

SECONDA UNIVERSITA' DEGLI STUDI DI NAPOLI



FACOLTA' DI INGEGNERIA

Dipartimento di Ingegneria dell'Informazione
Corso di Laurea Triennale in Ingegneria Elettronica

TESI DI LAUREA IN INGEGNERIA ELETTRONICA

SISTEMA DI ACQUISIZIONE DATI RADAR SORA

Relatore:

Ch.mo Prof. Luigi ZENI

Candidato:

Antonio Merola

Matricola 832/633

Correlatore:

Ing. Giovanni Alberti

Anno Accademico 2007/2008

*Alle persone a me care,
quelle che hanno sempre creduto in me,
quelle che mi hanno spronato a fare sempre di più*

Un cammino finisce un altro comincia

*Un ringraziamento va al mio correlatore Giovanni Alberti,
a Gianfranco Palmese e a Luca Ciofaniello,
splendidi tutor da cui apprendere la loro preziosa esperienza.*

*Non posso non ringraziare tutte le altre persone di cui si compone il
Co.Ri.S.T.A., che con la loro cortese disponibilità e con la giusta simpatia
hanno reso piacevoli i giorni trascorsi con loro.*

Infine ringrazio il mio relatore Luigi Zeni.

INDICE

INTRODUZIONE	6
1 ARCHITETTURA DEL PAYLOAD DI SORA	10
1.1 Acronimi e Definizioni	10
1.2 Architettura del Sistema.....	11
1.2.1 Descrizione Generale.....	11
1.2.2 Stati degli Strumenti e Modalità Operative	13
1.3 Interfacce Gondola-Radar.....	17
1.3.1 Interfacce Elettriche.....	17
1.3.1.1 Interfacce TC/TM.....	17
1.3.1.2 Definizione Protocollo SER 1.....	18
1.3.1.3 Definizione Protocollo SER 2.....	20
1.3.2 Interfacce Meccaniche	24
1.4 Sottosistemi del Radar	29
1.4.1 Computer di Bordo	29
1.4.1.1 Descrizione	29
1.4.1.2 Caratteristiche	31
1.4.1.3 Interfacce	32
1.4.2 Sw di Bordo	33
1.4.2.1 Descrizione	33
1.4.2.2 Caratteristiche	34
1.4.2.3 Interfacce	36
1.4.3 Unità Digitale.....	37
1.4.3.1 Descrizione	37
1.4.3.2 Caratteristiche	37
1.4.3.3 Interfacce	40
1.4.4 Unità RF.....	40
1.4.4.1 FGU	42
1.4.4.2 Trasmitter Chain.....	42
1.4.4.3 Receiver Chain.....	43
1.4.4.4 Duplexer.....	43
1.4.4.5 Antenna	44
1.5 Considerazioni ambientali	44
2 CENNI SUL TELERILEVAMENTO	47
2.1 Concetti Generali	47
2.2 Sensori RADAR	48
2.3 I chirp RADAR.....	52
3 CENNI SUL SOTTOCAMPIONAMENTO	58
3.1 Introduzione	58
3.2 Aliasing	59
3.3 Jitter	60
3.4 SFDR	66
3.5 Conversione di frequenza	67
4 DEMO BOARD DC996A E CONVERTITORE ANALOGICO DIGITALE LTC2208-14.....	71
Specifiche.....	72

4.1	Descrizione	73
4.2	Diagramma Temporale	74
4.3	Funzionamento.....	76
4.3.1	Prestazioni Dinamiche	76
4.3.1.1	<i>Signal-to-Noise Plus Distortion Ratio</i>	76
4.3.1.2	<i>Signal-to-Noise Ratio</i>	76
4.3.1.3	<i>Total Harmonic Distortion</i>	76
4.3.1.4	<i>Intermodulation Distortion</i>	76
4.3.1.5	<i>Spurious Free Dynamic Range (SFDR)</i>	77
4.3.1.6	<i>Full Power Bandwidth</i>	77
4.3.1.7	<i>Aperture Delay Time</i>	77
4.3.1.8	<i>Aperture Delay Jitter</i>	77
4.3.2	Funzionamento della Conversione.....	78
4.3.3	Sample/Hold ed Unità d'Ingresso	79
4.3.3.1	<i>Funzionamento del Sample/Hold</i>	79
4.3.3.2	<i>Polarizzazione di Modo Comune</i>	81
4.3.3.3	<i>Impedenza dell'Unità d'Ingresso</i>	81
4.3.4	Circuiti d'Ingresso	82
4.3.4.1	<i>Filtro d'Ingresso</i>	82
4.3.4.2	<i>Transformer Coupled Circuits</i>	82
4.3.4.3	<i>Circuiti di Accoppiamento Diretto</i>	84
4.3.4.4	<i>Funzionamento del Riferimento in tensione</i>	85
4.3.4.5	<i>Pin PGA</i>	86
4.3.4.6	<i>Pilotaggio degli Ingressi Codificati</i>	86
4.3.4.7	<i>Massimo e Minimo Rapporto di Codifica</i>	89
4.3.5	Uscite Digitali	89
4.3.5.1	<i>Modalità delle Uscite Digitali</i>	89
4.3.5.2	<i>Buffer delle Uscite Digitali (Modi CMOS)</i>	90
4.3.5.3	<i>Buffer delle Uscite Digitali (Modi LVDS)</i>	91
4.3.5.4	<i>Formato dei Dati</i>	92
4.3.5.5	<i>Bit di Overflow</i>	93
4.3.5.6	<i>Clock d'Uscita</i>	93
4.3.5.7	<i>Digital Output Randomizer</i>	93
4.3.5.8	<i>Output Driver Power</i>	94
4.3.5.9	<i>Dither Interno</i>	95
4.3.5.10	<i>Trasferimento di Calore</i>	96
5	ELABORAZIONE ED ANALISI DEI DATI ACQUISITI.....	97
5.1	Acquisizione dati in Complemento a 2.....	97
5.2	Data Formatting	98
5.3	Sora_proc	100
5.3.1	Funzionamento del sora_proc.m.....	101
5.3.2	Listato sora_proc.m	104
5.3.3	Acquisizione in modalità autocheck	108
	CONCLUSIONI	117
	BIBLIOGRAFIA	118

INTRODUZIONE

Il progetto SORA (SOunding RAdar) mira alla realizzazione di una serie di sperimentazioni di supporto alla comunità scientifica italiana nell'ambito della ricerca sui radar a penetrazione da utilizzare in missioni spaziali dedicate all'esplorazione del sistema planetario.

Per lo svolgimento delle attività, l'Agenzia Spaziale Italiana (ASI) e l'agenzia norvegese Andøya Rocket Range (ARR), hanno firmato un accordo per lo sviluppo e la gestione di un sito di lancio per missioni stratosferiche polari presso le isole Svalbard. Il sito, battezzato "Nobile/Amundsen Stratospheric Balloon Center", è situato oltre il Circolo Polare Artico, a circa 80 gradi di latitudine.

La missione stratosferica polare SORA, è stata programmata per il 2009. Si tratta di un pallone stratosferico che volerà a circa 36 km sul livello del mare, equipaggiato con un radar a penetrazione appositamente progettato.

Il volo, che partirà dalle Isole Svalbard, circumnavigherà l'Artico sorvolando Spitsbergen, la Groenlandia e l'Arcipelago Artico Canadese, e durerà circa una settimana, raccogliendo dati di grande interesse scientifico.

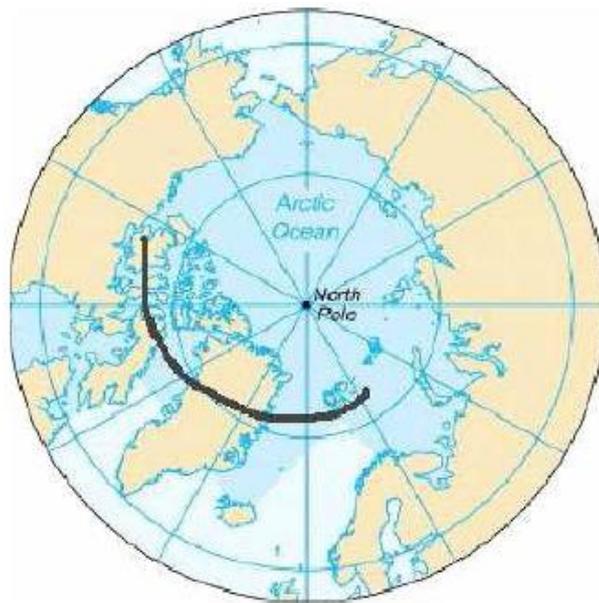


Figura 1

Attualmente, le misurazioni della topografia del letto di ghiaccio e della stratificazione del ghiaccio possono essere fatte dalla superficie, da aerei o dallo spazio. Ad ogni modo, le misurazioni da pallone forniscono miglior copertura dell'osservazione da spazio aereo, e migliore risoluzione dell'osservazione da satellite. Rispetto alle osservazioni da aereo, le osservazioni da pallone non sono affette dal meteo troposferico e non sono limitate in range. Invece, di contro le osservazioni da satellite, si realizzano con tempi di integrazione molto più lunghi. In oltre, lunghe durate delle operazioni da pallone sono molto meno costose delle operazioni da satellite, e sono meno costose delle operazioni da aereo.

Fondamentalmente si tratta di un sistema capace di acquisire dati radar per la creazione dei relativi modelli, per sperimentare nuovi algoritmi e per aumentare le capacità interpretative dei dati.

Le analisi del SORA saranno preziose nell'interpretazione ed inversione dei dati della sottosuperficie planetaria. Infatti, l'esperimento aiuterà ad acquisire dati dallo strato di ghiaccio allo scopo di avere un insieme di misure di riferimento da usare per eliminare possibili ambiguità nelle analisi dei dati Marziani dell'esperimento SHARAD (Shallow Radar). Quindi, SORA deve essere quanto più simile possibile a SHARAD in termini di larghezza di banda e risoluzione.

L'esperimento SORA è focalizzato sull'investigazione di strati di ghiaccio, aree di permafrost, zone stagionalmente ricoperte di neve e mare ghiacciato. Fornirà preziose informazioni sullo spessore del ghiaccio, stratificazione del ghiaccio, proprietà fisiche del bedrock, assorbimento del ghiaccio, acidità del ghiaccio, laghi sotto ghiaccio, e così via.

Le regioni scelte possono essere correttamente considerate come l'analogo terrestre della morfologia polare Marziana, per la presenza di strutture geologiche simili.

L'esperimento trasporterà una gondola riusabile resistente alla pressione ed all'acqua progettata e sviluppata dal CISAS, dotata di tutti i sottosistemi necessari che fornirà al team scientifico le informazioni necessari.

I principali sottosistemi della gondola sono: la struttura meccanica ed il sottosistema di controllo termico; il sottosistema di alimentazione, il sottosistema di manipolazione e gestione dei dati (insieme con le telemetrie) ed il sottosistema di governo. La trasmissione dei dati e l'alimentazione sono le più esigenti; il radar produce un enorme

carico di dati che saranno conservati a bordo mentre, a causa del basso bit rate di trasmissione di dati possibile, solo le informazioni di governo saranno inviate a terra.

In questo esperimento Co.Ri.S.T.A. (Consorzio di Ricerca su Sensori di Telerilevamento Avanzati) di Napoli è responsabile della progettazione e dello sviluppo del radar.

Velocità del Pallone	5 m/s
Altitudine di Volo	35 km
Frequenza Centrale	160 MHz
Banda Trasmessa	10 MHz
PRF	500 Hz
Durata dell'Impulso	3,2 μ s
Finestra di Ricezione	86 μ s
Range Gate Delay	175 μ s
Data Rate	8 KB (no presuming); 1 KB (presuming = 8)
Data Volume	1000 GB (no presuming); 130 GB (presuming = 8)
Massima Potenza Trasmessa	100 W
Risoluzione Verticale	15 m
Dimensioni	< 1000x1000x300 mm ³
Massa	<100 kg

Tabella 1: Parametri del Sistema Payload del SORA

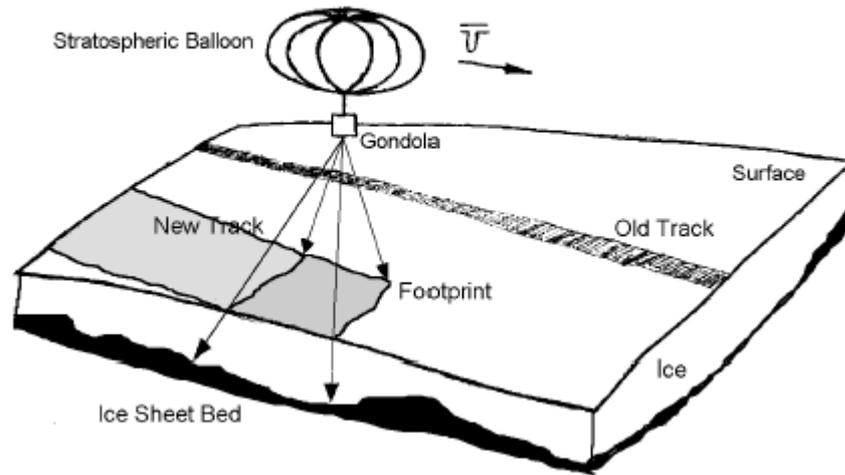


Figura 2: Concept Idea

L'attività di tesi svolta presso il Co.Ri.S.T.A, ha riguardato lo studio del Sistema di Acquisizione Dati del radar SORA e l'elaborazione e la manipolazione dei dati acquisiti.

Nel primo capitolo andremo ad illustrare i principali sottosistemi del payload di SORA. Nel secondo parleremo dei sensori di telerilevamento per poi introdurre il discorso sui segnali chirp di cui fa uso il radar. Nel terzo si accennerà alla tecnica del sottocampionamento di cui utilizzata per l'acquisizione dei dati. Nel quarto capitolo si tratteranno le caratteristiche ed il funzionamento del Convertitore Analogico-Digitale, cuore del Sistema di Acquisizione Dati. Nel quinto ed ultimo capitolo vedremo con l'utilizzo di un software sviluppato in Matlab come determinare dei parametri che stimano la bontà dell'acquisizione dati fatta.

1 ARCHITETTURA DEL PAYLOAD DI SORA

1.1 Acronimi e Definizioni

ADC	Analog To Digital Converter (Convertitore Analogico Digitale)
CDMU	Command and Data Management Unit (Unità di Gestione Dati e Comandi)
CGU	Chirp Generator Unit (Unità Generatore di Chirp)
DAC	Disk Array Controller (Controllore Array di Dischi)
DU	Digital Unit (Unità Digitale)
FGU	Frequency Generator Unit (Unità Generatore di Frequenza)
FPGA	Field Programmable Gate Array
INS	Inertial Navigation System (Sistema di Navigazione Inerziale)
LVDS	Low Voltage Differential Signal
MSB	Most Significant Bit
OBC	On-Board Computer (Computer Di Bordo)
PDU	Power Distribution Unit (Unità Distribuzione Alimentazione)
PRF	Pulse Repetition Frequency
PRI	Pulse Repetition Interval
RFU	Radio Frequency Unit (Unità a Radio Frequenza)
SORA	SOunding RAдар
SSD	Solid State Device (Dispositivo a Stato Solido)
TC	Telecommand (Telecomando)
TM	Telemetry (Telemetria)

1.2 Architettura del Sistema

1.2.1 Descrizione Generale

Il payload di SORA [1][2] che è attualmente in fase finale di messa a punto, è composto dai seguenti sottosistemi principali:

- **Unità di Distribuzione dell'Alimentazione:**
Questo sottosistema ha il compito di fornire alimentazione ai sottosistemi del RADAR.
- **Computer di bordo:**
Ha il compito di monitorare e controllare i sottosistemi del RADAR, gestire telecomandi e telemetrie, salvare i dati acquisiti ed eseguire il software di bordo. È la principale interfaccia con la gondola.
- **Unità a Radio Frequenza:**
Ha il compito di generare e ricevere segnali RADAR.
- **Unità Digitale:**
Ha il compito di fornire i segnali di sincronizzazione del RADAR, configurare ed immagazzinare i dati acquisiti e quelli ancillari.

Il payload si interfaccia con il sottosistema CDMU, la principale unità della gondola. Al CDMU è affidato il compito di accendere e spegnere il RADAR, inviare sequenze di telecomandi per commutare tra le varie modalità operative del RADAR, inviare telemetrie ancillari, ricevere telemetrie del RADAR e segnali di ALIVE.

In Figura 1-1, è riportata l'architettura generale del payload di SORA.

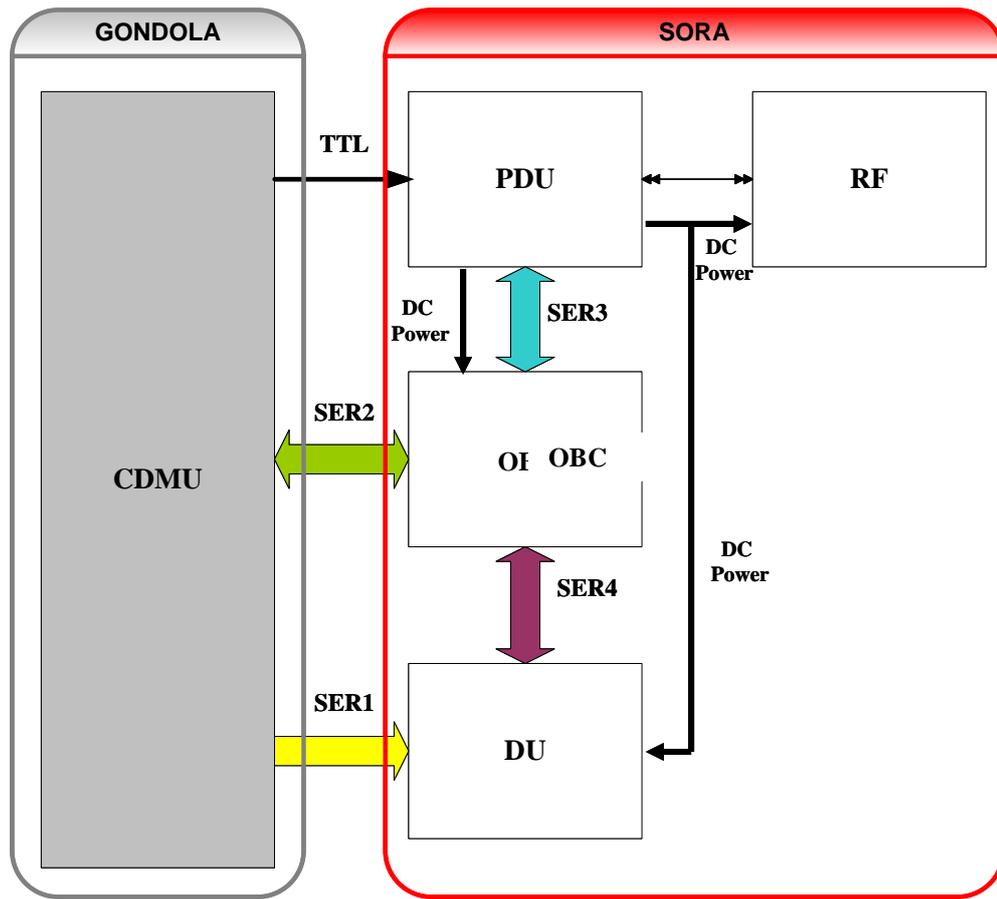


Figura 1-1: Architettura generale del payload di SORA

Le modalità logiche di funzionamento del RADAR sono descritte di seguito:

- La CDMU accende la PDU tramite segnali logici TTL di abilitazione
- La PDU alimenta il Computer di Bordo (tramite alimentazione in DC)
- L'OBC avvia ed esegue il software di bordo
- La PDU alimenta la DU e l'FGU (o RFU) (tramite alimentazione in DC)
- L'OBC invia periodicamente segnali di ALIVE alla CDMU (tramite connessione seriale SER2)
- La CDMU invia periodicamente telemetrie ancillari alla DU (tramite connessione seriale SER1)
- La CDMU invia comandi all'OBC (tramite connessione seriale SER2)
- L'OBC invia comandi di riconoscimento alla CDMU (tramite connessione seriale SER2)

- L'OBC esegue i comandi della CDMU: invia comandi corrispondenti alla DU (tramite connessione seriale SER4) ed alla PDU (tramite connessione seriale SER3)
- La PDU alimenta l'RFU
- L'OBC chiede e riceve i dati ancillari del RADAR dalla PDU (tramite connessione seriale SER3) ed invia le telemetrie ancillari del RADAR alla CDMU (tramite connessione seriale SER2)
- La CDMU spegne l'OBC (tramite connessione seriale SER2)
- La CDMU spegne la PDU con segnali logici TTL di disabilitazione
- La PDU disalimenta OBC, RFU e DU

1.2.2 Stati degli Strumenti e Modalità Operative

Per il payload di SORA sono stati identificati i seguenti stati del RADAR:

- POWER ON (ACCESO)
Questo stato è attivato automaticamente all'accensione del computer di bordo: rappresenta le operazioni necessarie per far partire il RADAR. Al completamento di queste operazioni, lo strumento va nello stato IDLE;
- IDLE (INATTIVO)
Questo stato è attivato alla fine di tutte le altre modalità operative. Lo strumento accetta comandi dalla CDMU per commutare tra gli stati MEASUREMENT, DUMP, PROCESSING o POWER-OFF .
- MEASUREMENT (MISURAZIONE)
Rappresenta una modalità operativa dello strumento ed è attivata per far partire il RADAR ed iniziare l'acquisizione dati.
- POWER OFF (SPENTO)
Questo stato ha il compito di spegnere il RADAR.
- DUMP (SCARICO)
Rappresenta una modalità operativa dello strumento con il compito di copiare i dati acquisiti del RADAR dal disk array controller al disk array (sistemi di dischi).
- PROCESSING (ELABORAZIONE)
Questa è una modalità operativa attivata solo nella configurazione di autocheck. Quando si trova in questa modalità, lo strumento acquisisce dati a ciclo chiuso e li processa per stimare determinati parametri e valutare se il RADAR sta funzionando correttamente.

MEASUREMENT, DUMP e PROCESSING sono le modalità operative che sono comandate dalla CDMU secondo le seguenti sequenze di comandi. Comandi e telemetrie sono riportati e descritti nel Par. 1.3.1.1.

Timing	Command	From	To	IF	Note
T0	SIGNAL	CDMU	PDU	TTL	TTL enabled
T0 + T _{2min}	ALIVE	OBC	CDMU	SER 2	ALIVE signal sent periodically each 2 min
...
T0 + T _{2min}	TM	CDMU	DU	SER 1	Ancillary data Telemetries sent each 2 min
...
T0 + T _{2min}	TM_:RAD	OBC	CDMU	SER 2	Radar data Telemetries sent each 2 min
...
T0 + A	DIS_RAD_ACHK	CDMU	OBC	SER 2	Autocheck disabling command
T0 + B	DIS_RAD_ACHK_A CK	OBC	CDMU	SER 2	Acknowledge of previous command
T0 + C	EN_PSUMMING /DIS_PSUMMING	CDMU	OBC	SER 2	Pre-summing enabling/disabling command
T0 + D	EN_PSUMMING_A CK /DIS_PSUMMING_A CK	OBC	CDMU	SER 2	Acknowledge of previous command
T0 + E	RAD_START	CDMU	OBC	SER 2	Start acquisition command
T0 + F	RAD_START_ACK	OBC	CDMU	SER 2	Acknowledge of previous command
T0 + G	RAD_START_OK	OBC	CDMU	SER 2	OK start acquisition message
T0 + H	RAD_START_OK_A CK	CDMU	OBC	SER 2	Acknowledge of previous command
T0 + I	RAD_STOP	CDMU	OBC	SER 2	Stop acquisition command
T0 +L	RAD_STOP_ACK	OBC	CDMU	SER 2	Acknowledge of previous command
T0 + M	RAD_STOP_OK	OBC	CDMU	SER 2	OK stop acquisition message

T0 + N	RAD_STOP_OK_ACK	CDMU	OBC	SER 2	Acknowledge of previous command
T0 + O	RAD_SHUT_DOWN	CDMU	OBC	SER 2	OBC shut-down command
T0 + P	SIGNAL	CDMU	PDU	TTL	TTL disabled

Tabella 1-1: sequenza MEASUREMENT

Timing	Command	From	To	IF	Note
T0	SIGNAL	CDMU	PDU	TTL	TTL enabled
T0 + T _{2min}	ALIVE	OBC	CDMU	SER 2	ALIVE signal sent periodically each 2 min
...
T0 + T _{2min}	TM	CDMU	DU	SER 1	Ancillary data Telemetries sent each 2 min
...
T0 + T _{2min}	TM_:RAD	OBC	CDMU	SER 2	Radar data Telemetries sent each 2 min
...
T0 + A	RAD_SAVE	CDMU	OBC	SER 2	Start dump command
T0 + B	RAD_SAVE_ACK	OBC	CDMU	SER 2	Acknowledge of previous command
T0 + C	RAD_SAVE_OK	OBC	CDMU	SER 2	Executed dump message
T0 + D	RAD_SAVE_OK_ACK	CDMU	OBC	SER 2	Acknowledge of previous command
T0 + E	RAD_SHUT_DOWN	CDMU	OBC	SER 2	OBC shut-down command
T0 + F	SIGNAL	CDMU	PDU	TTL	TTL disabled

Tabella 1-2: sequenza DUMP

Timing	Command	From	To	IF	Note
T0	SIGNAL	CDMU	PDU	TTL	TTL enabled
T0 + T _{2min}	ALIVE	OBC	CDMU	SER 2	ALIVE signal sent periodically each 2 min
...
T0 + T _{2min}	TM	CDMU	DU	SER 1	Ancillary data Telemetries sent each 2 min
...
T0 + T _{2min}	TM_:RAD	OBC	CDMU	SER 2	Radar data Telemetries sent each 2 min
...
T0 + A	EN_RAD_ACHK	CDMU	OBC	SER 2	Enabling Autocheck command
T0 + B	EN_RAD_ACHK_A CK	OBC	CDMU	SER 2	Acknowledge of previous command
T0 + C	EN_PSUMMING /DIS_PSUMMING	CDMU	OBC	SER 2	Enabling/disabling on-board pre-summing command
T0 + D	EN_PSUMMING_A CK /DIS_PSUMMING_A CK	OBC	CDMU	SER 2	Acknowledge of previous command
T0 + E	RAD_START	CDMU	OBC	SER 2	Start acquisition command
T0 + F	RAD_START_ACK	OBC	CDMU	SER 2	Acknowledge of previous command
T0 + G	RAD_START_OK	OBC	CDMU	SER 2	OK start acquisition message
T0 + H	RAD_START_OK_A CK	CDMU	OBC	SER 2	Acknowledge of previous command
T0 + I	RAD_STOP	CDMU	OBC	SER 2	Stop acquisition command
T0 + L	RAD_STOP_ACK	OBC	CDMU	SER 2	Acknowledge of previous command
T0 + M	RAD_STOP_OK	OBC	CDMU	SER 2	OK stop acquisition message
T0 + N	RAD_STOP_OK_AC K	CDMU	OBC	SER 2	Acknowledge of previous command
T0 + O	RAD_SAVE	CDMU	OBC	SER 2	Start dump command

Timing	Command	From	To	IF	Note
T0 + P	RAD_SAVE_ACK	OBC	CDMU	SER 2	Acknowledge of previous command
T0 + Q	RAD_END_SAVE	OBC	CDMU	SER 2	Executed dump message
T0 + R	RAD_END_SAVE_A CK	CDMU	OBC	SER 2	Acknowledge of previous command
T0 + S	RAD_PROC_START	CDMU	OBC	SER 2	Start on-board processing command
T0 + T	RAD_PROC_START _ACK	OBC	CDMU	SER 2	Acknowledge of previous command
T0 + U	RAD_PROC_END	OBC	CDMU	SER 2	Executed on-board processing message
T0 + V	RAD_PROC_END_A CK	CDMU	OBC	SER 2	Acknowledge of previous command
T0 + Z	TM_RAD	OBC	CDMU	SER2	Radar telemetries which include processing results data
T0 + AA	RAD_SHUT_DOWN	CDMU	OBC	SER 2	OBC Shut-down command
T0 + AB	SIGNAL	CDMU	PDU	TTL	TTL disabled

Tabella 1-3: sequenza AUTOCHK

1.3 Interfacce Gondola-Radar

1.3.1 Interfacce Elettriche

1.3.1.1 Interfacce TC/TM

Le interfacce TC/TM sono raffigurate in Figura 1-2 e sono costituite da due interfacce RS-232: SER1 e SER2.

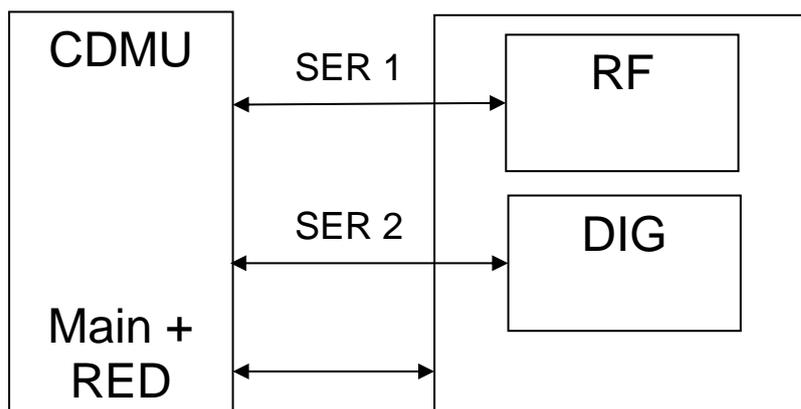


Figura 1-2 : interfacce TC/TM tra CDMU e SORA

Le caratteristiche principali sono:

- SER 1: per mezzo di questo collegamento RS-232 la CDMU invia al RADAR tutti i dati ancillari che saranno conservati per ulteriori post-elaborazioni off-line. Sono:
 - Tempo della Missione;
 - Posizione (Latitudine, Longitudine e Altitudine fornite dal GPS);
 - Orientamento della CDMU
- SER 2: per mezzo di questo collegamento RS-232 tutti i dati di telecomandi e di telemetrie sono scambiati tra la CDMU ed il payload di SORA.

1.3.1.2 Definizione Protocollo SER 1

I dati ancillari a pacchetto sono composti da un numero fissato di byte qui di seguito indicati:

Offset	Tipo di Dati	Dimensione [byte]	Dati Ancillari	Unità di Misura
0	unsigned int	4	Tempo della Missione	millisecondi
4	float	4	Longitudine Est	gradi
8	float	4	Latitudine Nord	gradi

12	float	4	Altitudine	km
16	float	4	Rollio	gradi
20	float	4	Beccheggio	gradi
24	float	4	Imbardata	gradi

Tabella 1-4 : Pacchetto dati ancillari

Ciascun tipo di dato è rappresentato in formato Little Endian.

Allo scopo di consentire una più facile sincronizzazione con i data deformatter, i dati a pacchetto mostrati nella Tabella 1-4 saranno preceduti da un marker composto da 4 + 4 byte. Ciascun gruppo di byte (in valore esadecimale) è ad esempio 47 47 50 53 (in ASCII sono niente altro che le rappresentazioni del GPS).

Riassumendo i dati a pacchetto saranno composti come segue (il primo byte trasmesso è quello da sinistra):

- 0x47 0x47 0x50 0x53
- 0x47 0x47 0x50 0x53

dati in Tabella 1-4.

Il protocollo RS-232 usato su SER 1 è il 9600 8N1. Il formato del pacchetto trasmesso è il seguente:

HEADER COUNT DATA ENDP BCC

dove:

- I byte HEADER e ENDP indicano, rispettivamente, l'inizio e la fine di ciascuna trasmissione di pacchetto; il loro valore è costante e 0x02 è il valore di default per HEADER e 0x03 il valore di default per ENDP;
- COUNT è il numero di byte che costituisce il campo DATA;

- BCC è il byte usato per controllare l'integrità del pacchetto dati. È calcolato secondo il seguente algoritmo: ciascun byte ricevuto è sommato al precedente (somma modulo 255); dopo che l'ultimo byte utile è stato ricevuto la somma viene negata e viene aggiunto 1.

I byte del pacchetto utili per il calcolo di BCC sono COUNT e DATA.

1.3.1.3 Definizione Protocollo SER 2

Tutti i telecomandi sono rappresentati per mezzo di un singolo byte. Per ciascun telecomando ricevuto dal RADAR un byte di riconoscimento sarà restituito alla CDMU; questo byte di ack è uguale al byte di comando ricevuto ma con l'MSB settato a 1.

Nella tabella seguente sono riportati tutti i possibili byte di comando e di ack:

Command Name	Value (hex)	NOTE
RAD_START	0x01	Radar MEASUREMENT operative mode start command (from CDMU to Radar via SER2)
RAD_START_ACK	0x81	ACK of RAD_START command (from Radar to CDMU via SER2)
RAD_STOP	0x02	Radar MEASUREMENT operative mode stop command (from CDMU to Radar via SER2)
RAD_STOP_ACK	0x82	ACK of RAD_STOP command (from Radar to CDMU via SER2)
RAD_SAVE	0x03	Radar DUMP operative mode start command (from CDMU to Radar via SER2)
RAD_SAVE_ACK	0x83	ACK of RAD_SAVE command (from Radar to CDMU su SER2)
RAD_END_SAVE	0x04	DUMP procedure ending message (from Radar to CDMU via SER2)
RAD_END_SAVE_ACK	0x84	ACK of RAD_END_SAVE command (from CDMU to Radar via SER2)

Command Name	Value (hex)	NOTE
NOP	0x05	No Operation command da CDMU verso RADAR
NOP_ACK	0x85	ACK of NOP command (from RADAR to CDMU via SER2)
ALIVE	0x06	ALIVE signal (sent each 2 min) (from Radar to CDMU via SER2)
EN_PSUMMING	0x07	Pre-summing enabling command (from CDMU to Radar via SER2)
EN_PSUMMING_ACK	0x87	ACK of EN_PSUMMING command (from Radar to CDMU via SER2)
DIS_PSUMMING	0x08	Pre-summing disabling command (from CDMU to Radar via SER2)
DIS_PSUMMING_ACK	0x88	ACK of DIS_PSUMMING command (from Radar to CDMU via SER2)
RAD_START_OK	0x09	Radar MEASUREMENT operative mode start executed message (from Radar to CDMU via SER2)
RAD_START_OK_ACK	0x89	ACK of RAD_START_OK message (from CDMU to Radar via SER2)
RAD_STOP_OK	0x0A	Radar MEASUREMENT operative mode stop executed message (from Radar to CDMU via SER2)
RAD_STOP_OK_ACK	0x8A	ACK of RAD_STOP_OK message (from CDMU to Radar via SER2)
EN_RAD_ACHK	0x0B	Radar AUTOCHECK operative mode enabling command (from CDMU to RADAR via SER2)
EN_RAD_ACHK_ACK	0x8B	ACK of EN_RAD_ACHK command (from RADAR to CDMU via SER2)
DIS_RAD_ACHK	0x0C	Radar AUTOCHECK operative mode disabling command (from CDMU to RADAR via SER2)

Command Name	Value (hex)	NOTE
DIS_RAD_ACHK_ACK	0x8C	ACK of DIS_RAD_ACHK command (from RADAR to CDMU via SER2)
RAD_PROC_START	0x0D	Radar PROCESSING operative mode start command (from CDMU to RADAR via SER2)
RAD_PROC_START_ACK	0x8D	ACK of RAD_PROC_START command (from RADAR to CDMU via SER2)
RAD_PROC_END	0x0E	Radar PROCESSING operative mode end message (from RADAR to CDMU via SER2)
RAD_PROC_END_ACK	0x8E	ACK of RAD_PROC_END message (from CDMU to RADAR via SER2)
RAD_SHUT_DOWN	0x0F	SHUT-DOWN command (from CDMU to RADAR via SER2)
RAD_SPARE_CMD	0x11	SPARE command (from CDMU to RADAR via SER2)
RAD_SPARE_CMD_ACK	0x91	ACK of RAD_SPARE_CMD command (from RADAR to CDMU via SER2)

Tabella 1-5 : Byte di Comando e di Ack

Il protocollo RS-232 usato su SER 2 è il 9600 8N1. Il formato del pacchetto trasmesso è il seguente:

HEADER #Byte Command DATA ENDP BCC

dove:

- I byte HEADER e ENDP indicano, rispettivamente, l’inizio e la fine della trasmissione di ciascun pacchetto; il loro valore è costante e 0x02 è il valore di default di HEADER e 0x03 è il valore di default per ENDP;

- #Byte il numero di byte che costituiscono i campi Command e DATA; il suo MSB è pari a 1 se il pacchetto è relativo alla comunicazione Terra-Radar, e pari a 0 se è relativo alla comunicazione CDMU-Radar. Per questa ragione il pacchetto più lungo avrà al più 128 byte nei campi Command e DATA (il minimo valore è 1 che almeno nel campo Command è sempre presente);
 - Command è l'operazione da eseguire ed è una di quelle indicate in Tabella 1-5.
 - BCC è il byte usato per controllare l'integrità del pacchetto dati. È calcolato secondo il seguente algoritmo: ciascun byte ricevuto è sommato al precedente (somma modulo 255); dopo che l'ultimo byte utile è stato ricevuto la somma viene negata e viene aggiunto 1. I byte del pacchetto utili per il calcolo di BCC sono #Byte Command e DATA.
- 1) la trasmissione di telecomandi tra CDMU e RADAR è ripetuta fino al periodo di timeout se nessun comando di riconoscimento è stato notificato fino ad allora
 - 2) un comando di ALIVE è trasmesso dal RADAR alla CDMU ogni 2 minuti. Se nessun comando di ALIVE viene ricevuto dalla CDMU, la CDMU spegne il RADAR.

Secondo ciò che è stato definito, tutti i dati sulle telemetrie che il RADAR invierà a terra, saranno inviate con l'MSB del campo #Byte pari a 1. In questo caso la CDMU eseguirà solo le verifiche del protocollo e dell'integrità dei dati e le invierà a terra senza eseguire qualsiasi altro compito. I campi #Byte+Command+DATA saranno deformati ed ulteriormente analizzati a terra.

Le telemetrie del RADAR includono dentro il campo Command+DATA 28 dati grezzi di tensioni e correnti così come sono stati misurati dall'ADC.

Poiché le telemetrie del RADAR sono inviate tramite SER2, il formato di TM_RAD è il seguente:

Header + #Byte + Command + DATA + TRAIL + BCC

Dove:

Header = 1 byte

#Byte = 1 byte (con MSB = 1)

Command + DATA = 28 byte (28 canali ad 1 byte ciascuno)

TRAIL = 1 byte

BCC = 1 byte

Questo corrisponde a 32 byte lungo il pacchetto di telemetria del RADAR.

1.3.2 Interfacce Meccaniche

Il payload di SORA sarà montato all'interno della sonda progettata e sviluppato dal CISAS.

La sonda è una gondola riusabile resistente alla pressione ed all'acqua. La sua struttura principale è realizzata in lega di alluminio eccetto per alcune componenti strutturali chiavi per le quali è stato utilizzato acciaio inossidabile. La sonda ospita l'antenna, l'equipaggiamento ASI per la telemetria, la zavorra di rilascio ed i sottosistemi per l'attuazione del paracadute all'esterno; all'interno ospita sul fondo le batterie e l'unità CDMU (con un software real time incorporato per l'acquisizione e la manipolazione dei dati ad hoc) sviluppata dal CISAS (Centro Interdipartimentale di Studi e Attività Spaziali) di Padova; l'unità di distribuzione dell'alimentazione viene montata sul piatto di alluminio di mezzo; mentre il payload del RADAR è posizionato sopra, sull'altro dei due piatti removibili di alluminio.

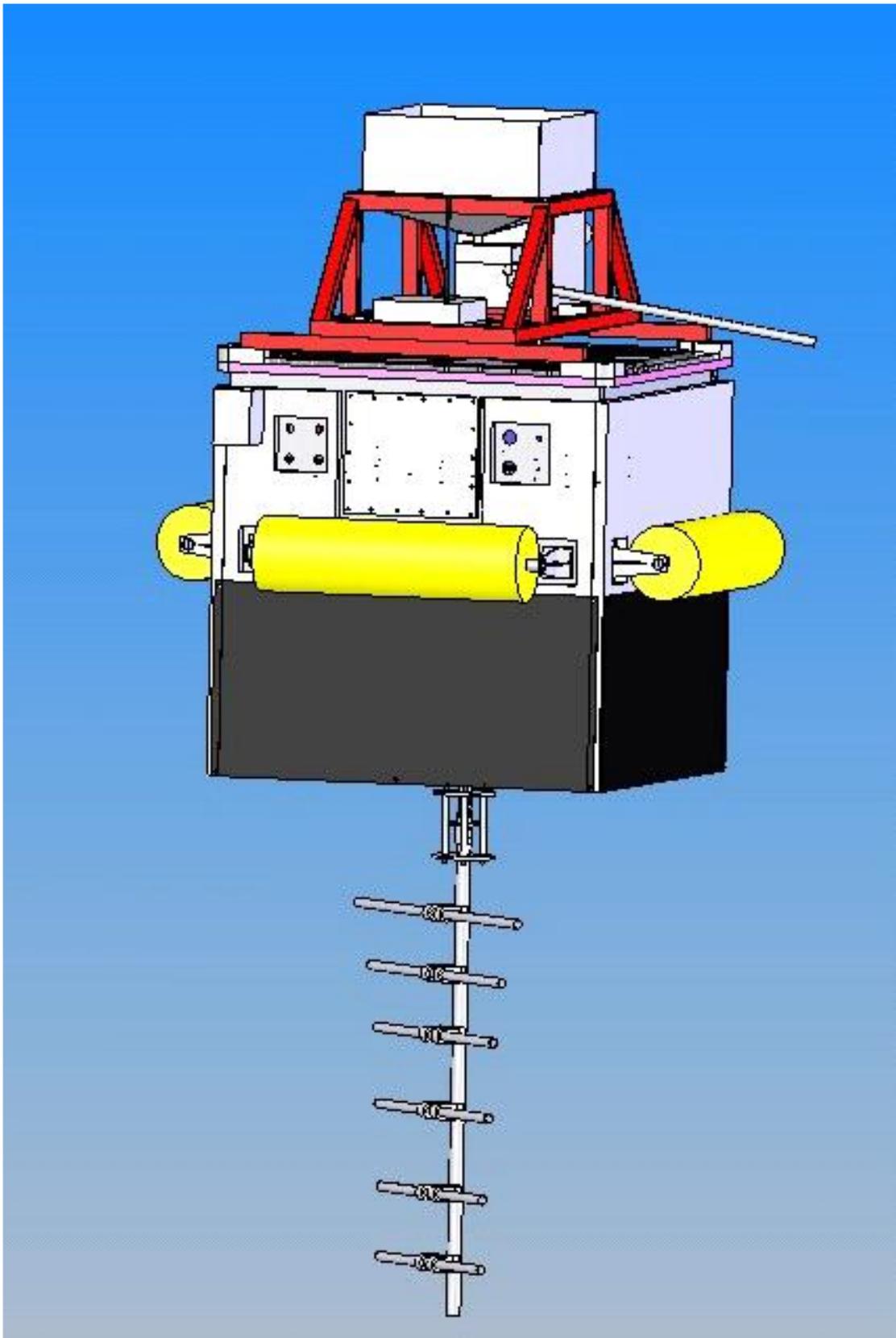


Figura 1-3: Progetto della sonda (dal CISAS)

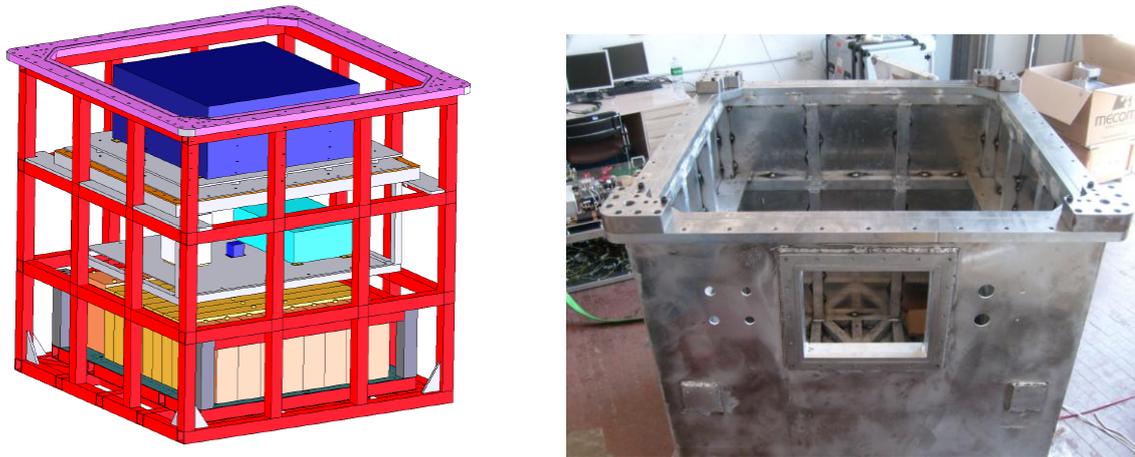


Figura 1-4: Struttura interna e Foto della struttura esterna della gondola

Il payload di SORA deve essere montato a bordo del piatto removibile della sonda, della quale è riportato il progetto meccanico in Figura 1-. il piatto ha le dimensioni $1000\text{mm}\times 850\text{mm}$ e presenta 15 fori M5 localizzati a $215\text{mm}\times 330\text{mm}$ di distanza tra loro. Questi fori saranno utilizzati per montare il piatto removibile nella sonda.

Nessuna limitazione specifica è stata identificata per quanto concerne il montaggio meccanico del RADAR, con l'eccezione del disk array del RADAR. Infatti, poiché è richiesto che il disk array del RADAR sia removibile così che i dati acquisiti possano essere prelevati subito dopo il volo della missione, il disk array deve essere montato vicino all'apertura della sonda localizzata al centro del lato lungo del piatto.

La disposizione meccanica ed il montaggio del payload di SORA è mostrato in Figura 1-6. Le dimensioni dei componenti sono riassunte nella Tabella 1-6.

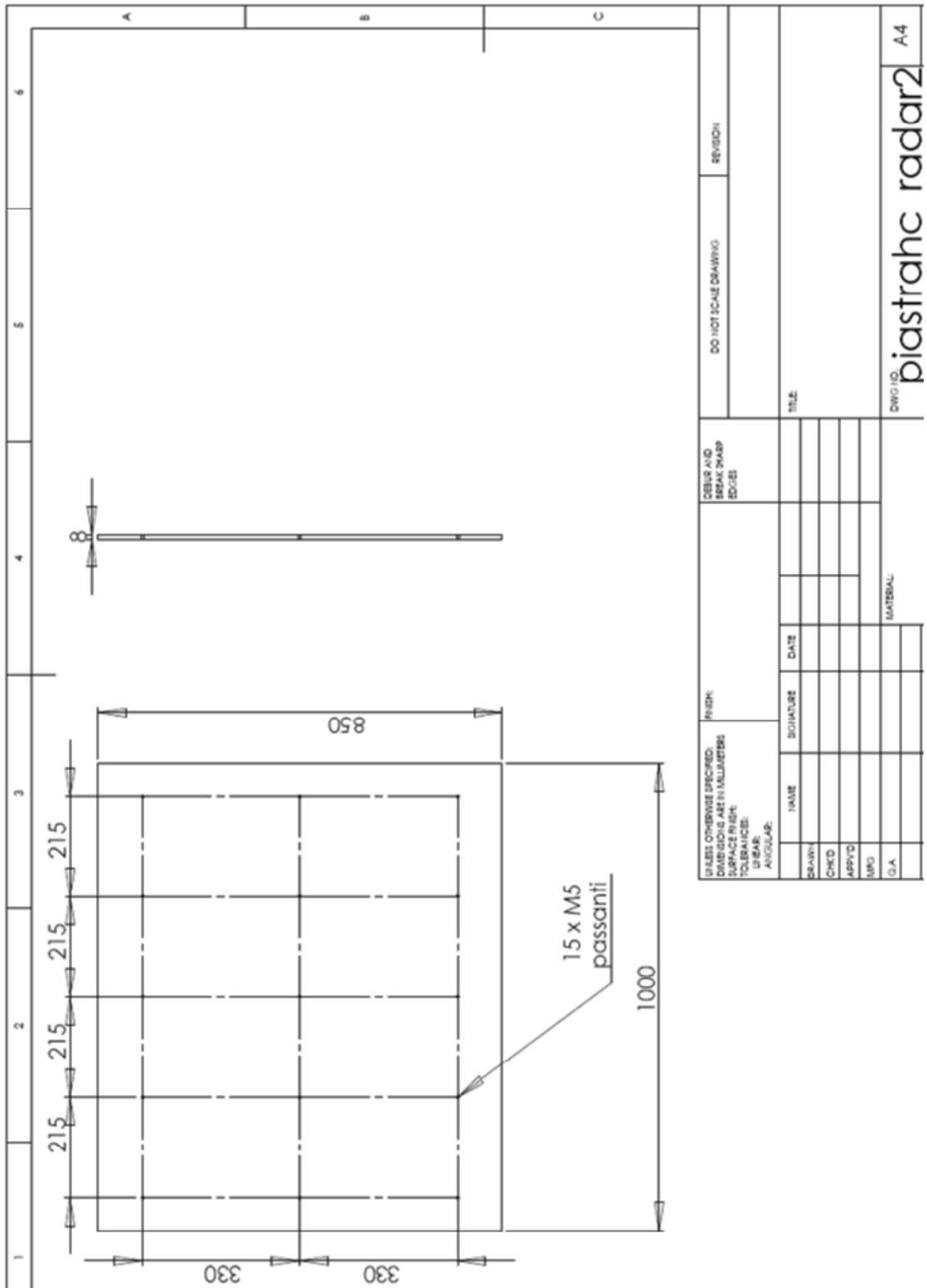


Figura 1-5: Progettazione della piastra di montaggio del RADAR

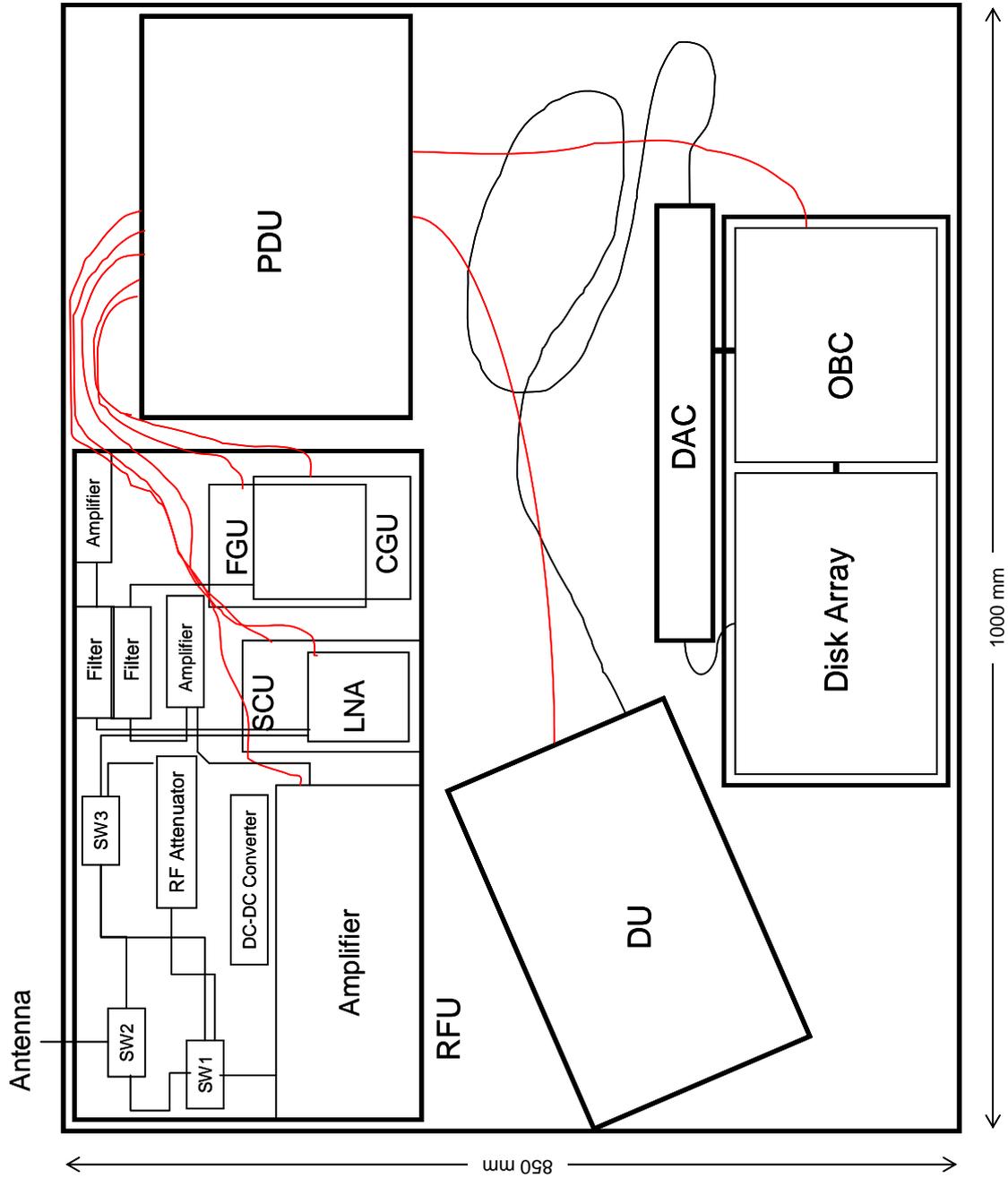


Figura 1-6: Disposizione meccanica e montaggio dei sottosistemi del RADAR

Descrizione	Dimensioni (mm)
Unità di Distribuzione dell'Alimentazione (PDU)	340×250×120
Computer di Bordo (OBC)	460×200×150
Unità Digitale (DU)	300×240×60
Unità Generatore di Frequenza (FGU)	110×200×60
Chirp Generator (CG)	110×140×70
Switch Control Unit (SCU)	110×140×70
Amplificatore a Basso Rumore (LNA)	100×90×30
Amplificatore	300×120×80
Attenuatore RF	140×80×80
Filtri	210×30Ø
Convertitore DC-DC	130×50×10
Switch 3	60×60×10
Switch 1 and 2	100×90×60

Tabella 1-6: Dimensione dei componenti del RADAR

1.4 Sottosistemi del Radar

1.4.1 Computer di Bordo

1.4.1.1 *Descrizione*

Il RADAR contiene un piccolo personal computer, con un disk array controller ed alcuni array di dischi. Il sottosistema del computer di bordo ha i seguenti scopi:

- Monitoraggio e controllo dei sottosistemi del RADAR;
- Gestione della comunicazione della gondola;
- Pilotaggio del disk array controller;
- Risparmio di dati ridondanti;

Uno schema del computer è mostrato in Figura 1-7:

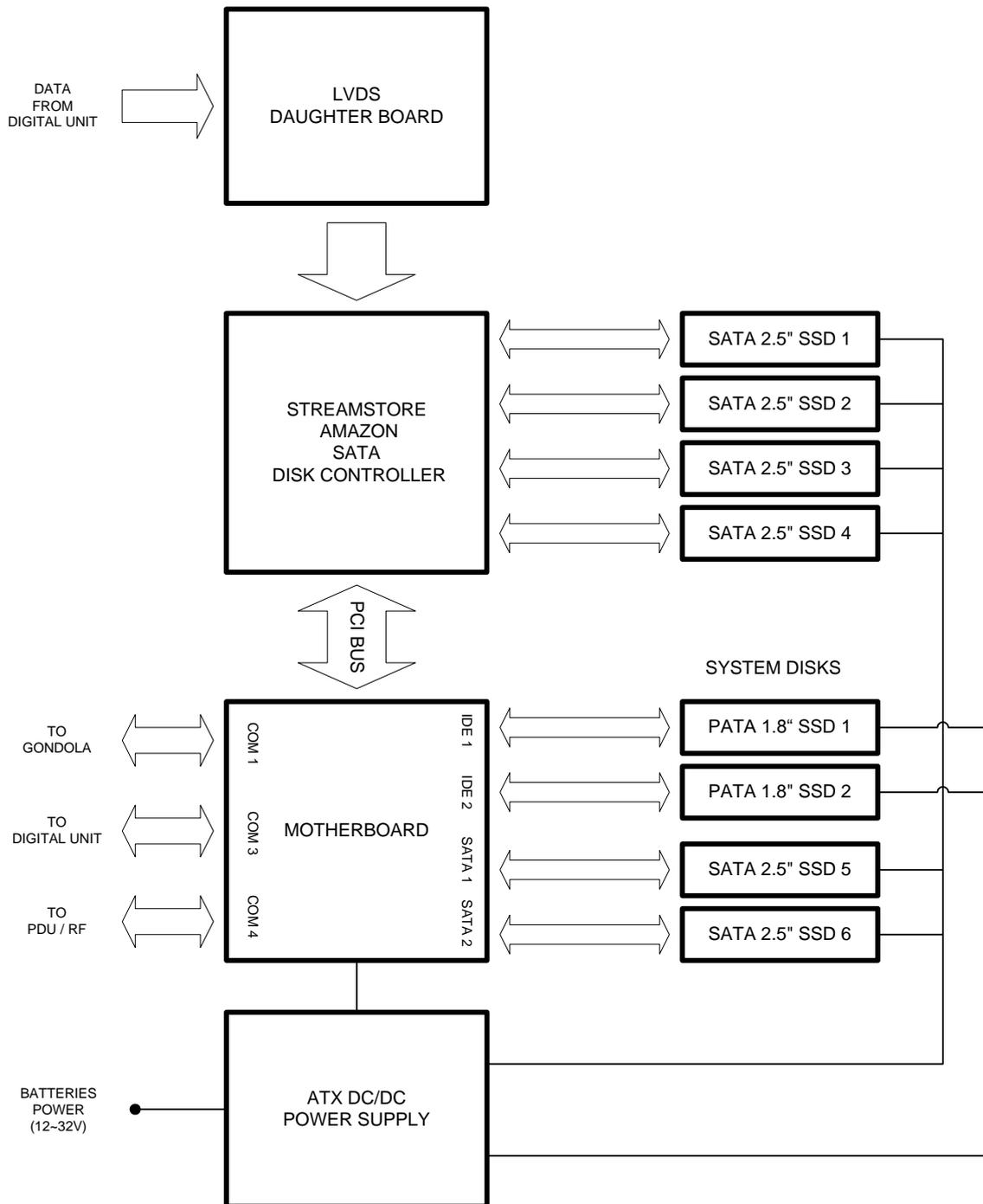


Figura 1-7: Schema del computer di bordo

Il disk array controller Streamstore Amazon SATA consente di catturare dati in tempo reale dall'unità digitale con il corretto data rate, durante l'acquisizione del RADAR. I

dati passano attraverso la daughter board LVDS che è montata sul disk array controller. Il controllore è provvisto di 4 dischi a stato solido SATA che memorizzano i dati. Sulla scheda madre del computer ci sono 4 dischi aggiuntivi contenenti il software di bordo. Su questi dischi saranno copiati i dati già catturati dal disk array controller. La scheda madre e tutti i dischi sono alimentati con un alimentatore ATX DC/DC.

1.4.1.2 Caratteristiche

Il PC di bordo è composto da differenti sottosistemi.

Sono elencati di seguito con le loro specifiche:

Streamstore Amazon SATA disk controller

- Up to 16 SATA conventional or SSD drives;
- Data rate of more than 400 MB/s sustained recording;
- Internal memory buffer of 1 GB SDRAM;
- Power failure recovery support;
- Modular mezzanine interface system;
- PCI 2.1 compatible / Universal PCI board

Daughter board LVDS

- 16 bit Low Voltage Differential Signal (LVDS) interface;
- Sustained recording of 400MB/s;
- High speed Infiniband connectors;
- Up to 4 independent data channels;
- Direct data interface to StreamStore recorders;

Motherboard

- MINI-ITX form factor (17 x 17 cm);
- Embedded VIA C7 EDEN 1.2 GHz Fanless Processor;
- VIA CN700 Northbridge + VT8237RP Southbridge Chipsets
- 1GB Single Channel DDR2 533 Memory DIMM memory;

- IDE controllers (UltraDMA 133/100/66);
- SATA controllers;
- serial RS232 ports;

Disk drives

2 x Sandisk 1.8" PATA SSD - 32 GB capacity;

6 x Sandisk 2.5" SATA SSD – 32 GB capacity;

Power Supply

ATX DC/DC power supply;

Maximum power 250 W;

Input voltage 6-32V.

1.4.1.3 Interfacce

LVDS daughter board

La daughter board LVDS è integrata sul disk array controller;

La scheda è connessa alla digital unit con uno cavo standard 12 lane Infiniband

Streamstore Amazon SATA disk controller

Il disk controller è connesso alla scheda madre con uno slot PCI standard. Sulla scheda ci sono sedici connettori SATA L. Quattro di essi sono connessi a quattro SATA SSD da 2.5".

Motherboard

I connettori usati sono:

- COM1 standard RS232 DB9 connesso alla gondola;
- COM3 standard RS232 DB9 connesso alla Digital Unit;
- COM4 standard RS232 DB9 connesso alla PDU/RFU;
- Connettore IDE standard 40 pin connesso al PATA SSD 1 tramite cavo flat standard IDE;

- Connettore IDE standard 40 pin connesso al PATA SSD 2 tramite cavo flat standard IDE;
- Connettore SATA L connesso al SATA SSD 5 tramite cavo standard SATA;
- Connettore SATA L connesso al SATA SSD 6 tramite cavo standard SATA;
- Connettore 20 pin standard ATX power connesso all'alimentatore ATX DC/DC;

Solo per scopi di collaudo, vengono usati:

- Connettore PS/2 per la tastiera;
- Connettore PS/2 o USB per il mouse;
- Video DB per il monitor;
- Connettore RJ-45 per la rete.

SATA 2.5" SSD drives

Oltre alle già descritte connessioni, usano connettori di alimentazioni standard SATA L.

PATA 1.8" SSD drives

Hanno un connettore ZIF e sono integrati per mezzo di un adattatore ZIF – 3.5" che fornisce un connettore IDE standard a 40 pin e un connettore di alimentazione maschio standard MOLEX a 4 pin, l'ultimo connesso all'alimentatore tramite un cavo di alimentazione femmina MOLEX a 4 pin.

1.4.2 Sw di Bordo

1.4.2.1 Descrizione

Il software di bordo ha lo scopo di tempificare ed attivare i vari componenti hardware del RADAR. Parte automaticamente all'avvio del computer.

Nel modo normale il software accetta comandi dalla CDMU tramite RS232: non usa tastiera, mouse e monitor. Ciò nonostante, c'è una speciale modalità di test locale che simula l'interazione della CDMU e consente al software di partire senza la presenza della CDMU. Può essere attivato mettendo a 0 il valore di un parametro della CDMU (normalmente settato a 1) nel file di configurazione.

Il software presenta, ad ogni modo, una GUI (Graphic User Interface) che consente all'operatore umano di monitorare e comandare: presenta pulsanti di comando per inviare comandi e finestre di testo che mostrano l'attività corrente.

C'è anche un server VNC che permette di connettersi al computer di bordo da un'altra stazione di lavoro tramite cavo standard RJ-45 cross-linked

1.4.2.2 Caratteristiche

Il sistema operativo è il Microsoft Windows XP Professional SP1.

Il software è sviluppato in Inprise/Borland C++ Builder 6.0 IDE ed un singolo file eseguibile (sorasw.exe) con un file di configurazione (sorasw.ini).

Richiede che il file xlrapi.dll (Conduant API interface to Streamstore Amazon SATA disk array controller) sia nel path della variabile d'ambiente del computer.

Il software di bordo presenta differenti stati, ciascuno corrisponde ad un modo operativo del RADAR, con i quali accetta e manda comandi alle varie interfacce. Sono:

- POWER ON
Questo modo è attivato automaticamente all'avvio del PC: rappresenta le operazioni necessarie per far partire il RADAR. Dopo aver completato queste operazioni, il software va nel modo IDLE;
- IDLE
Questo modo viene attivato al termine degli altri. Accetta comandi dalla CDMU che settano il software nei modi MEASUREMENT, DUMP, PROCESSING o POWER-OFF.
- MEASUREMENT
Fa partire il RADAR e comincia l'acquisizione dati; può cominciare nel modo standard o *autocheck*, con o senza *presumming*. Durante l'acquisizione, il software attende un comando di stop acquisizione dati.
- POWER OFF
Accende il RADAR.
- DUMP
Copia i dati acquisiti dal RADAR dal disk array controller al sistema di dischi.
- PROCESSING
Questo modo è attivato solo in configurazione di autocheck. Viene chiamata una

funzione esterna, passandole una certa quantità di dati precedentemente acquisiti. Questa funzione elabora i dati e ritorna dei valori che consentono di determinare se il RADAR sta funzionando correttamente.

In tutti i modi di funzionamenti, il software di bordo manda un segnale di ALIVE alla CDMU ogni due minuti. Se la CDMU non riceve questo segnale, spegne il RADAR.

Il software ha due thread aggiuntivi, oltre a quello principale: uno per le operazioni di scarico e l'altro per l'elaborazione.

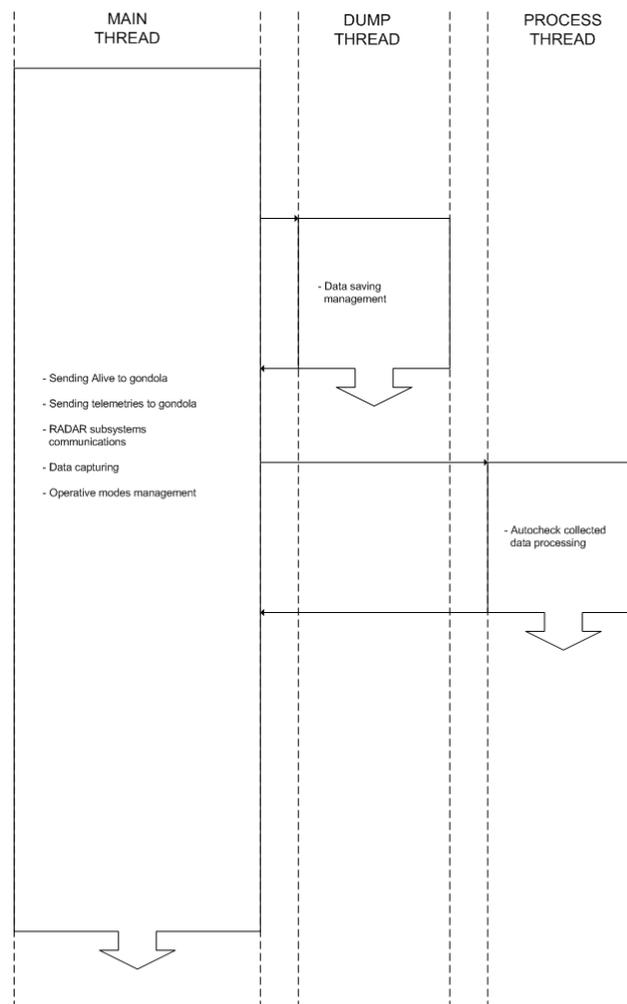


Figura 1-8: i Thread

Parte del codice di elaborazione è su un programma esterno, chiamato software principale.

1.4.2.3 Interfacce

Il software di bordo ha le seguenti interfacce:

- CMDU Line Interfaccia con la CDMU tramite apposito protocollo su RS232;
- PDU Line Interfaccia con la PDU/RFU tramite apposito protocollo su RS232;
- DU Line Interfaccia con la DU tramite apposito protocollo su RS232;
- DAC Interfaccia con la scheda Streamstore Amazon Disk Array Controller, tramite manufacturer's API;

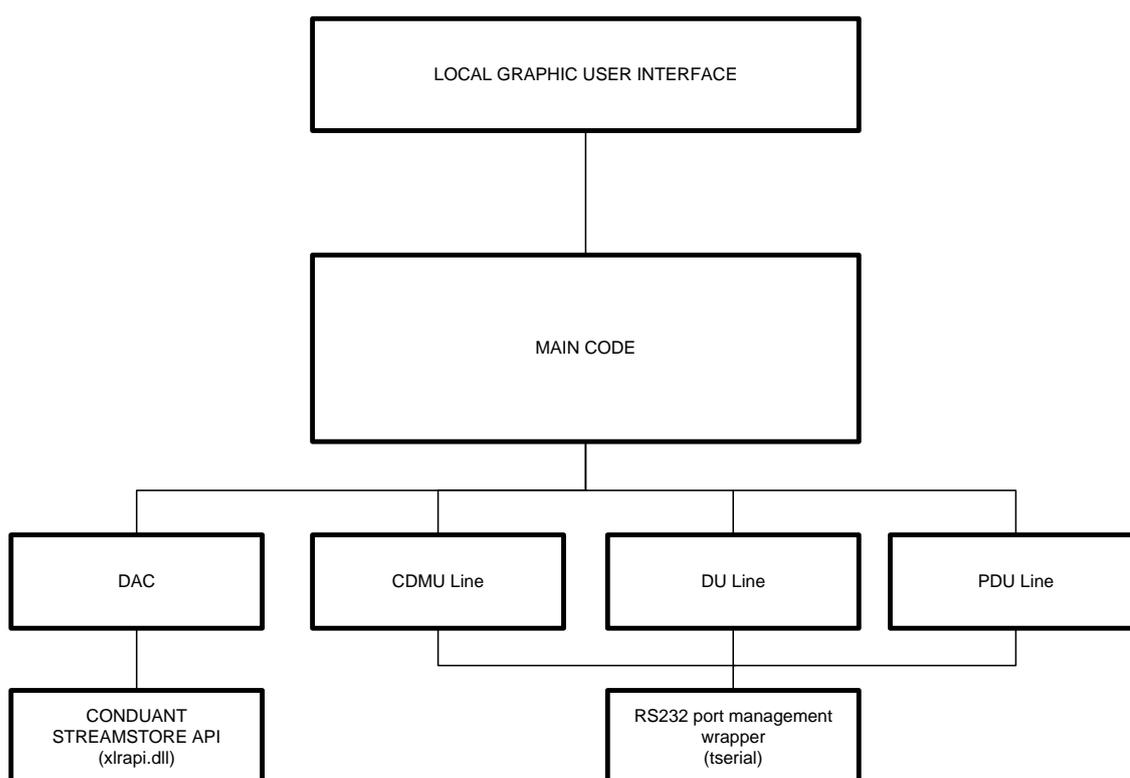


Figura 1-9: Componenti del Software

L'interfaccia del DAC è implementata in un modulo contenente funzioni che fanno operazioni necessarie allo scopo di controllare la scheda disk array controller. Tali funzioni chiamano appropriatamente il codice API contenuto nella xlrapi.dll della manufacturer's shared library.

Le altre interfacce ed i loro protocolli sono implementati con codice di Programmazione Orientata agli Oggetti in moduli contenenti classi e funzioni. Viene usata una classe base incapsulante per gestire la porta RS232.

Queste classi e funzioni consentono la dissociazione dalla funzione basso livello legata al hardware.

1.4.3 Unità Digitale

1.4.3.1 *Descrizione*

Gli scopi principali dell'unità digitale sono:

- acquisire e memorizzare echi RF;
- memorizzare dati ancillari che arrivano dalla CDMU;
- fornire segnali di tempificazione a tutto il RADAR.

Inoltre l'unità digitale è capace di operare in due modi differenti:

- **No Presumming**: in questa modalità il RADAR opera normalmente, gli echi sono campionati, formattati insieme con i dati ancillari ed immagazzinati nel disk array;
- **Presumming**: in questa modalità il RADAR opera esattamente come la precedente ma gli echi campionati sono sommati 8 a 8 e solo la somma viene formattata con i dati ancillari e memorizzata sul disk array.

1.4.3.2 *Caratteristiche*

L'architettura generale dell'unità digitale è riportata in Figura 1-10. I principali sottosistemi che compongono l'unità sono:

- Analog To Digital Converter;
- Timing and Control unit;
- Data Formatting and Disk Array Controller Board.

I sottosistemi menzionati sono implementati per mezzo di due schede: la prima è il Convertitore Analogico Digitale e la seconda è la scheda basata su FPGA.

L'ADC è un kit di valutazione dell'LTC2208-14 della Linear Technologies; il clock di campionamento a 50MHz viene ricevuto dal sottosistema RF ed è trasmesso alla scheda FPGA in formato elettrico LVDS insieme con i campioni a 14 bit del segnale

digitalizzato. L'uscita del clk LVDS viene usata dalla scheda FPGA allo scopo di sincronizzare tutti i sottosistemi digitali allo stesso clock di riferimento.

La scheda FPGA è principalmente composta da un FPGA Virtex-4 della Xilinx. I sottosistemi principali, tutti implementati nello stesso FPGA, sono:

- *Timing&Control Unit*, questo sottosistema genera i segnali di timing e di controllo necessari al RADAR allo scopo di operare correttamente: Start_CGU, Enable_Autocheck, RF_Switch, PRI, Start_ADC, Start_Formatting e così via. Tutti questi segnali sono derivati dal clock di riferimento ricevuto dall'ADC. Inoltre, questo sottosistema è responsabile di reagire ai comandi ricevuti dal Control PC e fornire le risposte adatte. Questo obiettivo è realizzato per mezzo di una custom board composta da un microcontrollore che converte comandi/risposte RS-232 ricevuti dal PC/FPGA in segnali elettrici;
- *CDMU IF-INS/GPS*: questo sottosistema è responsabile di ricevere dati ancillari trasmessi dalla CDMU, verificare la loro integrità e immagazzinarli in ordine FIFO per consentire loro la sincronizzazione al timing PRI;

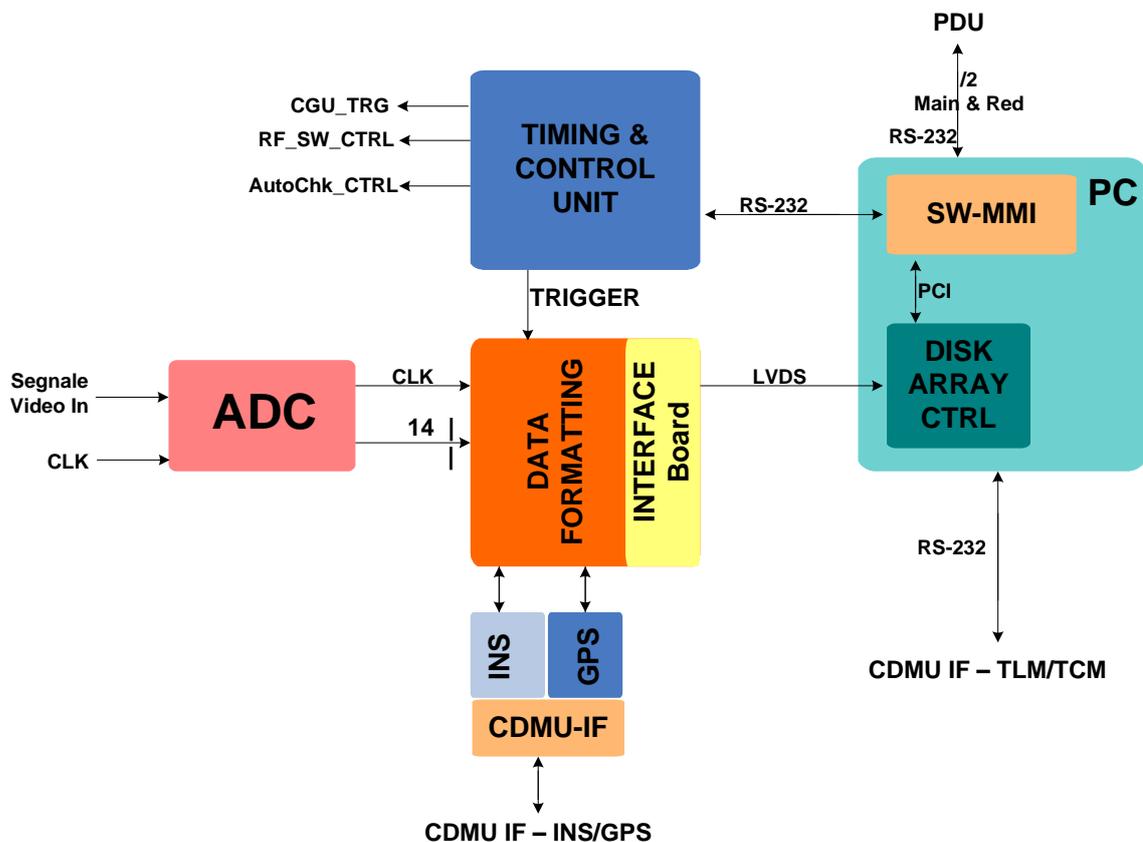


Figura 1-10: Architettura generale ed interfacce dell'unità digitale

- *Data Formatting*: i campioni a 14 bit del segnale digitalizzato vengono formattati insieme con i dati ancillari ricevuti dalla CDMU, con due PRI marker e due PRI counter. La struttura generale del formato dati PRI è mostrata di seguito (ogni word è lunga 16 bit):
 - A. Mark1 (0xAAAA)
 - B. Mark2 (0xBBBB)
 - C. Data Counter 1
 - D. Data Counter 2
 - E. 4095 Word dell'ADC
 - F. 18 word di dati ancillari dalla CDMU

dove:

Data Counter 1: I 10 bit meno significativi sono lo Scan Number; gli altri 6 bit sono i 6 bit meno significativi del PRI number;

Data Counter 2: sono gli ultimi 16 bit dei 22 bit di lunghezza del PRI Number;

ADC Word:

a) **No Presumming Mode**

I 14 bit LSB sono il campione in rappresentazione complemento a 2;

Equazione di conversione Campione-Volt:

$$\text{Valore In Volt} = (\text{Campione} \times 0,75) / \text{Massimo Valore}$$

$$\text{Massimo Valore} = 8192$$

b) **Presumming Mode**

la parola rappresenta i 16 bit, in rappresentazione complemento a 2, della somma coerente di 8 successivi echi dove ciascun eco è stato rappresentato con un campione a 13 bit in complemento 2.

Equazione di conversione Campione-Volt:

$$\text{Valore In Volt} = (\text{Campione} \times 0,75) / \text{Massimo Valore}$$

$$\text{Massimo Valore} = 4096$$

Ancillary Data: i 20 byte dei dati ancillari utili sono preceduti da 8 byte usati come marker di sincronizzazione.

- *Interface to Disk Array Controller:* questo sottosistema è responsabile dell'implementazione del Protocollo del Disk Array Controller e di trasmettere il pacchetto dati PRI formattato.

1.4.3.3 Interfacce

Le interfacce dell'unità digitale verso la CDMU sono:

- 1) *CDMU IF-INS/GPS:* questa interfaccia è una RS-232 operante a 9600 8N1. Questa interfaccia è unidirezionale in modo che i dati ancillari sono trasmessi dalla CDMU e ricevuti dall'unità digitale allo scopo di raccogliarli ed immagazzinarli in un modo sincrono insieme con i relativi echi digitalizzati;
- 2) *CDMU IF-TLM/TLC:* questa interfaccia è una RS-232 operante a 9600 8N1. Il controller PC è responsabile di decodificare e di deformatte correttamente i telecomandi ricevuti dalla CDMU e di trasmetterli all'unità digitale; allo stesso tempo è incaricata di raccogliere i dati trasmessi dall'unità digitale, di formattarli e di trasmetterli alla CDMU.

1.4.4 Unità RF

La RFU (Radio Frequency Unit) è il sottosistema del SORA che genera, trasmette e riceve gli impulsi radar. È costituita dai seguenti blocchi:

- FGU
- Trasmitter Chain (Catena di Trasmissione)
- Receiver Chain (Catena di Ricezione)
- Duplexer
- Antenna

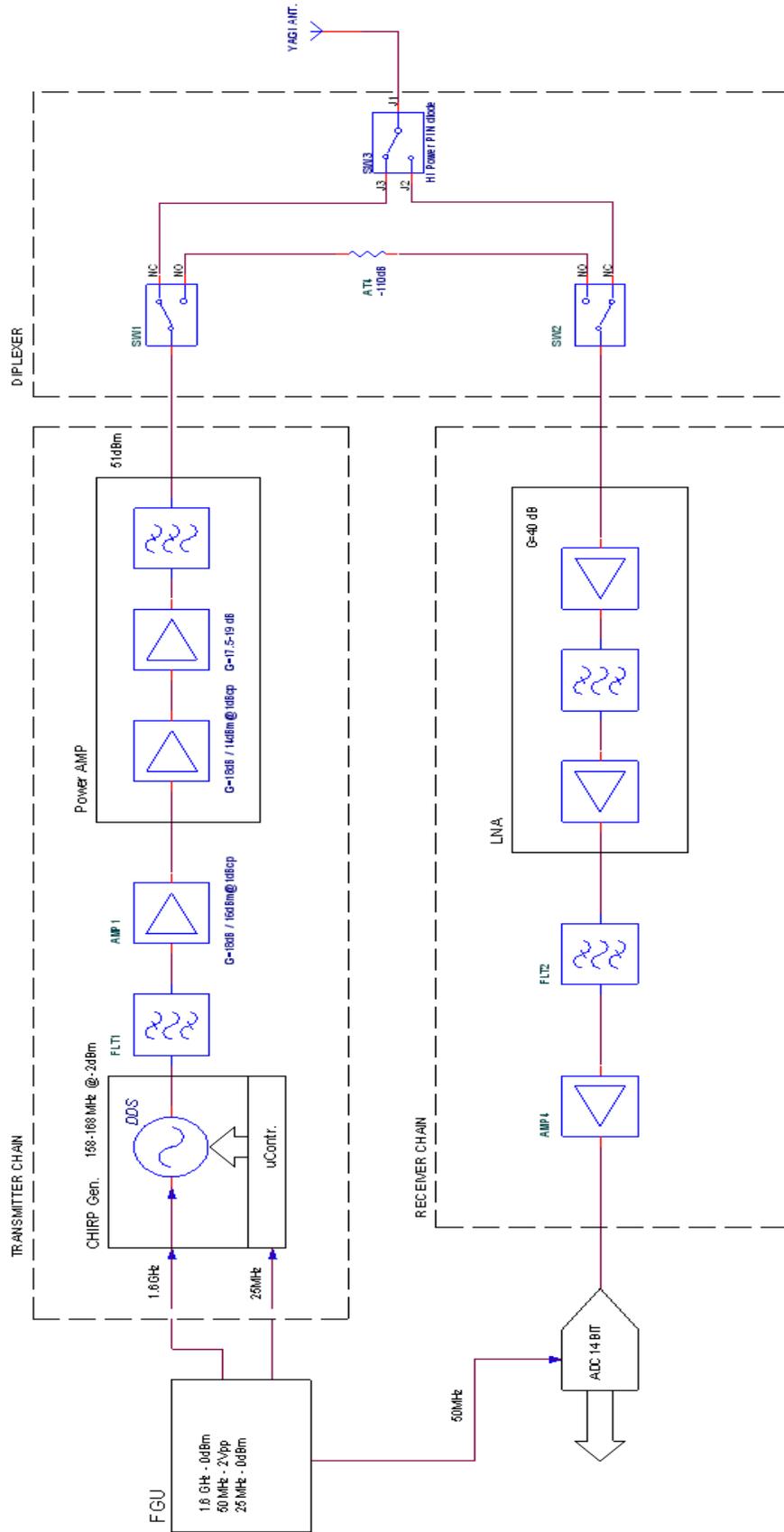


Figura 1-11: Schema a blocchi dell'RFU

1.4.4.1 FGU

L'FGU (Frequency Generator Unit) è l'unità che genera i segnali di sincronismo. L'oscillatore principale è basato su dispositivo al quarzo ultrastabile e termostato OCO (Oven-Controlled Crystal Oscillator) che genera un segnale a 100MHz. Attraverso divisori di frequenza e dispositivi a PLL, dal segnale di riferimento a 100MHz si ottengono i segnali di clock utilizzati per sincronizzare l'impulso radar trasmesso e i dispositivi digitali e di timing del sistema. Le frequenze di tali segnali sono: 1,6GHz; 50MHz; 25MHz.

1.4.4.2 Transmitter Chain

La catena di trasmissione ha il compito di generare il segnale di chirp, condizionarlo e amplificarlo opportunamente prima che venga mandato in antenna. È costituita da:

- a) chirp generator
- b) filtro passa banda
- c) amplificatore driver
- d) amplificatore di potenza

Il chirp generator è il circuito che genera i segnali di chirp che sono degli impulsi a modulazione lineare di frequenza di cui parleremo nel Capitolo 3.

Il cuore del chirp generator è il dispositivo AD9858 dell'Analog Devices che è un DDS (Direct Digital Synthesizer) ossia un generatore di frequenza a sintesi digitale diretta. Esso è completamente programmabile e può generare attraverso un suo DAC interno una forma d'onda periodica sinusoidale partendo da un singolo oscillatore di riferimento (1,6GHz). Nel nostro caso il DDS è programmato per sintetizzare un segnale sinusoidale con modulazione di frequenza lineare in banda VHF (Very High Frequency, banda 30 ÷ 300 MHz). La programmazione del AD9858 è affidata ad un microcontrollore che ne gestisce tutte le funzionalità ed è pilotato da un segnale di clock di 25 MHz prelevato dal FGU. Il chirp prodotto ha una banda di 10MHz intorno alla frequenza centrale di 163 MHz ed ha una durata di 3µs. il livello di potenza del segnale in uscita dal chirp generator è di -2dBm.

Il filtro passa banda FLT1 serve per eliminare le repliche del segnale di chirp che vengono prodotte con esso. L'amplificatore driver introduce una prima amplificazione

per pilotare correttamente l'amplificatore di potenza. Mentre l'amplificatore di potenza innalza il livello del segnale in modo da essere sufficiente a pilotare l'antenna.

1.4.4.3 Receiver Chain

La catena di ricezione ha il compito di condizionare il segnale catturato dall'antenna e portarlo all'ingresso analogico dell'ADC a 14 bit, di cui parleremo nel Capitolo 4, che preleva il clock di campionamento a 50 MHz dall'FGU. È costituito da un LNA (Low Noise Amplifier) cioè amplificatore a basso rumore, un filtro passa banda ed un amplificatore che adatta il livello del segnale in modo da poter pilotare opportunamente il convertitore.

1.4.4.4 Duplexer

Il duplexer o diplexer, permette di commutare la catena di trasmissione, quella di ricezione e l'antenna. Si compone di tre switch SW1, SW2, SW3 e un attenuatore. SW1 e SW2 sono degli interruttori elettromeccanici RF 12VDC 4GHz della RelComm Technologies, che se posti nella posizione NO (Normally Open) settano l'unità nella modalità **autocheck** isolando completamente l'antenna sia dall'apparato di ricezione che da quello di trasmissione. I due NO sono collegati tramite un attenuatore da 110dB che serve a simulare l'effetto della trasmissione e della ricezione sul segnale. In altre parole la modalità **autocheck** chiude il trasmettitore direttamente sul ricevitore attraverso l'attenuatore. Mentre nella posizione NC (Normally Close) connettono rispettivamente la catena di trasmissione all'ingresso J2 e la catena di ricezione all'ingresso J3 di SW3. SW3 è uno switch PIN diode cioè un relay a stato solido ed a secondo della sua posizione, connette l'antenna o alla catena di trasmissione o a quella di ricezione.

	SW1	SW2	SW3
TX	NC	NC	J3
RX	NC	NC	J2
Auto check	NO	NO	x

Tabella 1-7: Commutazioni degli switch del Duplexer

1.4.4.5 Antenna

L'antenna è di tipo direttivo Yagi-Uda a 6 elementi con guadagno di 13dB, essa serve ad irradiare i segnali di chirp prodotti dalla catena di trasmissione e catturare gli echi radar di ritorno per fornirli alla catena di ricezione. Non fa parte del payload ma è montata sul fondo della gondola in modo da consentire il nadir-looking della superficie sorvolata.

1.5 Considerazioni ambientali

La missione SORA avrà luogo a bordo di un pallone stratosferico poiché come discusso nell'introduzione si traggono vantaggi rispetto alle missioni con radar a penetrazione, sia da satelliti che da aerei. Ma la stratosfera presenta caratteristiche ben diverse da quella della troposfera, in particolare:

- bassa pressione 10mbar (e di conseguenza aria rarefatta)
- bassa temperatura, circa -2°C

Una pressione al di sotto di quella atmosferica potrebbe causare lo scoppio dei condensatori elettrolitici. L'alimentatore del PC di Bordo è stato perciò inglobato nella resina poliestere in modo da irrobustire l'involucro dei condensatori. Inoltre eventuali condensatori che non avrebbero superato il test a termovuoto sarebbe stati sostituiti con condensatori per applicazioni spaziali.

La rarefazione dell'aria risulta rilevante a fini della dissipazione del calore dei vari dispositivi elettronici di cui si compone il payload. Tali dispositivi in genere presentano sistemi alettati o ventole di raffreddamento che consentono uno scambio di calore per convezione, ma questo meccanismo è possibile solo se è presente un fluido. L'idea è quella di dissipare calore non per convezione ma per conduzione. Alette e ventole varie sono state perciò sostituite con dei ponti termici realizzati in rame o in alluminio,

collegati alla piastra (in alluminio) di montaggio del radar che considerando la bassa temperatura si può ritenere che si mantenga uniformemente intorno ai -2°C .

La Figura 1-12 mostra una foto recente del payload dalla quale si possono notare le modifiche di cui sopra.



Figura 1-12 : Payload SORA in fase di messa a punto

Il payload di SORA con le modifiche che consentono un corretto funzionamento anche in condizioni di bassa pressione e di bassa temperatura, è stato testato nella camera a termovuoto del C.I.S.A.S. di Padova (Figura 1-13) che consente di riproporre caratteristiche molto simili all'ambiente che la missione troverà.

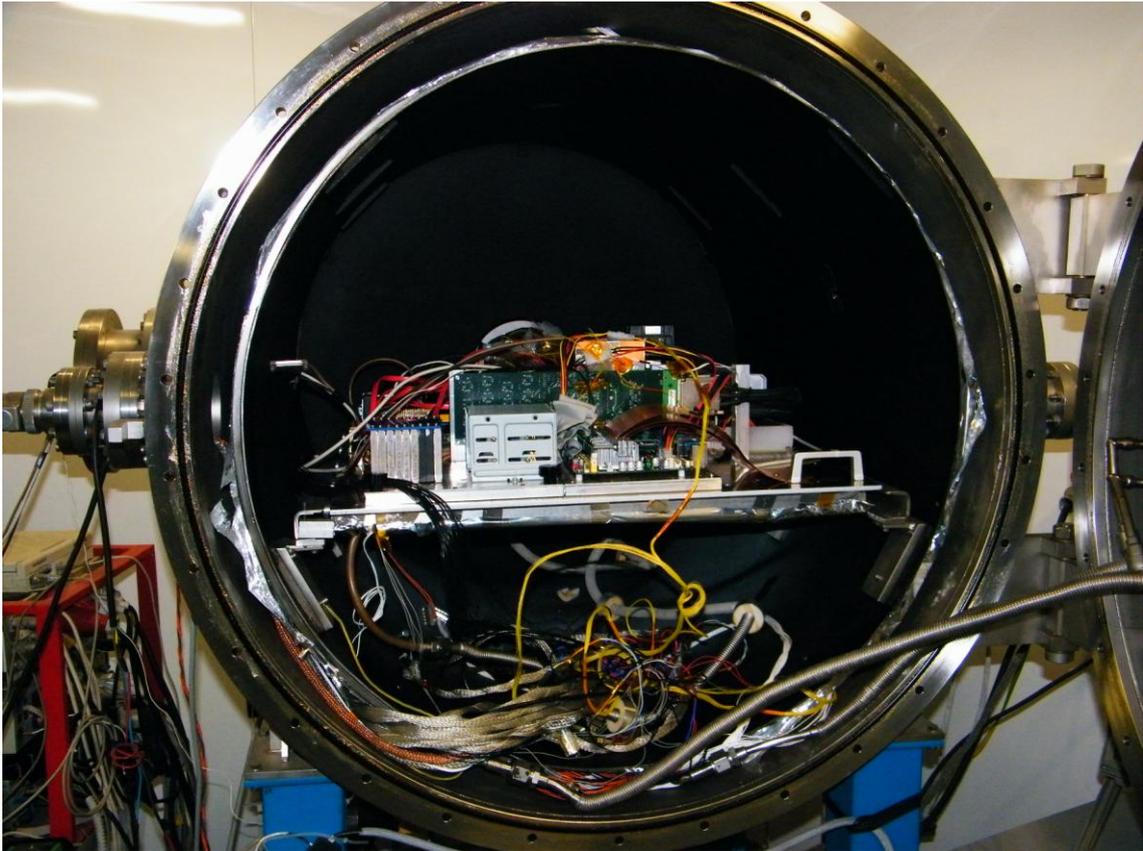


Figura 1-13 : Camera a termovuoto

I risultati dei giorni di test possono essere riassunti nel seguente modo:

1. Tutte le sequenze di comando e telecomunicazione tra CDMU e radar sono state provate e funzionano correttamente;
2. Le sequenze di accensione e spegnimento del radar tramite mappa precaricata sono state provate e funzionano correttamente;
3. Il radar è stato mantenuto a bassa pressione (10 mbar) per un totale di circa 24 h, non mostrando alcun problema;
4. Il radar è stato in funzionamento per circa 1.8 ore a bassa temperatura (piastra radar a circa $-2\text{ }^{\circ}\text{C}$). Le bandelle di rame riescono a trasferire gran parte del calore verso la piastra. Comunque saranno studiati ed implementati altri accorgimenti per cercare di ridurre ancora le temperature sulla scheda disk array controller.

2 CENNI SUL TELERILEVAMENTO

In questo capitolo vengono riportati brevemente i concetti principali dei sensori per il telerilevamento e la caratterizzazione dei segnali di cui fanno uso.

2.1 Concetti Generali

Il telerilevamento, è l'insieme delle tecniche per l'acquisizione a distanza e l'elaborazione di informazioni di vario tipo relative alla superficie della Terra o di altri pianeti.

Il telerilevamento quindi è la scienza per mezzo della quale le caratteristiche di un oggetto possono essere identificate, misurate e analizzate senza entrare in contatto diretto con esso.

Studia la radiazione elettromagnetica diffusa o emessa da un oggetto. Gli strumenti utilizzati per rilevare ed analizzare questa radiazione sono i cosiddetti sensori remoti.

Questi possono essere collocati su piattaforme terrestri, palloni, veicoli aerei e spaziali e si dividono in due categorie: sensori attivi e sensori passivi.

I sensori passivi servono a rilevare la radiazione elettromagnetica diffusa o emessa da fonti naturali, come, ad esempio, il sole.

L'energia del sole può essere o diffusa, ed è il caso delle lunghezze d'onda visibili, o assorbita ed emessa nuovamente, come per le lunghezze d'onda dell'infrarosso termico.

Per quanto riguarda l'energia diffusa, ciò può avvenire solo quando il sole illumina l'oggetto in osservazione, pertanto non di notte. L'energia emessa, come l'infrarosso termico, può essere invece misurata sia di giorno che di notte.

I sensori attivi rilevano la risposta riflessa da un oggetto irradiato da una fonte di energia generata artificialmente da loro stessi. Per questo motivo essi devono essere in grado di emettere una considerevole quantità di energia, in ogni caso sufficiente ad

illuminare il bersaglio. La radiazione emessa raggiunge l'oggetto in osservazione e la sua frazione diffusa viene rilevata e misurata dal sensore, come nel caso dei RADAR.

Tra i vantaggi dei sensori attivi, vi è la possibilità di effettuare misure ad ogni ora del giorno e della notte e, nel caso dei RADAR, anche in ogni condizione meteorologica.

I sensori per il telerilevamento sono tipicamente caratterizzati dal tipo di informazione che si vuole ottenere, a seconda della misura e dinamica dell'oggetto o del fenomeno in osservazione, e sulla base di tre parametri principali: spettrale, spaziale e di intensità. Inoltre, a seconda dell'intervallo di lunghezze d'onda da analizzare, i sensori, siano essi attivi o passivi, si suddividono in sensori elettroottici e a microonde.

2.2 Sensori RADAR

Il **RADAR** (**R**adio **D**etection **A**nd **R**anging) [3] è un sensore attivo operante nella banda di frequenza delle microonde e che ha, fondamentalmente, la funzione di misurare le caratteristiche elettromagnetiche di un oggetto "illuminato" da segnali inviati dal RADAR stesso, nonché la sua distanza, attraverso il calcolo del tempo di propagazione di un segnale inviato ad un bersaglio riflettente. La sorgente di questi segnali consiste in un trasmettitore, che genera la radiazione elettromagnetica, ed in un'antenna, che irradia nello spazio, in modo opportuno, questa radiazione, focalizzandola sulla superficie da osservare. Se questa superficie è scabrosa, l'onda elettromagnetica viene diffusa in tutte le direzioni. In particolare, una parte dell'energia elettromagnetica viene *retroirradiata*. Tale energia è captata dall'antenna, che è collegata al ricevitore, che rileva il segnale e lo registra in modo da poter essere, poi, elaborato. La discriminazione tra i segnali di ritorno (gli "echi RADAR"), ricevuti da differenti direzioni di propagazione, può essere realizzata con tecniche differenti, che coinvolgono la discriminazione temporale (echi che ritornano in istanti differenti appartengono a bersagli a differente distanza dal RADAR), la discriminazione angolare utilizzando gli angoli **azimuth** e di **elevazione** (echi provenienti da direzioni maggiori dell'ampiezza di fascio a 3dB dell'antenna non sono ricevuti), e la discriminazione in termini di scostamento Doppler (echi con scostamento Doppler differente appartengono a bersagli in posizioni differenti, perché la componente radiale della velocità relativa sensore-bersaglio è differente).

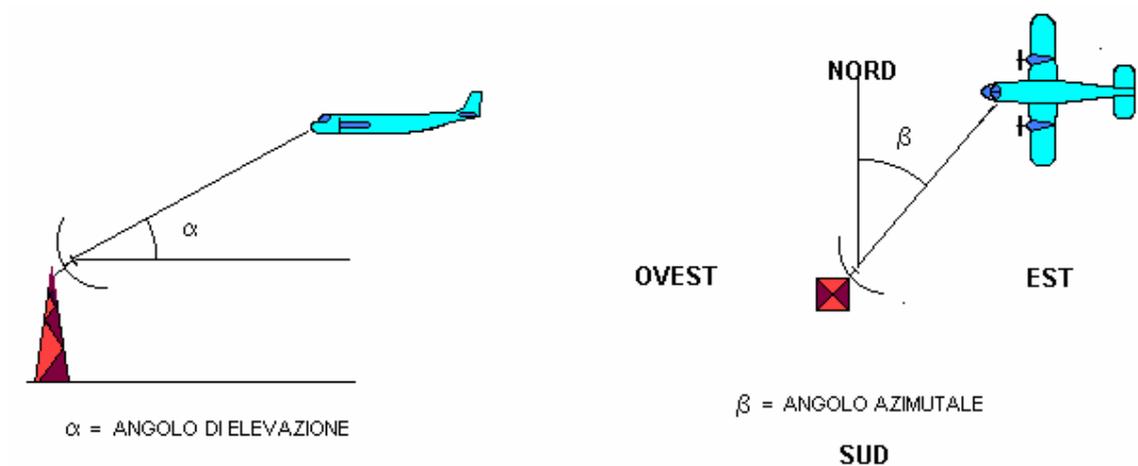


Figura 2-1: Angoli utilizzati dal RADAR per localizzare i bersagli

I componenti di un sistema RADAR sono quindi:

- un generatore di impulsi, che invia, con una certa frequenza di ripetizione, impulsi di energia a microonde;
- un trasmettitore;
- un duplexer (è un dispositivo che collega l'antenna al trasmettitore quando deve essere inviato l'impulso, e la collega al ricevitore quando deve essere ricevuto l'eco);
- un'antenna direttiva, che focalizza l'energia associata agli impulsi in un fascio;
- un dispositivo di immagazzinamento dati, per elaborazioni successive (*postprocessing*), o un dispositivo di visualizzazione *realtime* (un monitor, ad esempio).

I RADAR possono trasmettere segnali continui (si parla di RADAR *Continuous Wave*, o CW), oppure degli impulsi sinusoidali di durata finita (in questo caso si parla di RADAR *ad impulsi*). In questo secondo caso, il numero di impulsi trasmessi al secondo si chiama PRF (*Pulse Repetition Frequency*, frequenza di ripetizione dell'impulso). Un impulso (in inglese, *pulse*) di durata t e di frequenza f_0 , è un tronco di senoide:

$$s(t) = A \cos 2\pi f t_0 \quad -\frac{\tau}{2} \leq t \leq \frac{\tau}{2} \quad (2.1)$$

il cui involuppo (ossia la curva che unisce i massimi locali) è un impulso rettangolare di durata τ .

Un parametro di fondamentale importanza, nella comprensione della capacità del RADAR nel discriminare due bersagli in distanza (ovvero per determinare la risoluzione in *range*), è la **banda** del segnale trasmesso, ossia l'intervallo di frequenze (centrato intorno a f_0) "occupato" dall'impulso.

È possibile dimostrare che la banda di un impulso rettangolare monocromatico del tipo (2.1), con frequenza portante pari a f_0 , è pari a: $B=1/\tau$. La durata degli impulsi è molto piccola rispetto al periodo di tempo che intercorre fra due impulsi successivi (tipicamente, il prodotto $\tau(\text{PRF})$ è circa pari a 1/1000).

L'intervallo di tempo, fra emissione dell'impulso e ricezione dell'eco, può essere misurato con grande accuratezza e, quindi, essendo nota la velocità di propagazione delle onde elettromagnetiche nello spazio (pari a $c \cong 3 \times 10^8$ m/s nel vuoto), è possibile calcolare la distanza R dal RADAR dell'area osservata.

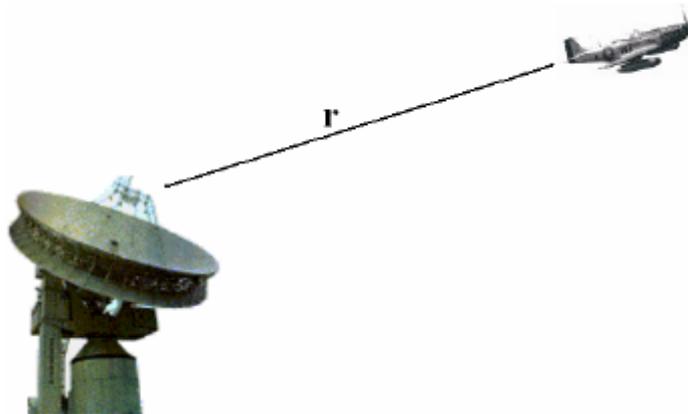


Figura 2-2: Distanza RADAR-Bersaglio

Dal momento che l'impulso trasmesso percorre una distanza pari a $2R$ (andata e ritorno) in un tempo T , è ovvio che: $R=cT/2$. Naturalmente la portata del RADAR, cioè la distanza massima oltre la quale non riesce a rilevare la presenza degli oggetti è limitata. Tale limite è dovuto sia alla potenza dell'impulso trasmesso, che ovviamente è una quantità finita, ma soprattutto è dovuto alla impossibilità di ricevere echi di un impulso dopo la trasmissione di uno successivo, poiché in quel caso è impossibile rilevare l'impulso di appartenenza dell'eco e quindi il tempo impiegato dallo stesso per tornare indietro.

Quindi le funzionalità del RADAR sono limitate alla sua portata, oltre la quale il RADAR è "cieco", quindi tutte le discussioni seguenti si limitano ad oggetti alla portata del RADAR e non possono essere estese oltre.

Alla luce di quanto detto, supponendo di avere due oggetti nello spazio di osservazione del RADAR, se tali bersagli sono separati da una distanza ΔR , due echi (dello stesso impulso) verranno ricevuti dall'antenna del RADAR, e torneranno con una separazione temporale:

$$\Delta T = 2 \frac{\Delta R}{c} \quad (2.2)$$

se l'impulso trasmesso ha durata τ , per fare in modo che i due echi non si sovrappongano è necessario che sia :

$$\tau > 2 \frac{\Delta R}{c} \quad (2.3)$$

pertanto possiamo definire la **Risoluzione geometrica** (o **spaziale**) come la minima distanza relativa alla quale due oggetti possono trovarsi ed essere identificati come oggetti distinti:

$$\rho = \Delta R_{\min} = \frac{c\tau}{2} = \frac{c}{2B} \quad (2.4)$$

2.3 I chirp RADAR

Si nota dalla (2.4) che, per avere una risoluzione in range adeguata (sull'ordine del metro), occorrerebbe trasmettere segnali di durata troppo piccola (sull'ordine dei nanosecondi, ma al più è possibile trasmettere segnali con durata dei microsecondi). Questo perché la durata di un impulso è anche legata alla sua energia ed alla sua potenza; l'energia di un impulso è a sua volta legata alla capacità di rivelare la presenza di un bersaglio, e sono desiderabili impulsi ad alte energie, cioè a grandi valori di τ . Questo peggiorerebbe però la capacità di discriminare i bersagli in *range* (un vincolo che "spinge" verso valori piccoli di τ , ovvero bande grandi).

L'esigenza di avere impulsi con bande grandi e durate sufficientemente grandi, impossibile da ottenere con impulsi rettangolari, è risolta con tecniche di *modulazione* dell'impulso (nei RADAR si usano impulsi con modulazione lineare di frequenza, detti *chirp pulses*, o semplicemente **chirp**, che rompono il legame fra τ e B). In definitiva, a parità di durata dell'impulso, con una modulazione lineare in frequenza si aumenta la banda del segnale trasmesso.

Un segnale chirp ha la seguente espressione:

$$s(t) = A\Pi\left(\frac{t}{T}\right)\text{Re}\left\{e^{j2\pi f_0 t} e^{j2\pi \frac{\alpha}{2} t^2}\right\} = A\Pi\left(\frac{t}{T}\right)\cos\left(2\pi f_0 t + 2\pi \frac{\alpha}{2} t^2\right) \quad (2.5)$$

ed è caratterizzato da una fase parabolica e frequenza istantanea lineare:

$$\varphi(t) = 2\pi f_0 t + 2\pi \frac{\alpha}{2} t^2 \quad (2.6)$$

$$f(t) = \frac{\dot{\varphi}}{2\pi} = f_0 + \alpha t \quad (2.7)$$

dove la costante α è detta **chirp rate**.

Eseguendo la Trasformata di Fourier del segnale e con l'ausilio del principio di fase stazionaria, si vede come lo spettro di un chirp è un chirp in frequenza di durata αT il cui chirp rate si inverte e la banda è direttamente proporzionale alla durata nel tempo del chirp:

$$B = \alpha T, \quad \alpha = \text{rate del chirp} \quad (2.8)$$

Quindi, un RADAR ad impulsi chirp permette di disporre di ottime risoluzioni in range (ordine del m) con impulsi di durata dell'ordine dei millisecondi e, quindi, con potenze di picco basse. Praticamente, non potendo trasmettere istantaneamente tutta la banda, in quanto occorrerebbe un impulso troppo stretto, si trasmettono le frequenze sfasate nel tempo. Occorrerà quindi in fase di ricezione un ricevitore, che comprima le fasi in modo che, alla sua uscita, le frequenze saranno tutte sovrapposte ad un certo istante di tempo, cioè si avrà un impulso stretto focalizzato in un istante di tempo. Tali ricevitori sono dei *filtri adattati* che, in pratica, una volta ricevuto il chirp e convertito in banda base, esso è soggetto all'operazione di compressione, la quale consiste nel filtrare il segnale $s(t-t_0)$ mediante un filtro la cui risposta impulsiva è data da:

$$h(t) = \sqrt{\frac{B}{T}} x^*(-t) \quad (2.9)$$

Tale filtro è detto filtro di compressione o filtro adattato, in quanto la sua risposta impulsiva è strettamente legata al segnale per cui è progettato, nel caso di un sistema RADAR $x(t)$ rappresenta il chirp trasmesso in condizioni ideali.

Le figure seguenti mostrano l'andamento di un segnale chirp in funzione del tempo ed il corrispondente andamento in frequenza.

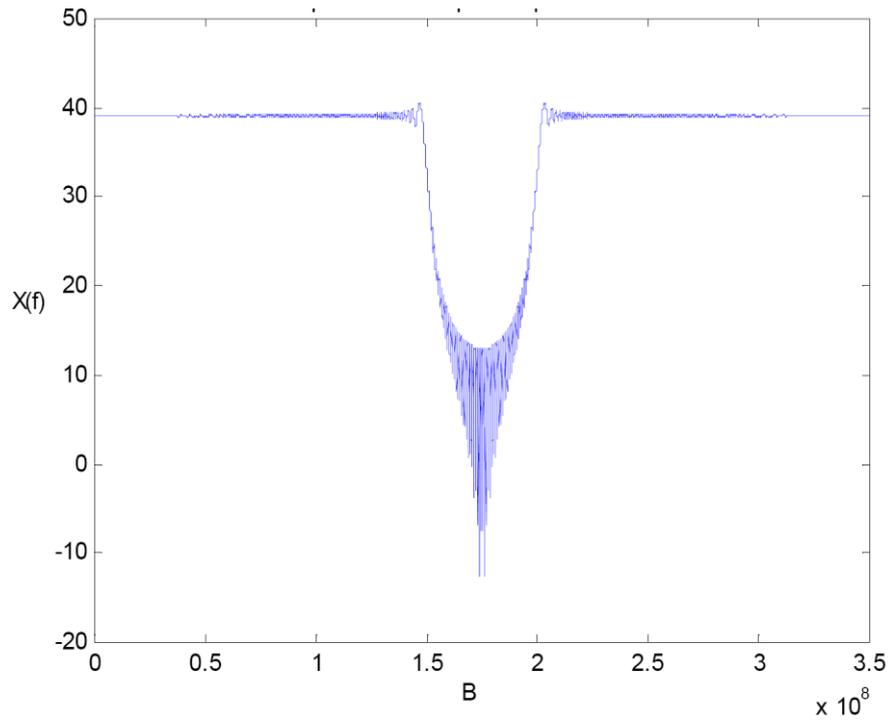


Figura 2-3: Modulo della FFT chirp

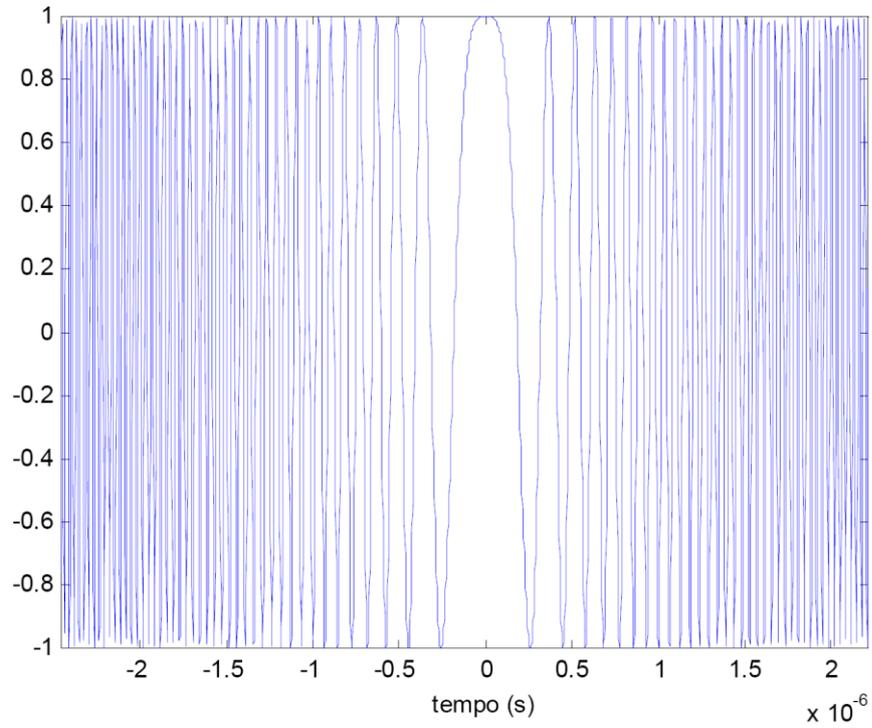


Figura 2-4: Parte reale del chirp

Osservando la Figura 2-5 si nota come le componenti della banda vengano trasmesse in tempi diversi rispetto a quello in cui dovrebbe essere concentrato l'impulso, e che ad ogni istante di tempo è associato una particolare componente del segnale.

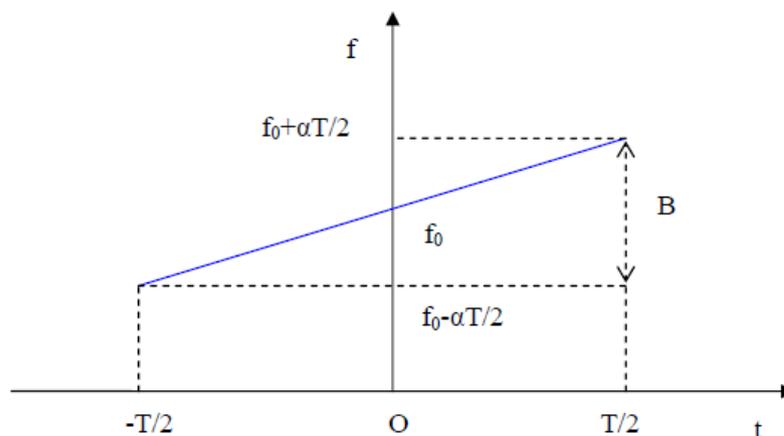


Figura 2-5: Caratteristica Frequenza-Tempo del chirp lineare

Tuttavia, poiché tale filtro ha una risposta impulsiva data dalla (2.9), quindi con rate opposto a quello del chirp trasmesso, accade che il filtro ritarda le componenti trasmesse prima e anticipa quelle trasmesse dopo in modo tale che le componenti frequenziali arrivino all'uscita del ricevitore tutte allo stesso istante così da sovrapporsi e convogliare l'energia del segnale nell'istante desiderato. Di conseguenza, il segnale di uscita dal filtro sarà un impulso stretto e alto, centrato su di un certo istante t_0 .

Infatti, analiticamente eseguendo la convoluzione tra l'impulso trasmesso e la funzione $h(t)$, si ricava che a valle di tale filtraggio il segnale prende la forma di un impulso sinc¹ alto \sqrt{TB} la cui durata da nullo a nullo è $2/B$ ma quella effettiva (cioè nella quale è concentrata la maggior parte dell'energia) è $1/B$.

Il segnale di uscita del filtro di compressione sarà quindi:

$$y(t) = \sqrt{TB} \operatorname{sinc}(B(t-t_0)) \quad (2.10)$$

¹ L'espressione analitica della funzione è $\operatorname{sinc}(x) = \frac{\sin(\pi x)}{\pi x}$

L'uscita, nel caso in cui $TB \gg 1$ e l'ampiezza dell'impulso trasmesso sia costante (unitaria per semplicità) ha un grafico riportato nella Figura 2-6.

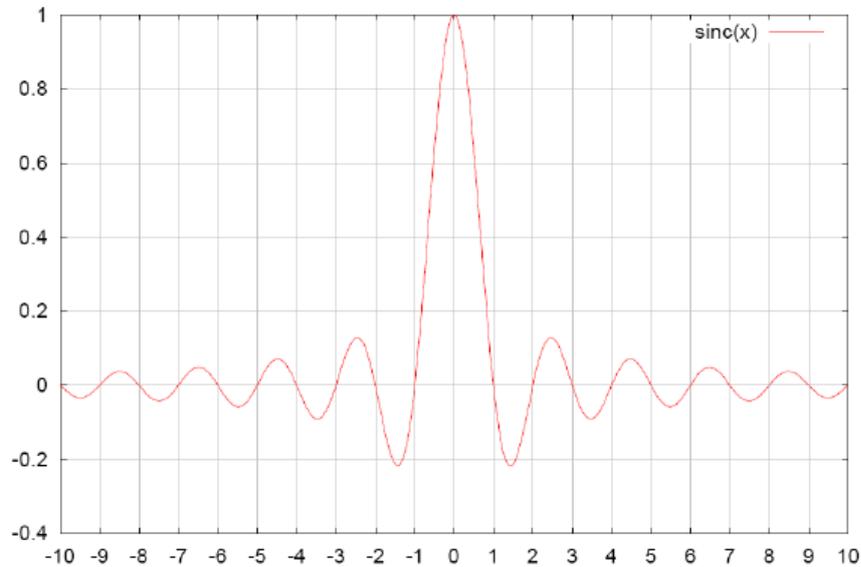


Figura 2-6: Grafico della funzione $\text{sinc}(x)$

Dunque, in conclusione, gli scopi di tale filtraggio sono quelli di:

Comprimere l'impulso ricevuto in modo che la risoluzione in range la cui nota espressione è:

$$\rho_R = \frac{c}{2} \tau \quad (2.11)$$

(con τ durata dell'impulso ricevuto) diventi dopo la compressione:

$$\rho_R = \frac{c}{2B} = \frac{c}{2\alpha T} \quad (2.12)$$

ovvero inversamente proporzionale alla durata dell'impulso trasmesso così da consentire la trasmissione di impulsi lunghi con basse potenze di picco.

La quantità:

$$BT = \frac{T}{1/B} \quad (2.13)$$

è detta fattore di compressione del filtro.

Consente di migliorare il rapporto segnale/rumore (SNR) di modo che, anche se a monte del filtro l'impulso ricevuto sta al di sotto del rumore, a valle emerge dal rumore, in quanto l'ampiezza dell'impulso passa da 1 a TB mentre il rumore resta costante. Nell'ipotesi di filtro conservativo, si può far vedere come SNR migliori proprio di BT .

3 CENNI SUL SOTTOCAMPIONAMENTO

L'acquisizione dati del SORA è incentrata sull'acquisizione degli echi RADAR di ritorno ossia l'aliquota riflessa dei segnali di chirp che incidono sulla superficie del bedrock del suolo della Groellandia.

La tecnica di campionamento adoperata è quella dell'undersampling o sottocampionamento che permette di ridurre la frequenza di campionamento con una conseguente riduzione del numero di campioni. Tale tecnica viene impiegata solitamente per comprimere la velocità di trasmissione e/o la dimensione dei dati in formato digitale. Il fattore di undersampling n è nella maggior parte dei casi un intero o una frazione razionale maggiore dell'unità che ne divide la frequenza. Siccome l'undersampling riduce la frequenza di campionamento, il Teorema del campionamento di Nyquist-Shannon naturalmente non viene infranto. Tale tecnica ha però lo svantaggio di andare incontro, durante l'inevitabile perdita di informazione, al famoso difetto di aliasing esigendo quindi filtri anti-aliasing con dei fianchi molto ripidi.

3.1 Introduzione

Il teorema di Shannon sul campionamento dice che condizione sufficiente, ma in generale non necessaria, perché un campionamento sia reversibile è che sia:

$$f_s \geq 2f_H \quad (3.1)$$

dove f_s è la frequenza di campionamento e f_H è la frequenza massima del segnale.

Per segnali passa basso tale condizione è anche necessaria, mentre per segnali passa banda si può scendere al di sotto della frequenza limite $2f_H$ continuando a garantire la reversibilità del campionamento.

Quello appena descritto è il concetto di sottocampionamento o undersampling. L'undersampling può essere realizzato purché valga la seguente relazione:

$$f_s \geq 2B \quad (3.2)$$

dove B è la larghezza di banda del segnale. La (3.2) esprime la condizione necessaria del sottocampionamento. Questo vuol dire che affinché si possa sottocampionare bisogna che la frequenza di sottocampionamento sia maggiore di $2B$, ma vedremo che non tutti i valori di f_s compresi tra $2B$ e $2f_H$ sono leciti.

3.2 Aliasing

È noto che ad un campionamento di un dato segnale analogico $x(t)$ nel dominio del tempo corrisponde una ripetizione periodica del suo spettro $X(f)$, a multipli di f_s , nel dominio delle frequenze. Rendere dunque il campionamento reversibile vuol dire evitare l'*aliasing* (sovrapposizione) delle repliche dello spettro nel dominio delle frequenze.

Il problema dell'*aliasing* diventa cruciale nello studio del sottocampionamento in quanto non sussiste la condizione dettata dal teorema di Shannon (3.1), dunque non è più garantita l'assenza di *aliasing*.

È possibile ricavare le relazioni che stabiliscono le frequenze di sottocampionamento "lecite", ovvero che garantiscono la reversibilità del processo. Si riportano in seguito le formule, ricavate in [4], che stabiliscono la condizione necessaria e sufficiente affinché il campionamento di un segnale passa banda sia reversibile:

$$n < \frac{f_L}{B} \quad (3.3)$$

$$\frac{2}{n+1} f_H < f_s < \frac{2}{n} f_L \quad (3.4)$$

con f_L frequenza minima del segnale e n intero positivo. Il caso $n = 0$ si riconduce al teorema di Shannon. Le formule appena riportate determinano n intervalli di frequenze a cui è possibile sottocampionare in maniera reversibile.

Nella scelta dell'opportuna (cioè minima possibile) f_s bisognerà poi tenere conto delle condizioni reali del sistema, ovvero del fatto che non esistono filtri ideali in fase di ricostruzione del segnale analogico e del fatto che quindi lo spettro del nostro segnale non è rettangolare ma avrà una discesa lenta ai bordi. Occorrerà cioè non spingersi su valori di n troppo bassi in modo da mantenere le repliche abbastanza distanziate, evitando che vadano a sovrapporsi. Il problema della sovrapposizione degli spettri implica anche un filtraggio del segnale da sottocampionare estremamente rigoroso onde evitare che bande al di fuori di quella di interesse, vadano a sovrapporsi con le repliche utili.

Lo spettro del segnale sottocampionato e quindi portato in banda base, sarà rovesciato o meno a seconda della posizione dello spettro del segnale originale rispetto alla frequenza f_s di campionamento scelta: se il corrispondente $n + 1$ è dispari l'ordinamento della banda si conserva, se invece è pari la banda è invertita.

3.3 Jitter

Per *jitter* si intende una piccola (rispetto al periodo del clock) variazione aleatoria degli istanti di campionamento. Il jitter è una problematica già presente nel campionamento "tradizionale", ma ha effetti ancor più sentiti nel sottocampionamento, dove riveste un ruolo determinante. Intuitivamente, basti pensare che avendo a che fare con segnali a frequenze molto elevate rispetto alla frequenza di campionamento, oscillazioni dell'ordine dei picosecondi degli istanti di campionamento possono portare a brusche variazioni dei valori del segnale come mostrato in Figura 3-1: un errore sull'istante di campionamento comporta un errore sul livello di tensione del segnale campionato tanto più grande quanto più elevata è la pendenza del segnale stesso.

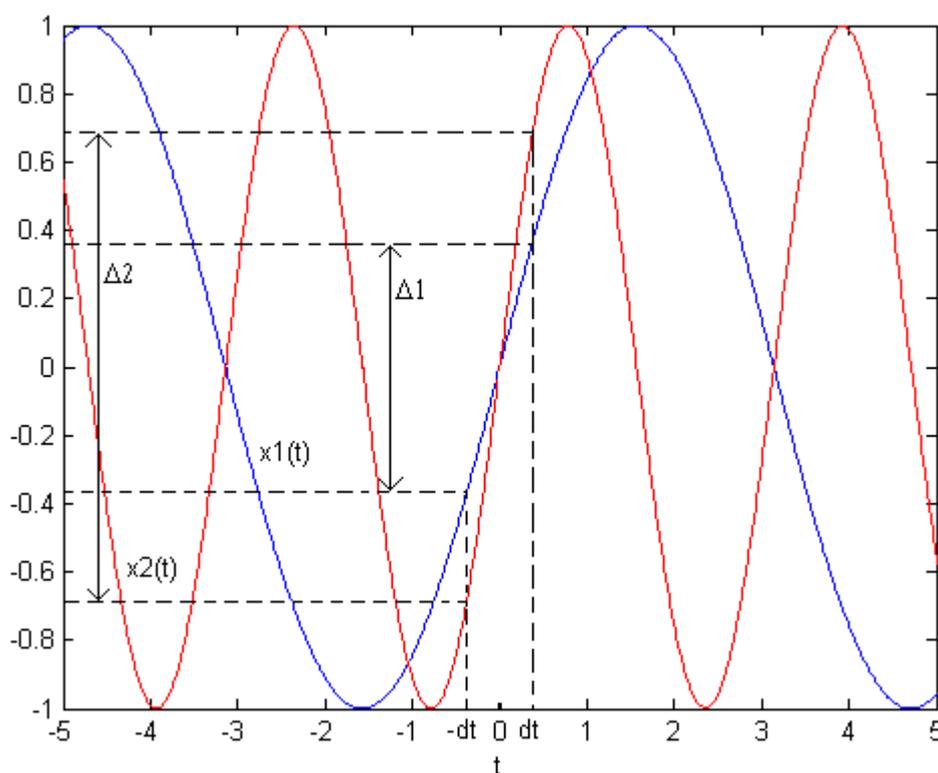


Figura 3-1: Al crescere della frequenza del segnale, aumenta l'errore dovuto al jitter

Il jitter totale di sistema si compone di due fattori: il jitter d'apertura dell'ADC, proprio del circuito di sample-and-hold interno al convertitore e riportato nei data sheet, e il jitter del clock; i due termini si combinano [5] in maniera RSS (*Root-Sum-Square*) secondo la formula:

$$t_j = \sqrt{t_{jCLK}^2 + t_{jADC}^2} \quad (3.5)$$

Mentre però il jitter d'apertura non può essere migliorato in quanto proprio dell'ADC, il jitter del clock influisce sulle prestazioni del sistema al variare delle caratteristiche del segnale analogico di ingresso.

La deviazione sugli istanti di campionamento provoca un degrado dell'accuratezza della conversione AD, che si manifesta nel dominio delle frequenze con l'introduzione di rumore di fase; conseguenza immediata è il peggioramento dell'SNR (*Signal to Noise*

Ratio) del sistema. La formula generale [6] che esprime l'impatto del jitter sull'SNR è la seguente

$$SNR_j = 10 \log_{10} \left(\frac{r_x(0)}{-r_x''(0) \cdot r_{t_j}(0)} \right) \text{ dB} \quad (3.6)$$

dove $r_x(t)$ è la funzione di autocorrelazione del segnale d'ingresso e $r_{t_j}(t)$ è la funzione di autocorrelazione del jitter del clock; è importante ricordare che la correlazione del segnale nell'origine coincide con la potenza, e che se t_j ha media nulla:

$$r_{t_j}(0) = \sigma_{t_j}^2 \quad (3.7)$$

dove $\sigma_{t_j}^2$ è la varianza del jitter.

La (3.6) è una formula del tutto generale; l'applicazione a due casi specifici porta ai seguenti risultati:

Segnale d'ingresso sinusoidale

In questo caso:

$$x(t) = A \sin(2\pi f_{IN} t) \quad (3.8)$$

e

$$r_x(t) = \frac{A^2}{2} \cos(2\pi f_{IN} t) \quad (3.9)$$

Segue che:

$$SNR_j = 20 \log_{10} \frac{1}{2\pi f_{IN} \sigma_{t_j}} = -20 \log_{10} (2\pi f_{IN} \sigma_{t_j}) \quad (3.10)$$

Nel caso di ingresso non sinusoidale, è possibile continuare ad utilizzare la (3.10) sostituendo ad f_{IN} la frequenza centrale della banda del segnale; in tal modo si giunge ad una stima approssimata dell'SNR reale.

Segnale d'ingresso con spettro di potenza rettangolare

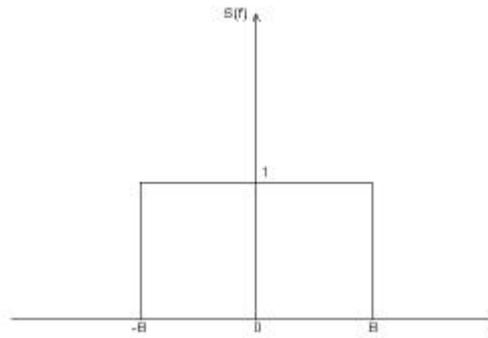


Figura 3-2

$$S(f) = \text{rect}\left(\frac{f}{2B}\right) \quad (3.11)$$

$$r_x(t) = \mathfrak{F}^{-1}\{S(f)\} = 2B \frac{\sin(2\pi Bt)}{2\pi Bt} \quad (3.12)$$

$$SNR_j = 20 \log\left(\frac{\sqrt{3}}{2\pi B \sigma_{ij}}\right) \quad (3.13)$$

In entrambi i casi il risultato importante è che la degradazione dell'SNR dovuta al jitter dipende, oltre che dal valore quadratico medio del jitter, dalla frequenza del segnale d'ingresso, come intuitivamente già visto in Figura 3-1.

L'SNR interno all'ADC (riportato nei data sheet) e quello dovuto al jitter del clock si combinano in maniera RSS:

$$SNR = -20 \log_{10} \sqrt{10^{\frac{SNR_{ADC}}{10}} + 10^{\frac{SNR_j}{10}}} \quad (3.14)$$

Il rumore generato all'interno dell'ADC che va ad incidere sul SNR_{ADC} , raccoglie 3 termini:

1. Rumore di quantizzazione: dovuto all'errore nell'assegnare un unico valore (discreto) a diversi valori del segnale che si trovano nello stesso intervallo di quantizzazione.
2. Rumore termico.
3. Rumore dovuto al jitter d'apertura.

E'importante sottolineare [7] che la (3.6), e di conseguenza la (3.10) e la (3.13), forniscono una stima pessimistica del reale SNR; in altri termini, dato come specifica di progetto il valore dell'SNR, i vincoli sul jitter possono essere rilassati.

Si consideri ad esempio il caso di ingresso sinusoidale: affinché l' SNR_j non vada sotto una soglia stabilita occorre che il jitter non superi il valore seguente:

$$\sigma_{j \max} = \frac{1}{2\pi f_{IN}} 10^{\frac{SNR_{j \min}}{20}} \quad (3.15)$$

La (3.15) altro non è che la formula inversa della (3.10).

Si vuole ora illustrare come possa avvenire il fenomeno di crescita del rumore.

Il campionamento di un segnale con un clock poco stabile introduce una modulazione sul segnale stesso:

$$v(t) = v(t - \Delta t) \quad (3.16)$$

dove Δt è la variazione temporale dovuta al jitter. Si consideri il caso di segnale d'ingresso sinusoidale con ampiezza A e pulsazione ω_I :

$$v(t) = A \cos(\omega_I t) \quad (3.17)$$

Nel caso di jitter di natura sinusoidale con ampiezza picco-picco J e pulsazione ω_I :

$$\Delta t = j(t) = \frac{J}{2} \sin(\omega_I t - \omega_I j(t)) \quad (3.18)$$

si ottiene una modulazione di fase del segnale d'ingresso:

$$v(t) = A \cos(\omega_I (t - \Delta t)) = A \cos(\omega_I t - \omega_I j(t)) \quad (3.19)$$

Attraverso manipolazioni trigonometriche e considerando che $J \ll T_I$ (T_I periodo del segnale) è possibile semplificare la (3.19), giungendo ad un'espressione molto interessante:

$$v(t) \cong A \cos(\omega_I t) + A \frac{J \omega_I}{4} \cos((\omega_I - \omega_j)t) - A \frac{J \omega_I}{4} \cos((\omega_I + \omega_j)t) \quad (3.20)$$

Dalla (3.20) si evince che per effetto del jitter, alla riga del segnale originario si sommano due componenti spurie allocate a $\omega_I \pm \omega_j$ (Figura 3-3), la cui ampiezza dipende, oltre che da A , da J e ω_I (ampiezza del jitter e frequenza del segnale).

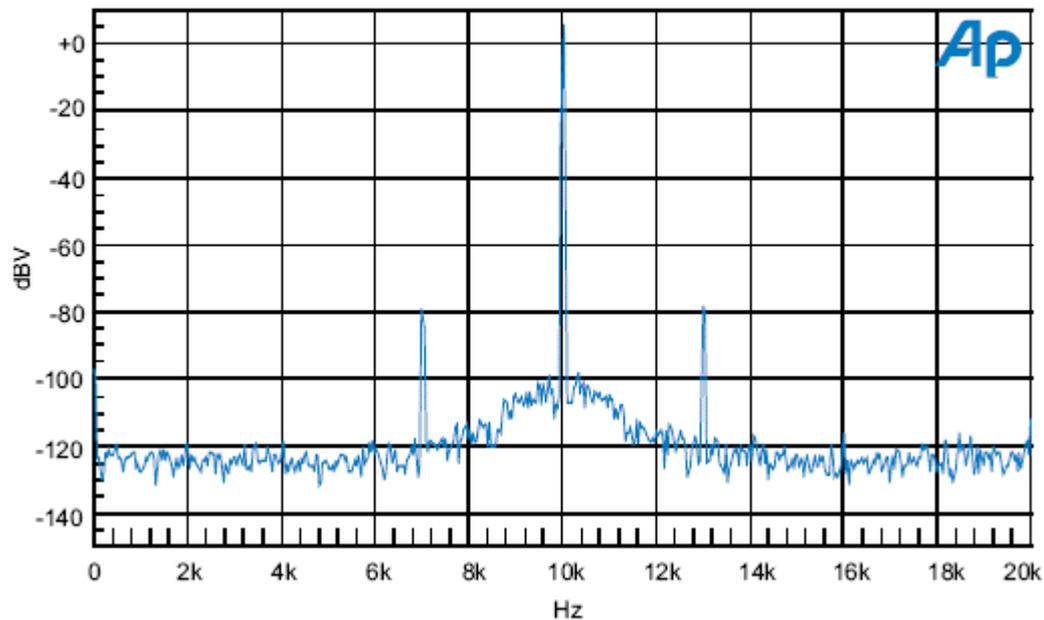


Figura 3-3: Esempio di jitter sinusoidale, con $f_I = 10\text{kHz}$, $f_j = 3\text{kHz}$

Questo accade nel caso di jitter sinusoidale. Nel caso generale, attraverso l'analisi di Fourier è possibile scomporre $j(t)$ nelle sue componenti spettrali.

Ognuna di esse modula in fase il segnale generando due righe nello spettro; le righe così ottenute si sovrappongono sull'asse delle frequenze dando origine ad un innalzamento del rumore di fondo.

3.4 SFDR

La caratteristica propria degli ADC di essere componenti non lineari porta inevitabilmente alla nascita di componenti spurie nello spettro del segnale campionato. Possono dunque nascere armoniche allocate a frequenze multiple della frequenza del segnale, o altre componenti spettrali generate da fenomeni quali la distorsione d'intermodulazione tra le diverse frequenze del segnale analogico.

Lo *Spurious-Free Dynamic Range* (SFDR) è un parametro importante nel valutare le prestazioni di un convertitore AD. Esso è definito [8] come il rapporto tra il valore

efficace dell'ampiezza del tono sinusoidale portante e il valore efficace della componente spettrale spuria con ampiezza maggiore all'interno della banda $[0, f_s/2]$, che può essere o non essere un'armonica.

L'SFDR può essere espresso in dBc, dB relativi alla portante (*carrier*), o in dBFS, dB relativi al fondo scala (*Full Scale*) dello strumento.

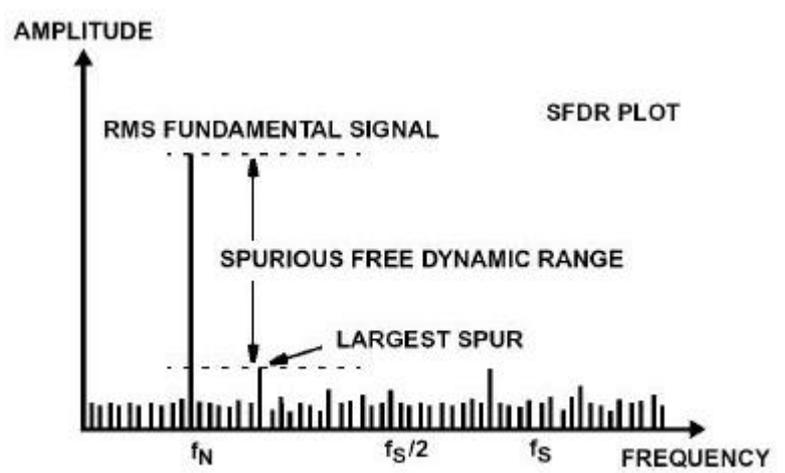


Fig 3-4: La largest spur è presa nella banda $[0, f_s/2]$ [9]

Le prestazioni in termini di SFDR non risentono del sottocampionamento, essendo tale parametro dovuto unicamente alle non linearità introdotte.

3.5 Conversione di frequenza

Le potenzialità dell'undersampling sono evidenti se si considera la semplificazione dell'operazione di conversione in banda base del segnale.

Come si è accennato nell'introduzione, nel campionamento tradizionale il segnale a radiofrequenza viene prima portato a frequenze intermedie mediante l'uso di mixer, e poi campionato. Il sottocampionamento permette la conversione del segnale in banda base (*down conversion*) senza l'utilizzo di mixer. Il segnale viene campionato a radiofrequenza; nelle Figure 3-5 e 3-6

ne sono riportati gli spettri:

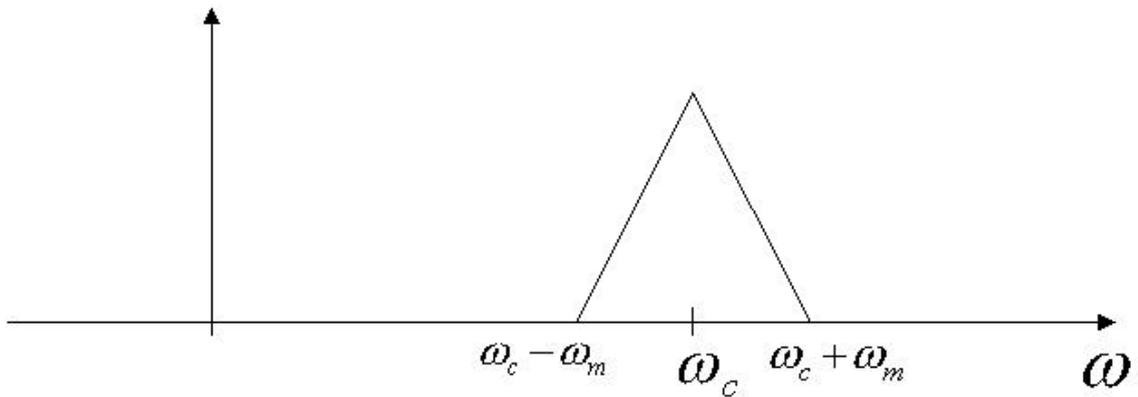


Figura 3-5: Spettro del segnale analogico

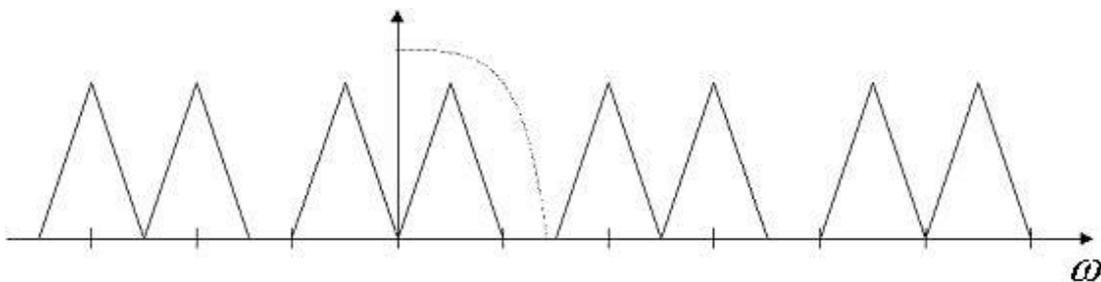


Figura 3-6: Esempio di spettro del segnale sottocampionato

In banda base è presente una replica del segnale originario; è possibile selezionarla ed eliminare le altre bande spurie tramite un filtraggio di tipo passa basso. Tale operazione può essere eseguita tramite un DDC (*Digital Down Converter*), il quale provvede a filtrare il segnale e a centrare la banda attorno allo zero, generando due segnali in fase e quadratura con banda dimezzata. Ad esempio, sottocampionando a 25 MHz un segnale di banda 10 MHz centrato in 30 MHz e programmando l'oscillatore locale interno al DDC ad una frequenza di 5 MHz, all'uscita del Digital Down Converter si ottengono due segnali (uno reale e l'altro complesso) ognuno con banda 5 MHz.

In Figura 3-7 è mostrata l'operazione complessiva di down conversion tramite DDC:

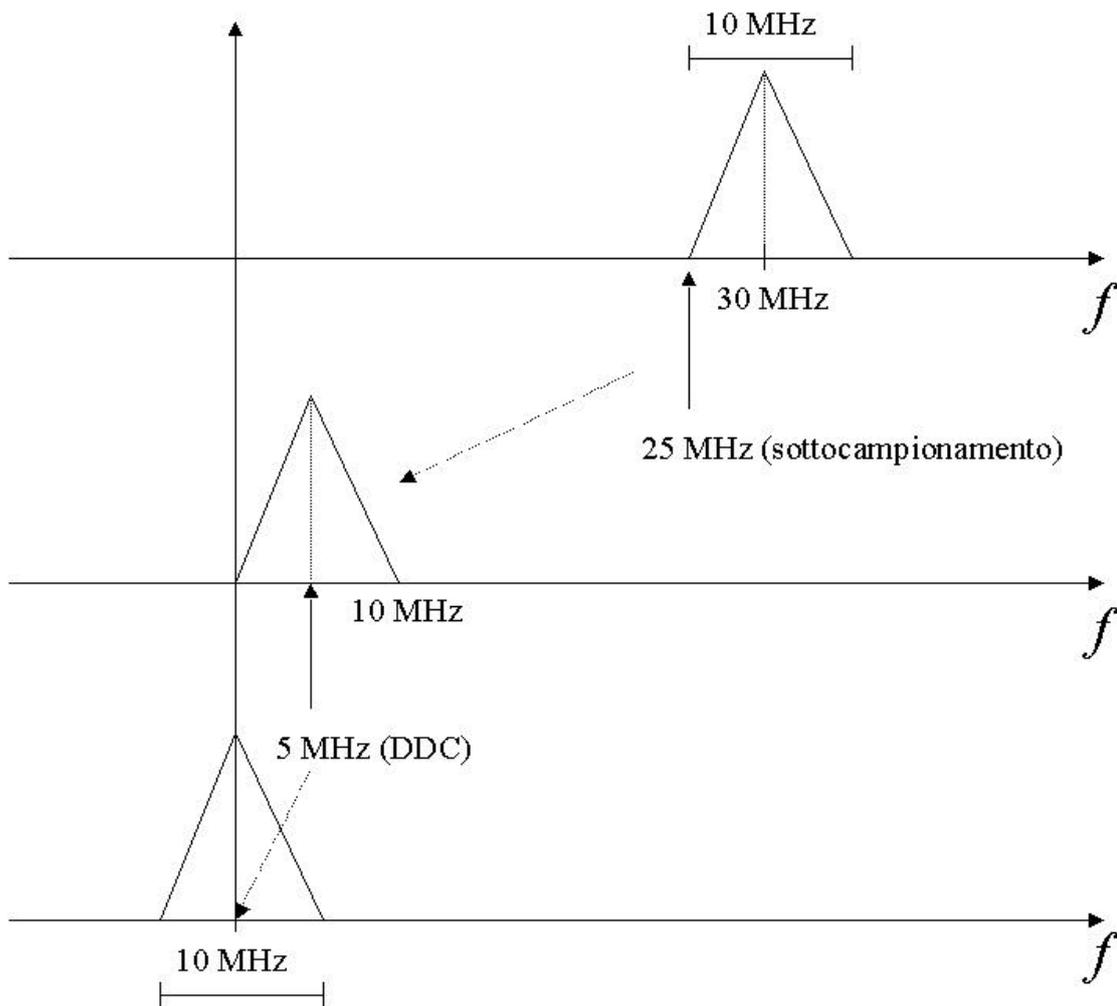


Figura 3-7: Down conversion tramite DDC

E' importante sottolineare come la conversione di frequenza viene fatta ora totalmente in digitale, in quanto il DDC opera sul segnale a valle dell'ADC.

L'assenza di mixer analogici comporta, come già accennato, un notevole miglioramento delle prestazioni ed una riduzione dei costi non indifferente.

Nella scelta della frequenza di campionamento, la soluzione ottimale sarebbe centrare la replica in banda base direttamente sullo zero in modo da ridurre la down conversion ad un semplice filtraggio passa basso. Questo porterebbe però ad "effetti collaterali" non desiderati: si consideri come esempio un segnale con banda centrata in $f_C = 30\text{MHz}$. Scegliendo $f_s = 30\text{MHz}$ la replica dello spettro sarebbe centrata nell'origine; nell'origine

però si troverebbe sovrapposta con lo spettro ribaltato (Figura 3-8) e con le armoniche introdotte dall'ADC, degenerate tutte a frequenza 0.

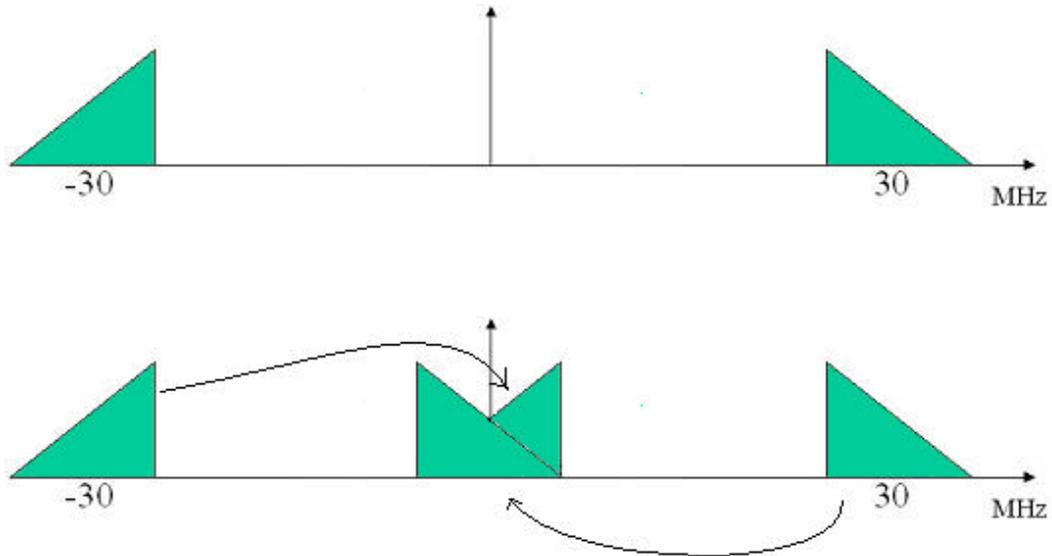


Figura 3-8: Sottocampionando con $f_s = f_c$ si ha aliasing attorno allo zero

A conferma di ciò, il valore $f_s = f_c$ non è mai compreso negli intervalli (3.3), (3.4). Si trovano infatti i range

$$\begin{aligned}
 n = 1 \quad & f_H < f_s < f_L \\
 n = 2 \quad & \frac{2}{3} f_H < f_s < f_L
 \end{aligned}
 \tag{3.21}$$

da cui è esclusa f_c , essendo $f_L < f_c < f_H$.

La situazione appena descritta si ripete per tutte le frequenze di campionamento $f_s = f_c/k$ (k intero positivo), cioè ogni qualvolta che f_c è multipla di f_s .

Le argomentazioni precedenti portano ad escludere una scelta di questo tipo.

4 DEMO BOARD DC996A E CONVERTITORE ANALOGICO DIGITALE LTC2208-14

Come già parlato nel primo capitolo l'unità digitale DU (Digital Unit) ha il compito di configurare ed immagazzinare i dati acquisiti e quelli ancillari, e di fornire i segnali di sincronizzazione del RADAR. I principali sottosistemi sono:

- 1) Analog To Digital Converter;
- 2) Timing and Control unit;
- 3) Data Formatting and Disk Array Controller Board

L'acquisizione dati viene realizzata con la Demo Board DC996A dell'ADC LTC2208-14 della Linear Technologies [10], con un clock di campionamento di 50MHz ricavato dall'unità RF che viene trasmesso alla scheda FPGA in formato elettrico LVDS insieme con i campioni a 14 bit del segnale digitalizzato. L'uscita del clk LVDS viene usata dalla scheda FPGA allo scopo di sincronizzare tutti i sottosistemi digitali allo stesso clock di riferimento (vedi Par. 1.4.3).

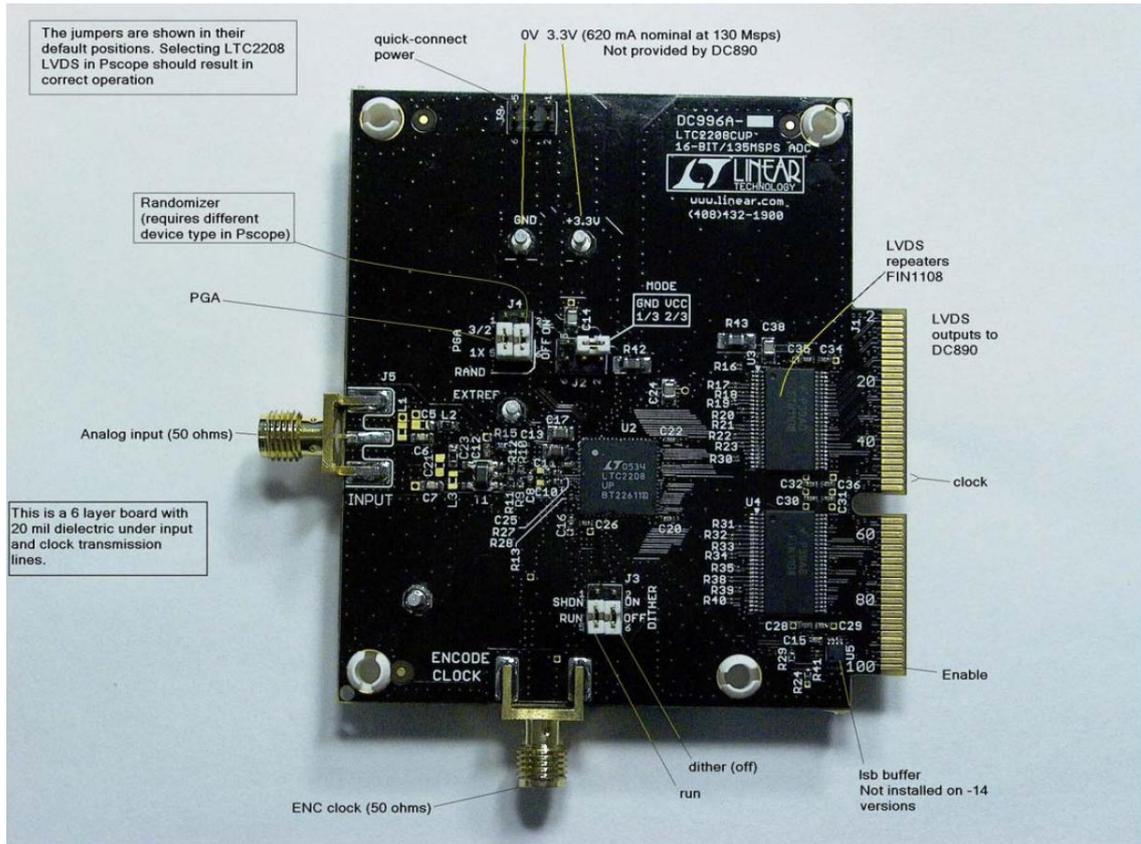


Figura 4-1: DC996A Setup

La Demo Board 996 supporta una famiglia di ADC 16/14 bit a 130Mps e supporta solo il funzionamento in formato LVDS, mentre per avere anche il funzionamento in formato CMOS va utilizzata la DC854 [11]. La 996 è già fornita dell'ADC prescelto, che viene selezionato in base alla risoluzione, alla frequenza di campionamento e alla frequenza d'ingresso. Inoltre fornisce un circuito di ottimizzazione degli ingressi analogici che si differenzia per frequenze d'ingresso al di sotto di 70MHz e per frequenze d'ingresso comprese tra 70MHz e 140MHz [10].

Specifiche

- 1) Sample Rate (Frequenza di Campionamento): 130Mps
- 2) 77.1dBFS Noise Floor (Rumore di Fondo)
- 3) 98dB SFDR

- 4) SFDR >81dB at 250MHz (1.5VP-P Input Range)
- 5) PGA Front End (2.25VP-P or 1.5VP-P Input Range)
- 6) 700MHz Full Power Bandwidth S/H
- 7) Optional Internal Dither
- 8) Optional Data Output Randomizer
- 9) LVDS or CMOS Outputs
- 10) Single 3.3V Supply
- 11) Power Dissipation: 1.32W
- 12) Clock Duty Cycle Stabilizer
- 13) Pin Compatible 16-Bit Version 130Msps: LTC2208 (16-Bit)
- 14) 64-Pin (9mm × 9mm) QFN Package

4.1 Descrizione

L'LTC[®]2208-14 [12] è un convertitore A/D a 14-bit con velocità di campionamento 130Msps, progettato per digitalizzare segnali ad alta frequenza, ad ampio range dinamico con frequenze di input fino a 700MHz. Il range d'ingresso dell'ADC può essere ottimizzato con un PGA (Programmable Gain Amplifier) a monte.

L'LTC2208-14 è perfetto per applicazioni di comunicazioni, con prestazioni in AC che comprendono un Rumore di Fondo di 77.1dBFS ed uno spurious free dynamic range (SFDR) di 98dB. Un jitter molto basso di 70fs_{RMS} consente il sottocampionamento di alte frequenze d'ingresso con eccellenti prestazioni in termini di rumore. Le specifiche massime in DC comprendono ±1.5LSB di INL, ±0.5LSB di DNL (senza codici mancanti).

L'uscita digitale può essere o differenziale in LVDS o single-ended in CMOS. Ci sono due opzioni di formato per le uscite CMOS: un singolo bus lavora a pieno data rate oppure bus demultiplexati lavorano a metà data rate. Un alimentatore d'uscita separato consente allo swing CMOS d'uscita di svariare tra 0.5V e 3.6V.

Gli ingressi ENC⁺ e ENC⁻ possono essere pilotati in modo differenziale o a singola uscita con ingressi PECL, LVDS, TTL o CMOS. Uno stabilizzatore opzionale di duty cycle del clock consente alte prestazioni alla massima velocità con un ampio range di clock duty cycle.

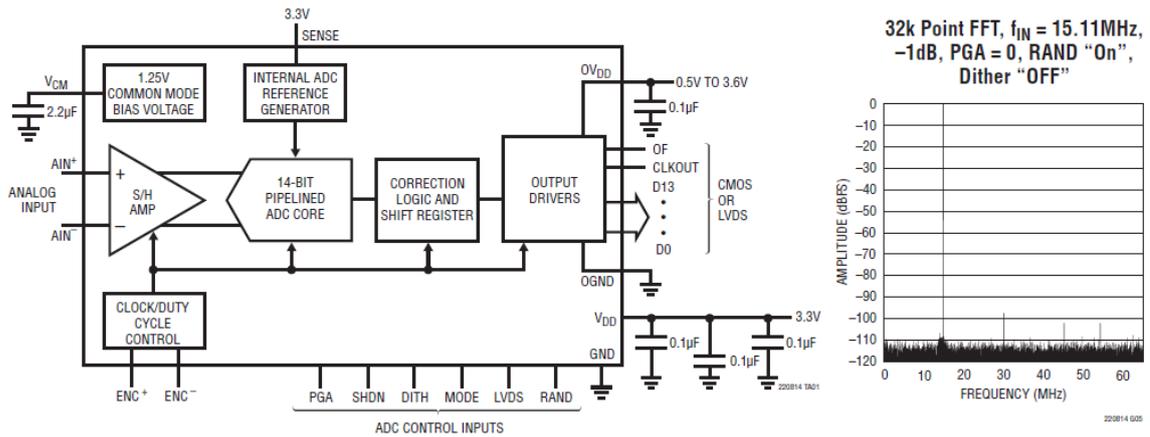
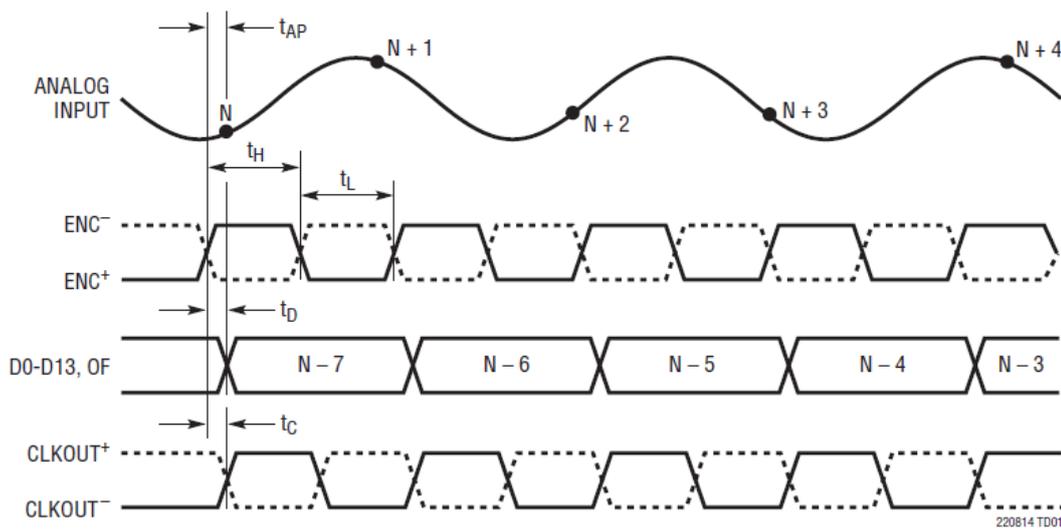


Figura 4-2: Applicazione Tipica

4.2 Diagramma Temporale

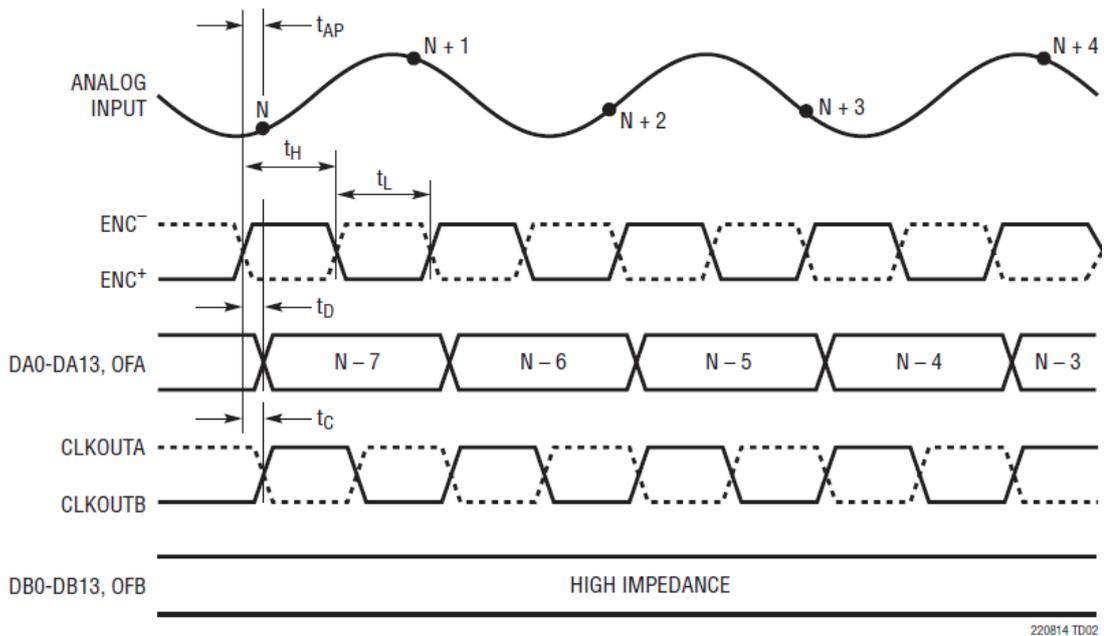
Tempificazione del Nodo d'Uscita LVDS

Tutte le uscite sono Differenziali ed hanno Livelli LVDS



Tempificazione del Modo d'Uscita Full-Rate CMOS

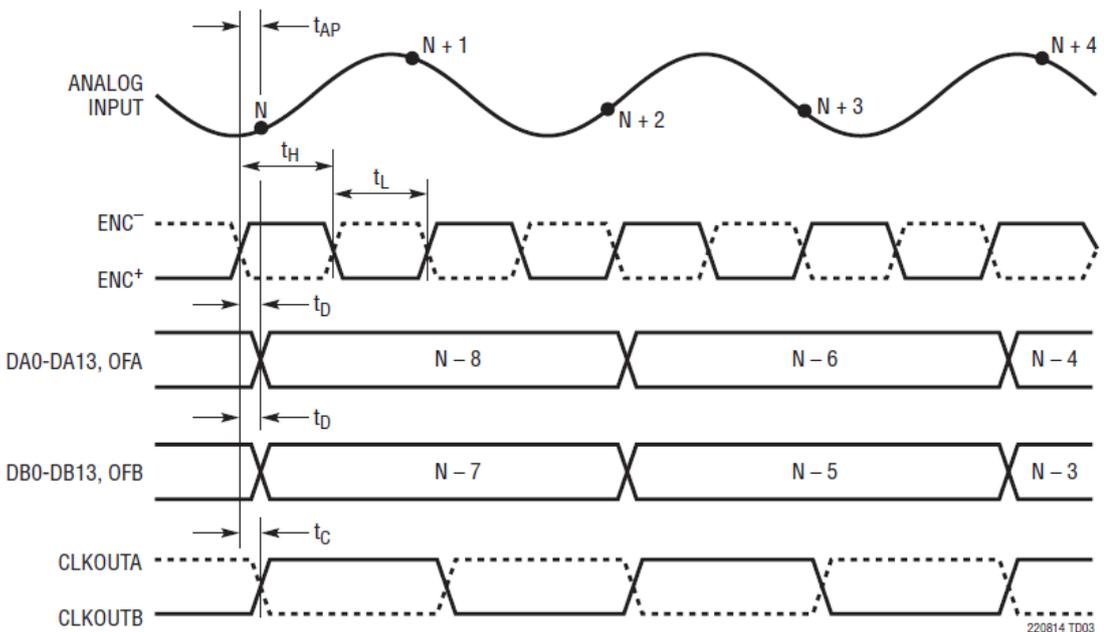
Tutte le uscite sono Single-Ended ed hanno Livelli CMOS



220814 TD02

Tempificazione del Modo d'Uscita Demultiplexed CMOS

Tutte le Uscite sono Single-Ended ed hanno Livelli CMOS



220814 TD03

4.3 Funzionamento

4.3.1 Prestazioni Dinamiche

4.3.1.1 Signal-to-Noise Plus Distortion Ratio

(Rapporto Segnale Rumore Più Distorsione)

Il signal-to-noise plus distortion ratio [S/(N+D)] è il rapporto tra l'ampiezza RMS della frequenza d'ingresso fondamentale e l'ampiezza RMS di tutte le altre componenti frequenziali dell'uscita dell'ADC. L'uscita è limitata alla banda di frequenze dalla DC fino a metà della frequenza di campionamento.

4.3.1.2 Signal-to-Noise Ratio

(Rapporto Segnale Rumore)

Il signal-to-noise ratio (SNR) è il rapporto tra l'ampiezza RMS della frequenza fondamentale e l'ampiezza RMS di tutte le altre componenti frequenziali, eccetto le prime cinque armoniche.

4.3.1.3 Total Harmonic Distortion

(Distorsione Armonica Totale)

La distorsione armonica totale è il rapporto della somma RMS di tutte le armoniche del segnale d'ingresso con la fondamentale stessa. Le armoniche fuori dalla banda si replicano nella banda di frequenza tra la DC e la metà della frequenza di campionamento. La THD è espressa come:

$$\text{THD} = -20\text{Log} \left(\sqrt{(V_2^2 + V_3^2 + V_4^2 + \dots + V_N^2)} / V_1 \right)$$

Dove V_1 è l'ampiezza RMS della frequenza fondamentale e V_2 fino a V_N sono le ampiezze delle armoniche dalla seconda all' N -esima.

4.3.1.4 Intermodulation Distortion

(Distorsione di Intermodulazione)

Se il segnale d'ingresso dell'ADC consiste di più di una sola componente spettrale, la non-linearità della funzione di trasferimento dell'ADC può produrre distorsione di intermodulazione (IMD) in aggiunta alla THD. La IMD è la variazione in un ingresso sinusoidale causato dalla presenza di un altro ingresso sinusoidale a frequenza differente.

Se due forme d'onde sinusoidali pure di frequenza f_a e f_b vengono applicate all'ingresso dell'ADC, le non-linearità nella funzione di trasferimento dell'ADC possono creare prodotti di distorsione a frequenze somme e differenze $m f_a \pm n f_b$, con m e $n = 0, 1, 2, 3$, etc.

Per esempio, i termini IMD del 3° ordine comprendono $(2f_a+f_b)$, (f_a+2f_b) , $(2f_a-f_b)$ and (f_a-2f_b) . La IMD del 3° ordine è definita come il rapporto del valore RMS di ciascun tono d'ingresso con il valore RMS del prodotto IMD del 3° ordine più grande.

4.3.1.5 Spurious Free Dynamic Range (SFDR)

(Intervallo di Dinamica Libera Spuria)

Il rapporto dell'ampiezza in RMS del segnale d'ingresso con il valore RMS del picco della componente spettrale spuria espresso in dBc. L'SFDR potrebbe anche essere calcolato relativamente al fondo scala ed espresso in dBFS.

4.3.1.6 Full Power Bandwidth

(Banda Passante alla Massima Potenza)

La banda passante alla massima potenza è quella frequenza d'ingresso alla quale l'ampiezza della fondamentale ricostruita è attenuata di 3dB per un segnale d'ingresso a fondo scala.

4.3.1.7 Aperture Delay Time

(Tempo del Ritardo di Apertura)

L'intervallo di tempo da quando un ENC^+ crescente raggiunge il livello di tensione ENC^- all'istante che il segnale d'ingresso è tenuto dal circuito sample & hold.

4.3.1.8 Aperture Delay Jitter

(Jitter del Ritardo di Apertura)

La variazione del tempo del ritardo di apertura da una conversione all'altra conversione. Questa variazione casuale risulterà in termini di rumore quando si campiona un ingresso AC. Il rapporto segnale rumore dovuto solo al jitter sarà:

$$SNR_{JITTER} = -20 \log (2\pi \cdot f_{IN} \cdot t_{JITTER})$$

4.3.2 Funzionamento della Conversione

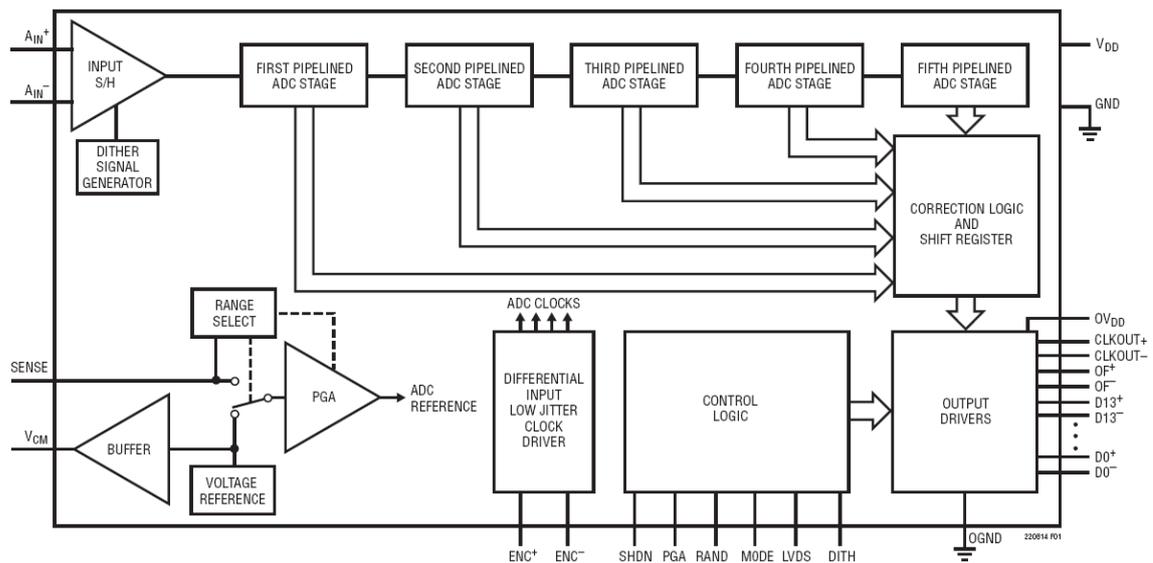


Figura 4-3: Schema a Blocchi

L'LTC2208-14 è un convertitore CMOS pipelined multistep con un PGA a monte. Come mostrato in Figura 4-3, il convertitore presenta cinque stadi ADC pipelined; un ingresso analogico campionato risulterà in valore digitalizzato sette cicli dopo (vedi Diagramma Temporale). L'ingresso analogico è differenziale per migliorare l'immunità al rumore di modo comune e per massimizzare il range d'ingresso. In oltre, l'unità d'ingresso differenziale ridurrà anche le armoniche del circuito di sample & hold. L'ingresso codificato anche è differenziale per migliorare l'immunità al rumore di modo comune.

L'LTC2208-14 ha due fasi di operazione, determinate dallo stato dei pin dell'ingresso differenziale ENC⁺/ENC⁻. Per brevità, il testo farà riferimento ad un ENC⁺ maggiore di ENC⁻ come ENC alto ed ad un ENC⁺ minore di ENC⁻ come un ENC basso.

Ciascun stato pipelined mostrato in Figura 1 contiene un ADC, un DAC di ricostruzione ed un amplificatore interstage. L'ADC quantizza l'ingresso dello stadio ed il valore quantizzato viene sottratto dall'ingresso con il DAC per produrre un residuo. Il residuo viene amplificato e mandato in uscita dall'amplificatore di residuo. Gli stadi successivi operano fuori fase così che quando gli stadi dispari producono il loro residuo, gli stadi pari acquisiscono quel residuo e viceversa.

Quando ENC è basso, l'ingresso analogico viene campionato in modo differenziale direttamente sui condensatori d'ingresso del sample & hold, all'interno dell' "input S/H" mostrato nello schema a blocchi. All'istante in cui commuta da basso ad alto, la tensione viene tenuta sui condensatori di campionamento. Mentre quando ENC è alto, la tensione d'ingresso tenuta viene bufferizzata dall'S/H amplifier il quale pilota il primo stadio ADC pipelined. Il primo stadio acquisisce l'uscita dell'S/H durante la fase alta di ENC. Quando ENC torna basso, il primo stadio produce il suo residuo che viene acquisito dal secondo stadio. Allo stesso tempo, l'input S/H torna basso per acquisire l'ingresso analogico. Quando ENC diventa alto, il secondo stadio produce il suo residuo che viene acquisito dal terzo stadio.

Un processo identico viene ripetuto per il terzo e per il quarto stadio, quindi il residuo del quarto stadio viene inviato al quinto stadio per una stima finale.

Ciascun stadio ADC successivo al primo presenta un range ulteriore per accogliere errori di flash e di amplifier offset. I risultati di tutti gli stadi ADC sono ritardati digitalmente in modo che i risultati possano essere correttamente combinati nella logica di correzione prima di essere inviati al buffer d'uscita.

4.3.3 Sample/Hold ed Unità d'Ingresso

4.3.3.1 Funzionamento del Sample/Hold

La Figura 4-4 mostra un circuito equivalente per il sample & hold differenziale CMOS dell'LTC2208-14. Gli ingressi analogici differenziali vengono campionati direttamente sui condensatori di campionamento (C_{SAMPLE}) attraverso dei transistor NMOS. I condensatori agganciati a ciascun ingresso ($C_{\text{PARASITIC}}$) sono le risultanti di tutte le altre capacità associate a ciascun ingresso.

Durante la fase di campionamento quando ENC è basso, i transistor NMOS connettono gli ingressi analogici ai condensatori di campionamento e li caricano, e seguono la

tensione d'ingresso differenziale. Quando ENC commuta da basso ad alto, la tensione d'ingresso campionata viene tenuta sui condensatori di campionamento.

Durante la fase di tenuta quando ENC è alto, i condensatori di campionamento sono disconnessi dall'ingresso e la tensione tenuta viene passata al cuore dell'ADC per l'elaborazione. Quando ENC commuta da alto a basso, gli ingressi vengono riconnessi ai condensatori di campionamento per acquisire un nuovo campione. Poiché i condensatori di campionamento tengono ancora il campione precedente, si vedrà in questo momento un glitch di carico proporzionale alla variazione di tensione. Se la variazione tra l'ultimo campione ed il nuovo campione è piccola, il glitch di carico visto all'ingresso sarà piccolo. Se la variazione dell'ingresso è grande, come la variazione vista con frequenze d'ingresso vicine a quella di Nyquist, allora si vedrà un glitch di carico maggiore.

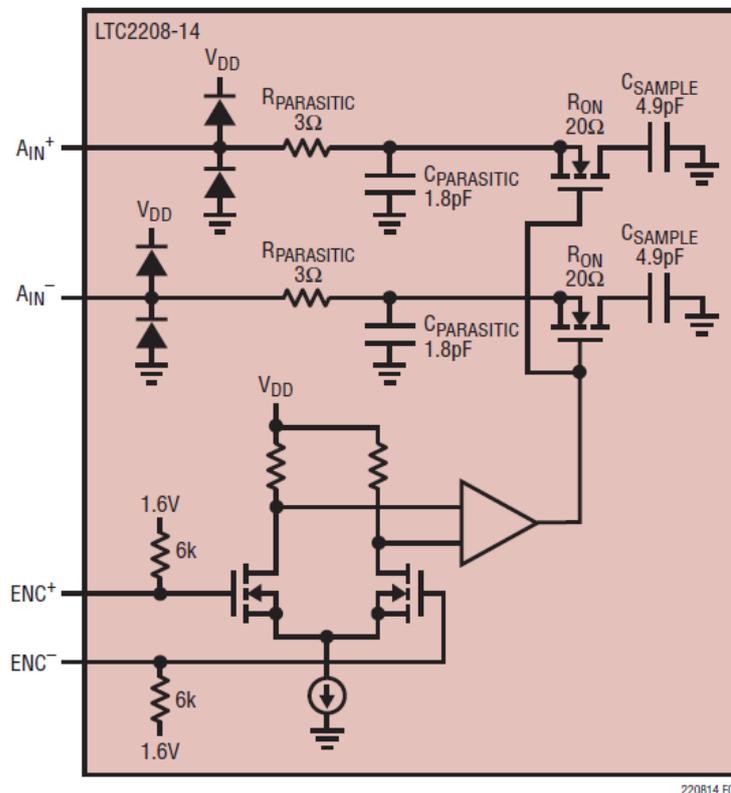


Figura 4-4: Circuito d'Ingresso Equivalente

4.3.3.2 Polarizzazione di Modo Comune

Il circuito di sample & hold dell'ADC richiede un'unità differenziale per ottenere specifiche prestazioni. Ciascun ingresso dovrebbe oscillare di $\pm 0.5625V$ per un range di $2.25V$ ($PGA=0$) o di $\pm 0.375V$ per un range di $1.5V$ ($PGA=1$), intorno ad una tensione di modo comune di $1.25V$. Il pin d'uscita V_{CM} (Pin 3) è progettato per fornire il livello di polarizzazione di modo comune. La V_{CM} può essere prelevata direttamente dalla presa centrale di un trasformatore per impostare il livello d'ingresso DC o come un livello di riferimento di un circuito di pilotaggio differenziale con amplificatore operazionale. Il pin della V_{CM} deve essere bypassato dall'ADC con $2.2\mu F$ o più.

4.3.3.3 Impedenza dell'Unità d'Ingresso

Come per tutti gli ADC ad alte prestazioni, ad alta velocità, le prestazioni dinamiche dell'LTC2208-14 possono essere influenzate dal circuito d'ingresso, in particolare la seconda e la terza armonica. L'impedenza d'ingresso e la reattanza d'ingresso possono influenzare l'SFDR. Al fronte di discesa di ENC il circuito di sample & hold conetterà il condensatore di campionamento da $4.9pF$ al pin d'ingresso ed inizierà il periodo di campionamento.

Il periodo di campionamento termina quando ENC si alza, tenendo l'ingresso campionato sul condensatore di campionamento. Idealmente, il circuito d'ingresso dovrebbe essere abbastanza veloce per caricare completamente il condensatore di campionamento durante il periodo di campionamento; comunque, questo non è sempre possibile e la risoluzione incompleta potrebbe degradare l'SFDR. Il glitch di campionamento è stato progettato per essere quanto più lineare possibile per minimizzare gli effetti della risoluzione incompleta.

Per una migliore prestazione è consigliato avere una impedenza d'ingresso di 100Ω o meno per ciascun ingresso. L'impedenza d'ingresso deve essere adattata per ingressi differenziali. Adattamenti non perfetti si tradurranno in un aumento delle armoniche di ordine pari, specialmente la seconda.

4.3.4 Circuiti d'Ingresso

4.3.4.1 Filtro d'Ingresso

Un filtro RC passa basso del primo ordine all'ingresso dell'ADC può provvedere a due funzioni: limitare il rumore dal circuito d'ingresso e fornire isolamento dalla commutazione del S/H dell'ADC. L'LTC2208-14 ha un circuito di S/H con una banda molto larga, dalla DC fino a 700MHz; può essere usato in una vasta gamma di applicazioni; pertanto, non è possibile fornire un singolo filtro RC da consigliare.

Le Figure 3, 4a e 4b mostrano tre esempi di filtraggio RC d'ingresso a tre intervalli di frequenze d'ingresso. In genere è preferibile fare i condensatori quanto più grandi possono essere ammessi; questo aiuterà a sopprimere il rumore random come pure il rumore accoppiato dal circuito digitale.

4.3.4.2 Transformer Coupled Circuits

(Circuiti Accoppiati al Trasformatore)

La Figura 4-5 mostra l'LTC2208-14 pilotato da un trasformatore RF con un secondario a presa centrale. La presa centrale del secondario è "polarizzata" al con la V_{CM} , impostando il segnale d'ingresso dell'ADC al suo livello DC ottimo. La Figura 4-5 mostra un trasformatore con rapporto di trasformazione 1:1. Possono essere usati altri rapporti di trasformazione; comunque, all'aumentare del rapporto di trasformazione aumenta anche l'impedenza vista dall'ADC. Un'impedenza d'ingresso maggiore di 50Ω può ridurre l'ampiezza di banda d'ingresso ed aumentare la distorsione in alta frequenza.

Uno svantaggio dell'uso del trasformatore è la perdita in termini di risposta in bassa frequenza. I trasformatori RF più piccoli hanno scarse prestazioni a frequenze al di sotto di 1MHz.

I trasformatori a presa centrale forniscono un comodo mezzo di polarizzazione del secondario; comunque, spesso presentano un bilanciamento scarso a frequenze d'ingresso alte, con conseguenti grandi armoniche del 2° ordine.

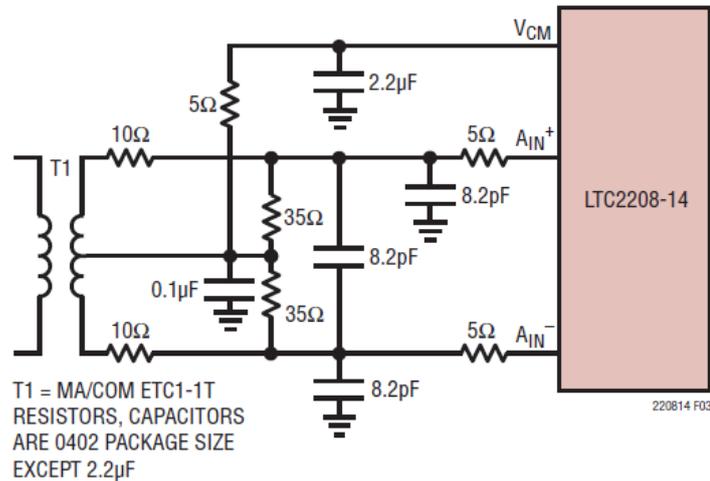


Figura 4-5: Conversione da Single-Ended a Differenziale con Trasformatore. Consigliato per Frequenze d’Ingresso da 5MHz a 100MHz

La Figura 4-6a mostra un trasformatore di accoppiamento che fa uso di una linea di trasmissione con trasformatore balun. Questo tipo di trasformatore presenta una risposta in alta frequenza ed un bilanciamento migliori di quelli dei trasformatori accoppiatori di flusso a presa centrale. I condensatori di accoppiamento sono aggiunti a massa ed ai terminali del primario d’ingresso per consentire di essere polarizzati a 1.25V. La Figura 4-6b mostra lo stesso circuito con componenti adatti per alte frequenze d’ingresso.

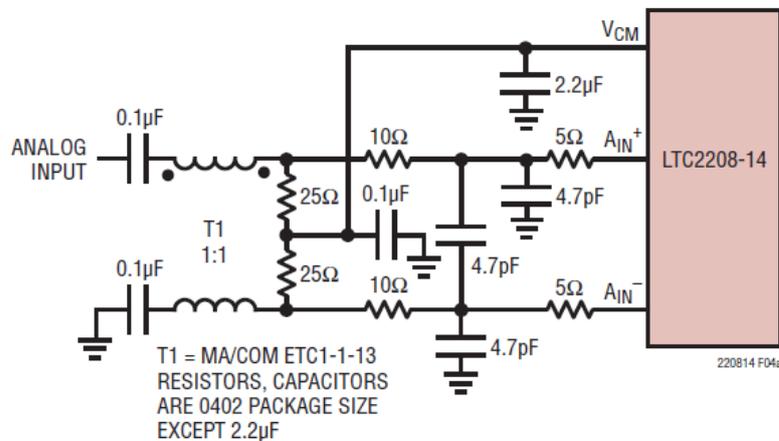


Figura 4-6a: Utilizzo di una Trasformatore Balun. Consigliato per Frequenze d’Ingresso da 100MHz a 250MHz

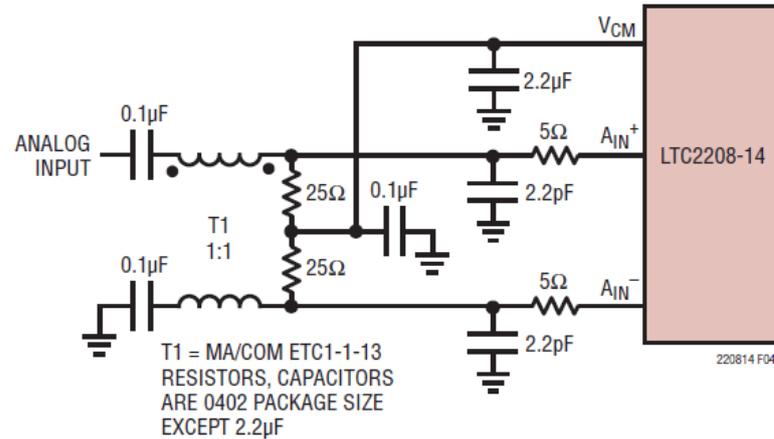


Figura 4-6b: Utilizzo di un Trasformatore Balun. Consigliato per Frequenze d’Ingresso da 250MHz a 500MHz

4.3.4.3 Circuiti di Accoppiamento Diretto

La Figura 4-7 mostra l’utilizzo di un amplificatore differenziale per convertire un segnale d’ingresso single-ended in un segnale d’ingresso differenziale. Il vantaggio di questo metodo è che fornisce risposta in bassa frequenza d’ingresso; comunque, la banda a guadagno limitato di qualsiasi amplificatore operazionale o amplificatore a ciclo chiuso degraderà l’SFDR dell’ADC alle alte frequenze d’ingresso. In oltre, gli amplificatori operazionali a larga banda o gli amplificatori differenziali tendono ad avere un elevato rumore. Di conseguenza, l’SNR sarà degradato a meno che la banda del rumore sia limitata prima dell’ingresso dell’ADC.

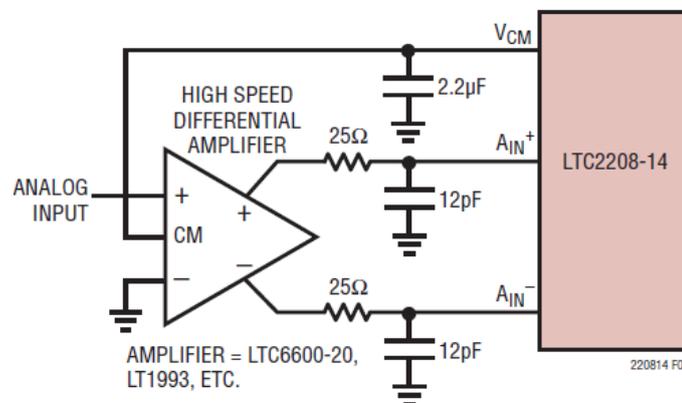


Figura 4-7: Ingresso Accoppiato DC con Amplificatore Differenziale

4.3.4.4 Funzionamento del Riferimento in tensione

La Figura 4-8 mostra il circuito del riferimento in tensione dell'LTC2208-14 consistente di un riferimento bandgap di 2.5V, un amplificatore a guadagno controllato ed un circuito di controllo. L'LTC2208-14 ha tre modi di funzionamento di riferimento: riferimento interno, riferimento esterno di 1.25V o riferimento esterno di 2.5V. Per usare il riferimento interno, collegare il pin SENSE alla V_{DD} . Per usare un riferimento esterno, applicare semplicemente una tensione di riferimento di 1.25V o di 2.5V al pin d'ingresso SENSE. Entrambi i livelli 1.25V e 2.5V applicati al SENSE produrranno un range a fondo scale di $2.25V_{P-P}$ ($PGA = 0$). Un'uscita di $1.25V V_{CM}$ viene fornita per una polarizzazione di modo comune per circuiti d'ingresso. È richiesto un condensatore esterno di bypass per l'uscita V_{CM} .

Ciò fornisce una bassa impedenza in alta frequenza che porta a massa per circuiti interni ed esterni. Questo è anche un condensatore di compensazione per il riferimento; non sarà stabile senza questo condensatore. Il minimo valore richiesto per la stabilità è $2.2\mu F$.

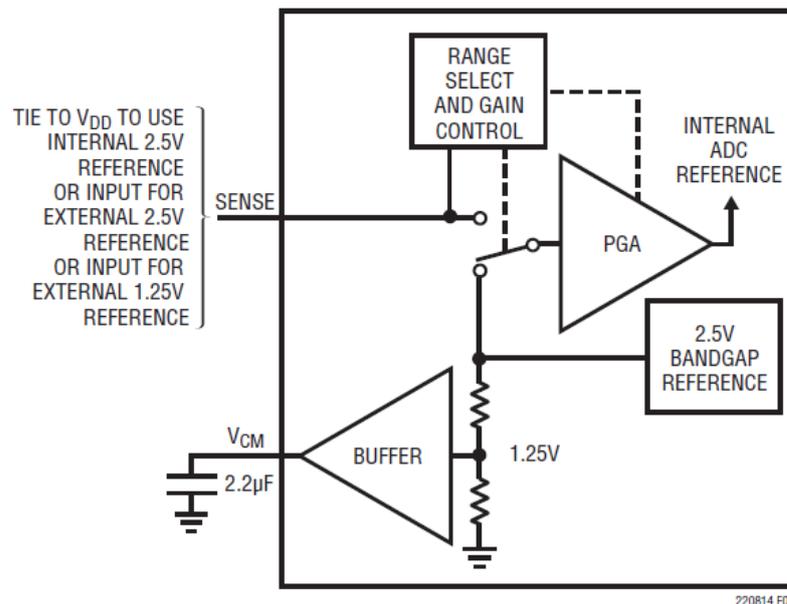


Figura 4-8: Circuito del Riferimento in tensione

L'amplificatore a guadagno programmabile interno fornisce la tensione di riferimento interna per l'ADC. Questo amplificatore ha dei requisiti di composizione rigorosi e non è accessibile per uso esterno.

Il pin SENSE può essere pilotato con tolleranza $\pm 5\%$ degli ingressi di riferimento esterni nominali di 2.5V o di 1.25V. Il range di regolazione può essere usato per regolare l'errore di guadagno dell'ADC o altri errori di guadagno del sistema. Quando si seleziona il riferimento interno, il pin SENSE dovrebbe essere collegato alla V_{DD} quanto più vicino possibile al convertitore. Se il pin SENSE è pilotato esternamente dovrebbe essere bypassato a massa quanto più vicino possibile al dispositivo con condensatore ceramico di $1\mu\text{F}$.

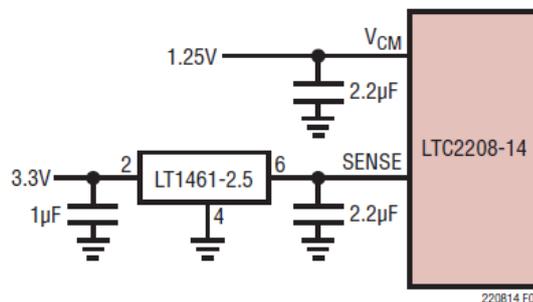


Figura 4-9: Un ADC con Range 2.25V e Riferimento Esterno di 2.5V

4.3.4.5 Pin PGA

Il pin PGA sceglie tra due guadagni per l'ADC a monte. $\text{PGA}=0$ seleziona un range d'ingresso di $2.25V_{\text{P-P}}$; $\text{PGA}=1$ seleziona un range d'ingresso di $1.5V_{\text{P-P}}$. Il range d'ingresso di 2.25V ha il migliore SNR; comunque, la distorsione sarà maggiore per frequenze d'ingresso al di sopra di 100MHz. Per applicazioni con alte frequenze d'ingresso, il basso range d'ingresso avrà una distorsione migliorata; comunque, l'SNR sarà approssimativamente 1.8dB peggiore.

4.3.4.6 Pilotaggio degli Ingressi Codificati

La prestazione in termini di rumore dell'LTC2208-14 può dipendere dalla qualità del segnale codificato tanto quanto dall'ingresso analogico.

Gli ingressi codificati sono destinati per essere pilotati differenzialmente, principalmente per l'immunità al rumore da sorgenti di rumore di modo comune. Ciascun ingresso è polarizzato attraverso un resistore da $6k\Omega$ ad un potenziale di $1.6V$. I resistori di polarizzazione impostano il punto di lavoro in DC per circuiti con trasformatore di accoppiamento e possono impostare la soglia logica per i circuiti single-ended.

Qualsiasi rumore presente sul segnale codificato genererà inoltre un jitter di apertura che sarà sommato in RMS con il jitter di apertura dell'ADC.

In applicazioni dove il jitter è fondamentale (alte frequenze d'ingresso), si fanno le seguenti considerazioni:

1. Usare lo stadio differenziale.
2. Usare un'ampiezza quanto più grande possibile. Se si usa il trasformatore di accoppiamento, usare un rapporto di trasformazione elevato per aumentare l'ampiezza.
3. Se l'ADC è clockato con un segnale sinusoidale di frequenza fissata, filtrare il segnale codificato per ridurre il rumore a banda larga.
4. Bilanciare le capacità e le resistenze serie ad entrambi gli ingressi codificati tale che qualsiasi rumore accoppiato apparirà ad entrambi gli ingressi come rumore di modo comune.

Gli ingressi codificati hanno un range di modo comune da $1.2V$ a $3V$. Ciascun ingresso potrebbe essere pilotato da massa a V_{DD} per stadi single-ended.

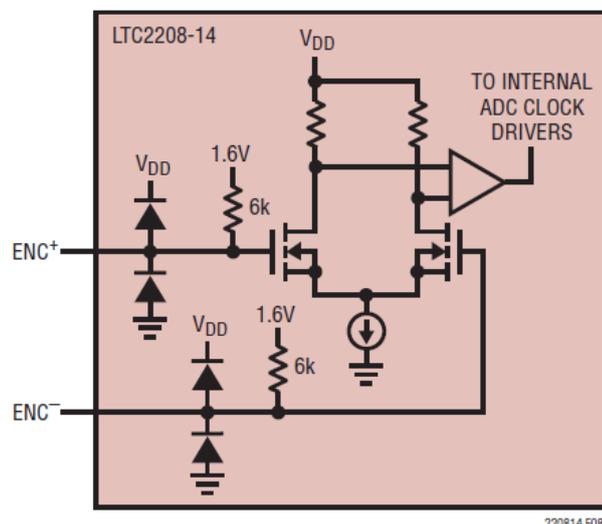


Figura 4-10a: Circuito Equivalente Ingresso Codificato

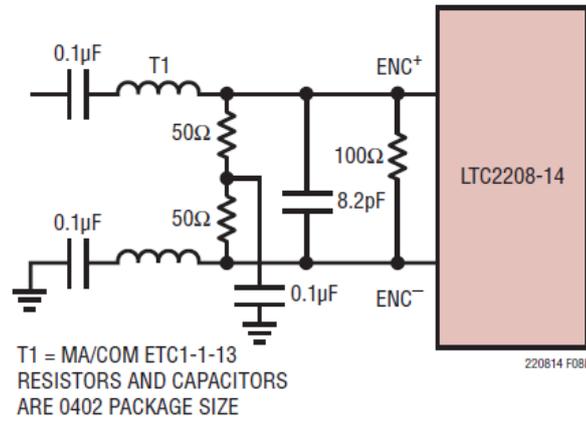


Figura 4-10b: Codifica Pilotata da Trasformatore

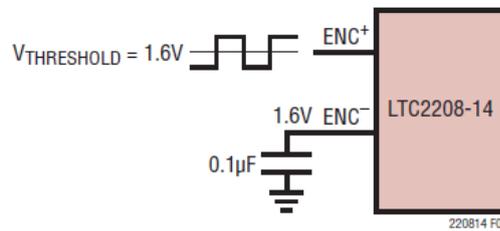


Figura 4-11: Unità ENC Single-Ended, Non Consigliata per Jitter Basso

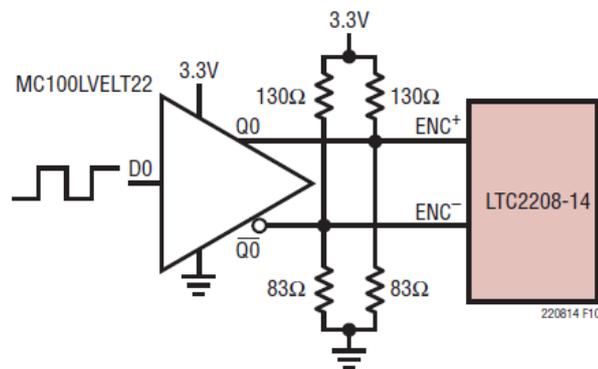


Figura 4-12: Unità ENC con Interfacciamento CMOS-PECL

4.3.4.7 Massimo e Minimo Rapporto di Codifica

Il Massimo rapporto di codifica per l'LTC2208-14 è 130Msps. Affinché l'ADC operi correttamente il segnale codificato dovrebbe avere un duty cycle del 50% ($\pm 5\%$). Ciascun semiciclo deve durare almeno 3.65ns per il circuito interno dell'ADC in modo da avere abbastanza tempo di assestamento per un corretto funzionamento. Ottenere un duty cycle del 50% preciso è facile con un segnale di pilotaggio sinusoidale differenziale utilizzando un trasformatore o utilizzando logiche differenziali simmetriche come PECL o LVDS. Quando si utilizza un segnale ENCODE single-ended asimmetrico nei tempi di salita e di discesa può risultare che i duty cycle siano lontani dal 50%.

Uno stabilizzatore di duty cycle opzionale può essere usato se il clock d'ingresso non ha un duty cycle del 50%. Questo circuito usa il fronte di salita del pin ENC per campionare l'ingresso analogico. Il fronte di discesa di ENC viene ignorato ed un fronte di discesa interno viene generato da un phase-locked loop. Il duty cycle del clock d'ingresso può variare dal 30% al 70% e lo stabilizzatore del duty cycle del clock manterrà un duty cycle interno costante del 50%. Se il clock è spento per un lungo periodo di tempo, il circuito stabilizzatore del duty cycle richiederà cento cicli di clock affinché il PLL si agganci al clock d'ingresso. Per utilizzare lo stabilizzatore del duty cycle del clock, il pin MODE deve essere connesso ad $1/3V_{DD}$ o $2/3V_{DD}$ utilizzando resistori esterni.

Il limite inferiore della frequenza di campionamento dell'LTC2208-14 è determinato dalla chiusura dei circuiti di sample & hold. L'architettura pipelined di questo ADC si basa sulla memorizzazione di segnali analogici su condensatori di piccolo valore. Le perdite di giunzione tendono a scaricare i condensatori. La minima frequenza di funzionamento specifica per l'LTC2208-14 è 1Msps.

4.3.5 Uscite Digitali

4.3.5.1 Modalità delle Uscite Digitali

L'LTC2208-14 può operare in quattro modalità per l'uscita digitale: standard LVDS, low power LVDS, full rate CMOS, e demultiplexed CMOS. Il pin LVDS seleziona la modalità operativa. Questo pin ha quattro livelli logici d'ingresso, centrati a 0, $1/3V_{DD}$,

$2/3V_{DD}$ e V_{DD} . Un resistore divisore esterno può essere usato per impostare i livelli logici a $1/3V_{DD}$ e $2/3V_{DD}$. La Tabella 4-1 mostra gli stati logici per il pin LVDS.

LVDS	DIGITAL OUTPUT MODE
0V(GND)	Full-Rate CMOS
$1/3V_{DD}$	Demultiplexed CMOS
$2/3V_{DD}$	Low Power LVDS
VDD	LVDS

Tabella 4-1: Funzioni del Pin LVDS

4.3.5.2 *Buffer delle Uscite Digitali (Modi CMOS)*

La Figura 4-13 mostra un circuito equivalente per un singolo buffer d'uscita nel Modo CMOS, Full-Rate o Demultiplexed. Ciascun buffer è alimentato da OV_{DD} e $OGND$, isolato dall'alimentazione e dalla massa dell'ADC. Il transistor aggiuntivo a canale N nel driver d'uscita consente il funzionamento a bassa tensione.

Il resistore interno in serie con l'uscita fa apparire l'uscita con una resistenza di 50Ω al circuito esterno ed elimina la necessità di resistori esterni di smorzamento.

Come con tutti i convertitori veloci ad alta risoluzioni, il carico dell'uscita digitale può influire sulla prestazione. Le uscite digitali dell'LTC2208-14 dovrebbero pilotare un carico capacitivo minimo per evitare possibili interazioni tra uscite digitali e circuiti d'ingresso sensibili. L'uscita andrebbe bufferizzata con un dispositivo come un latch CMOS ALVCH16373. Per un funzionamento alla massima velocità il carico capacitivo andrebbe tenuto sotto i 10pF . Si potrebbe utilizzare un resistore in serie con l'uscita ma non è richiesto poiché l'ADC ha un resistore in serie di 43Ω sul chip.

Le basse tensioni OV_{DD} aiuteranno a ridurre l'interferenza dalle uscite digitali.

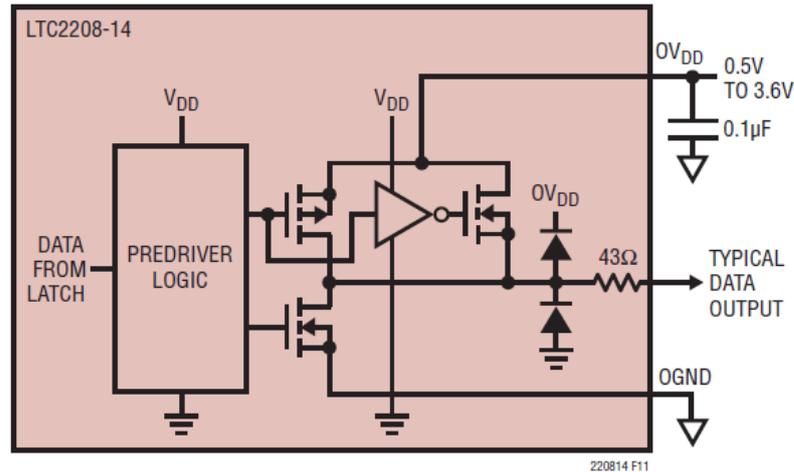


Figura 4-13: Circuito Equivalente per un Buffer dell'Uscita Digitale

4.3.5.3 *Buffer delle Uscite Digitali (Modi LVDS)*

La Figura 4-14 mostra un circuito equivalente per una coppia d'uscita LVDS. Una corrente di 3.5mA viene fatta fluire da OUT^+ a OUT^- o vice versa, la quale crea una differenza di potenziale di $\pm 350mV$ ai capi del resistore di terminazione di 100Ω al ricevitore LVDS.

Una catena in retroazione regola la tensione modo comune d'uscita a 1.20V. Per un corretto funzionamento ciascuna coppia d'uscita LVDS deve essere terminata con un resistore esterno di terminazione da 100Ω , anche se il segnale non è utilizzato (come OF^+/OF^- o $CLKOUT^+/CLKOUT^-$). Per minimizzare il rumore le piste del circuito stampato (Print Circuit board) per ciascuna coppia d'uscita LVDS andrebbero tracciate insieme. Per minimizzare il clock skew, tutte le tracce del circuito stampato LVDS dovrebbero avere più o meno la stessa lunghezza.

Nella Modalità Low Power LVDS circolano 1.75mA tra le uscite differenziali, causando $\pm 175mV$ sul resistore di terminazione da 100Ω al ricevitore LVDS. La tensione modo comune d'uscita è 1.20V, la stessa del Modo standard LVDS.

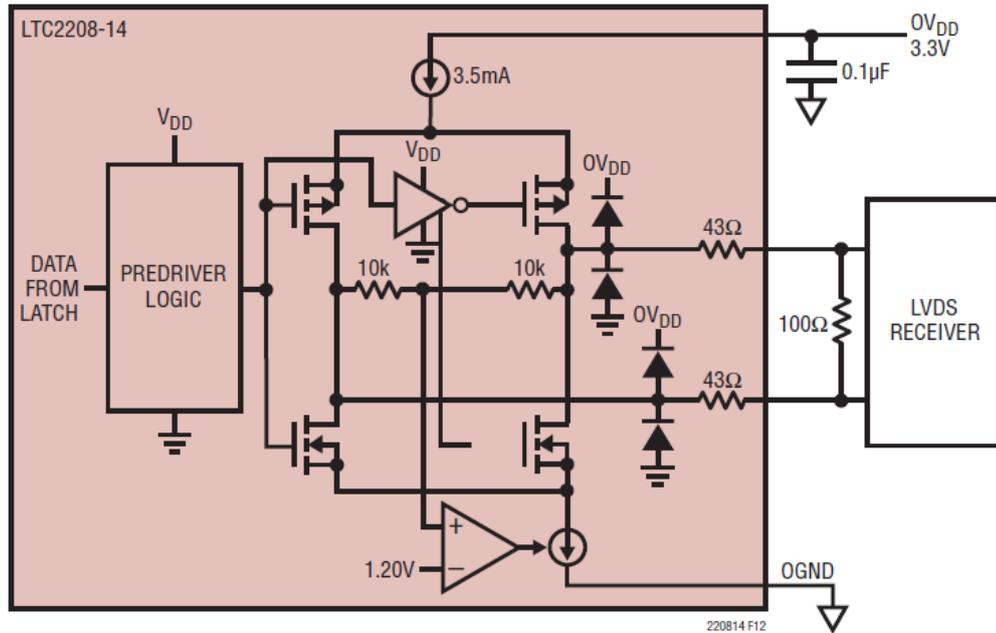


Figura 4-14: Buffer d'Uscita Equivalente nel Modo LVDS

4.3.5.4 Formato dei Dati

L'uscita digitale parallela dell'LTC2208-14 può essere selezionata per formato offset binary o complemento a 2. Il formato viene selezionato con il pin MODE. Questo pin ha un ingresso a Quattro livelli logici, centrati a 0, $1/3V_{DD}$, $2/3V_{DD}$ e V_{DD} . Si può utilizzare un resistore divisore esterno per impostare i livelli logici a $1/3V_{DD}$ e $2/3V_{DD}$. La Tabella 4-2 mostra gli stati logici per il pin MODE.

MODE	OUTPUT FORMAT	CLOCK DUTY CYCLE STABILIZER
0(GND)	Offset Binary	Off
$1/3V_{DD}$	Offset Binary	On
$2/3V_{DD}$	2's Complement	On
VDD	2's Complement	Off

Tabella 4-2: Funzione del Pin MODE

4.3.5.5 Bit di Overflow

Un bit overflow d'uscita (OF) indica quando il convertitore è over-ranged o under-ranged. Nel modo CMOS, un livello logico alto sul pin OFA indica un overflow o un underflow sul bus dati A, mentre un livello logico alto sul pin OFB indica un overflow sul bus dati B. Nel modo LVDS, un livello logico alto differenziale sui pin OF^+/OF^- indica un overflow o un underflow.

4.3.5.6 Clock d'Uscita

L'ADC ha una versione ritardata dell'ingresso codificato disponibile come uscita digitale, CLKOUT. Il pin CLKOUT può essere usato per sincronizzare il convertitore dati al sistema digitale. In entrambi i modi CMOS, il bus dati A sarà aggiornato secondo le discese di CLKOUTA e le salite di CLKOUTB. Nel modo demultiplexed CMOS il bus dati B sarà aggiornato secondo le discese di CLKOUTA e le salite di CLKOUTB.

Nel Modo Full Rate CMOS, solo il bus dati A è attivo; i dati possono essere latched sul fronte di salita di CLKOUTA o sul fronte di discesa di CLKOUTB.

Nel modo demultiplexed CMOS, CLKOUTA e CLKOUTB passeranno ad 1/2 della frequenza del segnale codificato. Sia il bus A che il bus B possono essere latched sul fronte di salita di CLKOUTA o sul fronte di discesa di CLKOUTB.

4.3.5.7 Digital Output Randomizer

L'interferenza dalle uscite digitali dell'ADC è a volte inevitabile. Le interferenze dalle uscite digitali possono provenire da accoppiamenti capacitivi o induttivi oppure da accoppiamenti attraverso il piano di massa. Anche un piccolo fattore di accoppiamento può far risultare toni distinguibili non voluti nello spettro in frequenza dell'uscita dell'ADC.

Randomizzando l'uscita digitale prima che venga trasmessa dal chip, quei toni non voluti saranno randomizzati, producendo un lieve aumento del rumore di fondo per ottenere un'ampia riduzione dell'ampiezza dei toni non voluti.

L'uscita viene "Randomizzata" applicando una funzione logica exclusive-OR tra l'LSB e tutti gli altri bit dell'uscita dati. Per decodificare, viene applicata la funzione inversa; che è una funzione exclusive-OR applicata tra l'LSB e tutti gli altri bit. L'LSB, le uscite

OF e CLKOUT non sono influenzate. La funzione Randomizer d'uscita è attiva quando il pin RAND è alto.

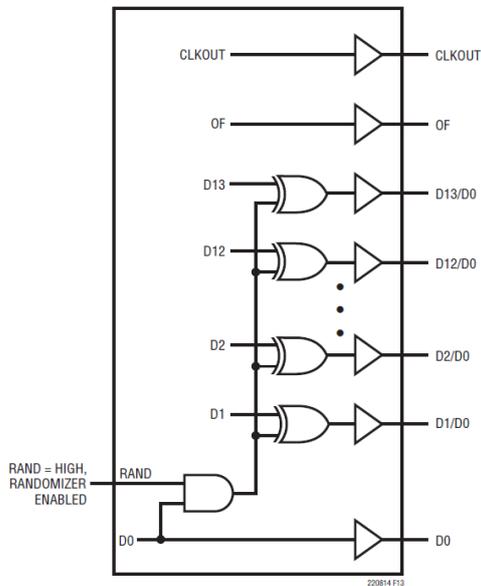


Figura 4-15: Equivalente Funzionale di un Digital Output Randomizer

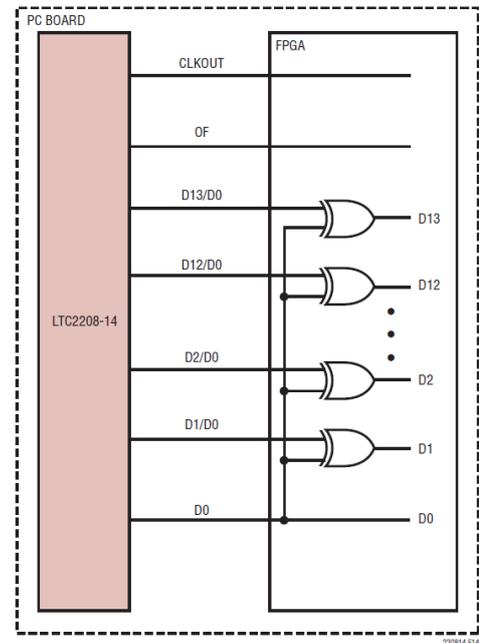


Figura 4-16: Derandomizzazione di un'Uscita Digitale Randomizzata

4.3.5.8 Output Driver Power

Separare i pin dell'alimentazione e della massa d'uscita consente ai driver d'uscita di essere isolati dal circuito analogico. L'alimentazione per i buffer dell'uscita digitale, OV_{DD} , dovrebbe essere collegata allo stesso alimentatore.

Per esempio, se il convertitore pilota un DSP alimentato ad 1.8V, allora OV_{DD} andrebbe collegata alla stessa alimentazione di 1.8V. Nel modo CMOS, OV_{DD} può essere alimentato con qualsiasi tensione logica fino a 3.6V. $OGND$ può essere alimentata con qualsiasi tensione con riferimento a massa fino a 1V e deve essere minore di OV_{DD} . Le uscite logiche oscilleranno tra $OGND$ e OV_{DD} . Nel Modo LVDS, OV_{DD} andrebbe connesso all'alimentazione da 3.3V e $OGND$ dovrebbe essere connessa a GND.

4.3.5.9 Dither Interno

L'LTC2208-14 è un ADC a 14 bit con una funzione di trasferimento molto lineare; comunque, a bassi livelli d'ingresso anche le piccole imperfezioni nella funzione di trasferimento produrranno dei toni indesiderati.

I piccoli errori della funzione di trasferimento sono di solito un effetto degli elementi dell'ADC. Una modalità opzionale di dither interno può essere abilitata per randomizzare la locazione dell'ingresso sulla curva di trasferimento dell'ADC, con conseguente un aumento dell'SFDR per bassi livelli di segnale.

Come mostrato in Figura 4-17, l'uscita del sample & hold amplifier viene sommata con l'uscita del dither DAC. Il dither DAC è pilotato da un generatore di sequenze di numeri pseudo-random; il numero random in ingresso al DAC del dither viene anche sottratto dal risultato dell'ADC. Se il DAC del dither è calibrato accuratamente all'ADC, si vedrà uscita ben poco dither. Il dither che lo attraversa apparirà come rumore bianco. Il dither DAC è calibrato per ottenere uno scostamento minore di 0.5dB da rumore di fondo dell'ADC senza dither.

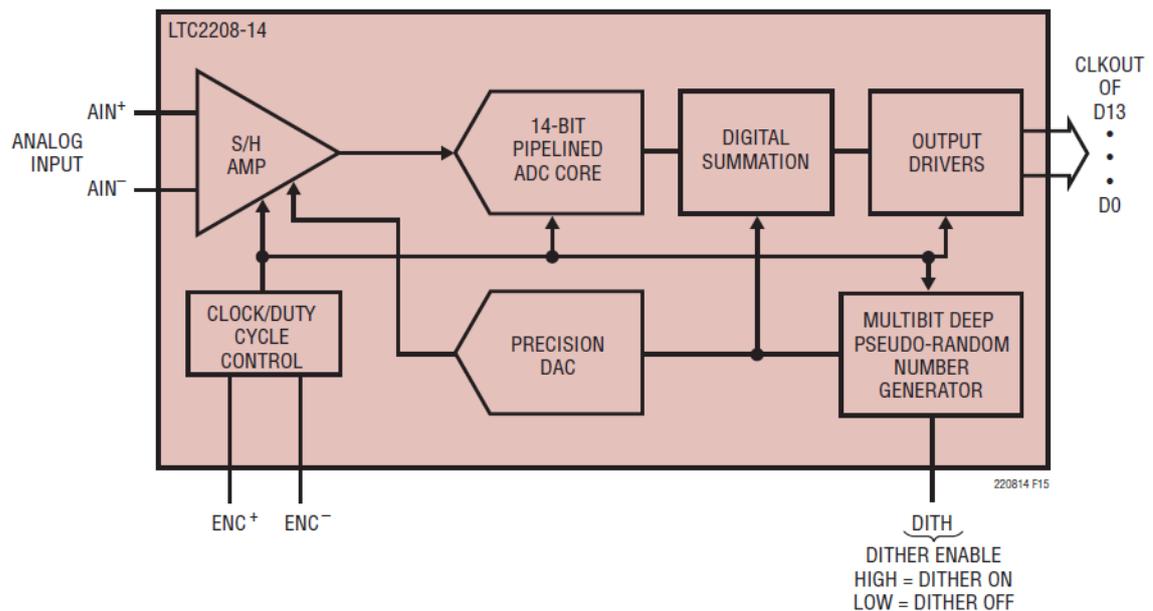


Figura 4-17: Schema a Blocchi dell'Equivalento Funzionale del Circuito di Dither Interno

4.3.5.10 Trasferimento di Calore

La maggior parte del calore generato dall'LTC2208-14 si trasferisce dal chip attraverso il fondo dell'exposed pad. Per buone prestazioni elettriche e termiche, l'exposed pad deve essere saldato ad un largo pad di massa sul circuito stampato. È fondamentale che l'exposed pad e tutti i pin di massa siano connessi al piano di massa con area sufficiente.

5 ELABORAZIONE ED ANALISI DEI DATI ACQUISITI

In questo 5° capitolo mostreremo la tipologia di segnali che interessa l'acquisizione dati del radar SORA. Faremo ricorso ad un software realizzato in Matlab che consente di elaborare ed analizzare i segnali radar sia nel dominio del tempo che in quello della frequenza ed estrapolare dei parametri che stimano la bontà dell'acquisizione.

5.1 Acquisizione dati in Complemento a 2

Nel primo capitolo abbiamo visto come gli echi radar attraversino la catena di ricezione ed arrivino in ingresso all'ADC a 14 bit. Viene quindi operata con LTC2208-14 una conversione degli ingressi analogici che vengono quindi riproposti in formato digitale. L'ADC è a 14 bit anche se in effetti poiché si lavora con i byte (8 bit) e di conseguenza con le word (2 byte = 16 bit), i dati binari vengono rappresentati su 16 bit, ma comunque le possibili combinazioni che possono assumere sono 2^{14} .

La rappresentazione binaria utilizzata è in complemento a 2, che consente di rappresentare sia valori positivi che negativi e facilita le operazioni matematiche. Per i valori non negativi si utilizzano le prime 2^N combinazioni, in modo analogo con quanto avviene con il BCD 8-4-2-1. Mentre per i valori negativi si converte l'opposto in BCD, si opera la negazione bit a bit e si aggiunge 1. Le rappresentazioni dei valori non negativi avranno l'MSB pari a 0 mentre quelle dei valori negativi avranno l'MSB pari a 1. Questi ultimi, per essere riportati in base 10, si convertono in BCD e vi si sottrae 2^N .

In Tabella 5-1 riportiamo la rappresentazione complemento a 2 per una codifica a 4 bit.

Complemento a 2	Decimale
0111	+7
0110	+6
0101	+5
0100	+4
0011	+3
0010	+2
0001	+1
0000	0
1111	-1
1110	-2
1101	-3
1100	-4
1011	-5
1010	-6
1001	-7
1000	-8

Tabella 5-1

5.2 Data Formatting

Il data formatting prevede che i campioni a 14 bit del segnale digitalizzato vengano impacchettati insieme con i dati ancillari ricevuti dalla CDMU, con due PRI Marker e due PRI Counter. Il PRI (Pulse Repetition Interval) è il tempo trascorso dall'inizio di un impulso all'inizio dell'impulso successivo, ed è definito come reciproco di PRF (Pulse Repetition Frequency). La struttura generale del formato dati di un record PRI è organizzata in word (16 bit) ed è mostrata di seguito:

- A. Mark1
- B. Mark2
- C. Data Counter 1
- D. Data Counter 2
- E. ADC Words: 4091 (presumming) o 4095 (no presumming)
- F. Dati Ancillari: 18 word del GPS dalla CDMU

Mark1 e **Mark2**: sono due word che in esadecimale si rappresentano rispettivamente con $AAAA_{16}$ ($1010101010101010_2=43690_{10}$) e $BBBB_{16}$ ($1011101110111011_2=48059_{10}$). Il record PRI comincia se vengono riconosciute queste due sequenze in successione.

Data Counter 1: i 10 bit meno significativi di questa word rappresentano lo Scan Number che conta il numero di volte che il radar viene acceso e spento. Mentre i restanti 6 bit, quelli più significativi, rappresentano i 6 bit meno significativi del PRI Number che conta il numero di record PRI.

Data Counter 2: rappresentano i 16 bit più significativi dei 22 bit di cui è composto il PRI Number.

ADC Word:

c) **Modalità No Presumming**

Nell'implementazione del firmware, in questa modalità si è scelto di utilizzare 4095 word per rappresentare le acquisizioni degli echi. Ciascuna word come abbiamo detto, è costituita da 16 bit, quindi per ciascuna di essa, i 14 bit LSB sono il campione in rappresentazione complemento a 2. Le combinazioni possibili sono $2^{14} = 16384$ ma poiché il complemento a 2 utilizza metà delle combinazioni per rappresentare valori non negativi e l'altra metà per rappresentare i valori negativi, il massimo valore risolubile è $2^{14}/2 = 2^{13} = 8192$.

Una volta fatte le acquisizioni, nel post-processing bisogna poi dare un significato in termini di tensione ai segnali digitalizzati. Basterà applicare una conversione Campione-Volt realizzabile con la seguente relazione:

$$\text{Valore in Volt} = (\text{Campione} \times 0,75) / 8192 \quad (5.1)$$

d) **Modalità Presumming**

In questa modalità, l'implementazione del firmware prevede la rappresentazione degli echi radar su 4091 word. Con il presumming non viene conservata ciascuna acquisizione ma tramite l'FPGA, si compie la somma di 8 echi successivi ed è questa che mediante il Disk Array Controller viene immagazzinata nel Disk Array. Le acquisizioni fatte in modalità presumming, sono in complemento a 2 ma hanno una risoluzione a 13 bit poiché, farle a 14 bit e sommarne 8 di esse, condurrebbe ad

un overflow dei 16 bit con cui lavora il sora_proc. All'uscita dell'FPGA avremo perciò dei dati a 16 bit con segno ma il massimo valore risolubile sarà però $2^{13}/2 = 2^{12} = 4096$.

L'equazione di conversione Campione-Volt è quindi:

$$\text{Valore in Volt} = (\text{Campione} \times 0,75) / 4096 \quad (5.2)$$

Dati Ancillari: i 28 byte dei dati ancillari utili sono preceduti da 8 byte usati come marker di sincronizzazione (vedi Par. 1.3.1).

5.3 Sora_proc

Per l'analisi degli echi radar acquisiti, è stato sviluppato un software realizzato in Matlab, il "sora_proc.m", che è stato installato sul computer di bordo ma che verrà utilizzato anche a terra nel post processing.

Il "sora_proc.m" attinge dai dati del SORA, i quali durante la modalità DUMP vengono memorizzati nel Disk Array, quindi vengono passati al disco di Windows dove viene generato un file con estensione ".dat". Il software ha il compito di individuare i record PRI contenuti nel file, determinarne il numero e distinguere i dati relativi agli echi radar acquisiti dai dati ancillari secondo le disposizioni del Data Formatting. Deve quindi manipolare le word dell'ADC che rappresentano gli echi di ritorno e individuare dei parametri.

Nel Par. 2.3 abbiamo visto che l'utilizzo di segnali chirp è dovuto all'esigenza di avere dei segnali con durate non brevi in modo da trasportare abbastanza energia e avere comunque buone risoluzioni in range. Ciò è possibile poiché sfruttano la modulazione lineare di frequenza che consente di svincolare la banda del segnale dalla sua durata, poiché non viene trasmessa istantaneamente tutta la banda ma si trasmettono le frequenze sfasate nel tempo. Questa tipologia di segnale richiede in ricezione un filtro adattato che comprima le fasi sovrapponendo le frequenze in un unico istante di tempo la cui risposta impulsiva è legata al segnale trasmesso (2.9). All'uscita di tale filtro ci si attende una sinc centrata in un istante t_0 la cui maggior parte d'energia è contenuta

nell'intervallo $1/B$. La ricezione con filtro adattato in passato era realizzato in analogico, ma come vedremo di seguito, è possibile realizzarla in digitale, implementandola direttamente in Matlab.

5.3.1 Funzionamento del `sora_proc.m`

Di seguito illustreremo il codice realizzato in Matlab del `sora_proc`, che consentirà di implementare quanto sopra richiesto.

Dopo aver lanciato il programma, viene aperto il file “.dat” di cui gli viene passato il nome come parametro ed un flag che identifica una delle quattro possibili combinazioni delle modalità. Viene scelto il fattore di conversione e si determina la dimensione del file. Col presumming le word ADC saranno 4091 e quindi con i Data Counter, i Marker e le 18 word del GPS, il record avrà 8226 byte, mentre nel modo no presumming avremo invece 4095 word ADC per un totale di 8234 byte nel record. A questo punto è possibile determinare il numero di record completi contenuti nel file.

Vengono impostate le specifiche del radar:

- Frequenza di campionamento $f_c = 50\text{MHz}$
- Durata dell'impulso chirp $T = 3,2\mu\text{s}$
- Banda dell'impulso chirp $B = 10\text{MHz}$
- Chirp rate $\alpha = B/T$
- Frequenza iniziale dell'impulso chirp $f_0 = 158\text{MHz}$

Adesso bisogna ricevere il segnale di ritorno acquisito con un filtro adattato. Se il segnale inviato è $x(t)$, allora il filtro adattato avrà risposta impulsiva $x^*(-t)$. Fare la convoluzione di due segnali nel dominio del tempo corrisponde a moltiplicare le loro trasformate di Fourier, ma i nostri segnali, in seguito alla conversione analogico-digitale, sono discreti. Si tratta di una convoluzione di segnali discreti che si traduce in un prodotto di FFT (Fast Fourier Transform). All'interno del programma implementiamo il filtro adattato che naturalmente deve essere discreto e deve essere legato al chirp inviato. Ci costruiamo allora un chirp di riferimento discreto che deve essere l'analogo della sua versione analogica:

$$\text{chirp}(t) = \cos(\varphi(t)) \tag{5.3}$$

dove:

$$\varphi(t) = 2\pi \int_{f_0+B}^{f_0} f(t)dt = 2\pi \int_{f_0+B}^{f_0} (f_0 + \alpha t)dt \quad (2.5)$$

Dopo avere realizzato il segnale nel dominio del tempo discreto, ci ricaviamo la sua FFT impostando il numero di punti a 4096 e scegliamo di andare a considerare 500 record PRI.

Possiamo ora scorrere i singoli record del file, individuare i dati acquisiti e calcolare la risposta impulsiva con un filtro ottimo. Tutto ciò è implementabile con un ciclo `while`. Il record comincia se vengono individuati i Marker 1 e 2 cioè le sequenze esadecimale AAAA e BBBB.

Una volta riconosciuto quelle sequenze, si individuano i due Data Counter da cui ricaviamo lo Scan Number e il PRI Number.

A queste word nel record seguono quelle dell'ADC che sono in numero diverso nelle due modalità. Nel no presumming vengono lette dal file 4095 word 16 bit con segno, ma le word ADC sono a 14 bit con segno. Allora si converte in un intero a 16 bit senza segno, quindi una rappresentazione 8-4-2-1 che viene convertita in complemento a 2. Con il presumming invece vengono lette 4091 word che sono a 16 bit con segno e che il `sora_proc` interpreta come interi a 16 bit con segno.

Infine legge 18 word GPS come interi con segno a 16 bit.

All'interno dello stesso ciclo `while`, eseguiamo la FFT dei dati acquisiti con lo stesso numero di punti della FFT del chirp di riferimento e trasformiamo secondo Hilbert le due FFT in modo da tagliare la parte a frequenze negative. Ma così facendo, quei segnali campionati a 50MHz con `N_FFT` punti, saranno campionati a $f_c/2=25\text{MHz}$ con `N_FFT/2` punti. Allora dopo aver moltiplicato tra loro le due trasformate di Hilbert, sovraccampioniamo il prodotto di un fattore `over = 4`. Questa operazione è uno **Zero Filling**, ossia un riempimento con degli zeri della FFT della risposta impulsiva, che consente di avere una maggiore risoluzione, con $(N_FFT/2) * over$ punti campionati a $(f_c/2) * over$. Infine basta applicare la IFFT (Inverse Fast Fourier Transform) alla FFT sovracampionata della risposta impulsiva.

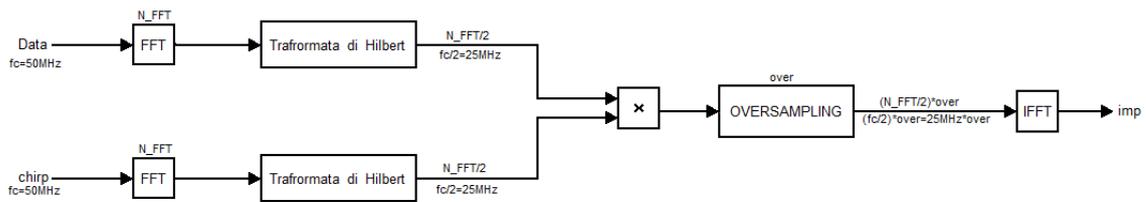


Figura 5.1

Come discusso nel Par. 3.2, dopo questa elaborazione che subisce l'eco acquisito, ci aspettiamo che la risposta impulsiva del radar, da non confondere con la risposta impulsiva del filtro adattato intesa come antitrasmessa della sua funzione di trasferimento, somigli ad un sinc.

Della risposta impulsiva ci calcoliamo il suo massimo, l'indice in cui lo raggiunge, l'ampiezza a 3dB nella quale è racchiusa la maggior parte dell'energia ed i massimi dei due lobi laterali principali.

Al termine del ciclo `while` calcoliamo:

- La media relativa all'over delle variazioni delle posizioni dei massimi della risposta impulsiva sovracampionata.
- La deviazione standard delle variazioni degli indici del massimo relativa alla frequenza di sovracampionamento
- Il numero di variazioni dell'indice del massimo diverse dalla media delle variazioni dell'indice del massimo
- La media espressa in dB dei massimi delle risposte impulsive
- La deviazione standard dei massimi dei massimi delle risposte impulsive normalizzata alla media dei massimi
- La media dell'ampiezza della risposta impulsiva normalizzata alla frequenza di sovracampionamento
- Peak Side Lobe Ratio destro: la media espressa in dB dei rapporti tra il massimo del lobo destro ed il massimo del lobo centrale
- Peak Side Lobe Ratio sinistro: la media espressa in dB dei rapporti tra il massimo del lobo sinistro ed il massimo del lobo centrale

5.3.2 Listato sora_proc.m

```

function sora_proc(varargin)

flag=str2num(cell2mat(varargin(1)));
filename=cell2mat(varargin(2));

if flag==0
    f_presum=0;           % no presuming
    f_autocheck=0;       % no autocheck
elseif flag==1
    f_presum=1;           % si presuming
    f_autocheck=0;       % no autocheck
elseif flag==2
    f_presum=0;           % no presuming
    f_autocheck=1;       % si autocheck
elseif flag==3
    f_presum=1;           % si presuming
    f_autocheck=1;       % si autocheck
end

if f_presum==1           % il fattore di conversione e il PRF cambiano con la modalit 
    conv_fact=0.75/4096;
    PRF=500/8;
else
    conv_fact=0.75/8192;
    PRF=500;
end

fprintf(1, '\nFlag value = %d\n', flag);
fprintf(1, '\nNome file = %s\n', filename);

FF=fopen(filename, 'r', 'l');                               % apre il file in lettura con
                                                            % ordinamento dei byte little endian

fseek(FF, 0, 'eof');                                         % Setta l'indicatore di posizione del file con
                                                            % offset 0 a partire dalla fine del file

Size=ftell(FF);                                              % ritorna la dimensione del file

fseek(FF, 0, 'bof');                                         % Setta l'indicatore di posizione del file con
                                                            % offset 0 a partire dall'inizio del file

if f_presum==1                                               % presuming
    Byte_rec=8226;                                           % numero di byte del record
    n_word=4091;                                             % numero di word ADC del record
else                                                         % no presuming
    Byte_rec=8234;                                           % numero di byte del record
    n_word=4095;                                             % numero di word ADC del record
end
n_word_GPS=18;                                              % numero di word GPS del record

```

```

n_rec_tot=floor(Size/Byte_rec);           % numero intero di record del file

fc=50e6;           % frequenza di campionamento 50MHz
T=3.2e-6;         % durata dell'impulso generato 3,2µs
B=10e6;           % banda in frequenza dell'impulso generato 10MHz
alfa = B/T;       % chirp rate = ((10^6)/(3,2*10^-6))s^-2 = 3.125*10^12
t=0:1/fc:T;       % tempo da 0 a T, con passo 1/fc = 1/50MHz = 20ns

f_start=158e6;    % frequenza da cui inizia ad oscillare il chirp

DURATA=1;        % in secondi

n_rec=floor(DURATA*PRF);           % numero di record presi in considerazione

tt=[0:n_rec-1]/PRF;               % finestra temporale

if f_autocheck==1
    N_FFT=4096;                   % numero di campioni della fft
    ff=0:fc/N_FFT:fc-fc/N_FFT;    % frequenza da 0 a fc-fc/N_FFT con passo fc/N_FFT

    f_clock=100e6;                % frequenza di clock 100MHz
    n_step=T*f_clock;             % numero di passi dell'fft del chirp di riferimento
    chirp_freq_step=B/n_step;     % frequenza dei passi del chirp di riferimento
    ff_step=f_start:chirp_freq_step:f_start+B; % frequenza in funzione del tempo
                                        % con passo chirp_freq_step

    fase_step = 2*pi*cumtrapz(ff_step(1:f_clock/fc:end))/fc; % fase del chirp Par 3.2
    chirp=cos(fase_step);         % chirp(t) = cos(fase(t))

    chirp_fft=(fft(chirp,N_FFT)); % fft del chirp
    i_max=zeros(1,n_rec);        % vettore di n_rec zeri
    over=4;                       % fattore di oversampling

    ww_1=0; % dichiara due word di confronto
    ww_2=0;
    ii=1;

    while ii<=n_rec
        ww_2=fread(FF,1,'uint16'); % legge dal file un uint16 (1 word)
        if (ww_1==43690)&(ww_2==48059) % ricerca le sequenze esadecimali AAAA e BBBB
            Counter=fread(FF,2,'uint16'); % legge dal file 2 uint16: i due data counter
            Scan(ii)=bitshift(bitshift(uint16(Counter(1)),6),-6); % Scan Number
            N_PRI(ii)=bitshift(uint32(Counter(1)),-10)+uint32(Counter(2))*2^6;
                                                    % PRI Number: i 6 MSB di Data
                                                    % Counter 1 più i 16 bit di
                                                    % Data Counter 2

            if f_presum==0

```

```

Data=cast(bitshift(bitshift(uint16(fread(FF,n_word,'int16')),2),-
2),'single');

                                % legge dal file n_word a 16 bit con segno,
                                % li converte in int16, shifta di 2bit verso
                                % l'MSB poi shifta verso l'LSB e fa il cast
i_change=find(Data>=2^13); % trova l'indice dei campioni
                                % rappresentati con valore maggiore
                                % uguale di 2^13
Data(i_change)=Data(i_change)-2^14; % assegna a tali campioni i
                                % corrispettivi valori negativi del
                                % complemento a 2

else
    Data=cast(fread(FF,n_word,'int16'),'single');
end

Data=Data*conv_fact;

GPS_data=fread(FF,n_word_GPS,'int16'); % legge dal file le word del
                                        % GPS come interi con segno
                                        % con a 16 bit
Data_fft=fft(Data.',N_FFT); % fft di Data su N_FFT punti

imp_fft=fftshift(Data_fft(1:N_FFT/2).*conj(chirp_fft(1:N_FFT/2)));
                                % è la fft della risposta impulsiva
                                % che si ottiene come prodotto tra
                                % la trasformata di Hilbert della
                                % fft di Data ed il complesso
                                % coniugato della trasformata di
                                % Hilbert della fft di chirp (sono
                                % entrambi a N_FFT/2 punti quindi
                                % 25MHz)

imp1=ifft(imp_fft); % FFT inversa della risposta impulsiva
imp=abs(spline([0:N_FFT/2-1],imp1,[0:1/over:N_FFT/2-1]));
                                % risposta impulsiva sovracampionata
                                % con (N_FFT/2)*over punti
[max_imp(ii) i_max(ii)]=max(imp); % Massimo valore di imp e indice in
                                % cui lo raggiunge
ii_last=find(imp>=max_imp(ii)/sqrt(2),1,'last');
                                % ultimo indice in cui
                                % imp raggiunge il
                                % valore efficace
                                % max(imp)/sqrt(2)
ii_first=find(imp>=max_imp(ii)/sqrt(2),1,'first');
                                % primo indice in cui
                                % imp raggiunge il
                                % valore efficace
                                % max(imp)/sqrt(2)

```

```

width_3dB(ii)=ii_last-ii_first+1;           % ampiezza a 3dB in indici
[max_imp_rigth(ii) i_max_rigth(ii)]=max(imp(ii_last+width_3dB(ii):end));
                                           % valore massimo e
                                           % scostamento in
                                           % indice da
                                           % ii_last+width_3dB
                                           % del lobo destro
                                           % del sinc

i_max_rigth(ii)=i_max_rigth(ii)+ii_last+width_3dB(ii)-1;
                                           % indice del massimo
                                           % del lobo destro

[max_imp_left(ii) i_max_left(ii)]=max(imp(1:ii_first-width_3dB(ii)));
                                           % valore Massimo e
                                           % suo indice del
                                           % lobo sinistro

ii=ii+1;
ww_1=0;
ww_2=0;
else
    ww_1=ww_2;
end
end

t_imp=0:(2/fc)*over:(length(imp)-1)*(2/fc)*over;
t_Data=0:1/fc:(n_word-1)*1/fc;

plot(tt,10*log10(max_imp.^2/B/T/50))
xlabel('tempo [s]')
ylabel('Potenza @ ADC [dBm]')

del_m=mean(diff(i_max))/over;               % media relativa all'over degli
                                           % indici del massimo di imp

del_std=std(diff(i_max))/fc*2/over*1e9; % deviazione standard degli i_max
i_del=length(find(diff(i_max)~=mean(diff(i_max))))); % numero degli indici la cui
                                           % variazione è differente
                                           % dalla media delle
                                           % variazioni

MAX_INP=db(mean(max_imp));                 % media dei massimi della risposta impulsiva in dB
MAX_STD=std(max_imp)/mean(max_imp)*100;   % deviazione standard dei massimi
                                           % della risposta impulsiva
                                           % normalizzata alla media di tali
                                           % massimi

width_3dB_m=mean(width_3dB)/fc*2/over*1e9; % ampiezza a 3dB
PSLR_rigth=db(mean(max_imp_rigth./max_imp)); % Peak Side Lobe Ratio destro
PSLR_left=db(mean(max_imp_left./max_imp)); % Peak Side Lobe Ratio sinistro

fclose(FF);                               % chiude il file

```

```

end

ff_out=fopen('autocheck.dat','w');

fprintf(ff_out,'\n\ndel_m*100=%d\ndel_std*100=%d\ni_del=%d\nMAX_INP*10=%d\nMAX_STD*1000=
%d\nwidth_3dB_m*10=%d\nPSLR_rigth*10=%d\nPSLR_left*10=%d\n
',floor(del_m*100),floor(del_std*100),i_del,floor(MAX_INP*10)/10,floor(MAX_STD*1000)/100
0,floor(width_3dB_m*10),floor(PSLR_rigth*10),floor(PSLR_left*10));

fprintf(1,'\n\ndel_m*100=%d\ndel_std*100=%d\ni_del=%d\nMAX_INP*10=%d\nMAX_STD*1000=%d\nw
idth_3dB_m*10=%d\nPSLR_rigth*10=%d\nPSLR_left*10=%d\n
',floor(del_m*100),floor(del_std*100),i_del,floor(MAX_INP*10)/10,floor(MAX_STD*1000)/100
0,floor(width_3dB_m*10),floor(PSLR_rigth*10),floor(PSLR_left*10));

fclose(ff_out);           % chiude il file autocheck

```

5.3.3 Acquisizione in modalità autocheck

Mostriamo un esempio dell'utilizzo del `sora_proc`, nel caso di acquisizione dati in modalità autocheck ed in configurazione no presuming, quindi `flag=2`.

Nel Par. 1.3.2, abbiamo visto come la CDMU, tramite protocolli su interfacce seriali, invia dei comandi al payload di SORA. Dopo l'accensione del radar (POWER ON), viene settata la configurazione di AUTOCHECK che si ottiene come abbiamo visto nel funzionamento della RFU, andando ad agire sugli switch elettromeccanici SW1 e SW2. Con la modalità DUMP si copiano i dati acquisiti dal disk array controller al disk array. Questi vengono passati al disco di Windows dove viene generato il file "00000001a.dat", dove la "a" indica l'acquisizione in modalità AUTOCHECK. Per l'AUTOCHECK, è prevista la modalità PROCESSING che consiste nell'acquisire dati a ciclo chiuso e processarli per stimare determinati parametri e valutare se il RADAR sta funzionando correttamente.

Con l'utilizzo di un breakpoint, fermiamo l'esecuzione del programma al termine del ciclo `while`. Ciò ci consente di poter analizzare i segnali presi in considerazione nel

tempo e in frequenza: i dati acquisiti dall'ADC (`Data`), la sua FFT (`Data_fft`), il chirp di riferimento che implementa la risposta impulsiva del filtro adattato (`chirp`), la sua FFT (`chirp_fft`), la risposta impulsiva del radar (`imp`) dopo che è stata sovracampionata.

```
K>> plot(t_Data*1e6,Data)
K>> xlabel('tempo [µsec]')
K>> ylabel('Data [Volt]')
```

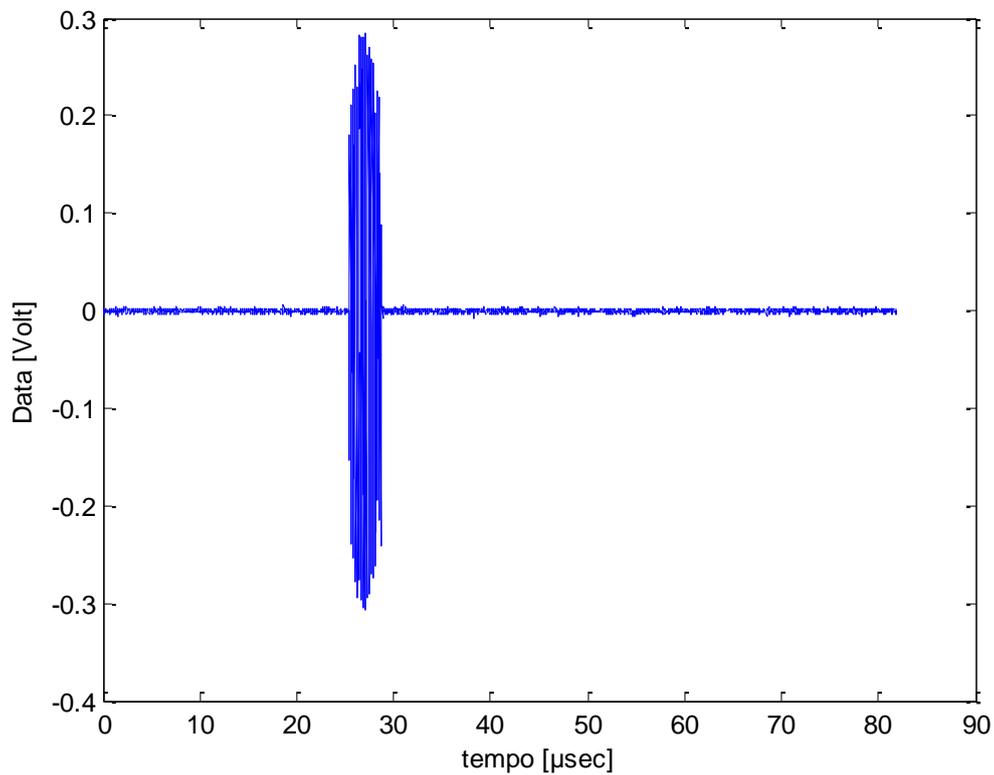


Figura 5-2: Dati acquisiti

Da questo plot possiamo notare che il segnale acquisito, campionato con 4095 punti è un impulso limitato nel tempo, affetto da rumore additivo.

Con il successivo plot zoommando sulla finestra temporale noteremo l'effetto della modulazione di frequenza.

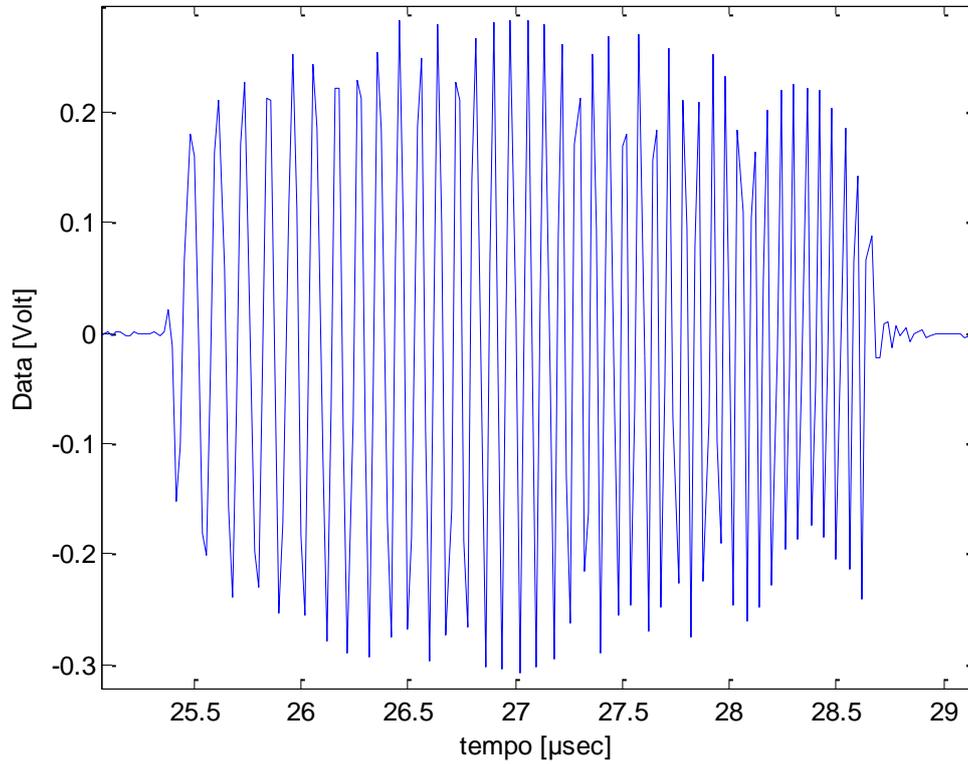


Figura 5-3: Zoom di Data

La sua FFT può essere visualizzata con il comando:

```
K>> plot(ff*1e-6,abs(Data_fft))  
K>> xlabel('frequenza [MHz]')
```

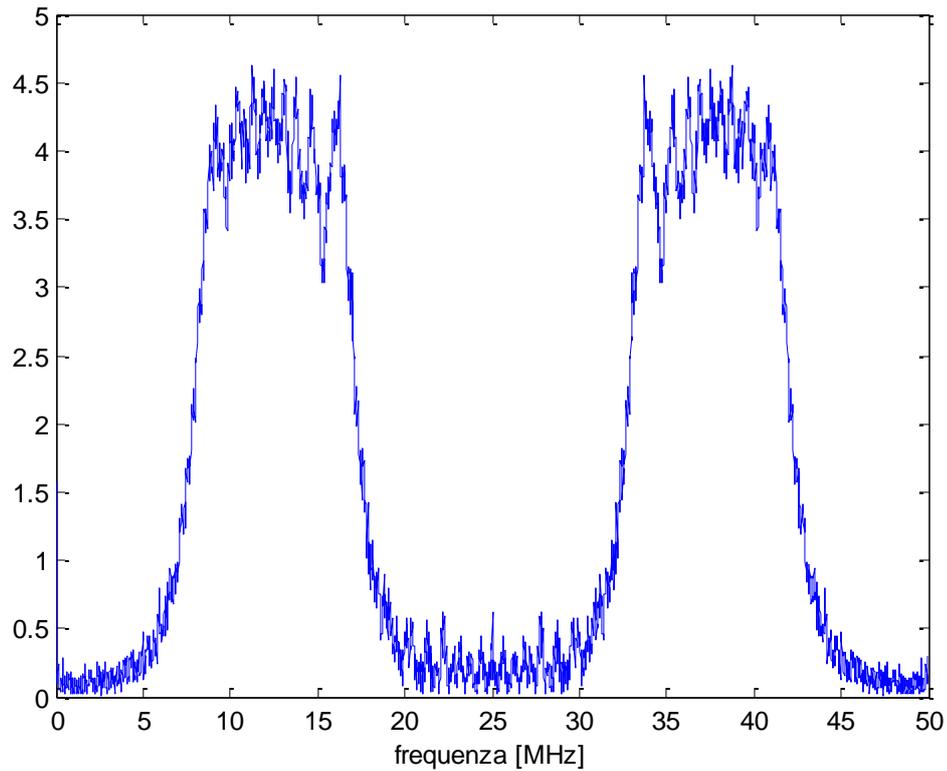


Figura 5-4: Modulo della FFT di Data

Della FFT del segnale acquisito convertito in volt, che è una funzione complessa, ci andiamo a considerare il modulo. Possiamo dire che il segnale ha una banda limitata in frequenza anche se i fianchi non sono perfettamente ripidi inoltre poiché è un segnale che è stato campionato presenta delle frequenze spurie.

Consideriamo ora il chirp di riferimento implementato nel codice che viene utilizzato come filtro adattato. Il segnale praticamente sarà la versione ideale discreta del segnale in uscita dal chirp generator.

```
K>> plot(t*1e6, chirp)
K>> xlabel('tempo [μs]')
```

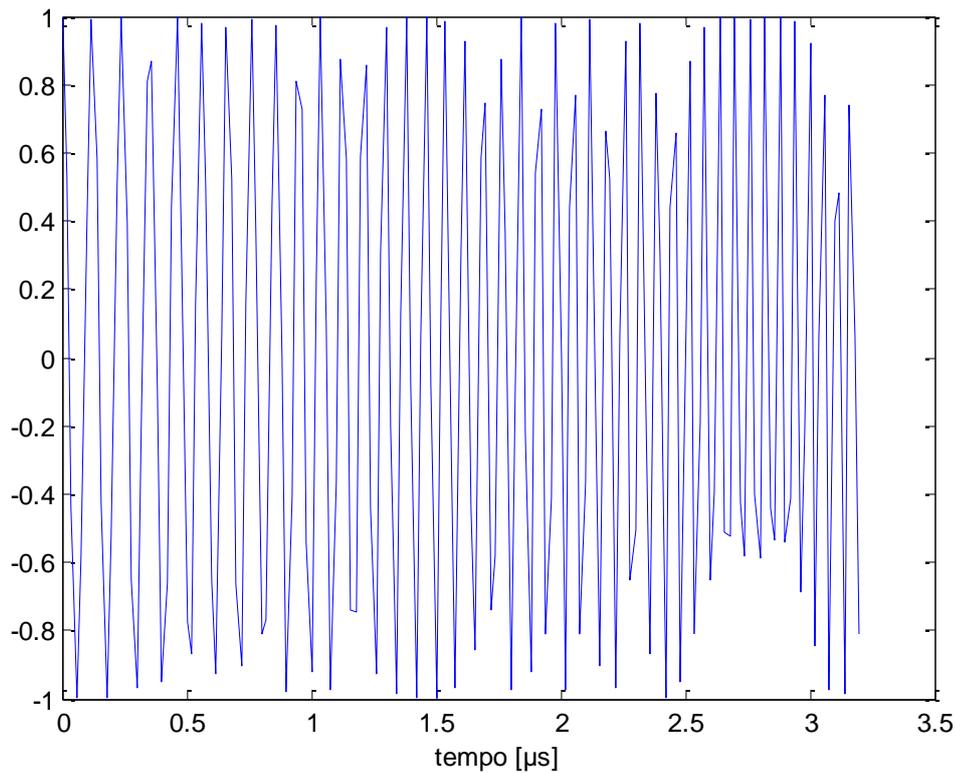


Figura 5-5: Risposta impulsiva del filtro adattato

È un segnale discreto, modulato in frequenza, di durata temporale limitata, naturalmente non affetto da rumore.

La sua FFT su 4096 punti implementerà la funzione di trasferimento del filtro adattato. Poiché si tratta di un coseno modulato a frequenza lineare, il suo spettro sarà a banda limitata.

```
K>> plot(ff*1e-6,abs(chirp_fft))
K>> xlabel('frequenza [MHz]')
```

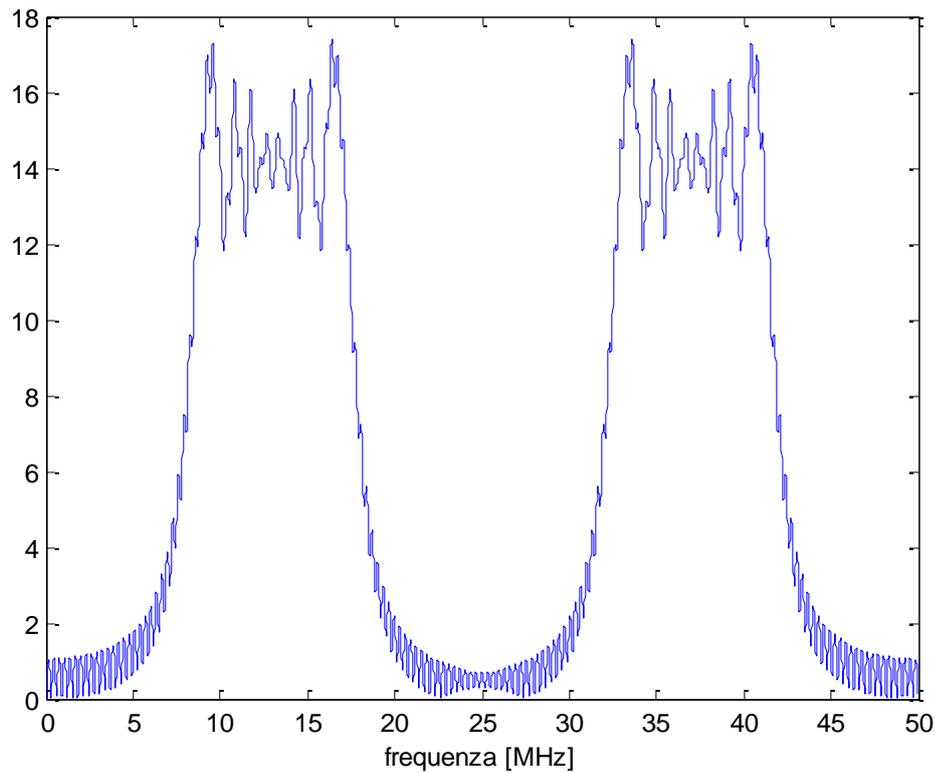


Figura 5-6: Funzione di trasferimento del filtro adattato

Se ora andiamo a sovrapporre la FFT degli echi acquisiti e la FFT del chirp di riferimento, normalizzati ciascuno al loro massimo assoluto, noteremo che hanno un andamento simile, ma il segnale ricevuto è affetto da frequenze spurie e da un decadimento.

```
K>> plot(ff*1e-6,abs(chirp_fft) /max(abs(chirp_fft)), 'r',ff*1e-6,abs(Data_fft)/max(abs(Data_fft)))  
K>> xlabel('frequenza [MHz]')
```

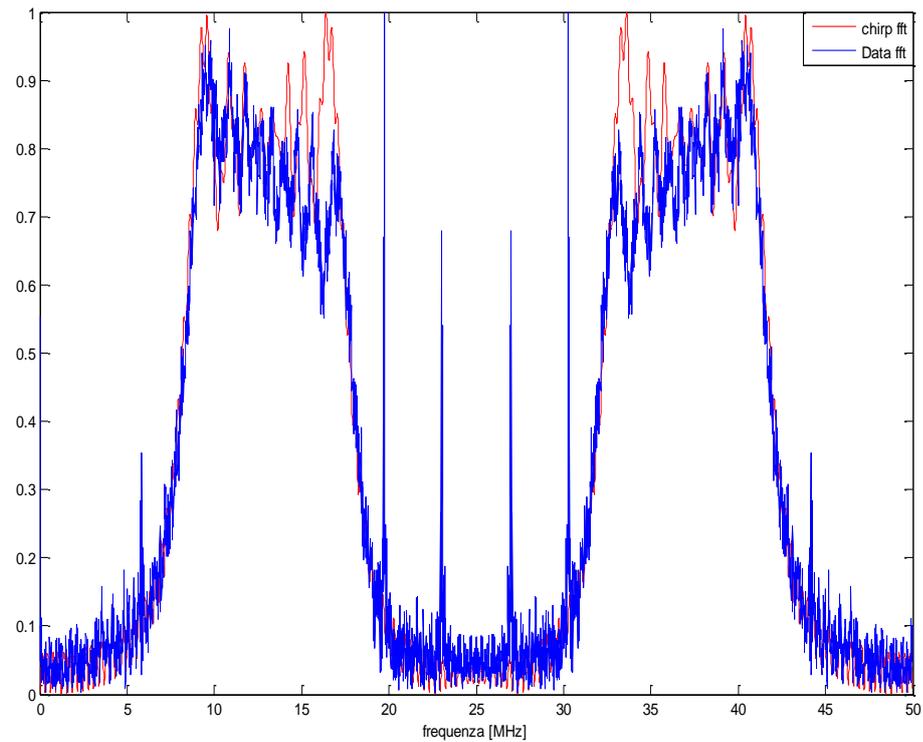


Figura 5-7: FFT dei dati e FFT del chirp di riferimento

Infine plottiamo la risposta impulsiva del radar rappresentata con $(N_FFT/2)$ over punti, che sarà pressappoco il modulo di un sinc.

```
K>> plot(t_imp*1e6, imp)
K>> xlabel('tempo [μs]')
```

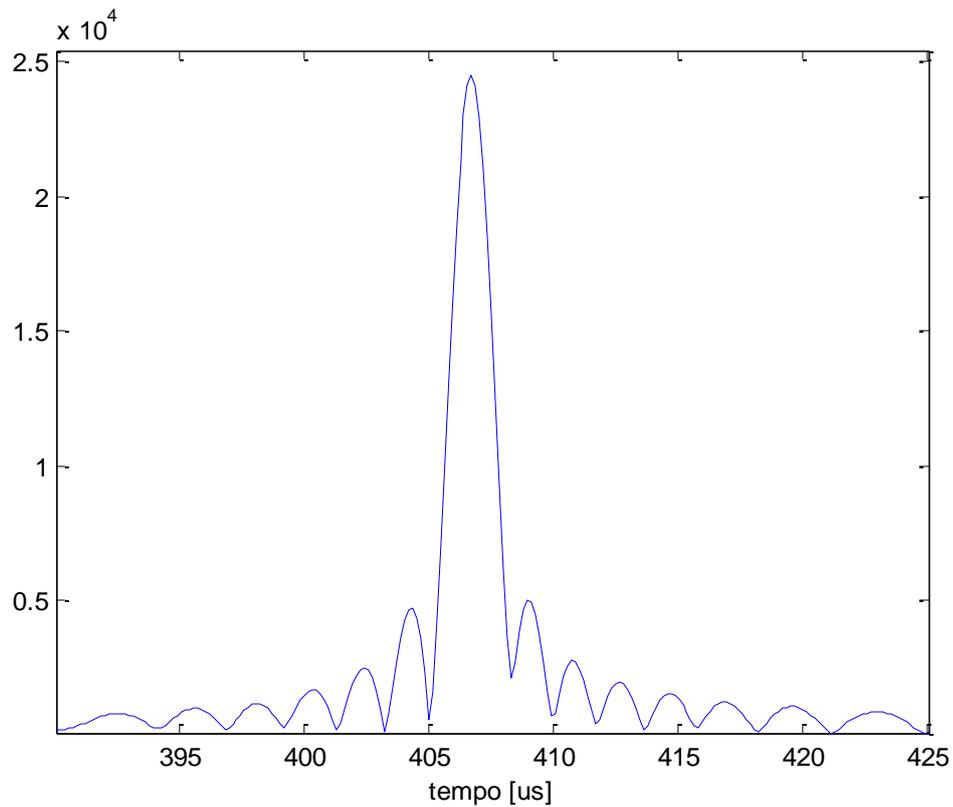


Figura 5-8: Impulso compresso

Eseguiamo adesso il `sora_proc` senza nessun breakpoint. Verrà creato un file “`autocheck.dat`” in cui verranno scritti i valori dei parametri richiesti per caratterizzare la risposta impulsiva del radar e verranno stampati anche a video.

```
del_m*100=0
del_std*100=0
i_del=0
```

Questi tre parametri rappresentano l'entità del jitter. Si evince che l'intervallo di campionamento non subisce variazioni e che il jitter ha una distribuzione molto stretta. Il fatto che siano nulli è segno che il campionamento avviene in assenza di jitter o che è presente praticamente in maniera influente. Ciò è fondamentale al fine di operare correttamente il sottocampionamento di segnali ad alta frequenza.

$\text{MAX_INP} * 10 = -5.600000e+000$

È il livello medio di potenza in dBm all'ingresso dell'ADC.

$\text{MAX_STD} * 1000 = 1.256000e+000$

È lo scostamento massimo in dBm della potenza dal suo valore medio.

$\text{width_3dB_m} * 10 = 911$

Si dimostra che l'ampiezza a 3dB è il reciproco della banda a 3dB. Nel nostro caso è un'ampiezza di 91,1ns cioè $1/91,1\text{ns} = 10,97\text{MHz}$, mentre nominalmente il nostro chirp dovrebbe avere una banda di 10MHz. Ciò significa che il chirp generator deve essere sottoposto ad una taratura.

$\text{PSLR_righth} * 10 = -150$

$\text{PSLR_left} * 10 = -145$

Questi due rapporti espressi in dB devono essere molto simili tra loro e tengono conto della distorsione che subisce il segnale e di quanto ciascuno dei due lobi laterali influisce sul lobo centrale.

CONCLUSIONI

Il lavoro di tesi realizzato ha riguardato lo studio del sistema di acquisizione dati del radar SORA.

Dapprima si è andato ad analizzare l'architettura del payload SORA, di cui si è appreso le varie modalità di funzionamento, le sue interfacce ed i suoi sottosistemi cercando di focalizzare maggiormente l'attenzione sugli aspetti riguardanti i segnali da acquisire e le relative modalità d'interesse, inoltre s'è discusso delle problematiche relative ad ambienti a bassa pressione ed a bassa temperatura. Si è poi affrontato l'argomento segnali di chirp che come abbiamo visto presentano notevoli vantaggi rispetto agli impulsi rettangolari comuni. Quindi si è accennato alla tecnica del sottocampionamento per poi passare al funzionamento dell'LTC2208-14. Infine è stato sviluppato il software Matlab "sora_proc.m" che consente di analizzare i segnali di ritorno acquisiti ed elaborarli in modo tale da ricavare dei parametri che permettano di caratterizzare l'acquisizione dati.

Maggiori conclusioni, non solo sul sistema di acquisizione ma in generale sull'intero payload, si potranno trarre nell'estate 2009 quando avrà luogo la missione SORA. Da quest'ultima si ricaveranno preziose informazioni sull'inversione dei dati e sull'eliminazione delle ambiguità di missioni come SHARAD che mirano ad investigare la superficie di Marte.

La possibilità di utilizzare un clock più basso offerta dalla tecnica del sottocampionamento e la produzione di un Data Volume più contenuto che si ottiene con la tecnica del presampling, consentono di utilizzare A/D Converter con risoluzioni maggiori. Eventuali sviluppi futuri di questa tesi potrebbero essere l'implementazione di queste tecniche in altri sistemi radar quali i S.A.R. (Synthetic Aperture Radar).

BIBLIOGRAFIA

- [1] **PRC_COR_ADD_08_003**
Alberti G., Bortone M., Galiero G., Mattei S., Palmese G.
- [2] **PRC_COR_NTE_08_002_01**
Giovanni Alberti, Stefania Mattei
- [3] **Introduction to Radar Systems**
MI Skolnik, McGraw-Hill, New York, (1985)
- [4] **Considerazioni sulla digitalizzazione e l'elaborazione dei segnali SODAR**
Nota Interna, IFA-CNR, Angelo Ricotta, Luglio 1983
- [5] **When undersampling, clock jitter does matter**
By Bonnie Baker -- EDN, 8/4/2005
www.edn.com/article/CA629307.html
- [6] **On the Jitter Requirements of the Sampling Clock for Analog-to-Digital Converters**
Nicola Da Dalt, Moritz Harteneck, Christoph Sandner and Andreas Wiesbauer
IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—I:
FUNDAMENTAL THEORY AND APPLICATIONS, VOL. 49, NO. 9,
SEPTEMBER 2002
- [7] **Effect of Jitter on Asynchronous Sampling With Finite Number of Samples**
Nicola Da Dalt

IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—II: EXPRESS
BRIEFS , VOL. 51, NO. 12, DECEMBER 2004

- [8] **Audio precision technote TN-23 Jitter theory**
Julian Dunn

- [9] **Measuring Spurious Free Dynamic Range in a D/A Converter**
Juan Garcia, Stephen G. LaJeunesse, Douglas Bartow
Intersil Technical Brief, January 1995 - TB326

- [10] **Linear Tecnology QUICK START GUIDE FOR DEMONSTRATION
CIRCUIT 996**

- [11] **Linear Tecnology QUICK START GUIDE FOR DEMONSTRATION
CIRCUIT 854**

- [12] **Linear Tecnology LTC2208-14 Data Sheet**