

ISTITUTO NAZIONALE DI FISICA NUCLEARE

Sezione di Genova

INFN/TC-85/21  
8 Novembre 1985

E. Spinolo, E. Torcello e S. Vitale:  
REGISTRATORE PER L'ANALISI DIGITALE IN FORMA DI IMPULSI

Servizio Documentazione  
dei Laboratori Nazionali di Frascati

## REGISTRATORE PER L'ANALISI DIGITALE IN FORMA DI IMPULSI

E. Spinolo, E. Torcello e S. Vitale

L'analisi di parametri di forma di impulsi generati da dispositivi elettrici è generalmente un utile metodo per comprenderne a fondo il funzionamento.

In presenza di rumore questa analisi richiede una strumentazione adatta. Per questo abbiamo realizzato uno strumento che consente la registrazione digitale di transienti con distribuzione casuale nel tempo. Le successive elaborazioni consentono, con gli opportuni algoritmi, il filtraggio in linea, l'estrazione di segnale dal rumore, la determinazione automatica di forme medie o parametri di forma etc.

Lo strumento realizzato consiste in un registratore digitale di transienti ottimizzato per consentire l'analisi in forma degli impulsi di rivelatori nucleari di vario tipo. Esso campiona, digitalizza e memorizza il segnale della sorgente. La frequenza di campionamento è 100 MHz con clock interno o variabile con clock esterno (max 140 MHz). Ogni transiente viene digitalizzato con 64 campionamenti.

I dati relativi ai campionamenti che costituiscono parole a 7 bit (6 bit + 0.F.) vengono caricati su una struttura a shift registers che li differisce di 64 cicli di clock. Durante questo tempo la logica di trigger può decidere la registrazione del transiente. Il trigger determina la registrazione del transiente, costituito da una successione di 64 campionamenti (a 7 bit) alternativamente su due schede (A e B) di memoria (buffer) a 4 livelli. Mentre si registrano quattro transienti su una scheda, l'altra viene letta via CAMAC.

In tale modo si riducono drasticamente i tempi morti per transienti con distribuzione casuale nel tempo. Per una frequenza dei transienti  $\nu$  ed un tempo di lettura  $T_0$

per transiente le perdite di conteggio sono per  $\nu T_0 = 1$  10% e per  $\nu T_0 = 0.7$  sono minori dell'1%.

In questa prima fase la lettura, l'elaborazione ed il filtraggio avvengono attraverso un collegamento in linea via CAMAC con un calcolatore LSI 11/23.

In una seconda versione attualmente in fase di preparazione lo strumento sarà realizzato su BUS WME e consentirà anche elaborazioni in tempo reale.

Passiamo ora ad un'analisi più dettagliata delle 5 schede di cui è composto il registratore e dei necessari comandi CAMAC.

### F.A.D.C. UNIT

L'unità funziona con un clock a 100 MHz che con 3 bistabili va a formare 4 clock a 25 MHz sfasati in sequenza di 90°; nella descrizione che segue questi clock verranno chiamati rispettivamente §1, §2, §3, §4. Il segnale analogico entra in parallelo su 4 F.A.D.C. SDA6020 che vengono sequenzialmente strobatati da §1, §2, §3 e §4.

Ogni FADC va poi a 7 shift register a 16 bit e ogni gruppo di 7 shift avranno come ck il § del corrispondente FADC. In questo modo nell'insieme degli SHIFT si avrà sempre 64 parole a 7 bit corrispondenti agli ultimi 640 nS che è il tempo che si è deciso prendere in esame per ogni transiente.

### Trigger (TTL)

Il trigger può venire dall'esterno o essere generato internamente da un comparatore a soglia variabile. Il trigger dà inizio con §1 all'incremento di un contatore le cui uscite inviate ad un identity comparator quando raggiungono il valore impostato sull'apposito dip switch generano il trigger ritardato per le schede di memoria. Il ritardo tra il trigger In. e il trigger inviato alle schede di memoria serve per fare in modo che l'inizio della memorizzazione del transiente avvenga quando in uscita dagli SHIFT cominciano ad aversi i dati relativi al transiente che ha generato il trigger.

### MEMORY UNIT

Ogni scheda di memoria è composta da 4 blocchi uguali più la logica di controllo per la scrittura. In ognuno dei 4 blocchi viene memorizzato 1 transiente (64 parole a 7 bit). La lettura viene invece fatta attraverso le apposite schede.

### Trigger

I trigger ritardati provenienti dalla FADC UNIT vanno ad uno SHIFT register a 4 bit che avanzando di un passo ad ogni trigger va ad abilitare il relativo blocco di

memoria.

Con la fine della scrittura del quarto blocco la scheda invia un segnale di fine scrittura alla rispettiva read unit che conseguentemente settando il LAM register segnala al calcolatore di essere pronta per la lettura. Il segnale di fine scrittura viene pure inviato alla FADC UNIT che commutando l'apposita elettronica invierà i prossimi 4 trigger ritardati all'altra memory unit. In questo modo le due memory unit vengono alternativamente scritte e lette. Ovviamente ogni Memory Unit dopo la fine scrittura è insensibile ad eventuali trigger in ingresso fino a che non avrà ricevuto dalla rispettiva read unit un segnale di fine lettura.

### READ UNIT

Ogni scheda di lettura è formata da 4 blocchi uguali composti ognuno da 4 drivers, 3 State che si affacciano sulle read lines più il registro LAM, l'elettronica di controllo e quella per la decodifica dei comandi CAMAC. Ogni scheda di lettura è collegata alla rispettiva scheda di memoria attraverso 3 flat cable che portano i dati e i vari segnali di abilitazione e controllo.

### CAMAC

#### Clear

Deve essere dato da CAMAC dopo l'accensione per predisporre l'elettronica alla acquisizione dei transienti.

#### LAM

Il registro LAM posto sulle "read unit" viene settato alla fine della scrittura dell'ultimo transiente (max 4).

Il LAM rimane però mascherato e andrà sulla relative linea LAM solo dopo che CAMAC avrà generato la funzione F26 x S2. Il LAM verrà tolto alla fine della lettura.

#### Q response

E' messo fuori dalla stazione di lettura indirizzata in risposta alle funzioni:

A1 + A2 + A3 + A4 + A5 + A6 + A7 + A8 x F2 x LAM STATUS

F8 x LAM REQ

F9, F24, F26 direttamente.

#### X response

E' messo fuori dalla stazione di lettura indirizzata in risposta alle funzioni:

A1 + A2 + A3 + A4 + A5 + A6 + A7 + A8 x F2

F8, F9, F24, F26 direttamente.

Read lines

In ognuna delle 2 schede di memoria possono venire memorizzati 4 transienti corrispondenti ognuno a 64 conversioni e cioè 64 parole da 7 bit. Essendo le read lines disponibili 16 le parole verranno lette a coppie di 2 per cui la lettura di ognuna dei 4 blocchi di memoria corrispondenti ai 4 transienti registrati richiederà 32 cicli CAMAC. Conseguentemente la lettura completa di una scheda di memoria richiederà 128 cicli CAMAC.

LETTURA SCHEDA MEMORIA (1° blocco)

Funzioni CAMAC	Read lines	Ordine in cui sono avvenute le registrazioni	Fase di clock (§) in registrazione
F2xA1	1-7	1	§1
	9-16	2	§2
F2xA2	1-7	3	§3
	9-16	4	§4
F2xA1	1-7	5	§1
	9-16	6	§2
F2xA2	1-7	7	§3
	9-16	8	§4
≈ ≈ ≈ ≈ ≈			
F2xA1	1-7	57	§1
	9-16	58	§2
F2xA2	1-7	59	§3
	9-16	60	§4
F2xA1	1-7	61	§1
	9-16	62	§2
F2xA2	1-7	63	§3
	9-16	64	§4

Dopo le 32 letture del primo blocco (1° transiente registrato) si procederà alla lettura degli altri 3 blocchi cambiando semplicemente i sotto indirizzi

Blocco 2 = F2 x A3 / F2 x A4

Blocco 3 = F2 x A5 / F2 x A6

Blocco 4 = F2 x A7 / F2 x A8.

A questo punto la scheda letta è pronta per un nuovo ciclo di acquisizione, mentre si potrà iniziare l'operazione di lettura dell'altra scheda di memoria. Si inizierà

ovviamente con F8 per testare il LAM e se presente si procederà come descritto.

Se con F8 non si verifica la presenza del LAM significa che la registrazione dei 4 transienti sulla scheda esaminata non è ancora terminata. Si dovrà quindi procedere con l'invio periodico di F8 sino a verificare la presenza del LAM.

#### Funzioni CAMAC

A1 + A2 + A3 + A4 + A5 + A6 + A7 + A8 x F2 = Lettura

F8 = Test LAM status

F9 = Clear

F24 = LAM mask

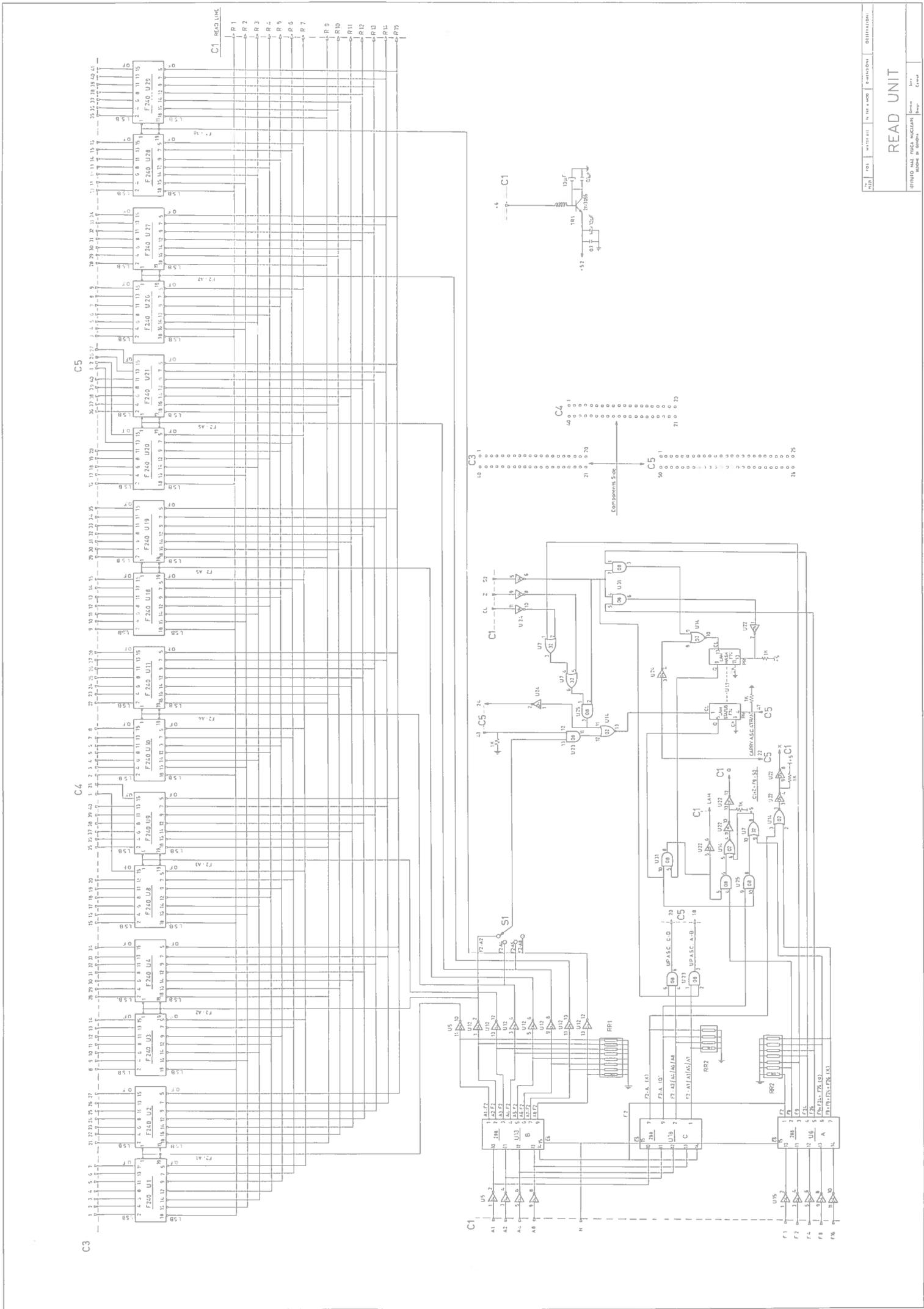
F26 = LAM unmask.

Il LAM è mascherato anche da F9, C, Z.









READ UNIT