-allarme
Sa = 68 °C : Temperatura di allarme

Contenuto del registro dopo il Reset: 0b000000??

Per accedere a questo registro, procedere come segue:

- selezionare il registro, scrivendo il valore 0x06 in REG0,
- accedere al registro REG1, con l'operazione Read (modalità "a byte") desiderata.

4.1.6 WDP - Watch-Dog Counter Preset Register (0x09)

Questo registro è di tipo *Write Only* e viene utilizzato per presetare il tempo di espirazione (count-down) del Watch-Dog.

x	x	x	x	T3	T2	T1	T0	Tesp
0	0	0	0	0	0	0	0	0 ms
0	0	0	1	0	0	0	1	250 ms
0	0	1	0	0	0	1	0	500 ms
0	0	1	1	0	0	1	1	750 ms
0	1	0	0	0	1	0	0	1000 ms
0	1	0	1	0	1	0	1	1250 ms
0	1	1	0	0	1	1	0	1500 ms
0	1	1	1	0	1	1	1	1750 ms
1	0	0	0	1	0	0	0	2000 ms
1	0	0	1	1	0	0	1	2250 ms
1	0	1	0	1	0	1	0	2500 ms
1	0	1	1	1	0	1	1	2750 ms
1	1	0	0	1	1	0	0	3000 ms
1	1	0	1	1	1	0	1	3250 ms
1	1	1	0	1	1	1	0	3500 ms
1	1	1	1	1	1	1	1	3750 ms

Dopo il RESET hardware dell'unità, il contenuto di questo registro assume il valore 0b00001111.

Allo scadere del tempo di espirazione, se il Watch Dog è abilitato, viene generato un Reset hardware dell'intera board.



ATTENZIONE Impostando il valore di preset 0000, il Watch-Dog genererà un RESET hardware dell'unità nell'istante stesso in cui viene abilitato.

Qualsiasi accesso in Write a questo registro fa ripartire il count-down dal valore di preset programmato dalla Write stessa.

Per accedere a questo registro, procedere come segue:

- selezionare il registro, scrivendo il valore 0x09 in REG0,
- accedere al registro REG1, con l'operazione Write (modalità "a byte") desiderata.

4.1.7 ***FLL - Flash Memory Low-byte Address Register (0x10)***

Questo registro è di tipo *Write Only* contiene il low-byte dell'indirizzo:

A7	A6	A5	A4	A3	A2	A1	A0
----	----	----	----	----	----	----	----

Contenuto del registro dopo il Reset: 0x00

Per accedere a questo registro, procedere come segue:

- selezionare il registro, scrivendo il valore 0x10 in REG0,
- accedere al registro REG1, con l'operazione Write (modalità "a byte") desiderata.

4.1.8 ***FLM - Flash Memory Mid-byte Address Register (0x11)***

Questo registro è di tipo *Write Only* contiene il mid-byte dell'indirizzo:

A15	A14	A13	A12	A11	A10	A9	A8
-----	-----	-----	-----	-----	-----	----	----

Contenuto del registro dopo il Reset: 0x00

Per accedere a questo registro, procedere come segue:

- selezionare il registro, scrivendo il valore 0x11 in REG0,
- accedere al registro REG1, con l'operazione Write (modalità "a byte") desiderata.

4.1.9 ***FLH - Flash Memory High-byte Address Register (0x12)***

Questo registro è di tipo *Write Only* contiene l' high-byte dell'indirizzo ed il bit di selezione del tipo di accesso (Read/Write) che si andrà ad eseguire:

R/W	x	x	x	A19	A18	A17	A16
-----	---	---	---	-----	-----	-----	-----

R/W = 0 → Write

R/W = 1 → Read

Contenuto del registro dopo il Reset: 0x00

Per accedere a questo registro, procedere come segue:

- selezionare il registro, scrivendo il valore 0x12 in REG0,
- accedere al registro REG1, con l'operazione Write (modalità "a byte") desiderata.

4.1.10 **FLD - Flash Memory Data Register (0x13)**

Questo registro è di tipo *Read/Write* e contiene il dato letto/da scrivere dalla/nella Flash Memory:

D7	D6	D5	D4	D3	D2	D1	D0
----	----	----	----	----	----	----	----

Contenuto del registro dopo il Reset: 0x00

Per accedere a questo registro, procedere come segue:

- selezionare il registro, scrivendo il valore 13 in REG0,
- accedere al registro REG1, con l'operazione Read/ Write (modalità "a byte") desiderata.

4.1.11 **FLC - Flash Memory Status/Control Register (0x14)**

Questo registro è di tipo *Read/Write* e contiene il byte di stato/controllo per le operazioni di lettura/scrittura dalla/nella Flash Memory:

x	x	x	x	x	x	x	SC
---	---	---	---	---	---	---	----

STATUS (Read) :

- SC = 0 → operazione terminata
- SC = 1 → operazione in corso

CONTROL (Write) :

- SC = 0 → nessuna operazione
- SC = 1 → start operazione di lettura/scrittura

Contenuto del registro dopo il Reset: 0x00

Per accedere a questo registro, procedere come segue:

- selezionare il registro, scrivendo il valore 0x14 in REG0,
- accedere al registro REG1, con l'operazione Read/ Write (modalità "a byte") come segue:

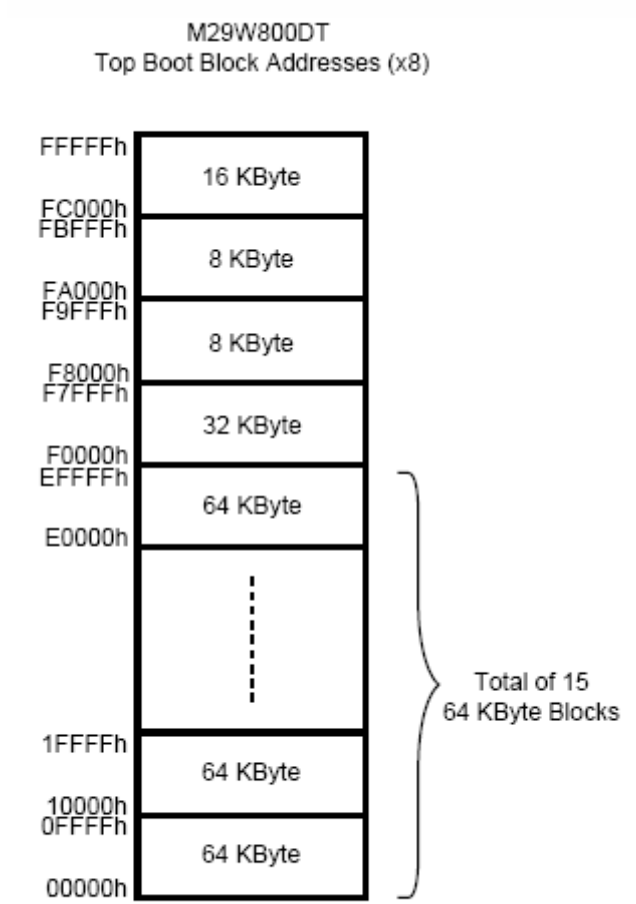
per eseguire la WRITE di un dato a 8-bit nella Flash Memory, eseguire la seguente sequenza:

- scrivere il low-byte dell'indirizzo della flash nel registro FLL (0x10)
- scrivere il mid-byte dell'indirizzo della flash nel registro FLM (0x11)
- scrivere il high-byte dell'indirizzo della flash nel registro FLH (0x12)
- e settare a 0 il bit 7 del registro stesso (operazione WRITE)
- scrivere il dato nel registro FLD (0x13)
- scrivere 0x01 nel registro FLC (0x14) – comando start operazione
- leggere ripetutamente FLC attendendo che ritorni 0x00 (operazione terminata)

per eseguire la READ di un dato a 8-bit dalla Flash Memory, eseguire la seguente sequenza:

- scrivere il low-byte dell'indirizzo della flash nel registro FLL (0x10)
- scrivere il mid-byte dell'indirizzo della flash nel registro FLM (0x11)
- scrivere il high-byte dell'indirizzo della flash nel registro FLH (0x12)
- e settare a 1 il bit 7 del registro stesso (operazione READ)
- scrivere 0x01 nel registro FLC (0x14) – comando start operazione
- leggere ripetutamente FLC attendendo che ritorni 0x00 (operazione terminata)
- leggere il dato nel registro FLD (0x13)

Il banco della Flash Memory è realizzato mediante il componente M29W800DT (1Mbyte) prodotta da ST, la cui organizzazione interna è quella indicata qui sotto.



L'allocazione del banco è come segue:

- 0x00000 ÷ 0x3FFFF : **FPGA bit stream** (256 Kbyte)
- 0x40000 ÷ 0x41FFF : **Authentication Code** (8 Kbyte)
- 0x42000 ÷ 0xFFFFF : disponibile per l'applicazione

Le operazioni di Erase, Read e Write sono così codificate:

Command	Length	Bus Write Operations											
		1st		2nd		3rd		4th		5th		6th	
		Addr	Data	Addr	Data	Addr	Data	Addr	Data	Addr	Data	Addr	Data
Read/Reset	1	X	F0										
	3	AAA	AA	555	55	X	F0						
Auto Select	3	AAA	AA	555	55	AAA	90						
Program	4	AAA	AA	555	55	AAA	A0	PA	PD				
Unlock Bypass	3	AAA	AA	555	55	AAA	20						
Unlock Bypass Program	2	X	A0	PA	PD								
Unlock Bypass Reset	2	X	90	X	00								
Chip Erase	6	AAA	AA	555	55	AAA	80	AAA	AA	555	55	AAA	10
Block Erase	6+	AAA	AA	555	55	AAA	80	AAA	AA	555	55	BA	30
Erase Suspend	1	X	B0										
Erase Resume	1	X	30										
Read CFI Query	1	AA	98										

Note: X Don't Care, PA Program Address, PD Program Data, BA Any address in the Block.

All values in the table are in hexadecimal.

The Command Interface only uses A-1, A0-A10 and DQ0-DQ7 to verify the commands; A11-A18, DQ8-DQ14 and DQ15 are Don't Care. DQ15A-1 is A-1 when BYTE is V_{IL} or DQ15 when BYTE is V_{IH} .

Per ogni ulteriore dettaglio, si fa riferimento al datasheet del componente M29W800DT.

4.1.12 *SND - “Silicon Serial Number” Data Register (0x20)*

Il Serial Number (memorizzato nel dispositivo DS2401) è costituito da una stringa di 64 bit così formattata:

D63		D0
8-bit CRC code	48-bit Serial number	8-bit family code

Al fine di semplificare le operazioni di lettura di questa stringa, è disponibile una apposita interfaccia mediante la quale è possibile leggere sequenzialmente i 64 bit della stringa partendo dal bit D0 (LSB).

Questa interfaccia consiste di due registri qui di seguito descritti.

Il registro SND è di tipo *Read-Only* e viene utilizzato per leggere sequenzialmente i bit del “Serial Number”.

x	x	x	x	x	x	x	Dn
---	---	---	---	---	---	---	----

Dn = bit *n* del Serial Number ($n = 0,1,\dots,63$)

Contenuto del registro dopo il Reset: 0x00

Per accedere a questo registro, procedere come segue:

- selezionare il registro, scrivendo il valore 0x20 in REG0,
- accedere al registro REG1, con l’operazione Read (modalità “a byte”) desiderata.

4.1.13 *SNC - “Silicon Serial Number” Status/Control Register (0x21)*

Questo registro è di tipo *Read/Write* e contiene il byte di stato/controllo per le operazioni di lettura sequenziale dei bit Dn:

x	x	x	x	x	x	CSR1	CSR0
---	---	---	---	---	---	------	------

STATUS (Read) :

CSR0 = 0 → lettura del bit Dn terminata
 CSR0 = 1 → lettura del bit Dn in corso

CSR1 = 0 → INIT terminato
 CSR1 = 1 → INIT in corso

CONTROL (Write) :

CSR0 = 0 → nessuna operazione
 CSR0 = 1 → start operazione di lettura del bit Dn

CSR1 = 0 → nessuna operazione
 CSR1 = 1 → start INIT

Contenuto del registro dopo il Reset: 0x00

Per accedere a questo registro, procedere come segue:

- selezionare il registro, scrivendo il valore 0x21 in REG0,
- accedere al registro REG1, con l’operazione Read/ Write (modalità “a byte”) come segue:

per effettuare la lettura del Serial Number, eseguire la seguente sequenza:

- scrivere 0x02 in SNC (setta a 1 CSR1 - **comando start INIT**)
- leggere ripetutamente SNC fino a quando il bit CSR1 torna a 0 (**INIT terminato**)

- ripetere per 64 volte la seguente sottosequenza:
 - o scrivere 0x01 in SNC (setta a 1 CSR0 - *comando start lettura bit Dn*).
 - o leggere ripetutamente SNC fino a quando il bit CSR0 torna a zero (*lettura del bit Dn terminata*).
 - o leggere SND; Dn = bit 0 di SND

I 64 bit verranno letti a partire da D0 (LSB).

5 Test dell'unità CRONOS

In questa sezione vengono riportate le informazioni specifiche per il test dell'unità CRONOS.

5.1 Console Seriale

La porta seriale asincrona RS232 COM1, con funzione di CONSOLE dell'unità CRONOS, è disponibile sul pannello frontale (connettore CONSOLE).

La porta seriale COM1 è impostata con i seguenti parametri di comunicazione:

- baud rate : **9600 baud**
- nr start bit : **1**
- nr data bit : **8**
- parity : **nessuna parità**
- nr stop bit : **1**
- flow control : **nessuno**

5.2 Dispositivi di Boot

L'unità CRONOS è equipaggiata con firmware BIOS Phoenix® EB(SF).005, nel quale è stato configurato il seguente ordine di default dei dispositivi di boot :

- USB
- IDE
- SATA

5.3 Procedura di test funzionale

5.3.1 *Materiale necessario*

Per l'esecuzione dei test funzionali dell'unità CRONOS, sono necessari i seguenti materiali:

- unità U0933 CRONOS
- Hard_Disk (oppure, una *usb_key*) con Sistema csLinux/U0933_CRONOS pre-installato
- monitor VGA, Tastiera USB e Mouse USB connessi all'unità CRONOS
- cavo UTP e convertitore Ethernet-FibraOttica
- telecamera PAL standard
- 1 connettore DB9 femmina con connessione "tappo" di loop-back (pin 2-3)
- 1 bretella di fibra ottica ST-ST

5.3.2 *Sistema di test*

Il sistema di test csLinux/U0933_CRONOS è costituito da:

- lilo boot-loader e memtest86+
- kernel Linux 2.6.30.2, compilato includendo tutti i drivers e moduli di base
- file system writeable in ramdisk, caricato da initrd
- tools di base implementati con BusyBox v1.14.2
- Bash
- drivers e applicativi specifici dell'unità U0933 CRONOS

I files costituenti il sistema sono caricati nella prima partizione del disco, con una occupazione di circa 12.4Mb.

5.3.3 *Boot del Sistema*

Il BIOS Phoenix® EB(SF).005 utilizza di default la Console Seriale (porta seriale COM1 settata a baud 115200, 8N1). Tuttavia, se durante il boot viene premuto qualche tasto sulla Tastiera, il BIOS automaticamente attiva questa come input principale, e invia l'output sulla porta VGA.

Con il tasto TAB premuto durante il boot, *lilo* consente all'operatore di selezionare tra

- *memtest* memory test
- *cslnx* sistema csLinux/U0933_CRONOS

5.3.4 *Memory Test*

Scopo del Memory Test è testare in modo funzionale la RAM DDR2 del U0933 utilizzando il programma Memtest86+.

La schermata in Figura 5-1 riporta le informazioni relative sia alla configurazione rilevata, che all'avanzamento delle fasi del test.

Il test prosegue indefinitamente; per arrestare il test occorre resettare l'unità mediante il pulsante Reset.

Il memtest può essere lanciato anche dopo che il sistema è stato avviato regolarmente con csLinux, nel qual caso si può riavviare con la sequenza

```
mount /dev/hda1 /boot lilo -C /boot/lilo.conf -R memtest
reboot
```

```

icav@sw05: ~
File Edit View Terminal Tabs Help
Memtest86+ v4.00 | Pass 28% #####
Atom (0.045) 1600 MHz | Test 3% #
L1 Cache: 24K 6250 MB/s | Test #6 [Moving inversions, 32 bit pattern]
L2 Cache: 512K 3721 MB/s | Testing: 176K - 1008M 1007M
L3 Cache: None | Pattern: 00000002
Memory : 1007M 1775 MB/s |-----
Chipset : Intel i945GME (ECC : Disabled) - FSB : 133 MHz - Type : DDR2
Settings: RAM : 222 MHz (DDR444) / CAS : 4-4-4-12 / Single Channel (64 bits)

WallTime  Cached  RsvdMem  MemMap  Cache  ECC  Test  Pass  Errors  ECC  Errs
-----
0:04:57  1007M      0K      e820    on  off  Std    0    0

Memory SPD Informations
-----
- Slot 0 : 1024 MB DDR2-333 - Micron Technology 8HTF12864HDY-66 G1

(ESC)Reboot (c)configuration (SP)scroll_lock (CR)scroll_unlock

```

Figura 5-1: Pagina di visualizzazione del Memory Test

N.B. I dati semi-oscurati si riferiscono a valori non correttamente calcolati dal programma Memtest86+, quindi non attendibili.

5.3.5 *csLinux login*

Nel sistema csLinux/U0933_CRONOS è impostato un login automatico con utente *root* ; non è richiesta alcuna password.



Figura 5-2: Pagina Welcome di csLinux

5.3.6 *Test funzionali*

In generale i test funzionali vengono eseguiti lanciando script/applicazioni specifici.

Di norma lanciando i test con l'opzione -h viene presentata la sintassi corretta del comando e, se applicabile, la lista di tutte le opzioni

5.3.6.1 **PCI-devices**

Lanciare il comando

[/opt/bin/test_pci.sh](#)

Lo script presenta una lista [bus/dev vendor_id/device_id descrizione] dei PCI-devices appartenente alla board, controllando che siano tutti presenti, evidenziando quelli mancanti e visualizzando anche quelli rilevati sul bus e ma non previsti (vedi Figura 5-3).

```

#
#
# test_pci.sh
::INFO test_pci.sh - ver 0.1
::INFO devices presenti
::INFO 0000 808627ac Intel 945GME GMCH Host bridge
::INFO 0010 808627ae Intel 945GME GMCH VGA controller
::INFO 0011 808627a6 Intel 945GME GMCH Display controller
::INFO 00d8 808627d8 Intel 82801 ICH7 Audio
::INFO 00e8 808627c8 Intel 82801 ICH7 UHCI controller #1
::INFO 00e9 808627c9 Intel 82801 ICH7 UHCI controller #2
::INFO 00ea 808627ca Intel 82801 ICH7 UHCI controller #3
::INFO 00eb 808627cb Intel 82801 ICH7 UHCI controller #4
::INFO 00ef 808627cc Intel 82801 ICH7 EHCI controller
::INFO 00f0 80862448 Intel 82801 ICH7 PCI bridge
::INFO 00f8 808627b9 Intel 82801 ICH7 ISA bridge
::INFO 00f9 808627df Intel 82801 ICH7 IDE controller
::INFO 00fa 808627c4 Intel 82801 ICH7 SATA controller
::INFO 00fb 808627da Intel 82801 ICH7 SMBus controller
::INFO 0110 109e036e Conexant Bt878 Video Capture
::INFO 0111 109e0878 Conexant Bt878 Audio Capture
::INFO 0118 80861209 Intel 8255x Ethernet controller
::INFO 0128 c1a0f1ca Xilinx FPGA
::INFO 0140 808627dc Intel 82801 ICH7 Ethernet controller
::STATUS OK
#
CTRL-A Z for help | 9600 8N1 | NOR | Minicom 2.3 | VT102 | Offline

```

Figura 5-3: Pagina Lista dei PCI Devices

5.3.6.2 Linee seriali

La line seriale COM1 viene verificata semplicemente collegando ad essa un terminale (o emulatore di terminale) ed eseguendo il login.

5.3.6.3 Porte Ethernet

La configurazione delle due porte Ethernet é la seguente

Porta	IP-address	connettori
eth0	192.168.2.193	LAN2
eth1	192.168.1.193	LAN1

Sulla rete devono essere attivi almeno due host: uno sulla LAN 192.168.1.x ed uno sulla LAN 192.168.2.x , sui quali siano attivi i servizi di *ping* e *ftp* (read/write per accesso anonimo).

Lanciare il comando

```
/opt/bin/test_net.sh host1 host2 ...hostN
```

Lo script

- verifica che gli host specificati siano raggiungibili
- genera un file di numeri casuali da 1Mb
- spedisce/rilegge il file suddetto su ciascun host, comparando la copia riletta con l'originale.

In Figura 5-4 è riportato un esempio, con l'indicazione dell'avanzamento del test.

Specificando nel comando l'opzione **-c N** il test viene ripetuto N volte (default 5), mentre specificando N=0 il test viene eseguito indefinitamente, e può essere interrotto con Control-C.

```

icav@sw05: ~
File Edit View Terminal Tabs Help
#
#
#
# test_net.sh 192.168.1.101 192.168.2.101
::INFO test_net.sh - ver 0.1 - 192.168.1.101 192.168.2.101
::INFO Host 192.168.1.101 -> eth1
::INFO 192.168.1.101/eth1 presente
::INFO Host 192.168.2.101 -> eth0
::INFO 192.168.2.101/eth0 presente
::INFO start transfer test
::INFO 192.168.1.101/eth1 pass #1
::INFO 192.168.2.101/eth0 pass #1
::INFO 192.168.1.101/eth1 pass #2
::INFO 192.168.2.101/eth0 pass #2
::INFO 192.168.1.101/eth1 pass #3
::INFO 192.168.2.101/eth0 pass #3
::INFO 192.168.1.101/eth1 pass #4
::INFO 192.168.2.101/eth0 pass #4
::INFO 192.168.1.101/eth1 pass #5
::INFO 192.168.2.101/eth0 pass #5
::INFO 192.168.1.101/eth1 done OK
::INFO 192.168.2.101/eth0 done OK
::STATUS OK
#
CTRL-A Z for help | 9600 8N1 | NOR | Minicom 2.3 | VT102 | Offline

```

Figura 5-4: Pagina Test Porte Ethernet

5.3.6.4 Interfacce ed Hard Disk IDE/SATA

Si assume che i dischi siano correttamente partizionati, con dimensioni sufficienti a contenere i files che verranno scritti (default 128Mb).

Lanciare il comando

```
/opt/bin/test_disk.sh part1 part2 ... partN
```

Lo script esegue sulle partizioni specificate sulla riga di comando (indicate come *hda1*, *sda1*, *sdb1*, ...), i test di lettura/scrittura.

NOTA: Il test crea nella partizioni di test files con nome *xdd.file*. Eventuali files omonimi presenti vengono sovrascritti senza nessun avviso.

L'output del programma usato (*xdd*) viene riformattato in modo opportuno e va confrontato con i tempi tipici (misurati sperimentalmente la prima volta).

In Figura 5-5 è riportato un esempio.

```

icav@sw05: ~
File Edit View Terminal Tabs Help
#
#
#
#
#
# test_disk.sh hda2 hda3
::INFO test_disk.sh - ver 0.1 - hda2 hda3
::INFO md5: 77e7d114ea1229bbcd206aedb2b14d26
::INFO md5 127: 77e7d114ea1229bbcd206aedb2b14d26
::INFO disk hda2 -> /mnt/disk/hda2
::INFO md5: 9383af41d364382d6a86e3cf95db45e7
::INFO md5 127: 9383af41d364382d6a86e3cf95db45e7
::INFO disk hda3 -> /mnt/disk/hda3

::INFO /opt/bin/xdd: Version 6.5.103106.000CDT - xdd - I/O Performance Inc. Cop6

Pass 0017 Op/AvgRate 001024/514.24 * 001024/508.40 *
::INFO disk= rate=140.897 IO/sec=1074.96
::INFO disk= rate=133.269 IO/sec=1016.76

::STATUS OK
#
#
CTRL-A Z for help | 9600 8N1 | NOR | Minicom 2.3 | VT102 | Offline

```

Figura 5-5: Pagina Test Hard-Disk

5.3.6.5 Porte USB

Tutte le porte USB, anche quelle utilizzate per KBD e Mouse, sono equivalenti.

La porta USB a cui fosse eventualmente connessa la Tastiera della Console è verificata automaticamente. Le altre porte USB vengono testate utilizzandole in modalità USB 2.0, inserendo nella/e porta/e stessa/e delle *usb-key* opportunamente partizionate e formattate. Alla prima partizione è stata associata una *label* unica (mediante i comandi [e2label/dosfslabel](#)), che serve a identificare le singole chiavette.

Lanciare il comando

```
/opt/bin/test_usb20.sh usbkey1 usbkey2 ... usbkeyN
```

Il test è analogo a quello relativo alle interfacce IDE/SATA, con la differenza che, in questo caso, si specificano le *label* anziché le partizioni (in quanto l'unità associata alle diverse *usb-key* non è predicibile).

```

icav@sw05: ~
File Edit View Terminal Tabs Help
#
#
#
#
#
#
#
#
#
#
#
# test_usb20.sh KEY1
::INFO test_usb20.sh - ver 0.1 - KEY1
::INFO USBkey KEY1 -> /dev/sdb1

::INFO /opt/bin/xdd: Version 6.5.103106.000CDT - xdd - I/O Performance Inc. Cop6

Pass 0101 0p/AvgRate 001024/1398.10 *
::INFO USBkey=KEY1 rate=572.601 IO/sec=4368.60
::STATUS OK
#
#
#
#
CTRL-A Z for help | 9600 8N1 | NOR | Minicom 2.3 | VT102 | Offline

```

Figura 5-6: Pagina Test Porte USB

5.3.6.6 DIP-Switch

Lo stato del DIP-switch CONFIG viene letto attraverso la PLD09008.

Il test consiste nel leggere la configurazione impostata sul DIP-switch.

Impostare in successione le diverse configurazioni del DIP-Switch e leggerle mediante il comando:

```
PLD09008-reg switch --read
```

che riporta lo stato degli switch (in codice esadecimale).

5.3.6.7 Status Display

Il display 4-caratteri STATUS viene controllato attraverso la PLD09008.

Il test consiste nel comandare la scrittura di un testo, verificando la corrispondente visualizzazione sul display.

Primo step consiste nell' abilitare il display, lanciando il comando

```
PLD09008 display --enable
```

poi é possibile comandare i vari test da visualizzare sul display (ovviamente, vengono mostrati solo 4 caratteri (gli ultimi del testo)

```
PLD09008 display --write --val=testo1  
PLD09008 display --write --val=testo2  
....  
PLD09008 display --write --val=testoN
```

5.3.6.8 Watch-dog

Il watch-dog viene controllato attraverso la PLD09008.

Il test consiste nell'abilitare il watch-dog, retriggerarlo qualche volta, verificando che non scada, e poi lasciarlo scadere verificando che l'unità venga resettata.

Lanciare i comandi

```
PLD09008-reg wdog --enable  
PLD09008-reg wdog --write --val=15  
PLD09008-reg wdog --enable  
for ((i=0; i<5; i++)); do  
    echo $i  
    sleep 1  
    PLD09008-reg wdog --write --val=15  
done  
echo "rebooting..."
```

5.3.6.9 Silicon Serial Number

Il Silicon Serial Number viene letto attraverso la PLD09008, col comando

```
PLD09008-reg ssn --read
```

5.3.6.10 Flash Memory

La flash memory é divisa logicamente in 5 zone, la prima dedicata al *bit-stream* del FPGA Xilinx, le altre a disposizione dell'utente.

E' possibile fare operazioni di lettura e scrittura di questa memoria mediante il seguente comando:

[/opt/bin/flshpld \[options\]](#)

options

-r, --read	lettura [def]
-w, --write	scrittura
-z, --zona=ZONA	zona logica (numero o nome)
-o, --offs=OFFS	offset da base-zona [def=0]
-s, --size=SIZE	dimensione dati [def=tutto]
-f, --file=FILE	file dati
-V	versione
-h	help

ZONA

0	BULK	960 Kb
1	Z32	32 Kb
2	Z8a	8 Kb
3	Z8b	8 Kb
4	Z16	16 Kb

5.3.6.11 Monitor VGA

Attivando la console grafica (su VGA), viene intrinsecamente verificata la funzionalità VGA.

Tuttavia, volendo eseguire un test in modalità grafica con risoluzione 1024x768, è necessario lanciare il comando

[/opt/bin/test_vga.sh](#)

che visualizza ciclicamente una serie di immagini.

5.3.6.12 Ingresso Video VID1

Collegare una telecamera PAL all'ingresso VID1 e lanciare il comando

[/opt/bin/test_video.sh](#)

Lo script acquisisce il flusso video dalla telecamera PAL e lo presenta sul monitor sotto forma di video-lento.

Specificando l'opzione -t N la visualizzazione termina dopo N secondi. Se N=0, la visualizzazione continua indefinitamente.

5.3.6.13 Real-Time I/O Subsystem – Porte seriali

I test consistono nelle prove di trasmissione in loopback sulle varie porte ottiche ed elettriche gestite dal Real-Time I/O Subsystem (MC68360).

Lanciare lo script

[/opt/bin/test_360.sh](#)

Lo script carica il firmware di diagnostica dedicato, poi presenta il menu di selezione dei test specifici.

Per testare una determinata porta, installare un connettore DB-9 "tappo" sulla porta per generare un loop TX-RX, quindi selezionare la voce corrispondente e controllare i messaggi informativi sull'andamento del test (vedi Figura 5-7).

```

icav@sw05: ~
File Edit View Terminal Tabs Help
-----
Menu diagnostica U0933 SerialIO/SB96ports
-----
0 - Exit
1 - Versione f/w diagnostico
2 - B0A1 - Porta SER2 (M SCC2)
3 - B0A3 - Porta SER3 (M SCC3)
4 - CTV1 - Porta SER7 (S SCC1)
5 - CTV3 - Porta SER8 (S SCC2)
6 - S2A - Porta SER9 (S SCC3)
7 - MDIR - Porta SER5 (M SMC2)
8 - SP1 - Porta SER12 (S SMC1)
9 - SP2 - Porta SER11 (S SMC2)
A - SP3 - Porta SER10 (S SCC4)
B - SP4 - Porta SER4 (M SCC4)
C - Cons - Porta SER6 (M SMC1)

->7
::INFO run360.sh - ver 0.1 - 1 2
U0933_CRONOS SerialIO/SB96ports - F/w DIAGNOSTICO v0.1
04 numMsgTx=7000 numMsgRx=7000 numErr=0

CTRL-A Z for help | 9600 8N1 | NOR | Minicom 2.3 | VT102 | Offline
  
```

Figura 5-7: Pagina Test Porte Seriali I/O

5.3.6.14 Real-Time I/O Subsystem – Porta Ethernet

Il Real-Time I/O Subsystem (MC68360) é connesso con porta Ethernet allo Switch ESW0, quindi anche alla porta eth1 della Main CPU (Intel Atom N270), sulla quale é attivo il Sistema Linux.

Il test consiste nello scambio di messaggi ICMP ECHO REQUEST/REPLY tra la Main CPU ed il microcontroller MC68EN360, avendo assegnato alla porta Ethernet di quest'ultimo l'IPaddress 192.168.2.36.

Lanciare il comando

`/opt/bin/ping360.sh`

che carica il firmware di diagnostica dedicato e poi visualizza lo scambio di messaggi con l'utilità *tcpdump*.

Il test prosegue finché non viene interrotto dall'operatore.

NOTA: É opportuno scollegare eventuali altre connessioni di rete dallo Switch ESW0, per evitare di contaminare l'output con messaggi di servizio estranei.


```
:::INFO ping360.sh - ver 0.1 -
U0933_CRONOS SerialI/O/SB96ports - F/w PING v0.1
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 96 bytes
00:06:08:03:06:00 (oui Unknown) > 00:94:30:a8:9c:ad (oui Unknown), ethertype IPv4 (0x0800),
length 74: 192.168.2.36 > 192.168.2.193: ICMP echo request, id 1, seq 2, length 40
  0x0000: 0094 30a8 9cad 0006 0803 0600 0800 4500
  0x0010: 003c 0001 4000 ff01 f589 c0a8 0224 c0a8
  0x0020: 02c1 0800 06fc 0001 0002 0001 0203 0405
  0x0030: 0607 0809 0a0b 0c0d 0e0f 1011 1213 1415
  0x0040: 1617 1819 1a1b 1c1d 1e1f
00:94:30:a8:9c:ad (oui Unknown) > 00:06:08:03:06:00 (oui Unknown), ethertype IPv4 (0x0800),
length 74: 192.168.2.193 > 192.168.2.36: ICMP echo reply, id 1, seq 2, length 40
  0x0000: 0006 0803 0600 0094 30a8 9cad 0800 4500
  0x0010: 003c 336c 0000 4001 c11f c0a8 02c1 c0a8
  0x0020: 0224 0000 0efc 0001 0002 0001 0203 0405
  0x0030: 0607 0809 0a0b 0c0d 0e0f 1011 1213 1415
  0x0040: 1617 1819 1a1b 1c1d 1e1f
00:06:08:03:06:00 (oui Unknown) > 00:94:30:a8:9c:ad (oui Unknown), ethertype IPv4 (0x0800),
length 74: 192.168.2.36 > 192.168.2.193: ICMP echo request, id 1, seq 3, length 40
  0x0000: 0094 30a8 9cad 0006 0803 0600 0800 4500
  0x0010: 003c 0002 4000 ff01 f588 c0a8 0224 c0a8
  0x0020: 02c1 0800 06fb 0001 0003 0001 0203 0405
  0x0030: 0607 0809 0a0b 0c0d 0e0f 1011 1213 1415
  0x0040: 1617 1819 1a1b 1c1d 1e1f
00:94:30:a8:9c:ad (oui Unknown) > 00:06:08:03:06:00 (oui Unknown), ethertype IPv4 (0x0800),
length 74: 192.168.2.193 > 192.168.2.36: ICMP echo reply, id 1, seq 3, length 40
  0x0000: 0006 0803 0600 0094 30a8 9cad 0800 4500
  0x0010: 003c 336d 0000 4001 c11e c0a8 02c1 c0a8
  0x0020: 0224 0000 0efb 0001 0003 0001 0203 0405
  0x0030: 0607 0809 0a0b 0c0d 0e0f 1011 1213 1415
  0x0040: 1617 1819 1a1b 1c1d 1e1f
4 packets captured
4 packets received by filter
0 packets dropped by kernel
```