

ASTROS
Software User Guide

**Note esplicative riguardo il funzionamento e la struttura hardware di ASTROS,
ad uso dei swaristi**

A. Memoria di Riordino

La Memoria di Riordino, denominata STAM (STrip Address Memory), e' una RAM da 256x10 bit il cui compito e' quello di fornire gli indirizzi al buffer di evento, in modo che, laddove e' necessario, viene fatto un Riordino di tutte le strip il cui indirizzo logico non corrisponde al proprio indirizzo fisico sul canale.

Pertanto tale memoria, è caricabile da Fastbus attraverso un'operazione di Block Transfer, con il pattern di indirizzi corretti al momento dell'inizializzazione del modulo, secondo lo schema di connessione delle strip del calorimetro, per ogni canale.

La STAM e' leggibile via Fastbus, in modo da consentire in qualsiasi momento una verifica del suo contenuto, qualora l'operazione di Riordino non dovesse svolgersi correttamente.

Un bit del CSR1800_H, viene usato per operare con le STAM:

FBOP_ON_MEM

questo ingresso viene tenuto ad uno ogniquale si vuole operare con lo spazio dati di memoria del modulo

La lunghezza della parola della STAM, deriva dal fatto che si dovranno indirizzare 256 locazioni di memoria del Buffer di Front-End, che si ha bisogno di un bit (il nono), per effettuare la soppressione di strip rumorose, ed infine che l'uso di un decimo bit, indicherà il termine della scrittura nella memoria d'acquisizione qualora la catena ad essa associata e' piu' corta di 256.

Sono previsti due modi di funzionamento dello scanner; un **modo Riordino** in cui il buffer d'evento (di seguito in dicato come SHIM), viene indirizzato tramite la STAM e un **modo senza Riordino**, in cui l'indirizzamento della SHIM, avviene tramite lo Strip Counter (vedi Fig.1).

In caso Riordino, una volta terminata la fase di indirizzamento e scrittura dell'evento nella SHIM, all'interno della STAM la fine catena viene individuata dalla presenza del decimo bit a uno.

Pertanto la presenza nella STAM di una parola del tipo 2XX, sta a significare procedura di esame del Marker (decimo bit a uno) e azzeramento della SHIM nelle locazioni il cui indirizzo e' indicato da XX (primi 8 bit della parola).

Questa procedura va avanti fino al raggiungimento della parola 2FF nella STAM.

Il nono bit della parola della STAM, viene usato per abbuiare delle strip o insieme di strip, che nel corso dell'acquisizione dovessero risultare rumorose. Questo bit deve essere posto a 1 dal processore, tramite una riscrittura Fastbus delle parole nella STAM, prima di effettuare un nuovo run.

Necessità di preparare un data-base, che tenga conto della lunghezza effettiva delle catene, canale per canale, che va caricato in fase di inizializzazione.

Tale operazione avverrà tramite un block transfer indirizzando in sequenza tutta la memoria presente in un modulo, per tutti i moduli presenti in un crate.

Una volta terminata l'acquisizione per ogni canale, indipendentemente dalla sua lunghezza, parte l'esame del Marker, che e' automatico qualora uso il Riordino, viceversa nel caso senza Riordino il check del Marker e' possibile se tratto il caso come un falso caso Riordino, cioe' caricando nella STAM gli indirizzi logici in sequenza e non gli indirizzi fisici reali e abilitando il Riordino.

Cosa identica vale per la soppressione delle strip rumorose, se voglio che questo funzioni anche nel caso senza Riordino, devo trattare questo caso come un finto Riordino.

E' possibile mascherare automaticamente una catena che risulti completamente a uno a seguito del primo bit bloccato a uno all'interno della catena, e che trasmette questo livello per tutti i 255 impulsi di clock, riempiendo cosi' il buffer di informazione inutile.

Ogni STAM presente nel modulo, è indirizzabile da Fastbus, sia in modo Random che in Block Transfer.

In totale si devono scrivere o leggere, 96 STAM per modulo, corrispondenti a 1 STAM x 24 canali x 4 sezioni di ciascun modulo.

Tutte le STAM di ogni sezione, occupano uno spazio dati contiguo, in modo da poter effettuare un' unica operazione di Block Transfer per il riempimento di tutta la memoria disponibile per sezione.

B. Memoria di acquisizione (SHift In Memory)

Questa è una Ram da 256x1 bit, che ha funzione di Buffer di Front-End e che viene indirizzata (vedi Fig.1) dalla STAM, nel modo Riordino, dallo Strip Counter, nel modo senza Riordino e dal Clustering Counter nella fase di riletture e clusterizzazione.

Nel caso di un'acquisizione con Riordino, una volta rivelata la fine della catena, se quest'ultima e' inferiore a 256, il resto della SHIM viene riempita con zero.

Vista l'uguale profondita' della SHIM rispetto alla STAM, è possibile leggere l'unico bit della SHIM, quando leggo la STAM via Fastbus e questo viene letto come bit <10> del bus DATA<12:0>.

C. Circuito di gestione del MARKER

Per ogni ASIC presente per canale sul modulo, si ha un Marker Register che contiene la parola di Marker che viene precaricata all'inizio dell'acquisizione. Non appena viene rilevata la fine della catena, parte il confronto tra il Marker precaricato in ognuno di questi registri e la parola di Marker che viene fatta shiftare lungo tutta la catena, alla fine dell'acquisizione dell'evento. Se il confronto ha successo, viene trascritto uno 0 sul 15-esimo bit della parola di ASTROS, ovvero un 1 qualora questo confronto dovesse fallire.

D. Registro di TEST

Questo e' uno shift register a 8 bit interno all' ASIC, che viene usato per testare il canale singolarmente e utile per aumentare la testabilita' stessa dell'ASIC durante la fase preliminare di verifica del prototipo. Esso è caricabile in parallelo sfruttando lo stesso bus dati usato per leggere/scrivere le memorie e le FIFO e la sua uscita condivide, con l'uscita delle catene, l'ingresso alla memoria d'acquisizione.

E. FIFO

Queste sono profonde 128x12 bit e due per canale per poter derandomizzare gli eventi, e la gestione di queste, avviene sotto il controllo del processore che le legge in Block Transfer mode via Fastbus.

La fase di acquisizione e di clusterizzazione di un evento impiega, alla frequenze di clock prescelte (1MHz e 10 MHz), circa 280us dal segnale di trigger, 256us per l'acquisizione e il resto per la clusterizzazione. In tutta questa fase l'Astros mantiene accesa la linea di BUSY.

A seguito del trigger, un modulo esterno che gestisce le informazioni relative al trigger (vedi "TSR Functional Specification" CERN/ECP 9 Oct.1992), invia una Service Request al processore degli Astros che, dopo un certo tempo, serve tale richiesta andando a leggere l'evento già pronto nelle FIFO (vedi Fig.2 e 4).

Prima di ogni lettura, il processore, dopo aver controllato che la linea di Busy di tutti gli Astros presenti nel crate sia spenta (questa è una linea su un bus esterno che collega in Wired-Or le linee di Busy di tutti gli Astros), effettua un'operazione di swap delle FIFO (ce ne sono infatti due per canale), abilitando opportunamente la FIFO da leggere, e conservando quest'informazione in un bit del CSR1800_H, in modo da sapere esattamente, quale evento il processore sta leggendo in quel momento.

Durante il tempo di lettura del processore di tutto il crate, il trigger è comunque abilitato, pertanto gli Astros sono in grado di accettare un secondo evento, di clusterizzarlo e porlo nella seconda FIFO disponibile, abilitata anch'essa preventivamente, per la scrittura.

Se ciò si verifica, ci sarà una seconda Service Request, che viene servita non appena il processore ha terminato di leggere il primo evento.

Durante tutta la fase di lettura, il processore risulta busy verso il TSR, il quale non accetta un ulteriore terzo evento, finché il processore non ha terminato di leggere il primo (vedi Fig.2 relativa al timing d'acquisizione e Fig 4 di sintesi delle operazioni di Data Acquisition).

Il servizio della seconda Service Request pendente, avviene con le stesse modalità della prima e così via.

Le FIFO, una volta scaricate del loro contenuto, emetteranno un segnale di EMPTY, che verrà usato per abilitare in lettura la FIFO del canale successivo.

Il bus d'uscita delle FIFO viene multiplexato con il bus dati delle STAM all'interno del ASIC e questo per poter gestire un numero inferiore di pin e inoltre su questo stesso bus, trova posto anche un tredicesimo bit (15-esimo nella parola di Astros), che dà l'informazione relativa all'operazione di check con la parola di Marker.

Alle FIFO di ogni sezione, è stato associato un Base Secondary Address pari 68000_H (Sezione 1 del modulo), 6C000_H (Sezione 2), 6D000_H (sezione 3) e 6E000_H (sezione 4).

Nel caso si voglia fare un Block Transfer di tutte le FIFO del modulo, l'indirizzo secondario di partenza è quello della sezione 0, nel caso si voglia fare un trasferimento di una sezione per volta, l'indirizzo da specificare è quello relativo alla sezione interessata. Viceversa qualora si voglia fare un trasferimento partendo da una sezione diversa dalla 0 e per un numero di parole superiore a quello relativo ad una sola sezione, si dovrà usare l'indirizzo della sezione 0 e attivare i bit di "Read Fifo" (vedi specifiche del CSR1800) corrispondenti alle sezioni da leggere e solo quelli. Il motivo di tutto ciò, discende dall'implementazione hardware del daisy-chain di lettura.

Comunque così come implementata, l'operazione di Block Transfer può partire indifferentemente da una qualsiasi di queste sezioni secondo quanto detto prima e procedere in sequenza con le sezioni di indirizzo più alto, lasciando le FIFO con l'indirizzo più basso rispetto a quello di partenza, non lette, questo risulta comodo nel caso si voglia lasciare la struttura hardware del calorimetro suddivisa in parte adronica e parte muonica, durante la lettura dell'evento.

Ad ogni sezione, è stato assegnato un "word counter" con il proprio indirizzo, il cui valore è indicato nella seguente tabella.

WCR Sezione 1	E1F00H
WCR Sezione 2	E1F01H
WCR Sezione 3	E1F02H
WCR Sezione 4	E1F03H

Questi registri verranno letti con un'operazione di Block Transfer, all'inizio di ogni trasferimento, per fissare il numero di parole da leggere nel trasferimento dati.

F. Struttura della parola nella FIFO: struttura della parola di Astros

Il significato dei bit all'interno della parola nella FIFO è mostrato di seguito:

1^o Byte bit da 0-7 → strip-address
2^o Byte bit da 8-15 → cluster size

gli altri due byte che compongono la parola di ASTROS, vengono costruiti sulla Mother Board secondo il solito significato:

3^o Byte bit da 16-23 → layer address
4^o Byte bit da 24-31 → module identification.

Per quanto riguarda il 2^o byte, c'è da notare che poichè i bit 12,13 e 14 non sono significativi ai fini della lettura del buffer d'uscita, questi devono essere mascherati a software in fase di lettura. Viceversa questi hanno significato in un'operazione di scrittura secondo il significato assegnato a questi bit, ai vari registri del modulo.

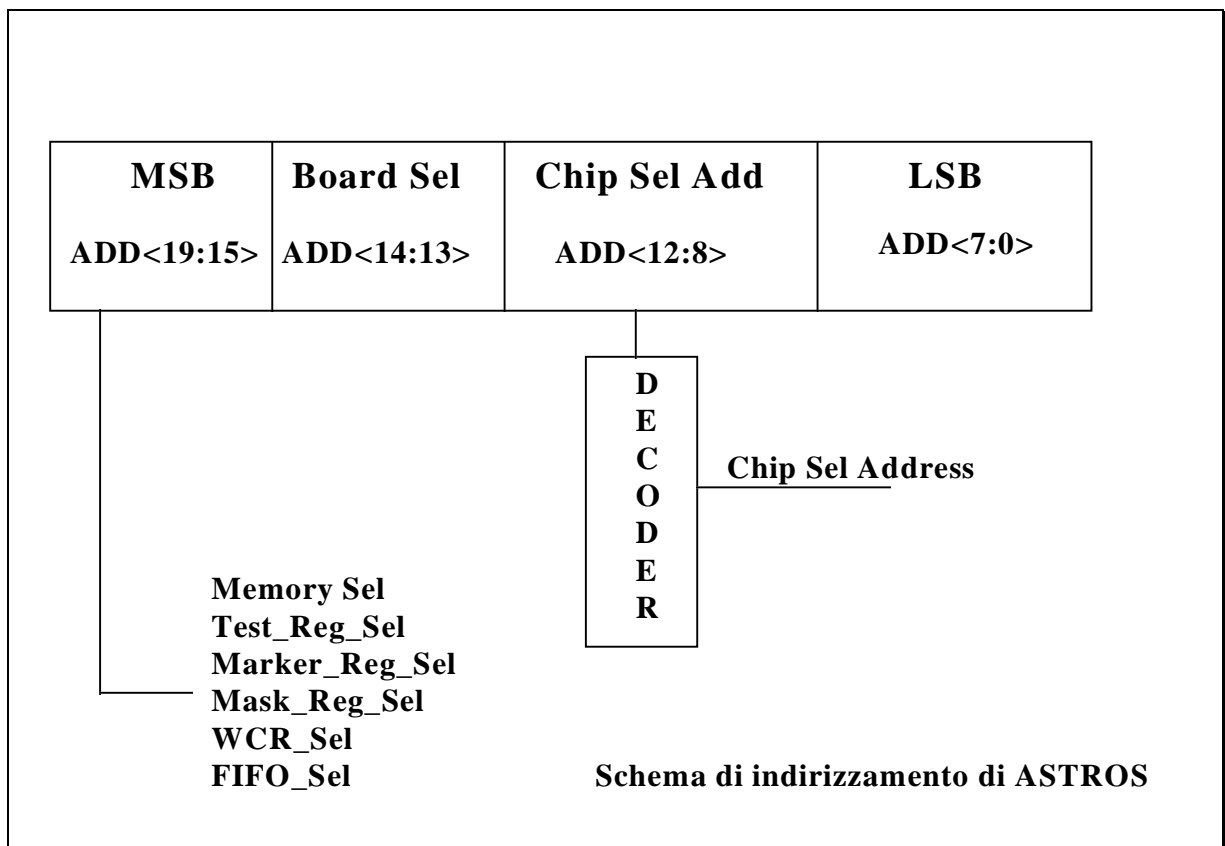
CAPITOLO II

REGISTRI E CONTATORI DEL MODULO ASTROS

A. Lo spazio indirizzi

Lo spazio indirizzi di ASTROS è uno spazio indirizzi a 20 bit, strutturato nella seguente maniera. I primi 8 bit, i meno significativi (ADD<7:0>), servono a indirizzare le 256 locazioni di memoria STAM usata per il riordino. I bit che vanno dall'ottavo al dodicesimo (ADD<12:8>), sono usati tramite un decoder, per indicare quale tra i 24 ASIC presenti su ogni sezione del modulo, voglio utilizzare.

I successivi due bit, (ADD<14:13>), vengono usati per selezionare quale sezione indirizzare e infine gli ultimi cinque bit, i più significativi (ADD<19:15>), sono usati per selezionare i vari registri interni agli ASIC, le FIFO, i Mask Register e i WCR, secondo lo schema di seguito indicato e la cui logica è stata posta all'interno di una PAL dell'ALTERA, del tipo E1800.



CAPITOLO II

Registri e Contatori del Modulo Astros

ADD<19:15>	<14:13>	<12:8>	<7:0>	REGISTRO	Campo Indirizzi
0 1 0 0 0	0 0	ASIC Sel	X X	STAM address 1^	40000-417FF
0 1 0 0 0	0 1	“	X X	STAM “ 2^	42000-437FF
0 1 0 0 0	1 0	“	X X	STAM “ 3^	44000-457FF
0 1 0 0 0	1 1	“	X X	STAM “ 4^	46000-477FF
0 1 0 0 1	0 0	“	0 0	TEST REG 1^	48000-49700
0 1 0 0 1	0 1	“	0 0	“ 2^	4A000-4B700
0 1 0 0 1	1 0	“	0 0	“ 3^	4C000-4D700
0 1 0 0 1	1 1	“	0 0	“ 4^	4E000-4F700
0 1 0 1 0	0 0	“	0 0	MARKER REG 1^	50000-51700
0 1 0 1 0	0 1	“	0 0	“ 2^	52000-53700
0 1 0 1 0	1 0	“	0 0	“ 3^	54000-55700
0 1 0 1 0	1 1	“	0 0	“ 4^	56000-57700
0 1 0 1 1	0 0	1 1 1 1 1	0 0	MASK REG 1^	59F00
0 1 0 1 1	0 1	“	0 0	“ 2^	5BF00
0 1 0 1 1	1 0	“	0 0	“ 3^	5DF00
0 1 0 1 1	1 1	“	0 0	“ 4^	5FF00
1 1 1 0 0	0 0	“	0 0	WCR_REG SEL 1^	E1F00
1 1 1 0 0	0 0	“	0 1	“ 2^	E1F01
1 1 1 0 0	0 0	“	0 2	“ 3^	E1F02
1 1 1 0 0	0 0	“	0 3	“ 4^	E1F03
0 1 1 0 1	0 0	0 0 0 0 0	0 0 0	FIFO 1^	68000-6B000
0 1 1 0 1	1 0	“	0 0	“ 2^	6C000-6CC00
0 1 1 0 1	1 0	1 0 0 0 0	0 0 0	“ 3^	6D000-6DC00
0 1 1 0 1	1 1	0 0 0 0 0	0 0 0	“ 4^	6E000-6EC00
0 0 0 0 0	0 0	1 1 0 0 0	0 0 0	GENERAL	1800
0 0 0 0 0	0 0	1 1 0 0 1	0 0 0	CHAIN LENGTH	1900
0 0 0 0 0	0 0	1 1 0 1 0	0 0 0	MARK PATTERN	1A00
0 0 0 0 0	0 0	1 1 0 1 1	0 0 0	IDENTIFICATION	1B00
0 0 0 0 0	0 0	1 1 1 0 0	0 0 0	DELAY SEZ 1	1C00
0 0 0 0 0	0 0	1 1 1 0 1	0 0 0	DELAY SEZ 2	1D00
0 0 0 0 0	0 0	1 1 1 1 0	0 0 0	DELAY SEZ 3	1E00
0 0 0 0 0	0 0	1 1 1 1 1	0 0 0	DELAY SEZ 4	1F00

B. Mark Register + Pattern Register

E' un registro a 8 bit circolare scrivibile e leggibile da Fastbus che, quando **in caso Test con Riordino**, contiene la parola di Test piu' quella di Marker, oppure solo il Marker quando in **modo acquisizione con Riordino** o ancora il Pattern di Test, quando **in modo acquisizione senza Riordino o in modo Test senza Riordino**. In ognuno di questi casi, bisogna far attenzione a scrivere nel registro in questione, sempre l'inverso del byte da impostare, in quanto sulle schedine della SGS, l'ingresso Test viene invertito.

In caso Riordino, la parola di Test scritta in tale registro, funge da Marker e pertanto viene trascritta anche nei Marker Register individuali di ogni canale d'acquisizione, presenti nell' ASIC.

Tale parola scritta in questo registro dalla duplice funzione, viene poi fatta circolare all'interno delle catene di shift register dell'apparato attraverso l'ingresso di Test delle schede e shiftata all'interno del Marker Register di canale, sul riconoscimento della fine della catena.

CAPITOLO II

Registri e Contatori del Modulo Astros

Pertanto volendo usare la modalita' Test con il Riordino, si deve aver cura di caricare le STAM in modo tale che queste tengano conto della lunghezza della catena sotto test, cosi come spiegato nel CAP.I paraf. A.

Questo stesso registro deve essere usato come Pattern Register durante il run nel modo senza Riordino, per trasferire uno 0 in coda a tutte le catene, ovvero per trasferire lungo la catena il Test Pattern, quando si opera in modo Test, ma senza Riordino.

E' prevista la possibilita' di usare una procedura di test che escluda il collegamento dell'Astros dalle catene e che consente di testare il modulo da solo anche se fisicamente collegato alle catene stesse. In tal caso un MUX sulla scheda provvede, tramite un opportuno bit di controllo all'interno del registro CSR1800_H (TEST_MOD, vedi dopo), a scollegare il PATTERN REGISTER dalle catene e a collegarlo direttamente all'ingresso delle SHIM di ogni canale del modulo in modo che il test di pattern alimenti direttamente l'ingresso di ogni canale dell'Astros.

C. Chain Length Register

E' un registro a **8 bit**, e serve per fissare, insieme al Contatore di Strip la durata dell'acquisizione. E' leggibile/scrivibile via Fastbus.

D. Mask Register

Questo e' un registro che ha un bit per canale e che maschera i canali corrispondenti a catene che durante l'acquisizione dovessero risultare rumorose. E' un registro sia scrivibile che leggibile da Fastbus. Tale registro puo' essere usato solo durante una normale acquisizione con le catene e in nessuna fase di test.

Si hanno quattro registri per modulo, uno per sezione secondo gli indirizzi indicati in tabella.

E. CSR ID

Questo e' un registro grande 6 bit, che contiene il codice identificativo del modulo e che assegna il modulo alla corrispondente parte dell'apparato. Ogni sezione all'interno di un modulo, viene a sua volta identificata attraverso altri due bit che insieme ai 6 bit precedenti, compongono il byte di identificazione del modulo. Questo byte viene letto come il piu' significativo, all'interno della parola a 32 bit letta dal Fastbus.

F. DELAY REGISTER

Questi registri sono previsti, per tener conto del differente ritardo con cui il segnale di "LOAD", arriva sul rivelatore, a seguito delle differenti lunghezze dei cavi, su cui viene trasmesso. Il loro intervallo temporale varia tra 0 e 100 ns, a step di 10 ns.

NOTA:

Sono previsti anche un CSR0 e un CSR1800_H "user defined" i cui bit vengono specificati di seguito e il CSR7, che specifichera' la "Classe di Broadcast" del modulo. Quest'ultimo viene implementato all'interno dell'interfaccia F/B, cosi' come vien fatto per l'NTA register. Fare riferimento alla nota esplicativa sull'interfaccia F/B montata sul modulo (FASP [3]).

G. Segnali di comunicazione tra ASTROS e il processore

H. BUSY (Out)

Questo segnale viene generato dal modulo e non viene trattato a livello di ASIC, e va attivo basso sull'inizio della scrittura dell'evento nella SHIM e viene riportato alto non appena conclusa la fase di clusterizzazione e scrittura dell'evento nella FIFO, circa 280us dopo il ricevimento del segnale di trigger. E' portato a uno tramite un bit di "reset" ad esso associato nel CSR1800_H e a clusterizzazione effettuata, e portato a zero all'inizio di ogni acquisizione dal segnale di TRIG (vedi dopo).

Questo segnale deve essere verificato dal processore, ogni qualvolta questi, dovendo servire una Service Request, inizia la lettura di un evento (vedi Fig.2 e 4).

Il segnale di BUSY inviato al processore, è formato dall'insieme dei BUSY di tutti i moduli presenti all'interno di un crate, attraverso un collegamento Wired-Or su di un bus esterno che collegherà tutti gli Astros di un intero crate con il processore, e la cui durata viene fissata dal tempo totale di scrittura evento e clusterizzazione.

Questo segnale alimenta un bit specifico del CSR1800_H (vedi dopo).

I. LOAD (In)

Il segnale di **LOAD** viene inviato alle catene di shift register presenti sul rivelatore e quindi è passante all'interno del modulo.

L. TRIGGER (In)

Il segnale di **TRIGGER** viene abilitato opportunamente sulla scheda del processore e inviato al modulo per far partire l'acquisizione.

M. EMPTY0, EMPTY1, EMPTY2, EMPTY3 (Out)

Questi quattro segnali, hanno il significato di "data ready" negato e hanno il compito di segnalare al processore l'avvenuta lettura di un evento all'interno di un modulo.

CAPITOLO III
CONTROL AND STATUS REGISTER DEL MODULO ASTROS

In totale abbiamo 11 registri di tipo CSR sul modulo a cui viene assegnato, nello spazio indirizzi riservato ai CSR, un indirizzo appropriato, secondo le specifiche del Fastbus.

I CSR, sono di seguito elencati:

- CSR0 User Defined Control Status Register
- CSR3 Logical Address Register,
- CSR7 registro che fissa la "Classe di Broadcast" del modulo.
- CSR1800_H User Defined Control Status Register
- CSR1900_H. Chain Length Register.
- CSR1A00_H. Marker+Pattern Register
- CSR1B00_H Registro di Identificazione della Sezione di ASTROS.
- CSR1C00_H Delay Register Sezione 1
- CSR1D00_H Delay Register Sezione 2
- CSR1E00_H Delay Register Sezione 3
- CSR1F00_H Delay Register Sezione 4

Specifiche dei bit del CSR0 di ASTROS

BIT DI CONTROLLO

- CLR (BIT 31)** Bit di clear generale del modulo (**attivo alto**). L'attivazione di questo bit determina l'azzeramento di tutti i contatori del modulo e della logica interna del ASIC e l'azzeramento di tutti i bit che compongono il CSR0, anche se poi ogni bit sarà del tipo "set and clear", secondo le specifiche FastBus, inoltre l'operazione di clear può essere effettuata oltre che a software attraverso questo bit, anche attraverso il segnale hardware di Power On Reset (POR), al momento dell'accensione del modulo. Questo azzeramento di tipo software, può effettuarsi anche tramite un'operazione Broadcast. Il CLR comunque non viene attivato qualora risulta attiva un'acquisizione (BUSY Astros attivo). Questo bit coincide con l'uscita denominata USR6_CLR_DATA della scheda FASP (FAStbus Slave Piggy-back sub-card), usata come interfaccia al Fastbus. (Vedi Ref. 4).
- CLR (BIT 30)** Reset generale del modulo, che genera un impulso positivo (**attivo alto**), pari a due periodi di clock, quando questo bit viene scritto. Coincide con l'uscita USR5_M_RESET, della scheda FASP.
- CSR0<15>** Module active bit e può venire solo letto. Coincide con l'uscita USR2_ACTIVE del FASP e viene usato per pilotare il led di indicazione di selezione del modulo.
- CSR0<16.31>** Module ID. Si possono solo leggere, ma sono scrivibili in fase di programmazione del CSR0, dovendo assegnare l'identificatore del modulo.

Specifiche dei bit del CSR1800_H di ASTROS

- FBOP_ONMEM** Bit per l'abilitazione delle operazioni Fastbus sulle STAM (**attivo alto**). Va mantenuto basso durante la lettura via Fastbus delle FIFO e durante l'acquisizione.
- REORDER** Indica l'operazione di Riordino effettuato su tutti i canali (**attivo alto**). Questo bit non viene resettato dal bit di CLR del CSR0, ma da un bit di "selective clear" nel CSR1800.
- FIFOSEL** Questo bit abilita, a secondo del suo stato la FIFO da scrivere sul successivo evento, all'interno del ASIC, pertanto restituisce in ogni istante, lo stato circa la FIFO indirizzata in quel momento. Inoltre questa operazione consente di porre sugli ingressi del Priority Encoder usato per la lettura del layer, citato nel paragrafo dedicato alla FIFO del capitolo I, le uscite di EMTPY delle FIFO da leggere. Il bit quando a zero, indica un'operazione di scrittura sulla FIFO0 e di lettura sulla FIFO1, viceversa se a uno. Non viene resettato dal bit di CLR del CSR0, ma da un bit di "selective clear" ad esso dedicato.
- PARALLEL_WRITE** Questo bit **se a uno logico**, consente l'indirizzamento unico e in parallelo sia del Marker Register che del Test Register, nel caso di una scrittura F/B per il loro setting, sia nel caso di un indirizzamento Geografico che Broadcast.
- TEST_SPLITTER** Questo bit corrisponde come funzionalita' al bit 9 del CSR0 del vecchio modulo, con il quale si mette in modalita' "Test" la nuova Splitter Board. Pertanto, dovendo corrispondere al bit succitato, ha sul connettore esterno del modulo, la posizione occupata dall'ex bit 9, corrispondente al vecchio "mask register load". Esso è **attivo alto**.
- TEST4_CHAN** Questo bit (**attivo alto**) se a uno, seleziona il modo test del ASIC.
Esso fa circolare una parola precaricata in uno Shift Register interno al ASIC, che alimenta l'ingresso delle SHIM e consente di verificare quindi il risultato ottenuto dopo l'acquisizione di tale parola di test.
Nel caso in cui questo bit venga selezionato, per effettuare il test descritto, si deve aver cura di caricare zero nel Mask Register.

- TEST_MOD** Questo bit (**attivo alto**) se a uno, seleziona il modo test di tutto il modulo. Questo consente di collegare gli ingressi SERIALIN del ASIC al Pattern Register di Astros, controllando un mux esterno (Fig.1). Quando questo bit e' a uno, il bit TEST4_CHAN, deve essere a zero.
- TEST_CHAIN** Questo bit (**attivo alto**) viene portato a uno se si vuole effettuare il test con le catene. Esso viene azzerato da CLR ed è di tipo read/write. Quando a uno gli altri due bit di TEST devono essere a zero.
- LAYER_COUNT** Questo bit (**attivo alto**) occupa sul connettore esterno del nuovo modulo la posizione occupata sul vecchio, dall'ex bit 4 del vecchio CSR10, corrispondente a "preset mask register". La nuova funzione implementata da questo bit sulla nuova Splitter Board, è quella di incrementare un contatore interno alla Splitter, per la selezione del Layer. Questo bit è di tipo write-only.
- TRIG_SOFT** Questo bit a solo scopo di test, serve a generare un impulso di trigger software, per il test dell'acquisizione.

I bit di controllo suddetti sono di tipo read/write secondo le specifiche dei CSR mostrate di seguito, tranne il bit di READ_FIFO e tutti i "selective clear", il bit di CLR compreso, che saranno solo di tipo write.

BIT DI CLEAR

Sono previsti per tutti i bit del CSR1800, in modo che tutti i bit siano del tipo "set and clear", secondo le specifiche FastBus.

BIT DI STATO

BUSY

Attivo basso, indica lo stato di Busy del modulo. Esso è portato a uno tramite un bit di "reset" ad esso associato e tramite la transizione 0-1 di STARCLUST a clusterizzazione effettuata, e azzerato dal segnale di TRIG.

EMPTY0, EMPTY1, EMPTY2, EMPTY3

Per ogni sezione di Astros, l'uscita Encoder Out (EO) piu' significativa, degli Encoder usati per abilitare a turno le FIFO in lettura, alimenta il bit di EMPTY corrispondente per sezione del CSR1800, indicando lo stato effettivo di "Empty" del modulo. Sull'impulso di CLEAR o all'accensione del modulo, questi bit vanno "on" stando a significare appunto lo stato di tutto vuoto del modulo.

CAPITOLO III

Control and Status Register del modulo ASTROS

Questi segnali, che sono **attivi alti**, corrispondono in un certo senso a un segnale di "data ready" negato e hanno il seguente significato. Se in un'operazione di Block Transfer in lettura delle FIFO, si verifica un "time-out" per qualche motivo legato a un malfunzionamento di un modulo, il processore, indirizzando geograficamente ogni modulo, può risalire all'offending module andando a leggere i bit di EMPTY e identificando il modulo incriminato, con il primo modulo nel crate, a partire da quello letto per primo nella catena di lettura, il cui bit di EMPTY e' rimasto settato a zero. E' comunque buona norma durante l'acquisizione, verificare che tali bit siano effettivamente attivati, altrimenti si potrà segnalare l'inconveniente con un messaggio d'errore a video.

Un'operazione di CLEAR effettuata in Broadcast, provvede a spegnere (in realta' a porre a uno logico), i bit di EMPTY di tutti i moduli nel crate.

Ricapitolando la situazione dei **bit del CSR0**, e' la seguente:

<u>Bit con funzioni di READ</u>		<u>Bit con funzioni di WRITE</u>
0	ERROR FLAG	SET ERROR FLAG
1		
14	PARITY ERROR	SET PARITY ERROR
16	MODULE ID (LSB)	CLEAR ERROR FLAG
17	"	
18	"	
19	"	
20	"	
21	"	
22	"	
23	"	
24	"	
25	"	
26	"	
27	"	
28	"	
29	"	
30	"	RESET MODULE
31	"	GENERAL RESET

La situazione dei bit del **CSR1800**, e' la seguente:

<u>Bit con funzioni di READ</u>		<u>Bit con funzioni di WRITE</u>
0	TEST4_CHAN	SET TEST4_CHAN
1	TEST_MOD	SET TEST_MOD
2	TEST_CHAIN	SET TEST_CHAIN
3	TEST_SPLITTER	SET TEST_SPLITTER
4	FB_OP_ONMEM	SET FB_OP_ONMEM
5	REORDER	SET REORDER
6	FIFOSEL	SET FIFOSEL
7	PAR_WRITE	SET PAR_WRITE
8	BUSY	SET_TRIG_SOFT
9	EMPTY0	READ_FIFO0
10	EMPTY1	READ_FIFO1
11	EMPTY2	READ_FIFO2
12	EMPTY3	READ_FIFO3
13		SET_LAYER_COUNT
16		CLEAR TEST4_CHAN
17		CLEAR TEST_MOD
18		CLEAR TEST_CHAIN
19		CLEAR TEST_SPLITTER
20		CLEAR FB_OP_ONMEM
21		CLEAR REORDER
22		CLEAR FIFOSEL
23		CLEAR PAR_WRITE
24		CLEAR BUSY

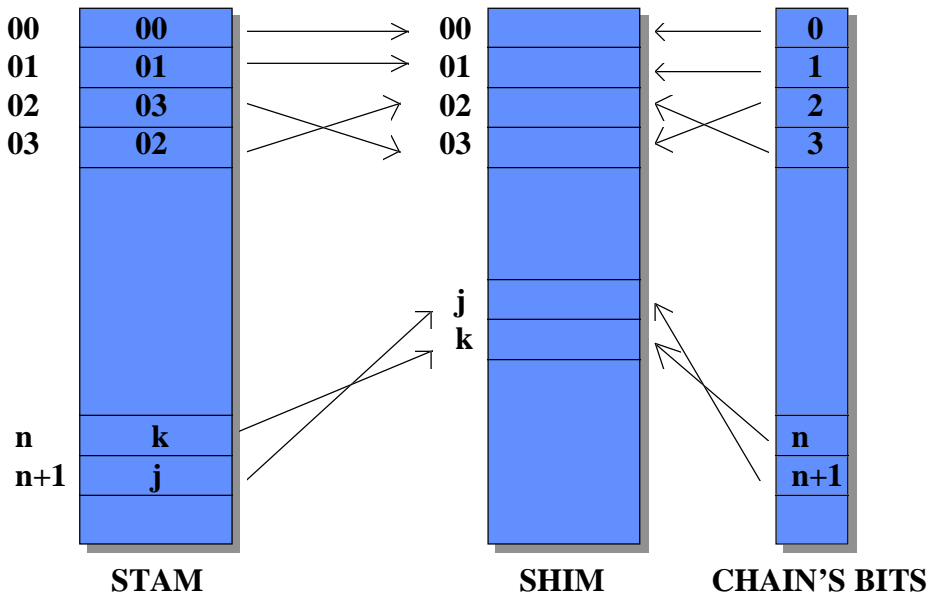


Fig. 1
Meccanismo di riordino delle parole, nella memoria di evento