

Seconda Università degli Studi di Napoli



FACOLTA' DI INGEGNERIA
CORSO DI LAUREA MAGISTRALE IN INGEGNERIA ELETTRONICA

TESI DI LAUREA
IN
ELETTRONICA DEI SISTEMI DIGITALI

**PROGETTAZIONE DI UN SINTETIZZATORE DIGITALE DI
FREQUENZA PER APPLICAZIONI RADAR**

Relatore:

Ch.mo Prof.
LUIGI ZENI

Candidato:

LUIGI DI MARTINO
Matricola n. A17/019

Correlatore:

Ing. LUCA CIOFANIELLO

Anno Accademico 2011-2012

A Marte

INDICE

Introduzione	IV
1. Cenni sul Telerilevamento	1
1.1 concetti Generali.....	1
2. Generalità sul Radar	5
2.1 Principio di funzionamento di un radar.....	5
2.2 RAR Real Aperture Radar.....	10
2.3 Risoluzione in azimuth di un sistema RAR.....	12
2.4 SAR Synthetic Aperture Radar.....	15
2.5 Limitazioni sulla scelta della PRF.....	20
2.6 Chirp Radar.....	21
2.7 Il SAR nelle applicazioni.....	24
3. Sintesi di frequenza	28
3.1 Introduzione alla sintesi di frequenza.....	28
3.2 Sintesi diretta analogica.....	32
3.3 Sintesi indiretta.....	38
3.3.1 Divisore.....	40
3.3.2 Frequency Phase Detector.....	42
3.3.3 Charge Pump.....	44
4. Sintesi diretta digitale	46
4.1 Introduzione alla sintesi digitale diretta.....	46
4.2 Architettura di base.....	48
4.3 Studio dello spettro del segnale di uscita.....	52
4.3.1 Effetti della risoluzione del convertitore D/A sulle spurie di uscita.....	54
4.3.2 Effetti del troncamento di fase.....	57
4.3.3 Considerazioni sul filtraggio di uscita del DDS.....	66

4.4 DDS Analog Devices.....	70
4.4.1 DDS Analog Devices AD9858	70
4.4.2 DDS Analog Devices AD9910.....	71
4.4.3 Confronto tra AD9858 e AD9910.....	73
5. Il Sintetizzatore frequenziale AD9858.....	75
5.1 Introduzione.....	75
5.2 Blocchi funzionali	79
5.2.1 DDS Core.....	79
5.2.2 DAC Output	79
5.2.3 PLL frequency Synthesizer	80
5.2.4 Phase- Frequency Detector PFD.....	80
5.2.5 Pompa di carico CP	81
5.2.6 Fast-Locking Logic	82
5.2.7 Mixer analogico	84
5.3 Modalità di Funzionamento	85
5.3.1 Single Tone	85
5.3.2 Frequency sweeping.....	86
5.4 Sincronizzazione	89
5.5 I registri dell'AD9858	90
5.5.1 Frequency Tuning Word1	91
5.5.2 Delta Frequency Word	92
5.5.3 Ramp Rate Clock.....	92
5.5.4 User Profile Registers	93
6. Gestione del chip AD9858 mediante microcontrollore microchip.....	95
6.1 Introduzione.....	95
6.2 MPLAB : ambiente di sviluppo integrato.....	102

<i>6.3 Programmazione del PIC</i>	103
<i>6.4 Gestione del sincronismo</i>	106
<i>6.5 Collegamento dei dispositivi</i>	109
<i>6.6 Firmware in Borland C++</i>	111
<i>6.7 Test sul dispositivo</i>	116
<i>6.7.1 Test Report</i>	119
Conclusioni	122
Bibliografia	125
Ringraziamenti	

INTRODUZIONE

L'osservazione della Terra dallo spazio sta sempre più imponendosi come uno dei mezzi maggiormente efficaci per il controllo del pianeta, a salvaguardia del suo patrimonio ambientale.

Le competenze sistemistiche e tecnologiche maturate nel corso di decenni si sono concretizzate nello sviluppo di sofisticati strumenti di osservazione come Radar Altimetri e Radar ad Apertura Sintetica, in grado di monitorare vari aspetti sullo stato del pianeta: dal clima all'inquinamento, dalla desertificazione alla deforestazione, dai ghiacci polari alle correnti marine, fino alle emergenze dovute alle catastrofi naturali.

Inoltre, nello scenario internazionale odierno una grande importanza è rivestita dai sistemi di informazione e di controllo dell'informazione stessa. Sia in ambito civile che in quello militare, si risente la necessità di avere un controllo globale del territorio con l'acquisizione nel tempo minore possibile di informazioni il più precise possibili.

Tutto ciò si può ottenere grazie a sistemi di telerilevamento sempre più avanzati in termini di risoluzione e di velocità di trasmissione.

In tale ambito la generazione di segnali periodici e le relative tecniche di sintesi, digitale e analogica, possono considerarsi un argomento chiave nel telerilevamento.

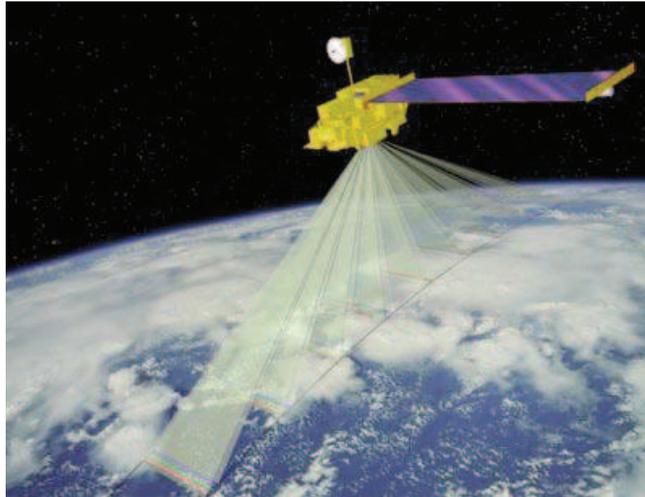


Fig 1.1- SAR: Synthetic Aperture Radar

Il presente elaborato di tesi ha come oggetto lo studio di un particolare sintetizzatore di frequenze per applicazioni radar, in fase di realizzazione presso il CO.RI.S.T.A. (Consorzio di Ricerca su Sistemi di Telesensori Avanzati).

Nei primo capitolo viene introdotto il concetto di telerilevamento.

Nel secondo capitolo viene descritto il principio di funzionamento di un radar tradizionale e quello di un radar ad apertura sintetica, illustrandone le funzionalità specifiche.

Nel terzo capitolo vengono descritti i metodi di sintesi analogica tradizionale, che sono ormai poco utilizzati, ma indispensabili alla comprensione delle tecniche moderne. Nel seguito si studia la sintesi indiretta.

Nel quarto capitolo viene descritta la sintesi digitale diretta, mentre nel quinto capitolo sarà analizzata la scheda utilizzata per l'implementazione del sintetizzatore di frequenza.

Nel sesto capitolo viene descritto la gestione del sincronismo dell'AD9858 effettuata tramite il microcontrollore PIC16F877A, e i test effettuati sul dispositivo.

Seguono le conclusioni.

1. CENNI SUL TELERILEVAMENTO

1.1 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.

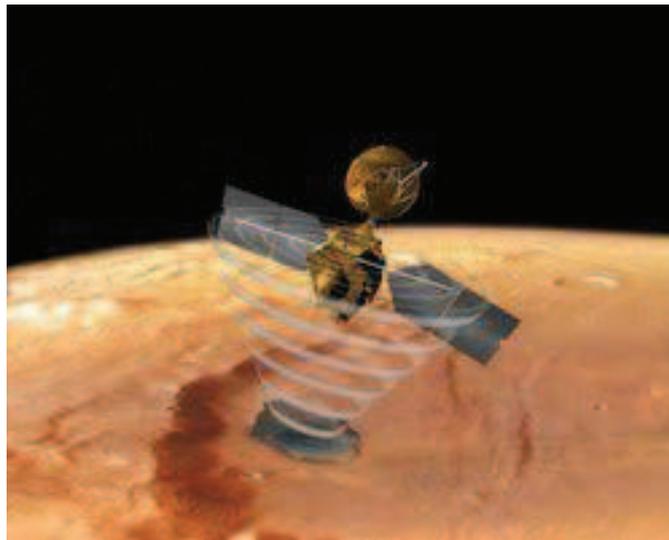


Figura 1.1 Rappresentazione digitale del MARS Reconnaissance Orbiter

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, studiando la radiazione elettromagnetica diffusa o emessa da un oggetto. In pratica ogni oggetto è caratterizzato da una propria capacità di riflessione,

assorbimento e trasmissione delle onde elettromagnetiche che ne permette la caratterizzazione.



Figura 1.2-Sicily, Italy - MERIS (MEdium Resolution Imaging Instrument) (ESA image)

Oltre agli oggetti situati sulla superficie terrestre, anche l'atmosfera può assorbire, riflettere o trasmettere in modo diverso nello spazio e nel tempo la radiazione elettromagnetica, comportandosi come un disturbo quando non rappresenta l'oggetto di studio della rilevazione. Di conseguenza la scelta delle lunghezze d'onda utilizzabili è limitata a particolari domini, quali il visibile ($\lambda = 0.4 \div 0.7 \mu m$), il vicino infrarosso ($0,7 \div 2,0 \mu m$), l'infrarosso termico (circa $5 \div 10 \mu m$) e il dominio delle microonde. Questi domini rappresentano, infatti, finestre di trasparenza dell'atmosfera terrestre alle radiazioni elettromagnetiche.

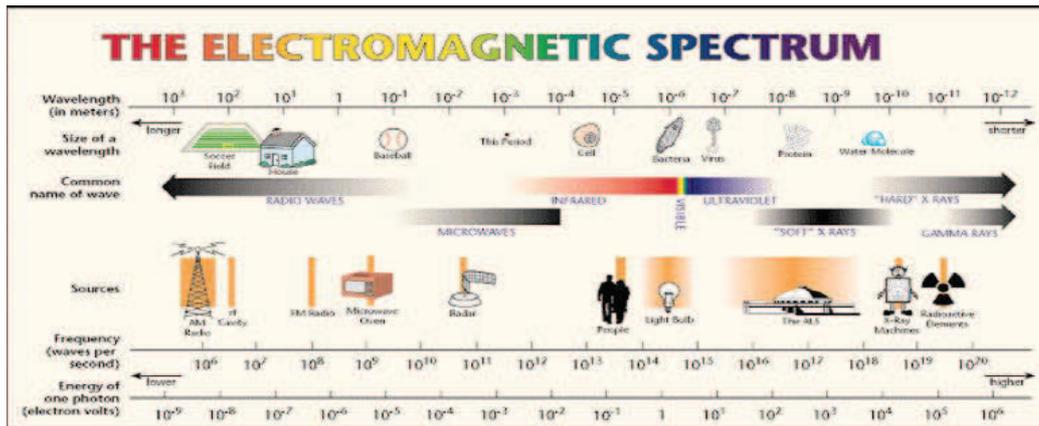


Figura 1.3 - Spettro Elettromagnetico

Ci sono essenzialmente due aspetti utilizzabili per descrivere la radiazione ricevuta:

1. Quanta radiazione è rilevata;
2. Quando la radiazione è rilevata.

Il primo aspetto permette di discriminare la superficie terrestre sulla base del diverso comportamento elettromagnetico dei diversi tipi di copertura, il secondo, invece, permette di determinare la distanza del sensore dal target ed è alla base del funzionamento di strumenti quali il LIDAR (acronimo di LIGht Detection And Ranging), il radar altimetrico e altri tipi di sistemi radar.

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. GENERALITÀ SUL RADAR

2.1 Principio di funzionamento di un radar

Il Radar (Radio Detection And Ranging) è essenzialmente un sistema di rilevamento funzionante sul principio dell'eco, in cui il trasmettitore irradia periodicamente energia sotto forma di impulsi di microonde di grande potenza, ma di durata molto breve.

Gli impulsi radar vengono irradiati da un'antenna fortemente direttiva (ad esempio una parabolica) che li trasmette alla velocità della luce verso il bersaglio.

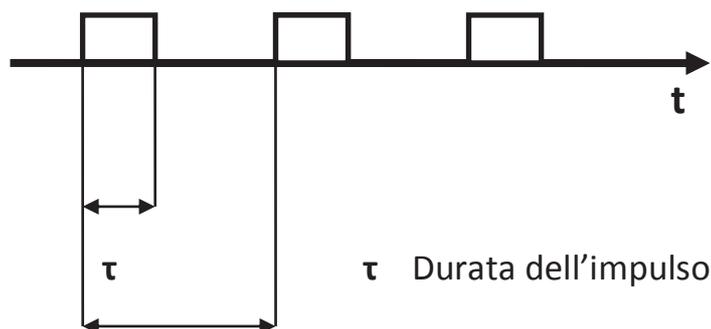


Figura 2.1 - Generazione degli impulsi radar

In realtà gli impulsi sono modulati perché le antenne funzionanti alle basse frequenze hanno una bassa direttività, il radar invece necessita di antenne fortemente direttive in quanto la potenza irradiata da un radar non si deve disperdere ma deve essere concentrata in una specifica direzione.

Se gli impulsi trasmessi non incontrano alcun ostacolo, non tornano più indietro, mentre se incontrano un aereo, una nave, una montagna, una piccola parte dell'energia irradiata ritorna all'antenna trasmittente dopo un tempo brevissimo sotto forma di eco.

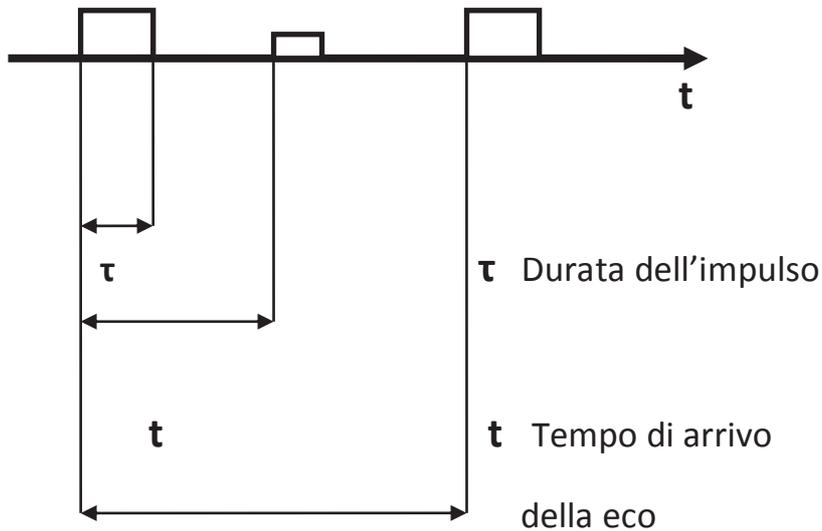


Figura 2.2 - Ritorno dell'eco del segnale emesso dal radar

Poiché la velocità di propagazione degli impulsi radar è nota, dal tempo impiegato dal segnale a raggiungere il bersaglio e a ritornare indietro, si può ricavare la distanza dallo stesso.

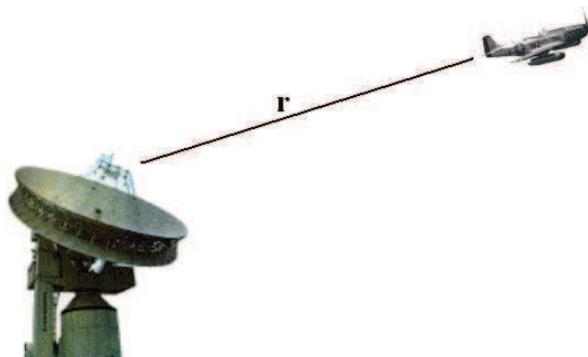


Figura 2.3 - Distanza Radar- bersaglio

Se c è la velocità della luce, r è la distanza dal bersaglio e t è il tempo dopo il quale ritorna l'eco dell'impulso all'antenna che lo ha trasmesso, si ha:

$$2 r = c t$$

Ovvero:

$$r = \frac{c t}{2}$$

La distanza massima quindi che possiamo coprire è pari a:

$$r_{MAX} = \frac{c T}{2}$$

La direzione del bersaglio è individuata dall'orientamento dell'antenna parabolica del trasmettitore per mezzo dell'angolo di *azimuth* e dell'angolo di *elevazione* che ne determinano la posizione rispetto all'antenna radar come è descritto nella figura seguente.

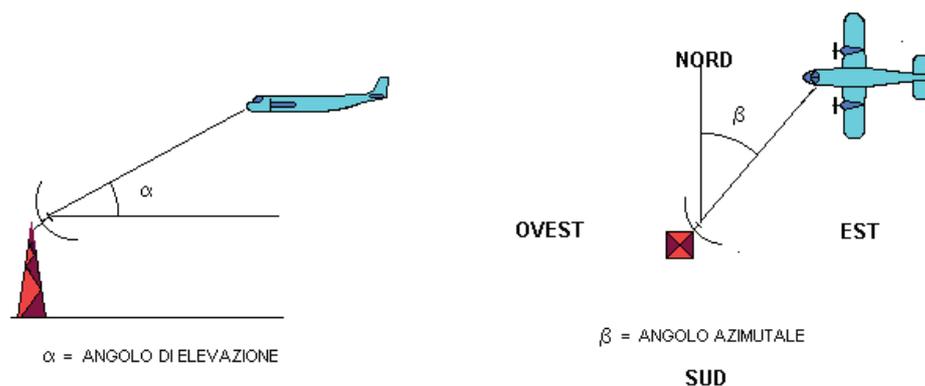


Figura 2.4 - Individuazione della direzione del bersaglio.

La durata degli impulsi τ è un parametro che va scelto in maniera opportuna perché da questo dipenderà la risoluzione spaziale del radar.

Ad esempio supponiamo di avere due oggetti vicini:

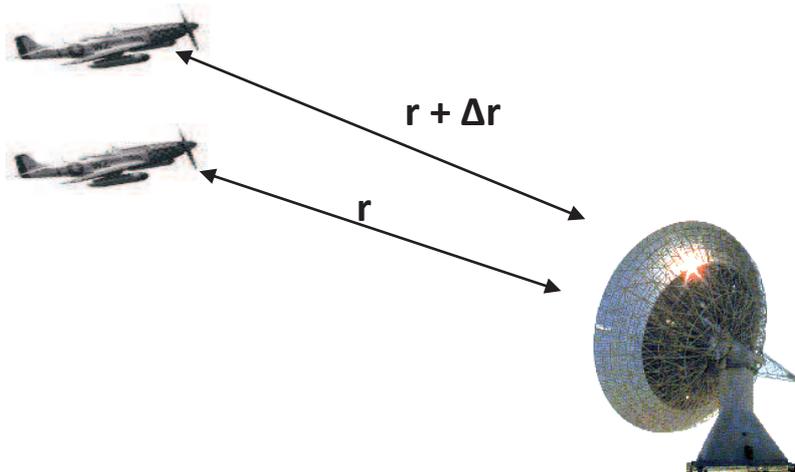


Figura 2.5 - Caso di oggetti vicini

In questo caso il radar riceverà due eco:

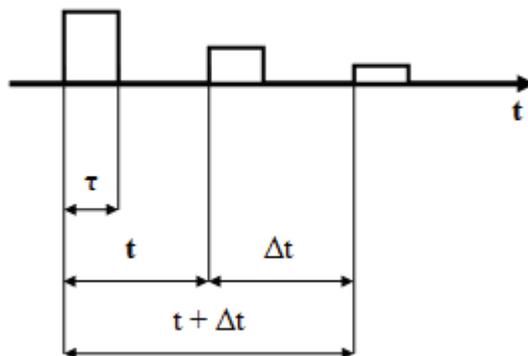


Figura 2.6 - Eco provenienti da due oggetti vicini

Per fare in modo che le due eco non si sovrappongano si deve imporre che:

$$\Delta t > \tau$$

Ma Δt è pari ad:

$$\Delta t = \frac{2 \Delta r}{c}$$

Quindi:

$$\tau \leq \frac{2 \Delta r}{c}$$

Il Δr è fissato dalle specifiche di progetto del radar.

Nel caso ideale di impulso di Dirac, la risoluzione è infinitamente alta (per “alta” si intende un “piccolo” valore della minima distanza tra due bersagli distinguibili), ed è quindi quello a cui si tende. Nel caso reale, l’impulso trasmesso ha comunque una durata nel tempo, seppur brevissima.

Per quanto riguarda invece, la risoluzione angolare questa dipende dalla direttività dell’antenna come verrà successivamente descritto.

2.2 RAR Real Aperture Radar

Per poter affrontare lo studio del principio su cui si fonda il Radar ad Apertura Sintetica è necessario introdurre la geometria di base del sistema di ripresa. Tale geometria è comunque comune ad un qualunque tipo di radar, anche quindi ad un sistema radar tradizionale, denominato RAR (Real Aperture Radar) che è portato a bordo di una piattaforma di volo.

Consideriamo la fig. 2.7.

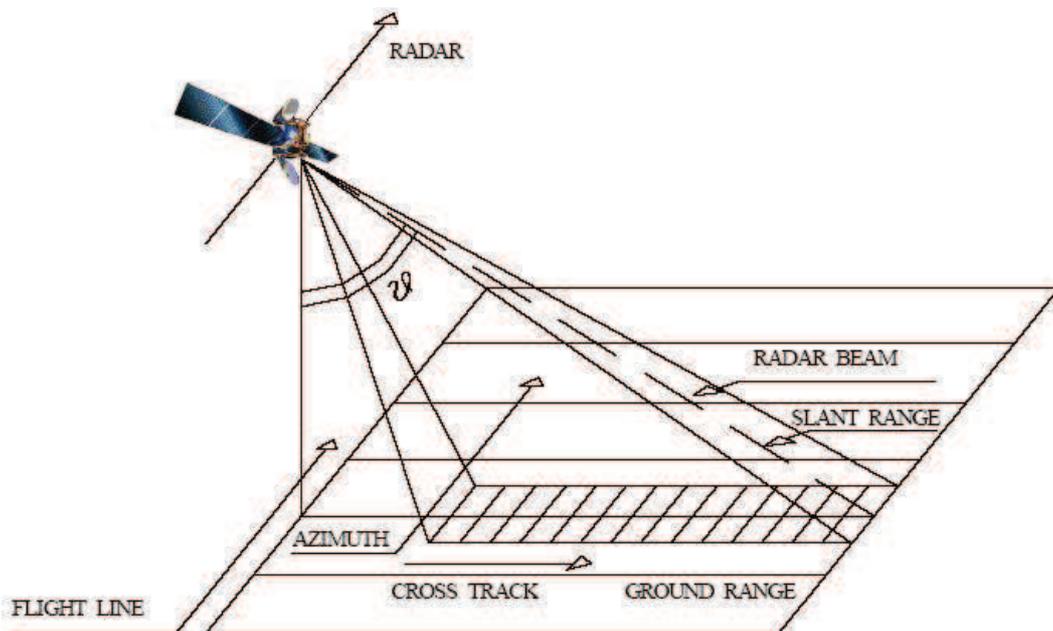


Figura 2.7 - Geometria di un sistema RAR

La direzione di volo del radar (along track) è detta di azimuth, invece, la direzione ortogonale a quest'ultima (cross track) è detta di range. Si parla di slant range riferendosi alla direzione di trasmissione in linea d'aria del segnale e di ground range riferendosi a quella riportata a

terra. Queste due direzioni sono chiaramente legate fra loro tramite il seno dell'angolo di "vista" θ .

Per quanto riguarda la risoluzione in range, come già precedentemente descritto, è possibile discriminare tra di loro due echi e quindi due target se la differenza dei loro ritardi è maggiore della durata τ degli impulsi.

La risoluzione in slant range è quindi pari ad:

$$\rho_{SR} = \frac{\tau c}{2}$$

Mentre quella in ground range:

$$\rho_{GR} = \frac{\tau c}{2 \sin \theta}$$

Per ottenere una migliore risoluzione in ground range, è opportuno utilizzare angoli di vista molto inclinati, in quanto per θ minori di 90° , il seno è crescente quindi ρ_{GR} diminuisce.

2.3 Risoluzione in azimuth di un sistema RAR

La risoluzione ρ_{az} lungo l'azimuth [XIV], corrisponde alla minima distanza a cui devono trovarsi, nella direzione di volo del radar, due oggetti affinché essi appaiano separati.

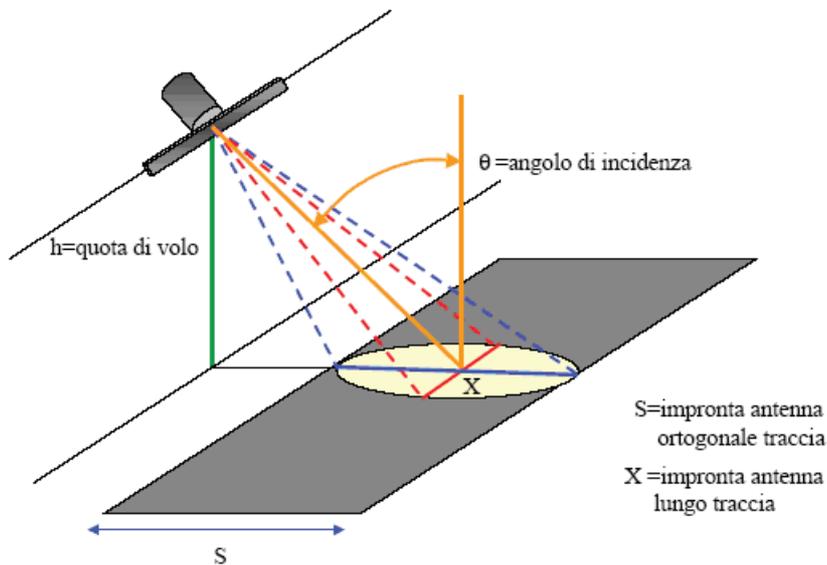


Figura 2.8 - Geometria di un Sistema RAR

Nella figura 2.8 è mostrata la geometria di un sistema radar montato a bordo di una piattaforma che si muove di moto uniforme lungo una traiettoria rettilinea. L'antenna reale del sistema radar è fissa ed è montata in modo tale che gli assi azimuthale e di ground sono diretti rispettivamente, parallelamente e ortogonalmente alla traccia della traiettoria. La direzione di puntamento del fascio forma l'angolo di incidenza Θ con il piano tangente alla superficie illuminata.

Ricordando che l'ampiezza del fascio a 3 dB per un'antenna ad apertura la cui lunghezza nella direzione di volo della piattaforma (in azimuth) sia L_{az} è esprimibile come:

$$\mathcal{G}_{AZ} = \frac{\lambda}{L_{AZ}}$$

(dove λ è la lunghezza d'onda della portante) avremo che la risoluzione lungo l'azimuth è pari ad:

$$\rho_{AZ} = r \mathcal{G}_{AZ}$$

Dove r è la distanza in linea d'aria dalla superficie irradiata.

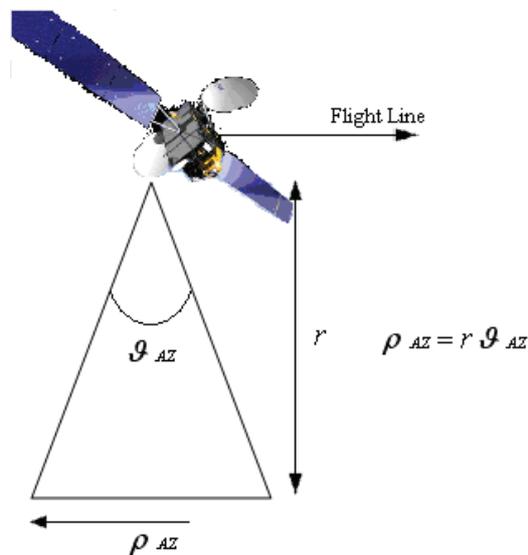


Figura 2.9 - Risoluzione in azimuth

Considerando h la quota di volo, abbiamo:

$$r = \frac{h}{\cos \mathcal{G}}$$

Avremo quindi:

$$\rho_{AZ} = \frac{h \lambda}{L \cos \mathcal{G}}$$

La risoluzione spaziale dell'immagine quindi è direttamente proporzionale alla quota ed inversamente proporzionale alla lunghezza dell'antenna.

Perciò, al crescere della quota di volo del sensore la risoluzione dell'immagine aumenta, quindi le prestazioni del radar diminuiscono, a meno che le dimensioni dell'apertura non siano aumentate. Per le lunghezze d'onda della luce visibile e dei raggi infrarossi si possono ottenere immagini ad alta risoluzione alle quote orbitali anche con dimensioni modeste dell'apertura. Ma per sensori a microonde, come il radar, dove le lunghezze d'onda sono centomila volte maggiori di quelle della luce, una risoluzione elevata non è possibile, se non con antenne aventi aperture esagerate, dell'ordine di dieci chilometri o più. L'elevata risoluzione geometrica intesa come minima distanza tra due oggetti sulla scena affinché siano rappresentati distintamente sull'immagine, rappresenta l'elemento più qualificante di un radar di immagine.

Consideriamo, ad esempio, un RAR posto su piattaforma aerea ed avente le seguenti caratteristiche:

- $\lambda = 3.1$ cm (Banda X)
- $L = 10$ metri
- $h = 7$ km
- $\theta = 29^\circ$

In tale caso la risoluzione lungo l'azimuth sarà pari a circa 24 metri. Lo stesso radar a bordo di un satellite, ad un'altezza di 700 km avrebbe una risoluzione lungo la direzione di moto 100 volte peggiore, ossia di circa 3 km.

E' questa dunque la ragione principale per la quale un RAR su piattaforma satellitare è improponibile.

2.4 SAR Synthetic Aperture Radar

Il radar ad apertura sintetica è un dispositivo che consente di aumentare la risoluzione in *azimuth* senza aumentare le dimensioni dell'antenna fisica. La risoluzione elevata, è ottenuta infatti in questo caso sintetizzando nel sistema di elaborazione del segnale un'apertura di antenna estremamente grande spostando una piccola antenna lungo una traccia (prestabilita e nota a priori) e poi elaborando coerentemente il segnale ricevuto. Ciò è realizzato in forma digitale da un computer a terra. Il risultato è che il SAR raggiunge una risoluzione in *azimuth* indipendente dalla quota a cui si trova il sensore (come la risoluzione in *range*), caratteristica che lo rende uno strumento eccezionale per l'osservazione dallo spazio.

Ogni sensore progettato per il telerilevamento ad alta risoluzione dallo spazio genera necessariamente una grande mole di dati. Per ottenere un'immagine SAR in tempo reale occorre un computer in grado di effettuare molte miliardi di operazioni al secondo. Perciò le potenzialità del SAR si sono potute sfruttare appieno solo in seguito agli eccezionali progressi dell'elettronica e dell'informatica realizzati negli ultimi decenni. La serie di procedure cui sono sottoposti i dati provenienti dal sensore è assai complessa.

La geometria di osservazione del SAR nella direzione azimutale (come già precedentemente descritto in *range* non cambia nulla rispetto al RAR) è schematizzata nella figura 2.10.

Se indichiamo con h la quota di volo, la generica distanza r tra una “stazione” della piattaforma (il punto x_i sulla traiettoria di volo) ed un

bersaglio posto nel punto P varierà al variare della posizione della piattaforma ($r = r(s)$), e varrà:

$$r = \sqrt{r_0^2 + V^2 s^2}$$

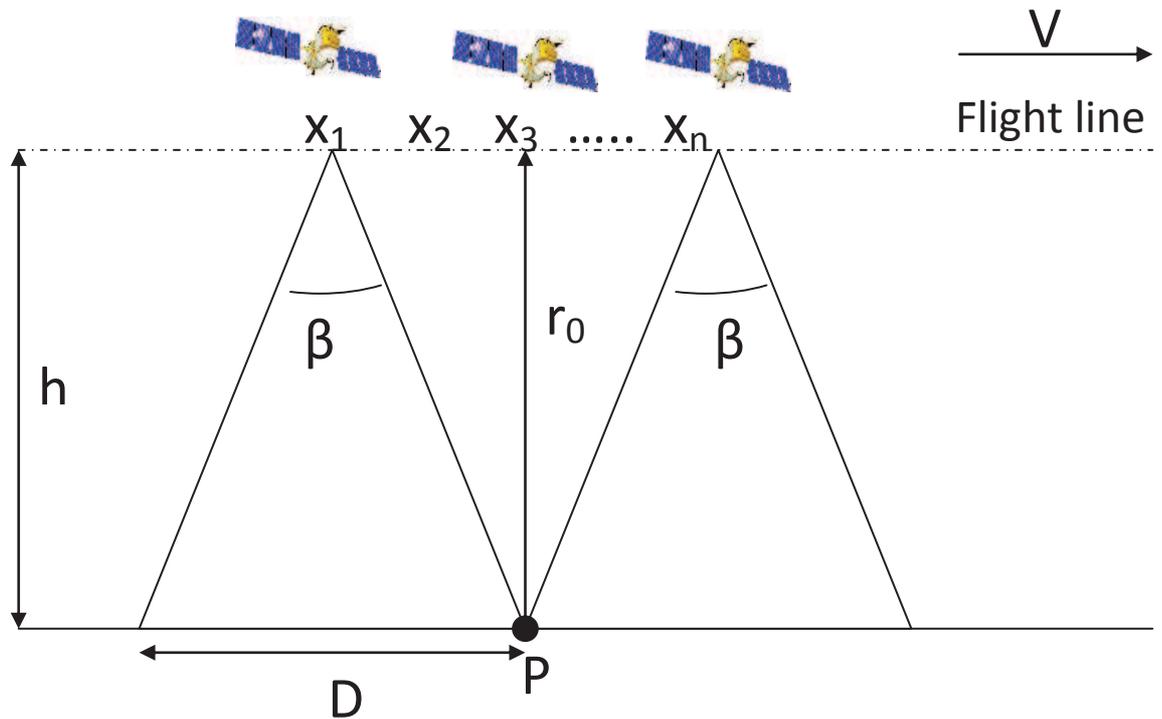


Figura 2.10 - Geometria mostrante la formazione di un array sintetico ottenuto dal moto di un'antenna reale lungo l'azimuth. Il punto P viene illuminato n volte dagli impulsi trasmessi in ogni stazione.

dove V è la velocità della piattaforma, e il tempo s è valutato a partire da una stazione iniziale, presa come riferimento (per esempio, la stazione x_1).

Man mano che il sensore si muove, è possibile registrare lungo la linea di volo i segnali di ritorno nei punti x_1, x_2, \dots, x_n . In corrispondenza di ogni stazione il radar emette un impulso, e subito dopo (la scala dei tempi è "veloce", nel senso che la propagazione è governata dalla velocità dell'onda elettromagnetica, cioè c) riceve

l'energia retro-irradiata dalla scena illuminata. E' conveniente utilizzare, nel modellare il funzionamento in azimuth del SAR, la cosiddetta *approssimazione stop-start* : si ritiene cioè che la piattaforma resti ferma durante il tempo necessario a trasmettere e ricevere l'impulso, e che si porti istantaneamente sulla posizione della stazione successiva.

Il tempo che intercorre tra una posizione e l'altra vale $1 / \text{PRF}$ dove PRF è la Pulse Repetition Frequency ovvero il numero di impulsi per unità di tempo.

A bordo della piattaforma un dispositivo provvederà a registrare "coerentemente" questi segnali, ossia tenendo conto dell'ampiezza e della fase in funzione del tempo.

La fase del segnale ricevuto è legata a $r(s)$:

$$\varphi(s) = \frac{4\pi}{\lambda} r = \frac{4\pi}{\lambda} \left(r_0 + V^2 \frac{s^2}{2r_0} \right)$$

avendo ragionevolmente considerato $V^2 s^2 / r_0^2 \ll 1$.

Si nota un termine di fase costante (non dipendente dal tempo) ed un termine quadratico (dipendente da s^2). Se quest'ultimo termine non esistesse, la frequenza ricevuta dal radar sarebbe uguale a quella trasmessa, e non ci sarebbe uno scostamento Doppler. Il secondo addendo è quindi il responsabile dello scostamento Doppler "registrato" dal radar in ricezione:

$$f_D = \frac{1}{2\pi} \frac{d\varphi}{dt} = \frac{2V^2 s}{\lambda r_0}$$

Il SAR registrerà coerentemente (cioè immagazzinando ampiezza e fase dell'eco ricevuto) tutti gli echi radar provenienti dal bersaglio per tutto il tempo in cui sarà "inquadrato" dall'antenna reale, e cioè in tutte le stazioni x_1, x_2, \dots, x_n . Questo processo corrisponde a "sintetizzare" n antenne in posizioni diverse, ed a combinare i segnali ricevuti in modo da formare un "allineamento" (o *array*) di antenne reali, che prende il nome di "Antenna sintetica".

Il corrispondente tempo di "inquadramento" (noto come tempo di integrazione) T_i sarà legato alla dimensione D (l'impronta dell'antenna a terra), che a sua volta dipende dall'ampiezza del fascio azimutale (β) dell'antenna reale.

$$T_i = \frac{r \beta}{V} = \frac{r \lambda}{V L}$$

Al variare di s , quindi l'eco ricevuto avrà una "storia Doppler", nel senso che lo scostamento Doppler $f_D(s)$ ad esso associato dipenderà dal tempo (linearmente, in questo caso).

Lo scostamento Doppler varierà tra un massimo ed un minimo, e sarà nullo in corrispondenza della distanza r_0 da P (r_0 si chiama *closest approach*).

E' importante valutare, a questo punto, l'intervallo di scostamenti Doppler (la banda Doppler B_D) dell'eco ricevuto durante la formazione dell'"antenna sintetica", cioè durante l'acquisizione degli echi delle n stazioni.

Valutando $f_D(s)$ per $s = T_i$ abbiamo:

$$B_D = f_D(T_i) = \frac{2V}{L}$$

La banda Doppler, dunque, esprime anche il massimo scostamento Doppler rilevato dal radar, ed il suo reciproco ha il significato fisico di

“minor tempo rilevabile”, ovvero di una *risoluzione temporale*: scostamenti Doppler maggiori di $2V/L$ non sono rilevabili dal sistema, ovvero, tempi più piccoli di $1/B_D$ non sono registrabili durante il movimento della piattaforma. Allora, due oggetti che in azimuth si trovano ad una distanza tale che i loro scostamenti Doppler siano maggiori del valore $2 V/ L$ non sono “risolvibili” dal SAR.

A questo tempo limite corrisponde una “distanza limite” che è appunto la *risoluzione in azimuth*:

$$\rho_{AZ} = \frac{V}{B_D} = \frac{L}{2}$$

Potendo utilizzare un’antenna reale di lunghezza L piccola, è possibile ottenere, quindi, una risoluzione ρ_{az} lungo l’azimuth molto fine. In aggiunta essa è anche indipendente dalla quota del sensore. Questo risultato sembra costituire un paradosso fisico: infatti diminuendo la lunghezza dell’antenna reale il suo fascio si allarga quindi si allarga “l’impronta” a terra ma contemporaneamente la risoluzione migliora.

Questo apparente assurdo fisico può essere spiegato nel seguente modo: se l’antenna reale si accorcia aumenta la lunghezza dell’intervallo di traiettoria in cui si riesce ad osservare lo stesso diffusore, pertanto si allunga l’allineamento che può essere sintetizzato memorizzando e successivamente sommando l’eco.

2.5 Limitazioni sulla scelta della PRF

La cadenza di ripetizione degli impulsi deve essere scelta in modo da evitare problemi di ambiguità. Si ha ambiguità in distanza (nella direzione cross track) quando la durata dell'eco dovuto all'intera superficie illuminata è maggiore del periodo di ripetizione T degli impulsi.

Si ha invece ambiguità in azimuth quando la PRF è maggiore dell'intervallo di tempo che il sistema impiega per muoversi lungo una "cella" di risoluzione pari al $\rho_{az} = L/2$. Deve infatti essere possibile inviare almeno un impulso in ognuna delle n posizioni x_i .

Pertanto deve essere:

$$T \geq \frac{2 d_{MAX}}{c}$$

$$T \leq \frac{\rho_{AZ}}{V}$$

2.6 *Chirp Radar*

Al fine di conseguire una risoluzione dell'ordine del metro lungo la direzione cross track (in ground range) si ricorre ad una particolare forma d'onda (Chirp) per il segnale trasmesso.

Per conciliare infatti l'esigenza di elevate risoluzioni in distanza con l'impiego di impulsi "lungi" con limitata potenza di picco, si ricorre alla cosiddetta codifica di impulso. Tale tecnica consiste nell'introdurre una qualche forma di modulazione in un impulso di lunga durata, allargandone la banda. In questo modo è possibile distinguere due echi parzialmente sovrapposti grazie alla modulazione presente nell'impulso.

Infatti, se viene utilizzato un segnale linearmente modulato in frequenza, tramite una correlazione del segnale di ritorno dal bersaglio con una replica del segnale trasmesso memorizzato nel sistema, si ottiene una sinc che meglio approssima il Dirac rispetto alla finestra rettangolare. Questo tipo di segnale è il *chirp*, da cui nasce il concetto di *chirp radar*.

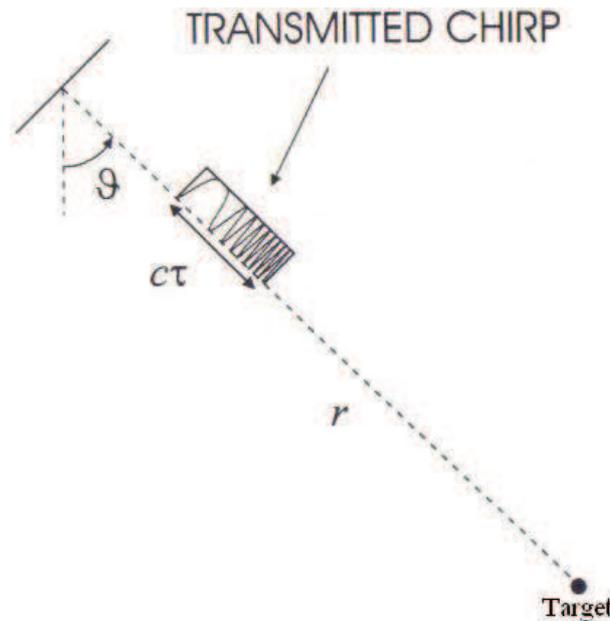


Figura 2.11 - Esempio di segnale chirp.

$$f(t) = \cos\left(\omega t + \frac{\alpha t^2}{2}\right) \text{rect}\left[\frac{t}{\tau}\right]$$

La risoluzione in tempo di un sistema del genere è in prima approssimazione pari a $T' = 1/B$, dove B è la banda del segnale chirp (che nel caso di un impulso non modulato di durata T , avendo questo una banda $1/T$, si riduce a $T' = T$), da cui si ricava che la risoluzione geometrica in range (cross track) di un chirp radar è:

$$\rho = \frac{c}{2B}$$

A titolo di esempio applicativo si riportano le specifiche di progetto del sistema radar ad apertura sintetica montato a bordo del satellite europeo ERS1/2 :

Il sistema opera in banda C, utilizza una portante a 5.3 GHz, trasmette e riceve in polarizzazione verticale, l'antenna è lunga 10 m in azimuth ed 1 m in elevazione. L'angolo di incidenza è $\theta = 23^\circ$. Il trasmettitore

trasmette un segnale impulsivo a larga banda di tipo chirp di durata $\tau = 37.1 \mu\text{s}$ e banda $B = 15.5 \text{ MHz}$. La cadenza di ripetizione è $T = 0.59 \text{ ms}$. Il segnale in ricezione è campionato con un periodo di campionamento di $T_c = 0.0527 \mu\text{s}$. Il sensore è posto su una piattaforma in orbita ad una altezza di 785 Km e la velocità con cui il satellite percorre la sua orbita è pari a $v = 7.5 \text{ Km/s}$.

Da questi dati si ricava quindi che la risoluzione in azimuth vale:

$$\rho_{AZ} = \frac{L_{AZ}}{2} = 5 \text{ m}$$

Mentre la risoluzione in ground range vale:

$$\rho_{GR} = \frac{c}{2B} = 9.68 \text{ m}$$

Si noti che qualora non si fosse utilizzato un chirp radar la risoluzione in ground range sarebbe stata di ben 14.2 Km .



Figura 2.12 - Immagine del satellite ERS1 con a bordo il sensore SAR descritto.

2.7 *Il SAR nelle applicazioni*

Come si è detto, le applicazioni del SAR sono innumerevoli: dal riconoscimento della presenza e delle caratteristiche della vegetazione allo studio delle onde oceaniche, a quello delle regioni vulcaniche, fino alla cartografia della superficie del pianeta Venere, coperta da una densa e opaca atmosfera. Lanciata nel 1989, la sonda Magellano venne inserita in un'orbita attorno a Venere che le consentiva di esaminare progressivamente tutta la superficie del pianeta. Nel corso di alcuni anni è stata così realizzata una mappatura della superficie di Venere con una ricchezza di dettagli molto superiore a quella raggiunta in precedenza: è stato possibile individuare caratteristiche superficiali con dimensioni minime dell'ordine di cento metri. Furono completati quattro cicli di mappatura, ognuno dei quali con una prospettiva leggermente diversa, il che ha permesso di realizzare un modello tridimensionale della superficie del pianeta.

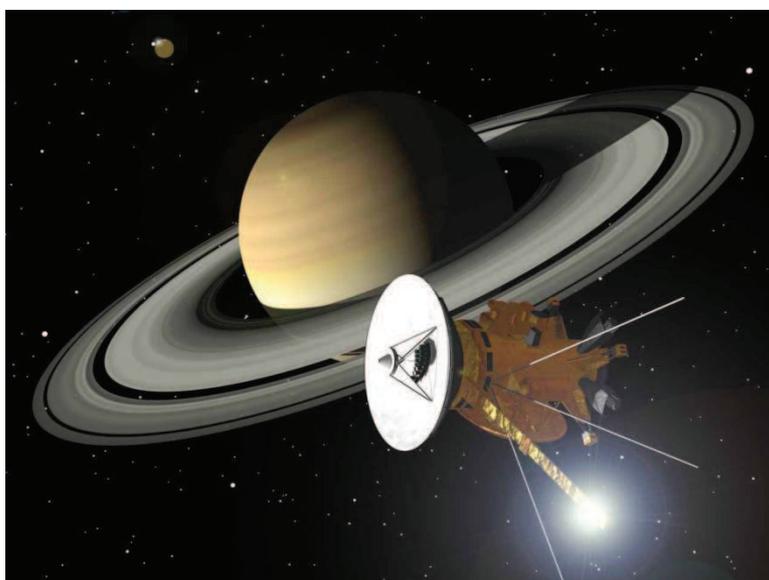


Figura 2.13 - Rappresentazione artistica della sonda Cassini.

L'impiego del SAR per sorvegliare i movimenti dei banchi di ghiaccio nelle regioni polari ha ampie applicazioni sia scientifiche che commerciali. Il ghiaccio polare è un mezzo disomogeneo, che contiene al suo interno tracce di sale, acqua salata e bolle d'aria. Le proprietà di riflessione delle onde elettromagnetiche possono cambiare notevolmente a seconda delle caratteristiche del ghiaccio, in particolare della sua età, e sono differenti da quelle dell'acqua. Dal punto di vista scientifico, la conoscenza dell'estensione del ghiaccio terrestre (la *criosfera*) è un'informazione essenziale per la climatologia: ad esempio, la crescita della criosfera è un meccanismo primario per la rimozione del biossido di carbonio dall'atmosfera. Le mappe che riportano la posizione e i movimenti dei banchi di ghiaccio sono naturalmente di grande interesse per l'industria della pesca, per il commercio marittimo e per altre attività che hanno luogo in mare alle alte latitudini. Ad esempio, l'industria petrolifera è interessata alle dimensioni e alla velocità degli iceberg in connessione all'installazione di impianti di perforazione temporanei.

Un'altra applicazione è la misura dell'umidità del suolo, grazie al fatto che la presenza d'acqua nel terreno influenza le proprietà di riflessione delle onde elettromagnetiche. Una stima accurata del contenuto d'acqua nei suoli è un parametro critico per lo studio del ciclo idrologico globale e consente di formulare modelli per comprendere il bilancio energetico del sistema climatico terrestre. Nel complesso, i dati forniti dal SAR avranno un ruolo primario nei prossimi anni per la determinazione dei fattori che influenzano le condizioni climatiche globali: ad esempio, ci consentiranno di seguire l'evoluzione dell'effetto serra atmosferico e dei processi di desertificazione che interessano alcune aree geografiche.

Negli ultimi anni il SAR ha trovato impiego anche nella sismologia. Si tratta della interferometria SAR differenziale (DInSAR), che consiste nel confrontare due immagini SAR della stessa area, acquisite in tempi diversi, cioè in due passaggi successivi del satellite sull'area considerata. Se tra i due passaggi si sono prodotti movimenti della superficie terrestre le differenze di percorso del segnale radar consentono di misurare lo spostamento di ogni elemento di superficie. La capacità di questa tecnica di misurare i movimenti del suolo con la precisione di pochi centimetri la rende uno strumento prezioso nello studio delle deformazioni della crosta terrestre connesse all'attività sismica. Ad esempio, l'analisi delle immagini SAR dell'Appennino umbro-marchigiano, riprese durante la sequenza sismica che ha colpito la regione nel 1997-98, sta consentendo di risalire alle caratteristiche di dettaglio delle sorgenti sismiche responsabili della sequenza.

3. SINTESI DI FREQUENZA

3.1 Introduzione alla sintesi di frequenza

La generazione di segnali periodici e' indispensabile per quasi tutte le applicazioni nel campo dell'elettronica, delle telecomunicazioni e dell'informatica. Per questo motivo lo studio e la progettazione di circuiti oscillatori e sintetizzatori di frequenza, risale agli albori dell'elettronica all'inizio del nostro secolo.

Il sintetizzatore è un apparato tale che se in ingresso gli si invia un segnale a frequenza fissa in uscita si possono prelevare segnali a frequenze diverse. In generale la gamma di frequenze prelevabili è molto ampia nei sintetizzatori per strumentazione da laboratorio, mentre è più ristretta in quelli per telecomunicazioni. In pratica, entrambi i tipi di sintetizzatori vengono progettati, prodotti e commercializzati completi di un oscillatore che diventa quindi parte integrante di essi.

I sintetizzatori da laboratorio quindi sono molto più complessi di quelli per telecomunicazioni dovendo coprire bande di frequenze molto vaste. In realtà sono costituiti da più sintetizzatori di tipi diversi che cooperano grazie ad una logica di controllo, che in funzione della banda e delle forme d'onda di esercizio commuta le circuiterie deputate a tale emissione.

Le forme d'onda generate sono rigorosamente sinusoidali nei sintetizzatori usati nelle telecomunicazioni; sono invece

sinusoidali, quadre, triangolari e impulsive nei sintetizzatori da laboratorio più vecchi o più economici; sono completamente determinabili dall'utente nei moderni sintetizzatori a sintesi digitale diretta.

Per confrontare i vari sintetizzatori sono stati definiti alcuni parametri standard che vengono utilizzati dalla maggior parte dei costruttori ed in alcuni casi sono misurati anche in maniera automatica.

Le caratteristiche tecniche sono ovviamente diverse da un sistema altro e mediante esse è possibile inquadrare il tipo di apparecchio:

- il campo di frequenze coperto;
- la risoluzione o passo del sintetizzatore (il minimo intervallo ottenibile tra due frequenze generate dallo stesso sintetizzatore; lo si può anche intendere come il numero di cifre utili per esprimere la frequenza in Hertz);
- la potenza di uscita che in molti casi è regolabile ed è espressa in decibel (si assume $0\text{dBm} = 1\text{mW}$);
- il campo delle potenze raggiungibili;
- la risoluzione nella regolazione di potenza anch'essa espressa in dB;
- l'impedenza nominale di uscita: essa è 50 o 75 W in ambito radio e telecomunicazioni, e 600 W nelle applicazioni audio.

In tutti gli apparecchi è richiesto l'uso dell'impedenza di carico per un ottimo adattamento nel trasferimento del segnale.

È molto importante notare che tutte le frequenze in uscita, qualunque sia il loro valore, hanno un'incertezza dipendente da quella dell'oscillatore fondamentale. Quindi la qualità di un segnale, prodotto da

un sintetizzatore, è sostanzialmente dipendente dalla qualità dell'oscillatore fondamentale.

I parametri di misura sono molteplici; alcuni di essi sono standard e semplici da misurare come ad esempio l'assorbimento di corrente, la potenza d'uscita, la stabilità della potenza d'uscita. Altri sono tipici proprio della sintesi di frequenza e sono utili per evidenziare i limiti di ogni singola strumentazione; sebbene non risultino facili da misurare.

Si illustrano qui di seguito i parametri secondo la notazione adoperata dai costruttori ed illustriamo le tecniche di misura usate per esse.

- ***piattezza dell'uscita***: indica al variare della frequenza quanto si discosta in ampiezza il segnale in uscita dalla potenza impostata.
- ***velocità di commutazione***: indica la velocità con cui il sistema passa da una frequenza ad un'altra, compreso l'eventuale tempo di stabilizzazione. Per misurare tale caratteristica è necessario valutare la stabilità della frequenza di partenza e della frequenza di arrivo poi si misura il tempo impiegato per la commutazione. Si ritiene un segnale in condizione di stabilità quando la variazione di fase è inferiore a 0,1 radianti. In molti casi questo parametro dipende dalla distanza relativa tra le frequenze.
- ***Armoniche***: indica il rapporto in decibel tra il segnale generato e le armoniche a frequenze superiori; questo parametro è significativo solo in modalità sinusoidale.

Attualmente le tecniche fondamentali di sintesi sono tre:

- Sintesi diretta analogica

- Sintesi indiretta
- Sintesi digitale diretta

I sintetizzatori più diffusi e più economici sono quelli presenti negli apparecchi per telecomunicazioni e sono di tipo indiretto con anello ad aggancio di fase. In particolare, quelli per ricevitori radio o televisivi generano le frequenze dell' oscillatore locale, che distano dalla frequenza ricevuta di una quantità ben precisa che è la frequenza di conversione intermedia. Quelli adoperati nei ricetrasmittitori generano contemporaneamente la frequenza dell'oscillatore locale e la frequenza base di trasmissione. Lo stesso vale per i sintetizzatori per telefonia cellulare e domestica senza filo. In tutti i sistemi FM trasmettenti la modulazione è un ulteriore compito affidato al sintetizzatore.

I sintetizzatori moderni per misura di laboratorio adoperano prevalentemente la sintesi digitale diretta e la sintesi ad anello ad aggancio di fase nelle gamme più alte. La sintesi diretta analogica è ormai sempre meno diffusa perché costosa ed ingombrante, tuttavia la sua conoscenza risulta una esperienza di base utile per una comprensione approfondita delle tecniche più attuali.

3.2 Sintesi diretta analogica

Il sintetizzatore diretto è il primo tipo di sintetizzatore che è stato inventato e può essere considerato come una immediata evoluzione di un banco di oscillatori al quarzo quale ad esempio quello contenuto nel commutatore di frequenza di un ricetrasmittitore a più canali. In questo apparecchio infatti inizialmente per ogni canale di trasmissione venivano adoperati due cristalli di quarzo le cui differenze in termini di frequenza di oscillazione equivalevano al valore di media frequenza del primo stadio di conversione della supereterodina ricevente.

In tutti i circuiti dove si fanno oscillare cristalli di quarzo il prelievo delle armoniche superiori è una variante di semplice realizzazione, e generare prodotti di intermodulazione, tra frequenze diverse è altrettanto facile. Per questo motivo, già da molto tempo, prima che i transistor avessero soppiantato i tubi a vuoto, i progettisti elettronici delle case produttrici leader nel settore della sintesi di frequenza si sono impegnati ad inventare svariate configurazioni circuitali che consentano di generare un certo numero di frequenze richieste con il minimo numero possibile di oscillatori al quarzo. Ad esempio uno dei più semplici, dei tanti artifici possibili, è quello di scegliere come frequenza di media conversione un valore multiplo intero del passo di canalizzazione, ovvero della differenza minima intercorrente fra due frequenze generabili. In tal modo lo stesso oscillatore viene adoperato in trasmissione ed in ricezione, ottenendo già un fattore di risparmio sugli oscillatori quasi del cinquanta per cento.

È facilmente immaginabile che in quell'epoca in cui gli interruttori erano soltanto meccanici si sono costruiti sintetizzatori dove la

complessità ed il costo dei commutatori potevano influenzare notevolmente il costo dell'apparecchio.

Nei sistemi a sintesi analogica diretta, le diverse frequenze vengono ottenute mediante operazioni elementari sulle frequenze:

- somma (e differenza) tra frequenze: si ottengono mediante un *mixer* e opportuno filtraggio dell'uscita; come è noto, infatti, dal prodotto di due segnali sinusoidali si ottengono la frequenza somma e la frequenza differenza:

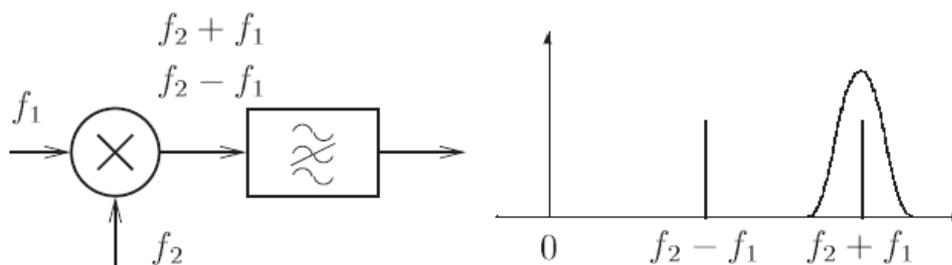


Figura 3.1 - Somma (o differenza) di frequenze.

Analiticamente:

$$V_1 \sin(2\pi f_1 t) \cdot V_2 \sin(2\pi f_2 t) = \frac{V_1 V_2}{2} \{ \cos[2\pi(f_1 - f_2)] + \cos[2\pi(f_1 + f_2)] \}$$

dunque operando, dopo il *mixer*, con un filtro opportunamente selettivo è possibile selezionare la componente di interesse.

- moltiplicazione di frequenza per un numero intero: si ottiene utilizzando una opportuna rete non lineare che dunque, pilotata da un segnale monocromatico, produce una molteplicità di armoniche e selezionando con un filtro l'armonica di interesse.
- divisione di frequenza per un numero intero ($f_{out} = \frac{f_{in}}{N}$): si ottiene impiegando contatori come divisori di frequenza. Pilotando infatti un contatore modulo N con un'onda quadra di

frequenza f_1 , è possibile ricavare in uscita un segnale (a onda quadra) di frequenza $\frac{f_1}{N}$. Se si ha interesse a che questo segnale abbia andamento sinusoidale, basta effettuare su di esso un filtraggio passa-basso per eliminare il contenuto armonico eccedente la frequenza fondamentale.

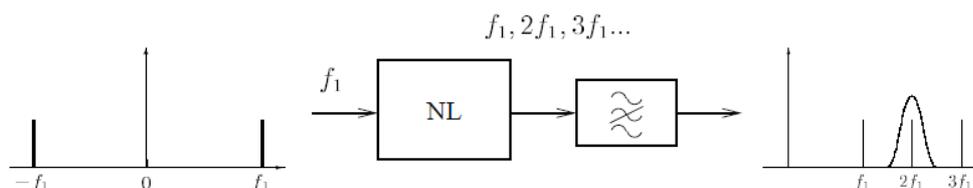


Figura 3.2 - Generazione di armoniche e selezione dell'armonica di interesse.

- combinazioni lineari tra frequenze: si ottengono combinando opportunamente i sistemi che effettuano somme/differenze tra frequenze e moltiplicazioni di frequenze per numeri interi.

Sebbene concettualmente semplici, queste tecniche richiedono l'utilizzo di filtri opportunamente selettivi allo scopo di selezionare la riga alla quale di volta in volta si è interessati, attenuando adeguatamente le altre (prodotte dal *mixer* o dalla medesima operazione di generazione di armoniche).

Una possibile realizzazione di sintetizzatore analogico di frequenza è schematizzato in Figura 3.4(a). Inviando al *mixer* la frequenza di riferimento f_r e una delle armoniche della riga (1 MHz) ottenuta per divisione di f_r è possibile ottenere in uscita le frequenze 11 MHz, 12 MHz, 13 MHz, ecc.

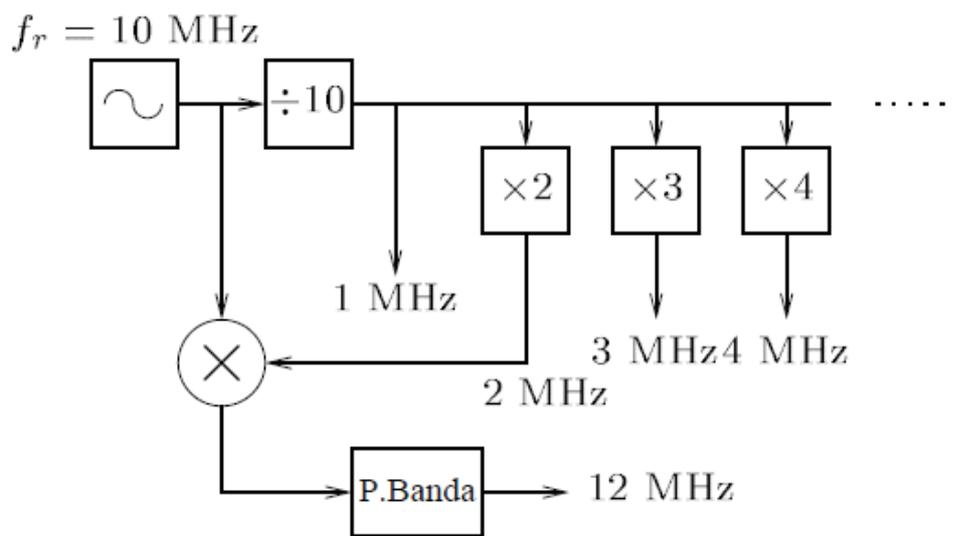


Figura 3.3 - Schematizzazione analogico di frequenza.

Il limite superiore è determinato dalle caratteristiche del sistema non lineare utilizzato per la generazione di armoniche (di 1 MHz): infatti l'ampiezza delle righe in uscita da questo decade all'aumentare della frequenza (e cioè dell'ordine di armonica) . Oltre un certo limite l'ampiezza della componente spettrale si riduce a valori tali da diventare difficilmente distinguibile da rumore e disturbi ed essere dunque inutilizzabile.

La tecnica descritta può essere impiegata per realizzare sintetizzatori di frequenza con struttura modulare, basati sulla cosiddetta "generazione di gruppi spettrali"; un esempio di tale sistema è rappresentato schematicamente in figura 3.4(a).

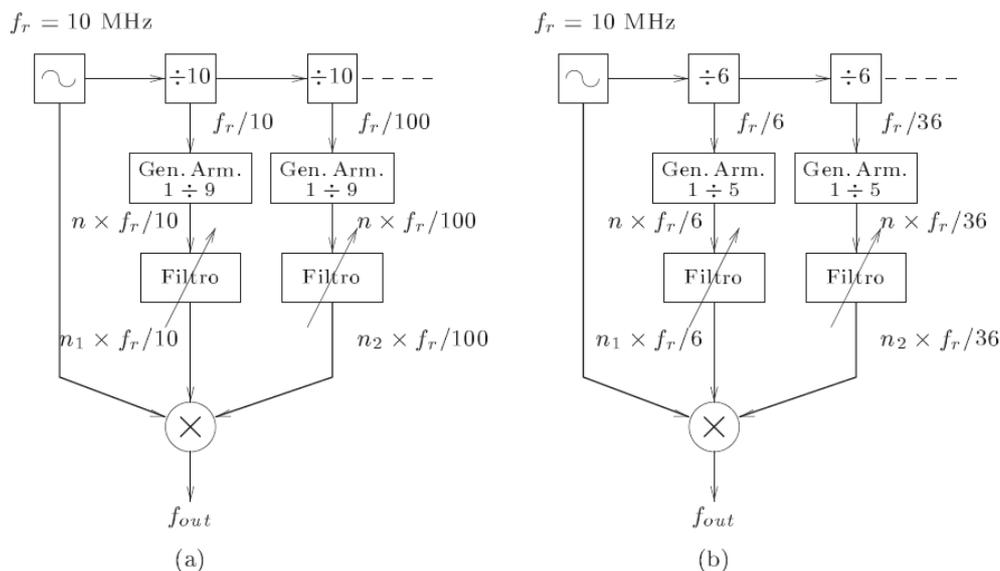


Figura 3.4 - Schematizzazione di sintetizzatori analogici di frequenza basati sulla generazione di gruppi spettrali

Nello schema ogni generatore di armoniche elabora la riga prodotta dopo un certo numero di divisioni successive a partire dalla frequenza di riferimento; dopo ogni generatore di armoniche, il filtro passa-banda seleziona quella di interesse che, inviata al *mixer* insieme con le righe provenienti dagli altri generatori di armoniche concorre a determinare la frequenza d'uscita. L'espressione "generatore di gruppi spettrali" si riferisce a ogni catena costituita da un divisore, generatore di armoniche e filtro passa-banda.

La frequenza prodotta ha la seguente espressione:

$$f_0 = f_r + n_1 \frac{f_r}{10} + n_2 \frac{f_r}{20}$$

L'utente, impostando il valore della frequenza di interesse, definisce i coefficienti n_1 e n_2 mediante i quali vengono selezionate le armoniche necessarie. così, per esempio, con $n_1 = 3$ ed $n_2 = 6$ si ottiene $f_0 = 13.6MHz$. Se tutti i generatori di armoniche sono in grado di produrre fino all'armonica di ordine 9 della riga di ingresso, allora il

sintetizzatore rappresentato in figura permette di ottenere frequenze da 100 kHz a 19.9 MHz, con la risoluzione di 100 kHz.

Il sistema ha evidentemente una struttura modulare, in quanto ogni generatore di gruppi spettrali è *concettualmente* identico agli altri (ma ovviamente lavora a frequenze diverse). L'aggiunta di ulteriori generatori gruppi spettrali permetterebbe una risoluzione in frequenza più spinta.

Lo spettro riportato in Figura 3.4(b) utilizza divisori di ordine inferiore ($\div 6$) e generatori a cui è richiesto di produrre fino alla quinta armonica della riga in ingresso; è facile verificare che, per ottenere la stessa risoluzione del sistema in Figura 3.4(a) e coprire lo stesso intervallo di frequenze è necessario un maggior numero di gruppi spettrali. La semplificazione dei generatori di armoniche comporta dunque un aumento della circuiteria necessaria.

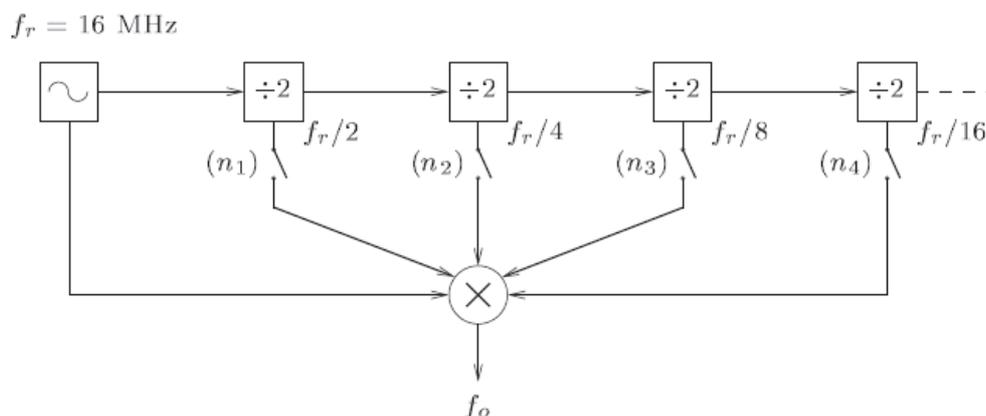


Figura 3.5 - Altro esempio di sintetizzatore analogico di frequenze.

In figura 3.5 è rappresentato schematicamente un analogo sintetizzatore che impiega solo divisori per 2 dunque non richiede generatori di armoniche. L'espressione della frequenza disponibile in uscita è del tipo:

$$f_0 = f_r + n_1 \frac{f_r}{2} + n_2 \frac{f_r}{4} + n_3 \frac{f_r}{8} + n_4 \frac{f_r}{16} + \dots$$

I cui coefficienti n_1, n_2, \dots assumono valore 0 se il corrispondente tasto è aperto e valore 1 in caso contrario.

3.3 Sintesi indiretta

Un'alternativa ai sintetizzatori basati sul processo di sintesi diretta sono i sintetizzatori indiretti, basati sull'uso dei PLL, ossia sugli anelli ad aggancio di fase (Phase-Lock Loop).

Osserviamo lo schema di funzionamento di un PLL, mostrato in figura 3.6.

I principali blocchi che lo compongono sono: il *Phase Detector*, indicato in figura con PD, il *Filtro d'Anello*, indicato in figura con LF dall'inglese *Loop Filter*, e l'*oscillatore controllato in tensione*, comunemente chiamato VCO dall'inglese *Voltage Controlled Oscillator*.

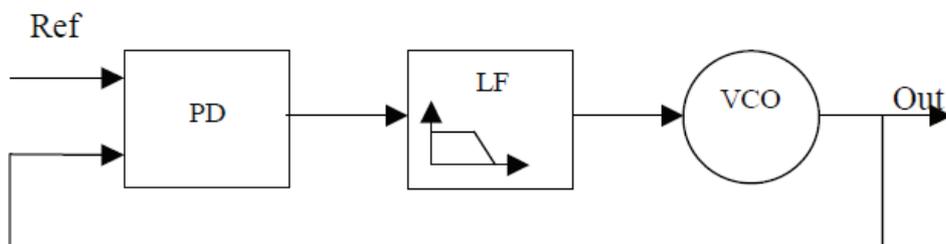


Figura 3.6 - PLL di "tipo I"

Il Phase Detector effettua un confronto tra la fase di ingresso e quella di uscita e genera in corrispondenza un segnale di tensione il cui valor medio è proporzionale allo sfasamento tra esse. Il segnale di uscita dal PD è un'onda quadra con duty cycle variabile. Il filtro d'anello ne fa passare solo la componente continua.

Il VCO è un oscillatore controllato in tensione, che genera un'oscillazione ad una frequenza proporzionale al valore della tensione

al suo ingresso. Dato che il sistema è retro-azionato negativamente, esso tende a stabilizzarsi in modo che la frequenza in uscita sia uguale a quella in ingresso. Occorre osservare che tra i due segnali vi sarà in genere una differenza di fase statica.

Ci sono varie uscite del PLL che possono essere utilizzabili secondo l'applicazione: nella demodulazione FM l'uscita del filtro fornisce il segnale modulante; in applicazioni di *timing recovery* l'uscita del VCO rappresenta proprio il segnale di clock rigenerato.

E' inoltre possibile utilizzare il PLL come sintetizzatore di frequenza nella generazione di un riferimento di frequenza.

Quando l'uscita del PLL oscilla esattamente alla stessa frequenza del riferimento, si afferma che il PLL è in condizione di *lock*, ossia che è “agganciato” al segnale di riferimento. Il PLL si aggancia, dunque, quando $f_{OUT} = f_{REF}$. In questo stato in genere vi è una differenza di fase $\Delta\phi$ *stazionaria* tra riferimento e uscita, corrispondente ad un certo errore di fase costante nel tempo. Viceversa, quando $\Delta\phi$ varia nel tempo, ci si riconduce ad una condizione indesiderata nella quale l'uscita del PLL non riesce ad agganciarsi all'ingresso. Questa situazione si definisce di “unlock”.

Lo schema illustrato in figura 3.6 rappresenta un caso classico di PLL, che in letteratura viene indicato come PLL di “tipo I”. Spesso nelle applicazioni si richiede di poter sintetizzare una frequenza *diversa* da quella del riferimento, in genere si richiede un fattore pari ad alcune centinaia o anche un migliaio tra le due frequenze, ossia $\omega_{out} = 1000\omega_{REF}$. In questo caso si ricorre ad un tipo diverso di PLL, indicato in letteratura come di “tipo II”, o *sintetizzatore di frequenza*. La figura 3.8 mostra questa struttura.

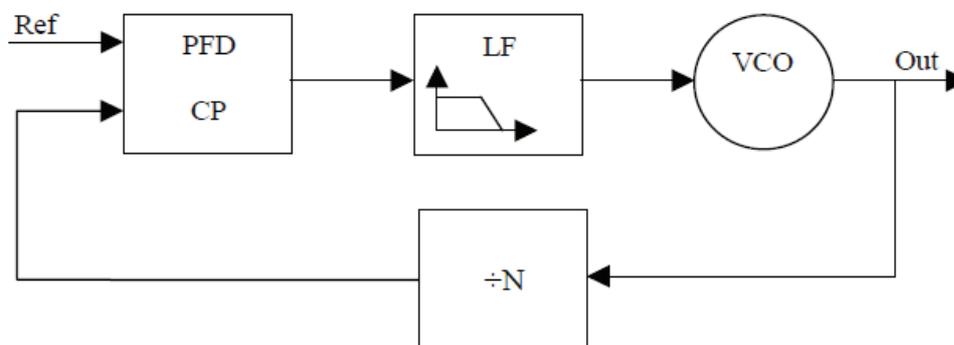


Figura 3.8 - Il PLL di "tipo II"

Rispetto al PLL di “tipo I”, la struttura viene modificata con l’aggiunta di un divisore di frequenza nel ramo di retroazione, indicato in figura come $\div N$. Indicando con N il fattore di divisione, per questo tipo di PLL vale la relazione:

$$F_{OUT} = N F_{REF}$$

Inoltre si utilizza al posto del Phase Detector una struttura costituita dal *Frequency Phase Detector*, indicato in figura con PFD, e dalla *Charge Pump*, indicata con CP.

Analizziamo in dettaglio il funzionamento di ogni singolo blocco.

3.3.1 DIVISORE

Generalmente il divisore viene reso programmabile, in modo da poter sintetizzare più frequenze, tutte multiple di quella di ingresso. In tal caso si ha che:

$$F_{OUT} = (N_{MIN} + k) \cdot F_{REF}$$

con $k = 0 \dots k_{MAX}$ ed N_{MIN} e $N_{MAX} = N_{MIN} + k_{MAX}$ sono gli estremi del rapporto di divisione. Un divisore programmabile, come si può vedere in figura 3.9, è composto di un Prescaler e da due contatori.

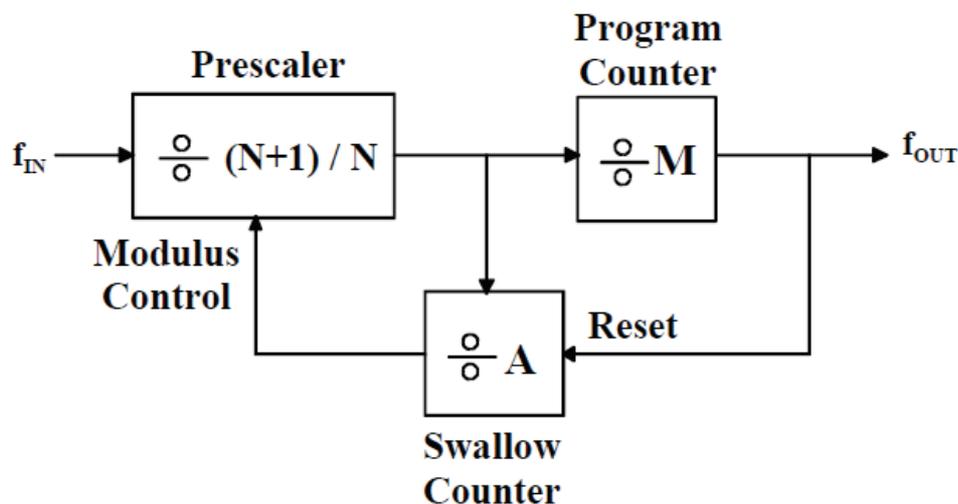


Figura 3.9 - Architettura classica del divisore di frequenza.

Il Prescaler è un divisore che divide per un numero N o per lo stesso numero aumentato di un'unità. Uno dei due contatori, detto Program Counter, conta un numero M di periodi d'uscita del Prescaler, dopodichè dà il comando di azzeramento all'altro contatore che è detto Swallow Counter. In Figura 2.10 si possono osservare gli andamenti dei segnali alle uscite del Prescaler e dei due contatori.

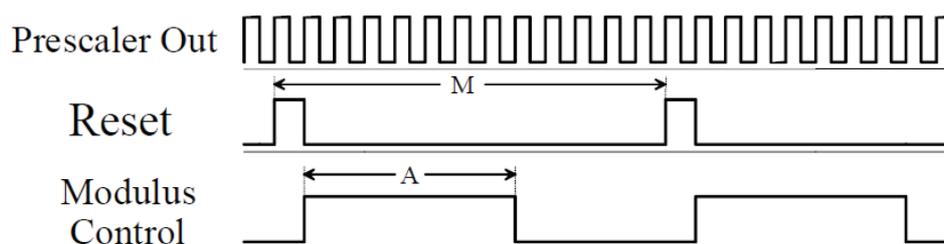


Figura 3.10 - Andamento di Reset e Modulus Control con $M=13$ e $A=7$

Lo Swallow Counter conta un numero A (minore-uguale a M) degli stessi periodi, e, durante questo numero di periodi, fornisce al Prescaler il segnale di controllo, detto Modulus Control, affinché questo divida per $N+1$. Durante gli altri $M-A$ periodi, il Prescaler divide per N . Quindi, se si esprime l'uscita del Program Counter come multiplo del periodo d'ingresso, si può ricavare il fattore di

moltiplicazione del periodo. Si osserva che il Prescaler divide il periodo d'ingresso per $N+1$ per A volte e per N per le restanti $M-A$ volte, in totale, l'uscita del Program Counter fornisce un comando di azzeramento dopo $A(N+1)+(M-A)N$ periodi d'ingresso. Semplificando possiamo ottenere il fattore di divisione della frequenza, cioè l'inverso del fattore di moltiplicazione del periodo, che vale $MN+A$. Questa formula è alla base del dimensionamento di tutti i divisori programmabili. Esiste anche la possibilità di sintetizzare frequenze scalate di un fattore non intero rispetto al riferimento, attraverso l'implementazione di divisori frazionari programmabili; questa architettura deriva dalla necessità di svincolare la spaziatura tra i canali dal valore della frequenza del riferimento. Infatti, fissata la banda del PLL, per considerazioni che verranno chiarite in seguito, si impone un vincolo inferiore alla frequenza del riferimento; dunque risulta chiaro come sia necessaria un'architettura frazionaria lì dove il valore di f_{REF} non può essere diminuito al di sotto della spaziatura tra canali desiderata.

3.3.2 Frequency Phase Detector

Nei PLL che utilizzano divisori interi l'anello reagisce a seguito di una variazione del rapporto di divisione variando la frequenza di uscita; lo studio della dinamica della transizione tra due frequenze è molto importante, in quanto spesso si richiede un tempo massimo di acquisizione entro il quale il dispositivo funziona correttamente. La situazione peggiore da questo punto di vista è chiaramente quando N varia da N_{MIN} a N_{MAX} . In questo caso, non appena il divisore varia N ,

al Phase Detector arrivano due segnali con frequenze tra loro scalate tra loro di un fattore N_{MIN}/N_{MAX} . Con l'uso di un Phase Detector non è assicurato che il PLL si agganci in questa condizione. A questo si aggiunga la indeterminazione del duty-cycle del segnale dopo il divisore per comprendere come un circuito combinatorio come il PD si riveli inefficace in architetture di PLL di "tipo II".

In queste strutture non viene più effettuato esclusivamente il confronto di fase tra il segnale di riferimento e quello all'uscita del divisore, ma, attraverso un comparatore di fase-frequenza, cioè il PFD dall'inglese Phase Frequency Comparator, si effettua dapprima una comparazione di frequenza e poi una di fase, permettendo di raggiungere l'aggancio in un minor tempo e prescindendo completamente dal duty-cycle dei segnali in ingresso. Si noti che non si tratta più di un circuito combinatorio ma di uno sequenziale. Il PFD può essere implementato ricorrendo a flip flop di tipo D azzerabili asincronamente mediante una AND, come mostrato in figura 3.10.

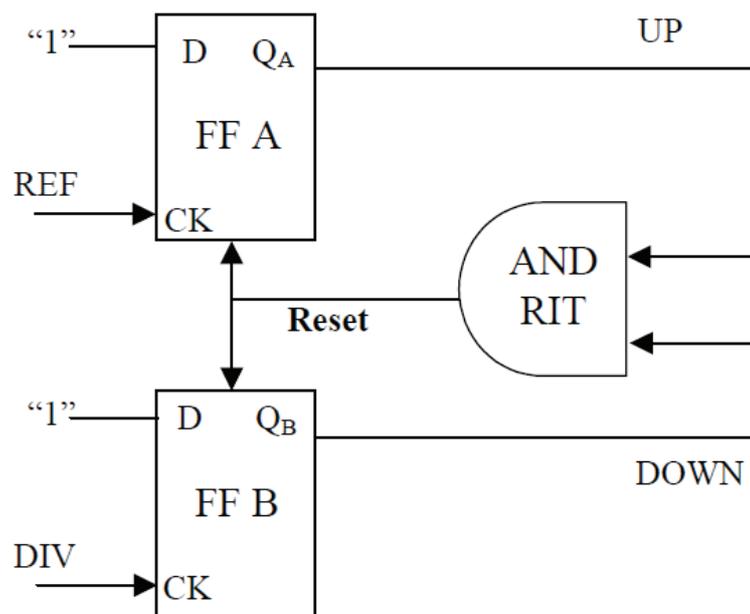


Figura 3.11 - Frequency phase detector

I due flip flop hanno gli ingressi del dato entrambi ad “uno” logico, mentre il clock è rappresentato dal riferimento per il flip flop superiore e dal segnale proveniente dal divisore quello inferiore. In questo modo i flip flop portano in uscita un “uno” logico solo quando leggono il fronte di salita del proprio segnale di clock, e lo mantengono anche dopo che il segnale ritorna basso. Infatti la struttura si resetta solo quando la AND si chiude, ossia quando sono arrivati entrambi i fronti, quello del riferimento e quello del divisore. Risulta chiaro che questo funzionamento è indipendente dai livelli, e quindi dal duty-cycle dei segnali, e risulta sensibile soltanto ai fronti di salita. Il Phase Frequency Detector ha due uscite, UP e DOWN, le quali controllano la carica o la scarica del filtro d'anello.

3.3.3 Charge Pump

Per interfacciarsi al filtro d'anello è opportuno utilizzare un blocco che converta i segnali di tensione provenienti dal Phase Frequency Detector in segnali di corrente; questa funzione viene svolta da un circuito chiamato Pompa di Carica o Charge Pump.

La Charge Pump è un generatore di corrente *three-state*: essa fornisce al filtro d'anello la corrente $\pm I_p$ oppure rimane ad alta impedenza. Quando il Phase Frequency Detector rileva che la frequenza di uscita è troppo bassa rispetto al riferimento esso genera degli impulsi di “UP” di durata Δt i quali comandano alla Charge Pump di iniettare corrente $+I_p$ nel filtro d'anello. La tensione su di esso aumenta e con essa la frequenza del VCO. Dopo un certo numero di cicli del reference il PLL si aggancia e il filtro d'anello rimane carico mantenendo ai suoi capi un valore costante di tensione. La funzione della Charge Pump è di trasferire sul filtro d'anello la carica

$\Delta Q = I_p \cdot \Delta t$. Nella condizione di aggancio la Charge Pump rimane nello stato ad alta impedenza, impedendo un cammino di scarica al filtro. Il PLL è un sistema tempo-discreto se si pensa soltanto alle commutazioni del PFD, dunque si può studiarne il comportamento dinamico attraverso l'uso, ad esempio, della trasformata Z . Comunque nell'ipotesi che lo stato del PLL vari molto lentamente rispetto al periodo del reference, il loop può essere studiato come se fosse un sistema lineare tempo-continuo. Quest'ipotesi risulta vera se la banda del loop è molto minore della frequenza di ingresso. In pratica si dimensiona la banda in modo da essere almeno 1/10 della frequenza di ingresso. Si capisce ora come sia importante l'uso di architetture frazionarie per applicazioni veloci in cui interessa mantenere bassa la spaziatura tra i canali.

4. SINTESI DIRETTA DIGITALE

4.1 Introduzione alla sintesi diretta digitale

La sintesi digitale di frequenza *Direct digital synthesis* (DDS) è una tecnica che utilizza dei blocchi di digital data processing per generare un segnale di uscita sintonizzabile in frequenza e fase a partire da una sorgente di clock a frequenza fissa. In pratica la frequenza di clock di riferimento viene divisa dall'architettura del DDS per un fattore di scala deciso attraverso la definizione di una parola binaria detta FTW (Frequency Tuning Word). La parola FTW è tipicamente formata da un numero di bit che varia da 24 a 48 e la sua lunghezza fornisce la risoluzione per la sintonizzazione della frequenza di uscita all'architettura DDS realizzata.

I prodotti DDS sono diventati oramai da un decennio una alternativa alle soluzioni di sintesi analogica di frequenza per il loro ingombro minimo e la possibilità di integrazione, le alte prestazioni ed i costi molto competitivi. La possibilità di integrare su un singolo chip convertitori digitale-analogico molto veloci e architetture DDS formando quello che viene chiamato *Complete-DDS solution*, permette a questa tecnologia di puntare ad un largo gruppo di applicazioni e fornisce, in molti casi, un'alternativa attraente ai sintetizzatori analogici basati su PLL. Per molte applicazioni, la soluzione DDS presenta molti vantaggi rispetto ad un sintetizzatore che impiega un circuito PLL:

- permette di ottenere una risoluzione in frequenza dell'ordine del milli-Hertz e una risoluzione al di sotto del grado dell'accuratezza della fase, con un completo controllo digitale.

- un'elevatissima velocità nella variazione della frequenza di uscita in quanto il sistema non richiede nessun tempo di assestamento nel cambiare la frequenza, infatti si va sull'ordine dei nanosecondi o poche decine di microsecondi.
- Tutti i cambiamenti di frequenza sono completati automaticamente con una variazione continua della fase, in questo modo la nuova frequenza generata riprende esattamente dall'ultimo valore di fase della frequenza precedente.
- elimina la necessità di una sintonizzazione manuale del sistema e di aggiustamenti dovuti all'età del componente e alle variazioni di temperatura come succede nelle soluzioni di sintesi analogica.
- l'interfaccia digitale di controllo dell'architettura DDS facilita un ambiente in cui i sistemi sono controllati da remoto e ottimizzati sotto il controllo di un processore.

La maggior parte delle circuiterie, necessarie per realizzare un sintetizzatore diretto digitale, è di tipo digitale e ciò ha innumerevoli benefici, come l'abbattimento dei costi, la ripetibilità, l'immunità al rumore, la realizzazione mediante macchine assemblatrici automatiche, la simulazione più semplice.

L'unica parte critica che spesso è l'elemento limitante per le prestazioni del sintetizzatore è il convertitore digitale analogico di uscita. È intuitivo che qualora si realizzasse un sintetizzatore totalmente digitale, le prestazioni sarebbero estremamente interessanti perché totalmente predicibili in quanto non affette dai tipici errori dei

circuiti analogici: scarsa precisione, dinamica limitata, non linearità e prodotti di intermodulazione.

4.2 Architettura di base

Nella sua forma più semplice un sintetizzatore digitale diretto può essere implementato da un generatore di clock di riferimento, un contatore modulo N, una memoria PROM (*programmable read only memory*), un convertitore D/A come si può vedere dalla figura 3.1:

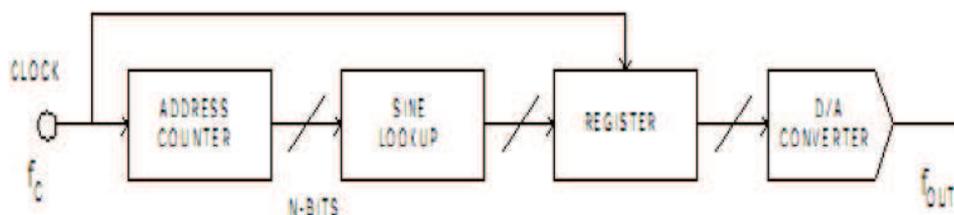


Figura 4.1 -Schema semplificato di un sintetizzatore di frequenza DDS

La frequenza di uscita di questo tipo di implementazione dipende da:

- la frequenza della sorgente di clock di riferimento.
- il passo di campionamento della forma d'onda sinusoidale memorizzata nella PROM.

Ma come si può notare, mentre la qualità del segnale di uscita, il jitter, le prestazioni in AC di questa semplice implementazione possono essere alquanto buone, questa manca di flessibilità. La frequenza di uscita può essere cambiata solo cambiando la frequenza del clock di riferimento o riprogrammando la memoria PROM ma nessuna di queste opzioni supporta salti in frequenza di uscita ad alta velocità. Con l'introduzione dell'accumulatore di fase nel percorso del segnale, l'architettura diventa un oscillatore controllato numericamente che è il

componente principale di una architettura DDS molto flessibile. Come mostrato in figura 4.2 prima della Sine Look-Up Table vengono inseriti un contatore modulo-N e un registro di fase per rimpiazzare il registro degli indirizzi.

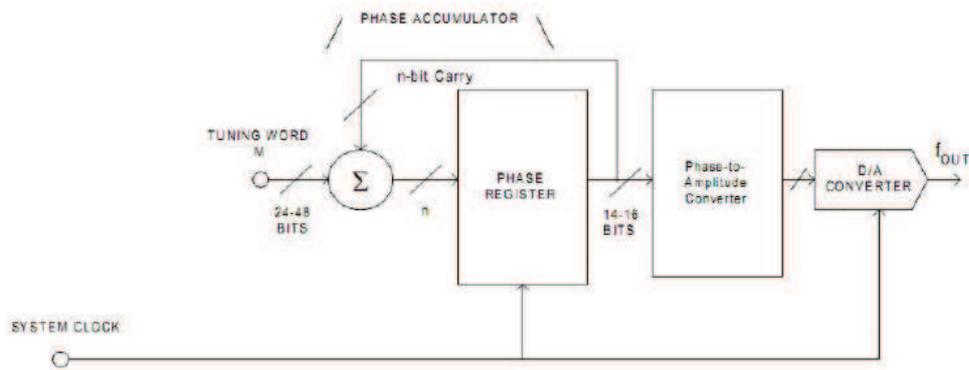


Figura 4.2- Schema di un DDS sintonizzabile in frequenza

Per comprendere la funzione base di questa struttura, visualizziamo l'onda sinusoidale come un vettore rotante sul cerchio delle fasi(come mostrato in figura 4.3).

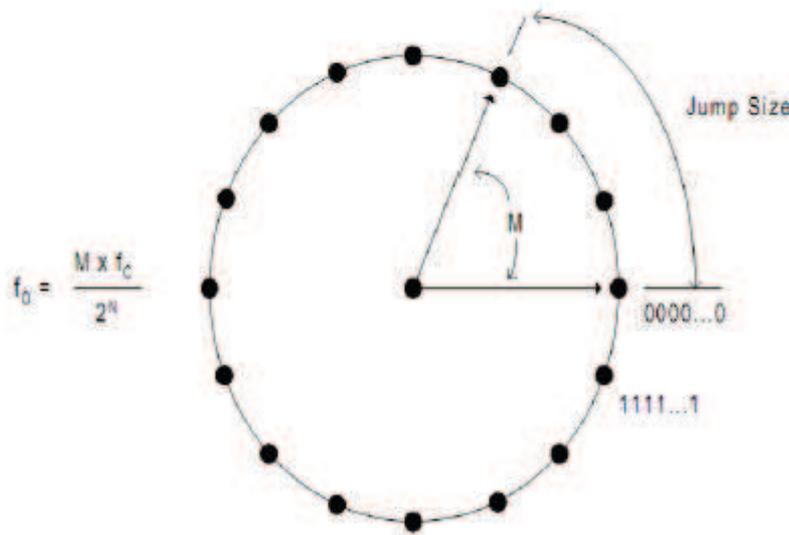


Figura 4.3- Ruota delle fasi

Ogni punto definito sul cerchio delle fasi corrisponde ad un punto esatto del ciclo di un'onda sinusoidale quindi il vettore ruotando a velocità costante descrive in una rivoluzione un periodo completo della sinusoide d'uscita. L'accumulatore di fase viene utilizzato per fornire il valore equivalente della rotazione lineare del vettore sulla ruota delle fasi. I valori contenuti nell'accumulatore di fase corrispondono ai punti di un periodo della sinusoide in uscita. Il numero discreto dei valori di fase del cerchio delle fasi è determinato dalla risoluzione, ovvero il numero di bit N , dell'accumulatore di fase. L'uscita dell'accumulatore di fase però è lineare e non può essere direttamente utilizzato per generare una sinusoide o qualsiasi altra forma d'onda eccetto una rampa. Perciò è necessario utilizzare una *phase-to-amplitude lookup table* per convertire una versione troncata del valore istantaneo di uscita dell'accumulatore di fase in un valore di ampiezza che andrà poi all'ingresso del convertitore D/A .

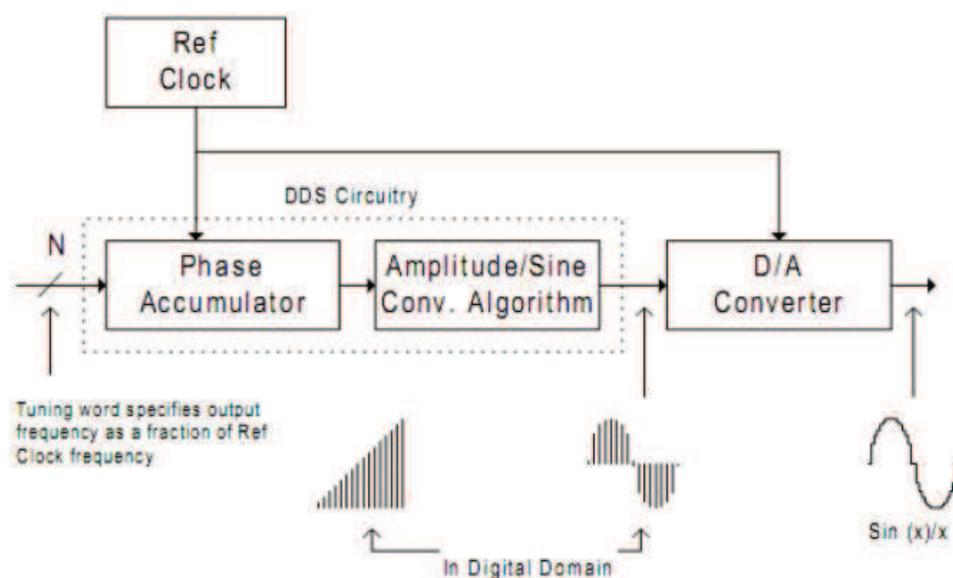


Figura 4.4- Segnale in ogni stadio della sintesi

Molte architetture DDS sfruttano la natura simmetrica della forma d'onda sinusoidale e utilizzano una logica di mappamento per

sintetizzare un periodo completo della sinusoide a partire da 1/4 di periodo dai valori in uscita dall'accumulatore. Attualmente l'accumulatore di fase è un contatore modulo-M che incrementa il valore in esso contenuto ogni volta che riceve un impulso di clock. L'ampiezza dell'incremento è determinata da una parola digitale lunga M bit contenuta in un *delta phase register* che viene sommata con l'*overflow* del contatore. La parola contenuta nel registro *delta-phase* rappresenta lo *step size* di fase tra successivi colpi di clock; in pratica definisce quanti punti si saltano sul cerchio delle fasi. Più grande è il salto più velocemente l'accumulatore di fase va in overflow e completa il periodo equivalente della sinusoide. Per un numero di bit N dell'accumulatore di fase, un numero M in rappresentazione binaria 000...0001 (uno) darà un overflow dell'accumulatore dopo 2^{32} colpi di clock (incrementi di valore). Se il valore di M cambia in 0111...1111, l'accumulatore di fase andrà in overflow dopo soli 2^1 colpi di clock. Questo controllo sul *jump size* costituisce la risoluzione di sintonizzazione della frequenza di uscita dell'architettura DDS. La relazione che lega il numero di bit e il delta phase dell'accumulatore di fase è detta *equazione base di sintonizzazione* dell'architettura DDS:

$$f_{out} = \frac{M * f_{clk}}{N}$$

con:

- f_{out} frequenza di uscita del DDS
- M valore della parola di sintonizzazione binaria
- f_{clk} frequenza di clock di riferimento (del sistema)
- N numero di bit dell'accumulatore di fase

Un cambiamento del valore di M nell'architettura del DDS provoca un immediato cambiamento senza salti di fase della frequenza di uscita. Nelle applicazioni pratiche, il valore di M , o *frequency tuning word* FTW , viene caricato in un registro seriale interno che precede il registro *delta-phase* con uscita parallela. Questo viene generalmente fatto per ridurre il numero di pin del dispositivo DDS. Una volta che il buffer del registro viene caricato, l'uscita parallela del registro *delta-phase* è temporizzato e la frequenza di uscita del DDS cambia. Generalmente i limiti sulla velocità di variazione della frequenza di uscita del DDS vengono imposti dal tempo che occorre per caricare il buffer del registro e perché il cambiamento abbia effetto. Naturalmente un ingresso parallelo per la FTW migliora la capacità di variare la frequenza in uscita velocemente. Il vantaggio della tecnologia DDS è insito nella sua natura. Proprio perché è un dispositivo digitale è possibile applicarvi ulteriori blocchi completamente digitali consentendo la generazione di forme d'onda differenti da quella sinusoidale come comparatori (per la generazione di un'onde quadra), integratori e derivatori.

4.3 Studio dello spettro del segnale di uscita

Per capire come è fatto lo spettro del segnale digitale di uscita, bisogna far ricorso alla teoria del campionamento. La teoria di Nyquist indica che sono necessari un minimo di due campioni per ciclo per poter ricostruire la forma d'onda desiderata in uscita. Per il teorema del campionamento, il singolo spettro della forma d'onda voluta, viene replicata e centrata in multipli della frequenza di campionamento. Nel caso di Sample & Hold, che è il metodo che

utilizza il convertitore DAC, tali repliche vengono poi pesate per la funzione $\frac{\sin x}{x}$ che si annulla proprio per valori interi multipli della frequenza di campionamento (figura 4.5).

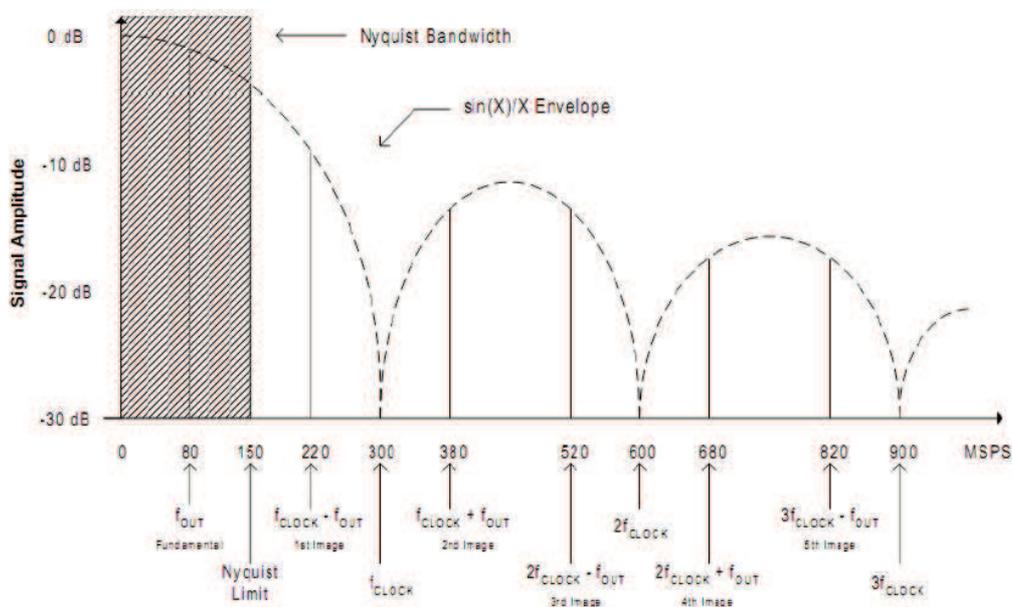


Figura 4.5- Esempio di spettro di uscita dal DDS

Per questo motivo risulta evidente, che per ottenere una sinusoida pura in uscita lo spettro ottenuto andrà ulteriormente filtrato attraverso un filtro passa basso. Tale filtro andrà ad eliminare le repliche della frequenza f_{out} (frequenze immagini) poste a multipli di $f_{clk} \pm f_{out}$. Per facilitare la progettazione di tali filtri e migliorare la purezza dello spettro di uscita è ragionevole limitare la frequenza f_{out} d'uscita al 40% della frequenza di campionamento, in modo da poter utilizzare filtri con costi competitivi.

In aggiunta al problema delle immagini, altri fattori degradano la purezza spettrale del segnale, quali :

- Effetti di non linearità del convertitore D/A (*Integral/Differential Linearity Errors*).
- Troncamento di Fase.

- Rumore associato alla segnale di Clock (*Clock feed-through noise*).

Tali anomalie non seguono l'andamento della $\frac{\sin x}{x}$; appaiono come armoniche spurie e generalmente hanno ampiezze inferiori rispetto ai contributi delle immagini dovute al campionamento.

4.3.1 Effetti della risoluzione del convertitore D/A sulle spurie di uscita

La risoluzione di un DAC (*Digital to Analog Converter*) è specificata dal numero dei suoi bits (D-Bits) in ingresso. Guardando la ricostruzione di un'onda sinusoidale (figura 3.6) si capisce facilmente che importanza abbia la risoluzione di un DAC. Come si vede dalla figura 3.6 nel quale un DAC a 4 bit di risoluzione è utilizzato per ricostruire una forma d'onda sinusoidale perfetta. Le linee verticali sono gli istanti di tempo durante i quali l'uscita del DAC viene aggiornata. Le distanze orizzontali tra le righe verticali rappresentano il periodo di campionamento. La distanza verticale tra due righe orizzontali è proprio l'errore introdotto dal DAC e dipende dalla risoluzione di quest'ultimo. Questo errore è detto *errore di quantizzazione* e dà origine ad un effetto noto come *distorsione da quantizzazione*. Per comprendere la distorsione da quantizzazione dobbiamo guardare le rapide variazioni del DAC, queste brusche variazioni implicano la presenza di componenti ad alta frequenza sulla fondamentale. Sono queste frequenze che costituiscono la distorsione dovuta all'errore di quantizzazione per cui nel dominio della frequenza appaiono come spurie discrete interne alla banda di Nyquist nello spettro di uscita del DAC.

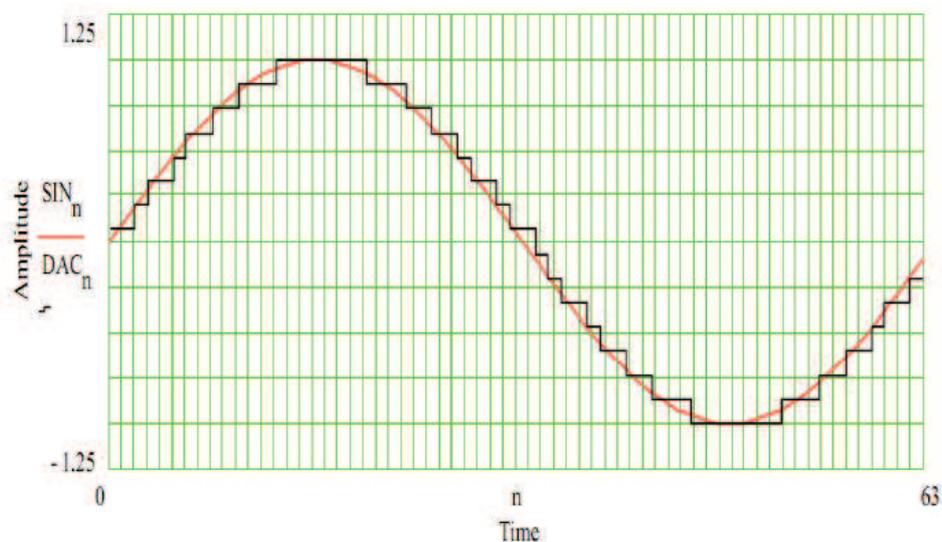


Figura 4.6- Effetto della risoluzione del DAC sulla sinusoide generata

È evidente che aumentando la risoluzione l'errore di quantizzazione diminuisce e le componenti spurie nello spettro all'uscita del DAC diminuiscono (figura 4.7). La relazione tra la risoluzione del DAC e l'errore di quantizzazione è quantificabile tramite il parametro SQR (*Signal power to Quantization noise power Ratio*) definito come:

$$SQR_{dB} = 1.76 + 6.02 N$$

Dove N rappresenta il numero dei bit del DAC.

Come si nota in figura 3.7, avere più bit a disposizione comporta un netto miglioramento sulla qualità dello spettro di uscita. I livelli delle spurie dovuto all'errore di quantizzazione sono più bassi se si aumenta il numero di bit. Il valore del SQR visto prima è valido solo se il DAC opera in fondo scala. Se il DAC lavora al di sotto del fondo scala la potenza associata alla fondamentale è minore, ma l'errore di quantizzazione rimane costante.

Questo effetto riduce il SQR che assume la forma:

$$SQR_{dB} = 1.76 + 6.02 N + 20 \log_{10}(FFS)$$

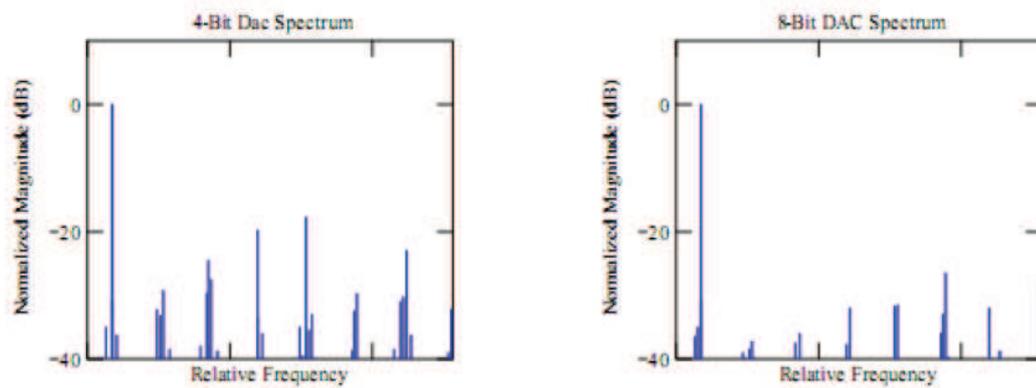


Figura 4.7- Comparazione dello spettro di uscita da un DAC a 4 e uno a 8 bits.

Dove FFS rappresenta la frazione del fondo scala a cui il DAC opera. Un altro effetto che produce il miglioramento di tale valore è invece dovuto al sovra-campionamento. Ricordando che Nyquist richiede che la banda del segnale campionato deve essere al massimo $1/2$ della frequenza di campionamento, in questo caso la banda del segnale campionato viene intenzionalmente ridotta ad una frazione della banda di Nyquist richiesta.

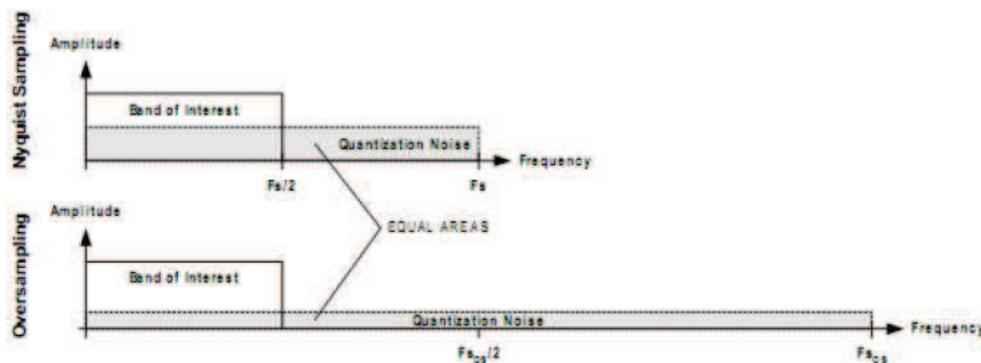


Figura 4.8- Effetto del sovracampionamento sul valore del SQR

Poiché la potenza del rumore di quantizzazione è legata alla risoluzione del DAC, aumentando la banda su cui viene preso in considerazione il suo livello diminuisce nella banda del segnale (figura 4.8). Questo significa che nella banda di interesse il parametro migliora ed abbiamo:

$$SQR_{dB} = 1.76 + 6.02 N + 20 \log_{10}(FFS) + \log_{10} \frac{F_{S_{os}}}{F_s}$$

con $F_{S_{os}}$ è la frequenza di *over-sampling* cioè di sovraccampionamento e F_s rappresenta la banda di Nyquist richiesta per non avere aliasing tra le repliche dello spettro del segnale.

4.3.2 Effetti del troncamento di fase

Il troncamento di fase è un importante aspetto delle architetture DDS. Consideriamo un DDS con accumulatore di fase a 32 bit. Per convertire 32 bit di fase nei corrispondenti valori dell'ampiezza avremmo bisogno di una tabella LUT con 2^{32} locazioni di memoria, ciò implicherebbe che se ogni locazione contenesse un valore codificato con 8 bit avremmo bisogno di una LUT di 4 GB di memoria. Chiaramente ciò è impensabile (da un punto di vista costruttivo) perché l'ingombro dovuto ad una memoria così grande andrebbe a ledere una delle caratteristiche peculiari del DDS ossia la compattezza e indirizzare una memoria così grande comporterebbe una riduzione della velocità operativa. Per porre rimedio a questa problematica si utilizzano soltanto una frazione dei bit che rappresentano l'uscita dell'accumulatore di fase. Per esempio su 32 bits prendiamo soltanto i primi $P = 12$ bit dell'informazione di fase e i restanti 20 bit vengono ignorati, troncati appunto. Per capire l'implicazione del troncamento di fase occorre far riferimento al concetto di ruota di fase digitale già introdotto precedentemente. Supponiamo di avere un DDS con un accumulatore di fase a 8 bit cioè 256 punti, ma di utilizzare soltanto i primi 5 bit per accedere alla *look-up table* cioè 32 punti (vedi figura 3.9).

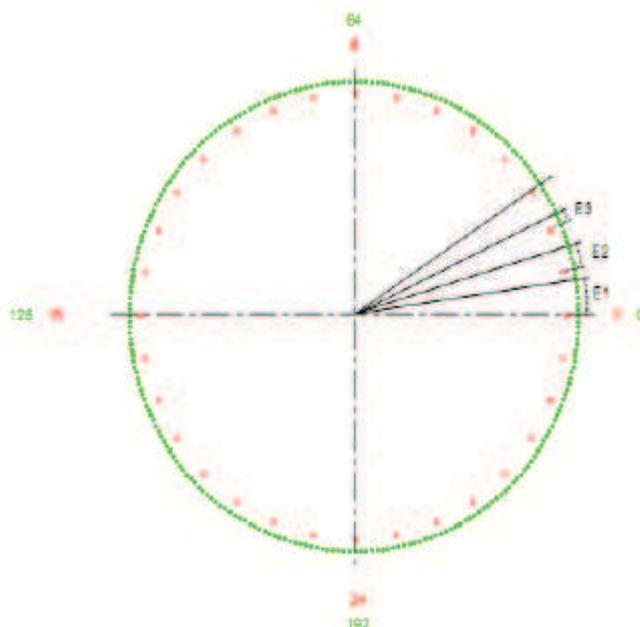


Figura 4.9 - Effetto del troncamento di fase sulla ruota delle fasi

Nel primo caso ho $1/256$ di risoluzione ($360^\circ/2^8 = 1.41^\circ$), mentre nel secondo caso la risoluzione di fase è di $1/32$ ($360^\circ/2^5 = 11.25^\circ$). Nel primo caso però avrei una LUT di 2^8 bytes mentre nel secondo avrei 2^5 , chiaramente ciò rappresenta una semplificazione della circuiteria del DDS. Supponiamo di avere $M = 6$ come FTW (*Frequency Tuning Word*). Questo significa che l'accumulatore conta con incrementi di 6. Al primo passo, nel cerchio esterno ho percorso 6 punti, mentre in quello interno non mi sono ancora mosso. Questo comporta un errore di fase di $(6 \times 1.41^\circ)$. Al secondo passo mi muovo ancora di 6 punti nel cerchio esterno, supero così il primo punto del cerchio interno. Questo comporta un ulteriore errore di $(4 \times 1.41^\circ)$. Questo procedimento continua, fino a che non ritorno al punto di partenza. Si intuisce che dopo un certo numero di giri si ripresentano gli stessi errori, tali errori sono quindi periodici. Essendo periodici nel dominio

del tempo, si presentano come spurie spettrali nel dominio della frequenza e sono conosciute come *Phase Truncation Spurs*.

Queste spurie dipendono da tre fattori :

1. Le dimensioni dei bit dell'accumulatore e del Frequency Control Register(N-Bits);
2. Il numero dei bit di fase dopo il troncamento (P-Bits);
3. Il valore della Frequency Tuning Word (M).

Alcune FTW non comportano nessun troncamento di fase e il livello delle spurie risulta essere contenuto, mentre altre FTW hanno un livello massimo di spurie. Nelle applicazioni pratiche se la quantità (N- P) Bit è maggiore o uguale a 4, il massimo livello delle spurie è approssimabile a $-6:02 \cdot P$ dBc (cioè $-6:02 \cdot P$ dB al di sotto del livello della portante a F_{out}). Per cui sperimentalmente gli errori nello spettro risultano accettabili. Se ho un DDS a 32 NBits che utilizza 12 PBit il livello delle spurie è -72 dBc rispetto al valore della fondamentale.

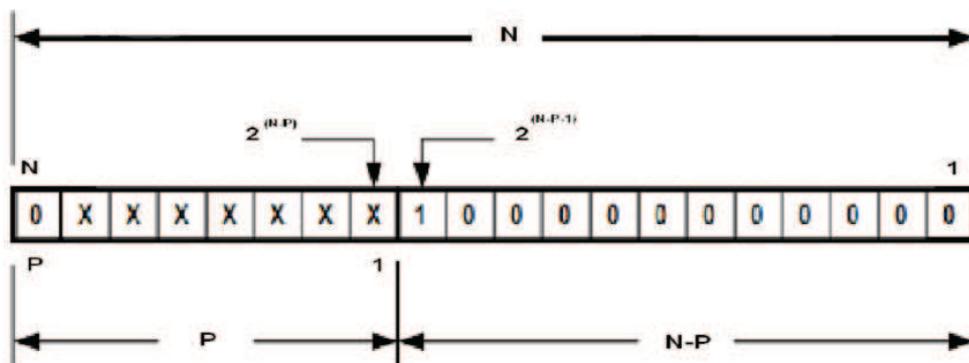


Figura 4.10- Forma delle FTW che forniscono il massimo livello di spurie

Le FTW che hanno un elevato livello di spurie sono quelle che, soddisfano la seguente relazione:

$$MCD(FTW, 2^{N-P}) = 2^{N-P-1}$$

dove $MCD(X; Y)$ è il Massimo Comune Divisore di X e Y. Affinché la relazione sia soddisfatta le FTW devono avere la forma mostrata in figura 3.10, cioè hanno un bit ad 1 nella posizione N-P-1.

I primi P bit sono quelli utilizzati per l'informazione di fase detti anche *Phase Word*. I restanti N-P bits vengono quindi ignorati. La FTW è composta da N-1 bits (questo perché il primo bit, ossia il *Most Significant Bit* MSB, deve essere 0 per poter rispettare la condizione di Nyquist e quindi risolvere il problema di aliasing).

Altre FTW invece, vengono troncate, ma il troncamento non comporta nessun danno. Queste particolari FTW sono quelle che soddisfano la seguente equazione :

$$MCD(FTW, 2^{N-P})=2^{N-P}$$

Come mostrato in figura 3.11 queste parole presentano un bit ad 1 nella posizione N-P e 0 nei restanti bit che vengono troncati, ma essendo zeri non portano informazione aggiuntiva per cui non si ha perdita di dati. Queste due caratteristiche rappresentano gli estremi, tutte le altre FTW si collocano all'interno di queste due categorie. Possiamo capire a questo punto come vengono generate le spurie e le loro implicazioni nello spettro di uscita.

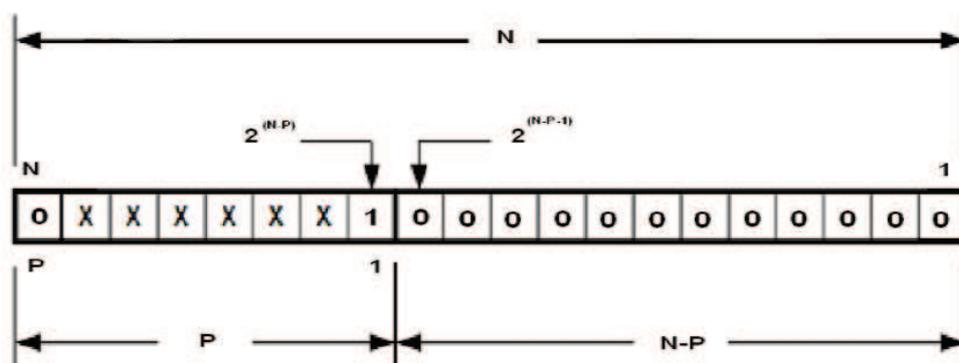


Figura 4.11- Forma delle FTW che non forniscono spurie da troncamento

In figura 4.12 vediamo la differenza dei diversi casi :

In figura 4.12a la $FTW_{bin} = [0 1 0 1 1 1 0 0 0 0]$ codificata su $N = 10$ bit. La frequenza da sintetizzare è 107.8 MHz. Per l'informazione di fase prendiamo i primi $P = 5$ bit. Questo è il caso peggiore che si può verificare nel troncamento poiché il bit nella posizione $N - P - 1$ è un 1. Nel caso peggiore poiché la quantità $N - P = 5$ quindi maggiore di 4 il FSDR (*Free Spurious Dynamic Range*) è di $-6:02 \cdot P = -30$ dBc. In figura 4.12b invece la $FTW_{bin} = [0 1 0 1 1 0 0 0 0 0]$ a cui corrisponde una frequenza di 100.3 MHz, troncando i primi 5 bit non ho perdita di informazione e si nota che non ho spurie nello spettro di uscita.

Nella figura 4.20c la $FTW_{bin} = [0 1 0 1 1 1 0 1 0 1]$ a cui corrisponde una frequenza di 109.2 MHz rappresenta un caso intermedio tra i due estremi, il FSDR rimane comunque al di sotto dei -30 dBc. Per analizzare e capire come si formano le spurie, ricordiamo innanzitutto, che il cuore del DDS consiste in un accumulatore ad N-Bits che ricorsivamente somma la FTW. Inizialmente il valore dell'accumulatore sarà uguale alla FTW che ho scelto. Ad ogni impulso di clock il valore della FTW viene sommato al precedente valore contenuto nell'accumulatore. Dopo un certo numero di cicli di clock il valore iniziale (cioè quella della FTW stessa) si ripresenterà. Il numero di cicli di clock richiesti per cui si verifichi tale fenomeno è detto GRR (*Ground Repetition Rate*) ed è uguale a:

$$GRR = \frac{2^N}{MCD(FTW, 2^N)}$$

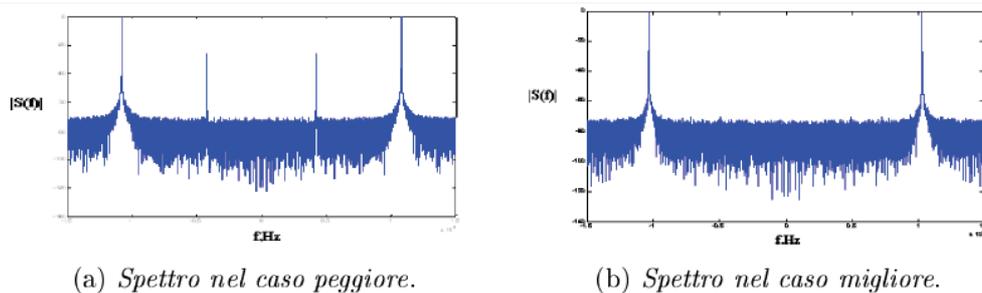


Figura 4.12- Spettro di uscita nei diversi casi

Nel caso mostrato in figura 3.13 N-Bits è 20 e la $FTW_{dec} = 182989$ e richiede un GRR di 524288. Per cui dopo circa mezzo milione di cicli di clock l'accumulatore riparte dal valore iniziale. Alcuni DDS hanno valori di N-Bits molto elevati (ad es. 48) per cui occorre un numero elevatissimo di cicli di clock prima che la sequenza riparta. Come dicevamo prima però, soltanto P bit vengono utilizzati per essere inviati al convertitore fase/ampiezza utilizzato per produrre la forma d'onda d'uscita. I restanti N-P bit quindi non vengono passati attraverso il convertitore. Comunque sia, se l'accumulatore è composto da N bit, ma soltanto P bit vengono utilizzati per determinare l'ampiezza, il segnale di uscita è essenzialmente in errore. Per cui il vero segnale di uscita è una composizione di due segnali, quello che ha massima risoluzione (cioè senza il troncamento di fase) ed un segnale che corrisponde all'errore dovuto ai bit troncati. Il segnale errore è la sorgente delle spurie. Per l'analisi di questo segnale errore faremo attenzione soltanto ai bit che vengono troncati e non ai bit della parola di fase.

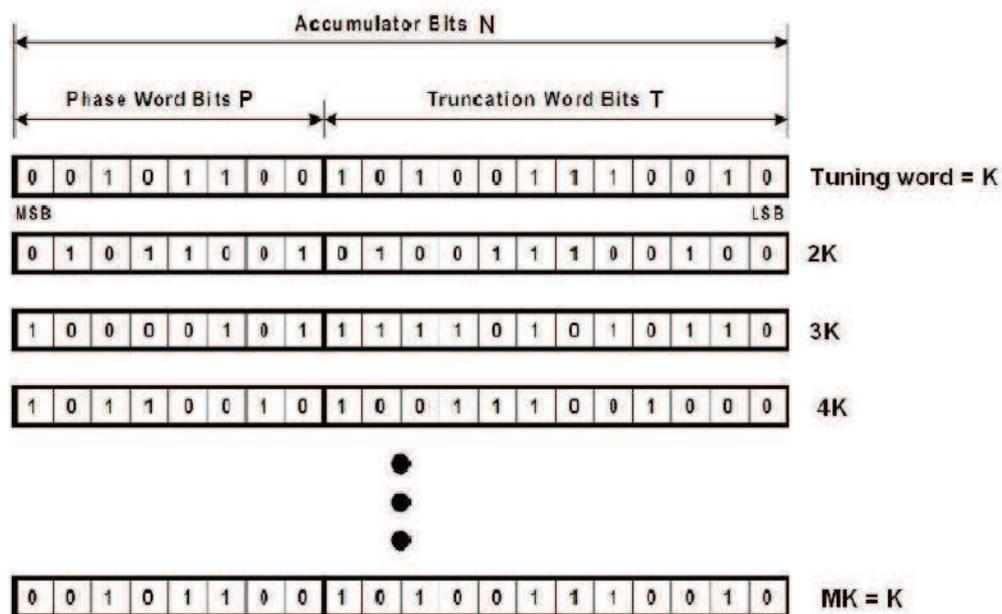


Figura 4.13- Sequenza delle FTW all'interno dell'accumulatore

È possibile quindi determinare il periodo entro il quale la Truncation Word (la codifica dei bit troncati) si ripete. Per esempio, considerando i valori presi in considerazione prima, il valore dei bit troncati che chiameremo $T = N - P$ è uguale a 12 bit. Introduciamo quindi una nuova tuning word detta ETW (*equivalent tuning word*) data da:

$$ETW = FTW \bmod(2^T)$$

Il risultato di questa operazione non è altro che una porzione della FTW originale. Sempre per l'esempio dato, la nostra ETW diventa uguale a 2674 e il GRR della parola troncata diventa 2048. Per cui abbiamo un segnale di errore che si ripete ogni 2048 cicli di clock. Ma quale è il comportamento della ETW all'interno di questo periodo?

Per trovare risposta a tale domanda dobbiamo considerare la capacità della parola troncata ETW che è 2^T . Dividendo il valore della capacità della ETW e il suo valore, ricaviamo il numero dei cicli di clock che impiega l'accumulatore per causare un overflow. Nel nostro caso la capacità dell'accumulatore è $2^{20} = 4096$. Prima di dividere il valore

della capacità per la ETW dobbiamo andare a vedere se il bit più significativo della ETWbin è uguale a 1.

In questo caso, il periodo di overflow sarebbe minore di 2 cicli di clock, per cui la frequenza prodotta sarebbe un'immagine. Se si verifica questo caso, dobbiamo aggiustare la ETW, sottraendo al valore della capacità la ETW stessa. Nell'esempio otteniamo $4096 - 2674 = 1422$. Questo perché la codifica binaria di 2674 ha nel suo MSB un 1. Se il bit più significativo è a 0 questa scalatura della ETW non è necessaria. Adesso possiamo determinare il numero di cicli di clock che determinano l'overflow della ETW.

$$overflow = \frac{2^T}{ETW} = \frac{4096}{1422} = 2,88045$$

Questo valore è la media dei cicli di clock impiegati per generare overflow della ETW. Se noi conosciamo il ciclo di ripetizione GRR della nuova ETW e conosciamo il numero dei cicli di clock che servono per generare overflow, possiamo calcolare il numero degli overflow come:

$$Numero\ f\ Overflow = \frac{GRR}{Overflow} = \frac{2048}{2,88045} = 711$$

Con queste informazioni è possibile quindi visualizzare l'effetto del troncamento di fase come mostrato in figura 4.14. Per cui ogni 2048 cicli, il segnale dente di sega si ripete dopo aver effettuato 711 overflow. Poiché risulta un segnale periodico nel dominio temporale, la sua trasformata di Fourier sarà periodica anche nel dominio della frequenza. In più essendo un sequenza reale la sua trasformata sarà simmetrica rispetto all'origine quindi può essere rappresentata con metà dei campioni, ad esempio se ho 2048 campioni nel tempo, avrò 1024 frequenze discrete positive e 1024 frequenze negative.

La frequenza fondamentale del nostro segnale dente di sega è :

$$f_{SawTooth} = \frac{f_{clock} \cdot ETW}{2T}$$

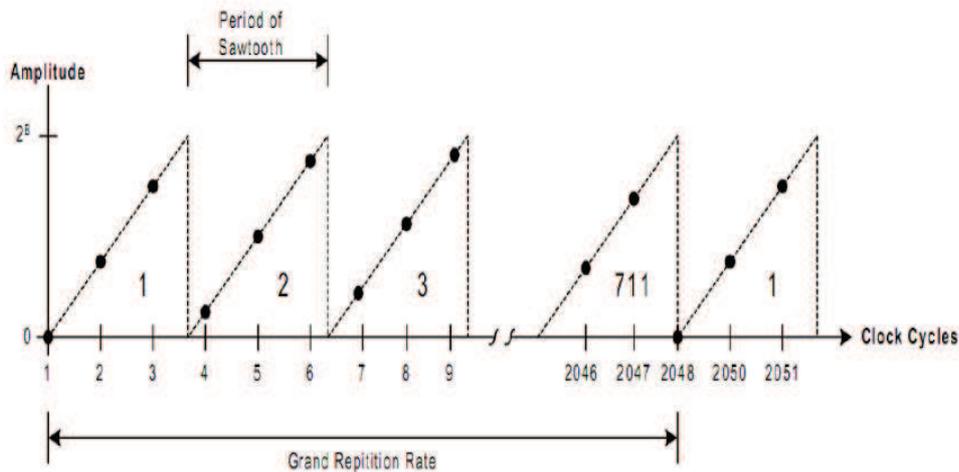


Figura 4.14- Andamento del segnale errore dovuto al troncamento di fase

Per cui lo spettro della sinusoide che cerco di sintetizzare sarà affetto da un segnale d'errore che terrà conto delle spurie introdotte dalla trasformata di Fourier del segnale dente di sega dovuto al troncamento di fase. Come si può vedere in figura 4.15 prendendo in considerazione una porzione dello spettro del segnale dente di sega e andando a vedere alcune delle spurie che vanno a rimapparsi nella banda di Nyquist di interesse, si capisce l'effetto delle spurie introdotte dal troncamento di fase.

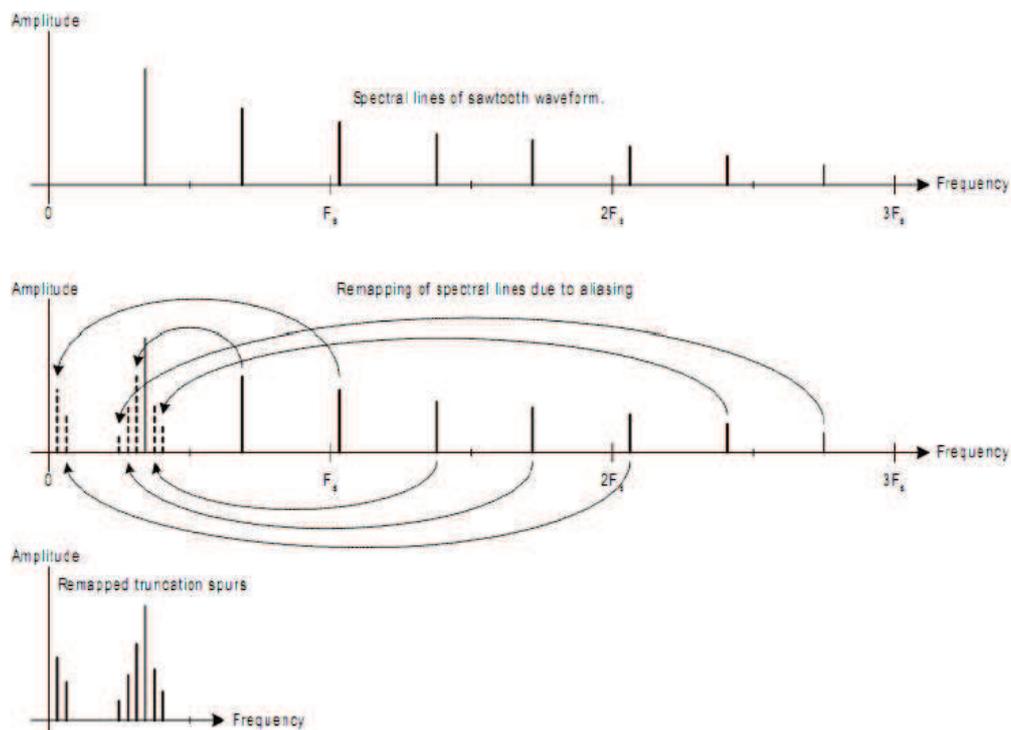


Figura 4.15- Spettro del segnale errore dovuto al troncamento di fase

4.3.3 Considerazioni sul filtraggio di uscita del DDS

Il DDS è un sistema campionato, per cui lo spettro del segnale di uscita è infinito. Come si nota anche in figura 4.16, lo spettro comprende non solo la frequenza fondamentale di nostro interesse f_{out} , ma anche le sue repliche che decrescono in ampiezza seguendo l'involuppo della $\text{sinc}(x)$. In figura, per una maggior chiarezza esplicativa, sono stati omessi i disturbi frequenziali dovuti alle problematiche analizzate precedentemente. Nella maggior parte delle applicazioni le repliche sono indesiderate, perciò un dispositivo DDS è generalmente seguito da un filtro passa-basso (detto appunto anti-immagine).

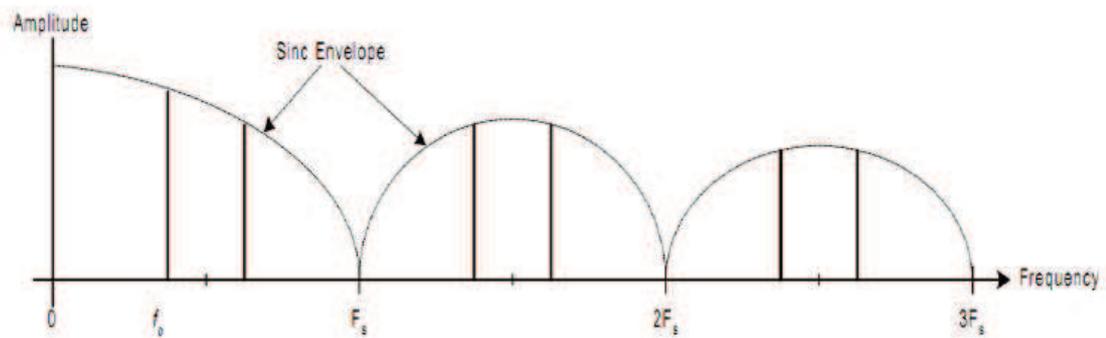


Figura 3.16- Spettro di uscita del DDS

La risposta in frequenza ideale di questo filtro è unitaria sull'intera banda di Nyquist ($0 < f < \frac{F_s}{2}$) e nulla al di fuori (come mostra la figura 4.17).

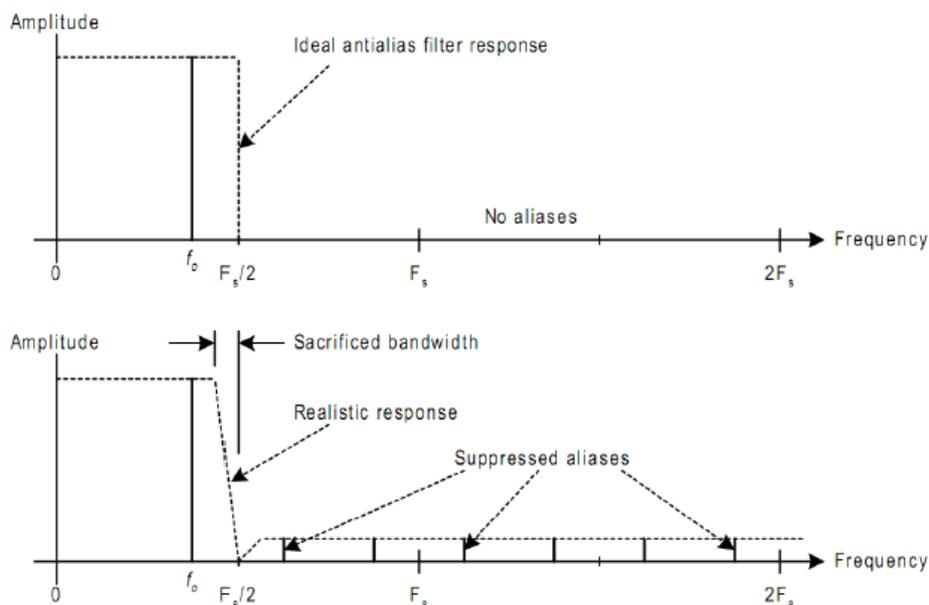


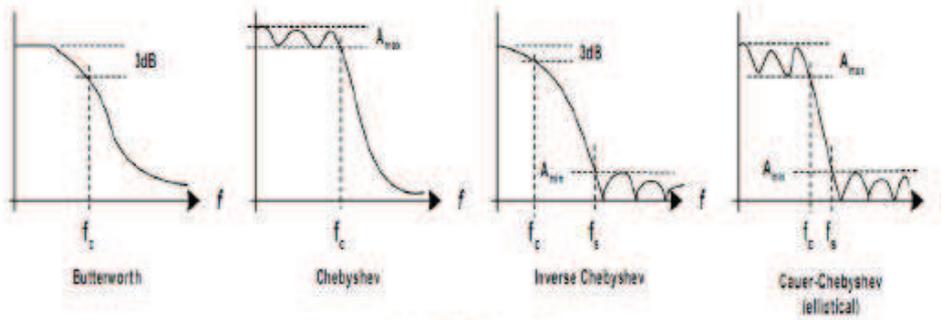
Figura 4.17 - Risposta in frequenza del filtro passa-basso anti-immagini

La realizzazione di tale filtro è però fisicamente irrealizzabile. La cosa migliore che si può è una risposta ragionevolmente buona per una certa percentuale della banda di Nyquist (detta banda passante, circa il 90%) con un rapido aumento dell'attenuazione fino a $\frac{F_s}{2}$, e sufficiente attenuazione nella restante parte dello spettro (banda

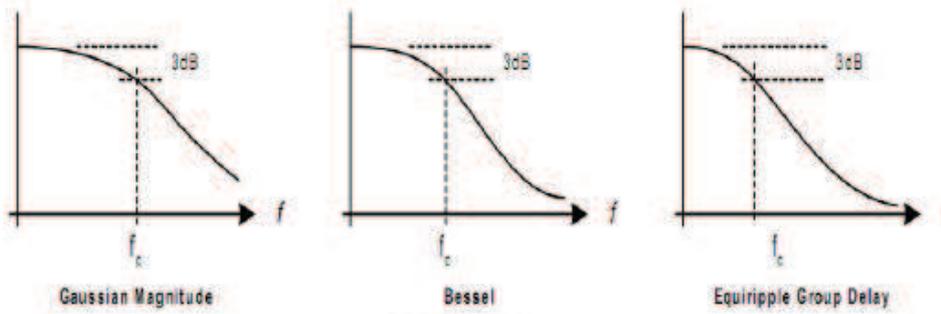
oscura). Questo sfortunatamente, implica il sacrificio di alcune porzioni di banda disponibile per poter realizzare la risposta del filtro anti-immagine. La realizzazione di tale filtro è un elemento critico nei sistemi DDS. Le caratteristiche che devono essere imposte alla progettazione del filtro dipendono fortemente dal tipo di sistema DDS voluto. Ci sono molte tipologie di filtri in letteratura ma tuttavia nella maggior parte delle applicazioni possiamo suddividerle in tre grandi categorie :

- Chebyshev
- Gaussiani
- Legendre

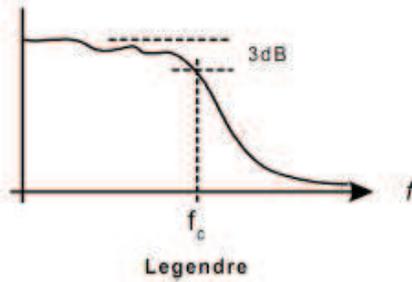
Per le applicazioni che richiedono una risposta in frequenza abbastanza ripida sono indicati i filtri della famiglia Chebyshev. Comunque, bisogna tener conto che si avranno dei *ringing* e degli *overshoot* nel dominio del tempo che potrebbero disturbare il funzionamento. Le applicazioni che richiedono una risposta nel tempo regolare e liscia sono indicate per l'utilizzo dei filtri gaussiani che però non hanno caratteristiche molto ripide in frequenza. Per le applicazioni che si trovano nel mezzo ai due estremi una buona scelta sono i filtri della famiglia Legendre. Le caratteristiche delle varie famiglie di filtri si possono vedere in figura 4.18.



(a) *Chebyshev.*



(b) *Gaussiano.*



(c) *Legendre.*

Figura 4.18- Risposta in frequenza delle varie famiglie di filtri

4.4 DDS Analog Devices

L'Analog Devices è leader a livello mondiale nella produzione di DDS (*Direct digital synthesis*) grazie alla loro capacità di integrare le varie funzioni di bordo come ad esempio comparatori, RAM, PLL, mixer e registri, rendendoli delle soluzioni ideali di sintetizzatore di frequenza nei sistemi di comunicazione per testare apparecchiature e nelle applicazioni radar. Offre inoltre una vasta gamma di dispositivi DDS a seconda delle specifiche di progetto. Per i nostri scopi i più adatti risultano l'AD9858 e l'AD9910.

4.4.1 DDS Analog Devices AD9858

L'AD9858 è un sintetizzatore digitale diretta (DDS) che sfrutta un convertitore digitale-analogico (DAC) a 10 bit che opera fino a 1 GSPS. Usa una tecnologia DDS avanzata, accoppiata con un DAC ad alte prestazioni molto veloce per creare un sistema completo di sintesi ad alta frequenza programmabile digitalmente capace di generare una forma d'onda sinusoidale in uscita analogica Fino a 400MHz.

Il dispositivo AD9858 è progettato per fornire salti di frequenza molto veloci e una risoluzione in frequenza molto alta.

Le caratteristiche principali sono:

- Velocità di clock a 1 GSPS
- DAC integrato a 10 bit
- 32 bit per la codifica della parola di frequenza
- Risoluzione in frequenza di 0.23 Hz
- 14 bit per la codifica della parola di fase
- 10 bit per la codifica della parola di ampiezza

sintesi ad alta frequenza programmabile digitalmente capace di generare una forma d'onda sinusoidale in uscita analogica fino a 400MHz. L'AD9910 impiega un' avanzata tecnologia che fornisce una significativa riduzione della potenza consumo senza sacrificare le prestazioni. L'utente ha accesso ai tre parametri del segnale di controllo che controllano frequenza, fase e ampiezza.

Le caratteristiche principali sono:

- Velocità di clock a 1 GSPS
- DAC integrato a 14 bit
- Risoluzione in frequenza di 0.23 Hz
- 32 bit per la codifica della parola di frequenza
- 16 bit per la codifica della parola di fase
- 14 bit per la codifica della parola di ampiezza
- Rumore di fase $\leq -125 \frac{dB_c}{Hz} @ 1kHz \text{ offset}$ (400MHz carrier)
- Prestazioni dinamiche eccellenti $> 80dB_c$ con a banda stretta SFDR
- Moltiplicatore PLL del clock di riferimento
- Tensione di alimentazione che può variare tra 1.8 - 3.3 V
- Potenza dissipata tipicamente 1.70mW
- Controllo di ingresso uscita seriale
- Oscillatore interno può essere pilotato da un singolo cristallo
- Capacità di modulazione di fase
- Capacità di modulazione di ampiezza

Lo schema a blocchi di tale dispositivo è mostrato in figura 4.20. Per maggiori informazioni è possibile consultare il riferimento[9].

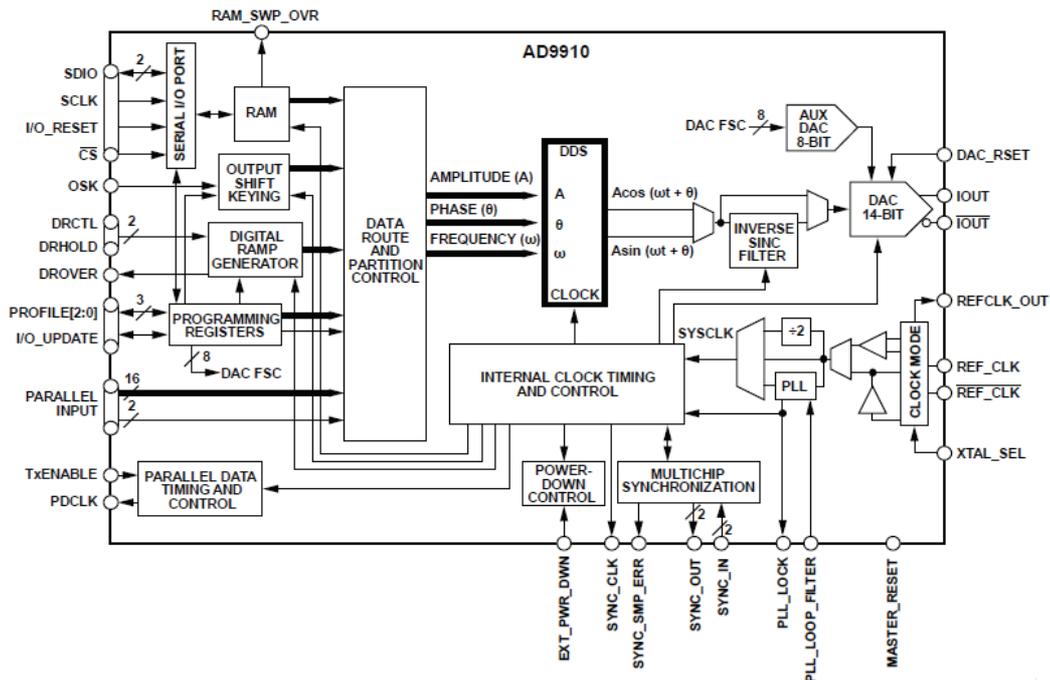


Figura 3.20 - Schema a blocchi dell'AD9910

4.4.3 Confronto tra AD9858 e AD9910

Passiamo ora ad effettuare un confronto tra questi due componenti al fine di effettuare una scelta efficiente.

Dalla tabella 3.1 è possibile evincere che l'AD9910, oltre ad avere un consumo di potenza inferiore, è molto più performante avendo una risoluzione di 14 bit rispetto ai 10 bit per l'AD9858; ciò equivale a dire che l'SNR dell'AD9910 è maggiore rispetto all'AD9858.

Un'altro parametro da considerare, non mostrato nella tabella 3.1, è la modalità di programmazione. In particolare se per l'AD9858 abbiamo la possibilità di effettuare una programmazione sia in parallelo che in seriale ciò non è vero per l'AD9910. Infatti tale dispositivo consente esclusivamente la modalità di programmazione seriale con frequenza massima pari a 70 Mbps.

Sigla Dispositivo	AD9858	AD9910
Master f_{clk}	1GHz	1GHz
Resolution n [Bits]	10	14
Tuning Word Width	32	32
I/O interface	Serial/Parallel	Serial/ Parallel
RefClk Multiplexer	Yes	Yes
Compliance Range	3.8 V	500mV
Nominal Supply	Multi(+3.3, +5)	Single(+1.8), Single(+3.3)
Price	45.66 \$	35.33 \$

Tabella 4.1 - Confronto tra AD9858/AD9910

Per quanto riguarda l'AD9858 sono possibili entrambi le modalità di programmazione; per la modalità seriale la frequenza massima pari a 10 MHz. Per la modalità parallela consente l'invio di 8 bit simultaneamente ad una frequenza di 100MHz.

Dall'analisi dei dati mostrati in questo paragrafo e dai dati raccolti mostrati in questa sessione si evidenzia che:

- la scelta ottimale risulta essere l'AD9858 se l'obiettivo è quello di ottimizzare i tempi di programmazione del dispositivo.
- la scelta ottimale risulta essere l'AD9910 se l'obiettivo è quello di ridurre i consumi di potenza e soprattutto avere una risoluzione maggiore.

5. IL SINTETIZZATORE FREQUENZIALE AD9858

5.1 Introduzione

In questo capitolo analizzeremo l'*evaluation board* per l'AD9858. Come descritto nel capitolo precedente, l'AD9858 è sintetizzatore di frequenza digitale con frequenza di clock massima di 2 GHz (divisore On) che include un DAC 10bit, un mixer RF e blocca la sintesi di chip PLL.

La schema a blocchi dell'*evaluation board* è mostrato nella figura 5.1.

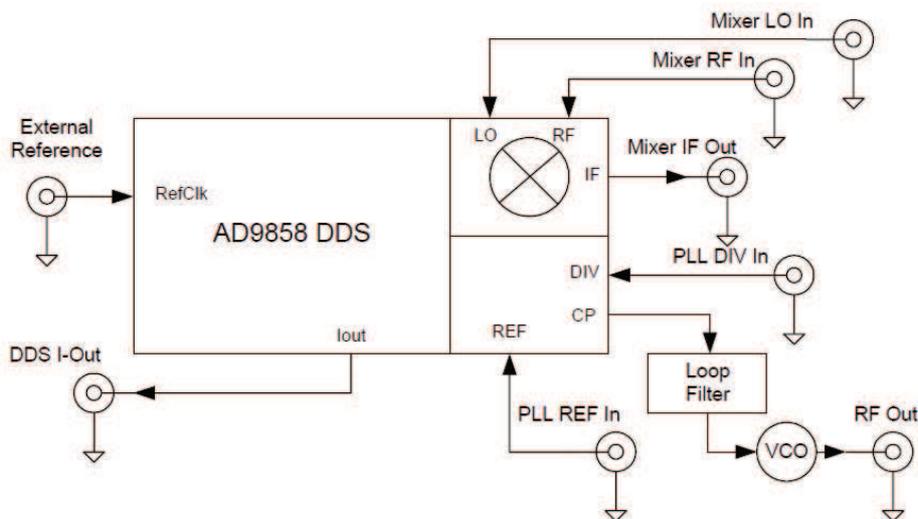


Figura 5.1 - Schema a bordi evaluation board AD9858

L'uscita del DAC della scheda di valutazione è stata progettata per permettere il pieno controllo dell'utente su tutti i blocchi funzionali dell' AD9858. Ciascun blocco funzionale ha i suoi ingressi e le uscite. L'utente può configurare e collegare il blocco DDS, il blocco PLL e il blocco mixer in qualunque modo, rispettando ovviamente i limiti del

dispositivo. Ad esempio, se un utente volesse valutare solo il mixer RF, il riferimento esterno (REFCLK) per il DDS e gli ingressi PLL possono essere scollegati. Ciò consente all'utente la possibilità di valutare le prestazioni del mixer come un singolo componente.

La tabella 5.1 descrive le modalità in cui il dispositivo può lavorare:

Parameter	Rating
Maximum Junction Temperature	150° C
Vs	4 V
Digital Inputs	-0.7 to +Vs
Digital Output Current	5 mA
Storage Temperature	-65°C to 150°C
Operating Temperature	-40° C to 85 °C
Lead Temperature (Soldering 10 s)	300° C
Maximum Clock Frequency	2 GHz

Tabella 5.1 - Parametri operativi per l'AD9858

La figura 5.2 mostra come sono disposti i pin del dispositivo.

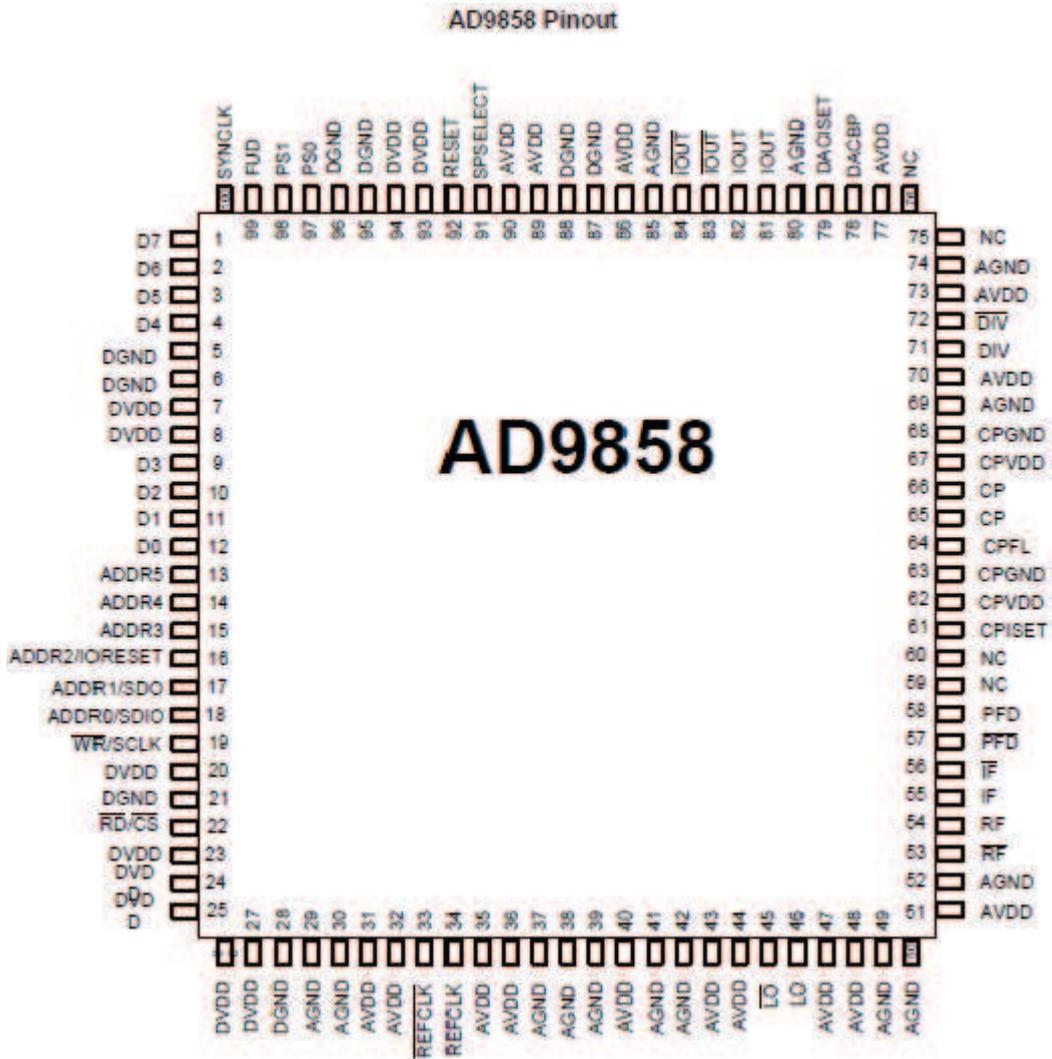


Figura 5.2 - Pin configuration

Mentre in tabella 5.2 è descritta la funzione di ciascun pin:

Pin No.	Mnemonic	I/O	Description
1-4, 9-12	D7-D0	I	Parallel Port DATA. Note that the functionality of these pins is only valid when the I/O port is configured as a parallel port.
5, 6, 21, 28, 95, 96	DGND		Digital Ground.
7, 8, 20, 23-27, 93, 94	DVDD		Digital Supply Voltage.
13-18	ADDRS- ADDR0	I	When the I/O port is configured as a parallel port, these pins serve as a 6-bit address select for accessing the on-chip registers (see the IORESET, SDO, and SDIO pins below for serial port mode).
16	IORESET	I	Note that this is only valid for serial programming mode. Active high input signal that resets the serial I/O bus controller. It is intended to serve as a means of recovering from an unresponsive serial bus caused by improper programming protocol. Asserting an I/O reset does not affect the contents of previously programmed registers nor does it invoke their default values.
17	SDO	O	Note that this is only valid for serial programming mode. When operating the I/O port as a 3-wire serial port, this pin serves as a unidirectional serial data output pin. When operated as a 2-wire serial port, this pin is unused.
18	SDIO	I or I/O	Note that this is only valid for serial programming mode. When operating the I/O port as a 3-wire serial port, this pin is the serial data input. When operated as a 2-wire serial port, this pin is the bidirectional serial data pin.
19	WFSCLK	I	When the I/O port is configured for parallel programming mode, this pin functions as an active low write pulse (WR). When configured for serial programming mode, this pin functions as the serial data clock (SCLK).
22	RDVCS	I	When the I/O port is configured for parallel programming mode, this pin functions as an active low read pulse (RD). When configured for serial programming mode, this pin functions as an active low chip select (CS) that allows multiple devices to share the serial bus.
29, 30, 37-39, 41, 42, 49, 50, 52, 69, 74, 80, 85, 87, 88	AGND	I	Analog Ground.
31, 32, 35, 36, 40, 43, 44, 47, 48, 51, 70, 73, 77, 86, 89, 90	AVDD	I	Analog Supply Voltage.
33	REFCLK	I	Reference Clock Complementary Input. (Note that when the REFCLK port is operated in single-ended mode, REFCLK should be decoupled to AVDD with a 0.1 μ F capacitor.)
34	REFCLK	I	Reference Clock Input.
45	LO	I	Mixer Local Oscillator (LO) Complementary Input. Note that when the LO port is operated in single-ended mode, LO should be decoupled to AVDD with a 0.1 μ F capacitor.
46	LO	I	Mixer Local Oscillator (LO) Input.
53	RF	I	Analog Mixer RF Complementary Input. Note that when the RF port is operated in single-ended mode, RF should be decoupled to AVDD with a 0.1 μ F capacitor.
54	RF	I	Analog Mixer RF Input.
55	IF	O	Analog Mixer IF Output.
56	IF	O	Analog Mixer IF Complementary Output.
57	PFDC	I	Phase Frequency Detector Complementary Input. When the PFD port is operated in single-ended mode, PFD should be decoupled to AVDD with a 0.1 μ F capacitor.
58	PFD	I	Phase Frequency Detector Input.
59, 60, 75, 76	NC		No Connection.
61	CPISSET	I	Charge Pump Output Current Control. A resistor connected from CPISSET to CFGND establishes the reference current for the charge pump.
62, 67	CPVDD	I	Charge Pump Supply Voltage.
63, 68	CPGND	I	Charge Pump Ground.
64	CPFL	O	Charge Pump Fast Lock Output.
65, 66	CP	O	Charge Pump Output.
71	DIV	I	Phase Frequency Detector Feedback Input.
72	DIV	I	Phase Frequency Detector Feedback Complementary Input. Note that when the DIV port is operated in single-ended mode, DIV should be decoupled to AVDD with a 0.1 μ F capacitor.
78	DACBP		DAC Baseline Decoupling Pin, Typically Bypassed to Pin 77 with a 0.1 μ F Capacitor.
79	DACISSET	I	A resistor connected from DACISSET to AGND establishes the reference current for the DAC.
81, 82	IOUT	O	DAC Output.
83, 84	IOUT	O	DAC Complementary Output.
91	SPSELECT	I	I/O Port Serial/Parallel Programming Mode Select Pin. Logic 0: serial programming mode. Logic 1: parallel programming mode.
92	RESET	I	Active High Hardware Reset Pin. Assertion of the RESET pin forces the AD9858 to its default operating conditions.
97, 98	PS0, PS1	I	Used to select one of the four internal profiles. These pins are synchronous to the SYNCLK output.
99	FUD	I	Frequency Update. The rising edge transfers the contents of the internal buffer registers to the memory registers. This pin is synchronous to the SYNCLK output.
100	SYNCLK	O	Clock Output Pin that Serves as a Synchronizer for External Hardware. SYNCLK runs at REFCLK/8.

Tabella 5.2 - Pin configuration descriptions

5.2 Blocchi funzionali

5.2.1 DDS Core

Il nucleo DDS genera valori numerici che rappresentano una sinusoide nel dominio digitale. A seconda della modalità di funzionamento del DDS, tale sinusoide può essere modificata in frequenza, fase, o anche modulato da un segnale portante.

La frequenza del segnale di uscita è determinata da una programmata FTW (*frequency tuning word*) impostata dall'utente. La relazione della frequenza di uscita del dispositivo per il clock del sistema (SYSCLK) è determinato dalla seguente equazione:

$$F_0 = \frac{FTW \cdot SYNCLK}{2^N}$$

dove per l'AD9858, N=32.

5.2.2 DAC Output

L'AD9858 incorpora un sistema integrato a 10-bit DAC con uscita in corrente. Tale dispositivo è dotato di due uscite complementari e la loro combinazione fornisce, in scala, la corrente di uscita (I_{OUT}). Uscite differenziali riducono la quantità di rumore di modo comune che potrebbe essere presente all'uscita DAC, offrendo il vantaggio di un rapporto segnale-rumore maggiore. La scala di corrente è controllato per mezzo di un resistore esterno (R_{SET}) collegato tra il pin DACISET e la massa analogica. La scala di corrente è proporzionale al valore della resistenza secondo la relazione:

$$R_{SET} = \frac{39.19}{I_{OUT}}$$

Il fondo scala della corrente di uscita per il DAC è di 40 mA, ma l'uscita è limitata a 20 mA per garantire un SFDR migliori e quindi

prestazioni migliori. Il range d'uscita è (AVDD - 1,5 V) a (AVDD + 0,5 V). Le tensioni sviluppate al di là di questo intervallo causa distorsioni eccessive per il DAC e potrebbe potenzialmente danneggiare i circuiti di uscita DAC.

5.2.3 PLL frequency Synthesizer

Il sintetizzatore di frequenza PLL è un gruppo di blocchi di sintesi indipendenti, progettato per essere utilizzato con la DDS per ampliare la gamma di applicazioni di sintesi. I blocchi che costituiscono il PLL sono un comparatore di fase e frequenza digitale (PFD) che aziona una pompa di carica (CP). La pompa di carica, a rapida chiusura, verrà descritto di seguito. In base ai requisiti del sistema, l'utente fornisce un filtro ad anello esterno e uno o più VCO. Viene inserito anche un mixer analogico ad alta velocità per effettuare traslazioni in frequenza. Usando i diversi blocchi del sintetizzatore di frequenza PLL in combinazione con il DDS, l'utente può creare cicli di traslazione, un divisore frazionario ciclico.

5.2.4 Phase- Frequency Detector PFD

Il rivelatore di fase è dotato di due ingressi, PDIN e DIVIN. Entrambi sono ingressi analogici che possono essere gestite in differenziale o single-ended mode. Sono progettati per operare a frequenze fino a 150 MHz, mentre per i segnali fino a 400 MHz vengono utilizzati i divisori. Il livello di ingresso previsto sia per il PD e ingressi DIV è nell'intervallo di 800 mV_{pp} (differenziale), 400 mV_{pp} (single-ended). Il divisore programmabile offre rapporti di divisione di

$M, N = \{1, 2, 4\}$. Il rapporto di divisione è controllato per mezzo del registro *Control Function*.

5.2.5 Pompa di carico CP

Il riferimento di corrente in uscita alla pompa di carica è determinato da una resistenza esterna (circa 2,4 k Ω), che stabilisce un 500 μ A di corrente di base interna massima (I_{CP0}). La corrente di base viene scalata per fornire all'unità la corrente appropriata per diverse modalità di funzionamento della CP. La quantità di scala in ciascuna modalità è programmabile tramite i valori memorizzati nel registro *Control Function*, fornendo la massima flessibilità d'uso della capacità di frequenza del PLL.

La polarità della CP può essere configurata come positivo o negativo rispetto all'ingresso PD. Quando la polarità CP è positiva, se l'ingresso DIV conduce l'ingresso PD, la pompa di carica tenderà di diminuire la tensione al nodo di controllo VCO. Se l'ingresso DIV ritarda l'ingresso PD, la pompa di carica aumenta la tensione sul nodo di controllo del VCO. Quando la polarità CP è negativa, avviene il contrario. Questo consente all'utente di definire sia input come il percorso di retroazione. Ciò consente all'AD9858 di utilizzare VCO con un qualsiasi riferimento compreso un riferimento a massa. Tale modalità è definita dalla pompa di carica polarità (CPP) bit accedendo al registro *Control Function*. Quando CPP = 0 (default), la pompa di carica è impostato per il funzionamento con un VCO riferito a massa. Quando CPP = 1, la pompa di carica è impostato per un VCO per il riferimento di tensione prefissato.

Interno alla CP, la corrente I_{CP0} viene scalata per fornire diversi valori di corrente di uscita per le varie modalità di funzionamento. Nella sua modalità di funzionamento normale, l'ultimo anello chiuso può essere programmato per scalare I_{CP0} di 1, 2, 3 o 4. Impostando il bit pompa di corrente di carica di offset, CFR <13>, si applica 2 mA di offset alla corrente della pompa di carica, consentendo valori scalati di I_{CP0} di 5, 6, 7, o 8. Le diverse modalità di funzionamento, e quindi i diversi valori di I_{CP0} , saranno discussi nella sezione successiva.

La CP ha un set di pin per l'alimentazione indipendente in grado di operare con tensioni fino a 5,25 V.

5.2.6 Fast-Locking Logic

La pompa di carica comprende un algoritmo di bloccaggio rapido che consente di superare i limiti tradizionali del PLL per quanto riguarda il tempo di commutazione di frequenza. L'algoritmo Fast Lock funziona in combinazione con il filtro di loop mostrato in figura 5.3 per fornire prestazioni in frequenza estremamente elevate.

Sulla base dell' errore tra il segnale di retroazione ed il segnale di riferimento, l'algoritmo di bloccaggio rapido mette la pompa di carica in uno dei tre stati:

- modalità rilevamento di frequenza
- modalità ad "ampio" anello chiuso.
- modalità finale ad anello chiuso.

Nella modalità di rilevamento di frequenza, i segnali di retroazione e di riferimento registrano la fase effettiva e gli errori di frequenza. Piuttosto che operare in un modo continuo chiuso retroazione, la pompa di carica fornisce una corrente fissa alla polarità del nodo di controllo VCO, il quale pilota l'anello di bloccaggio verso la

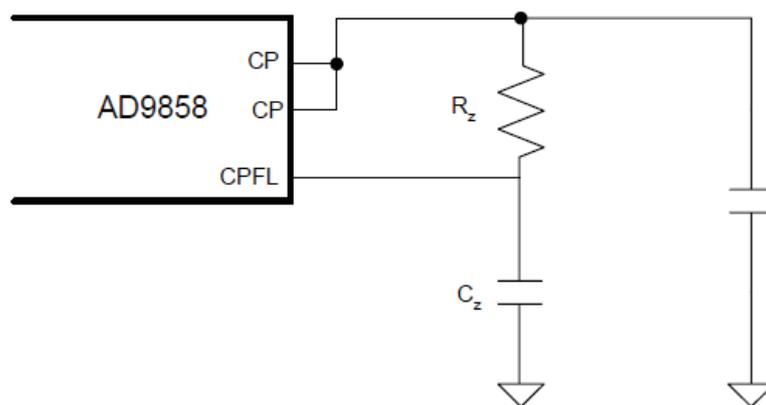


Figura 5.3 - Rappresentazione simbolica della pompa di carico connessa al filtro loop

frequenza. Una volta che la frequenza viene rilevata, la logica di bloccaggio rapido sposta la parte in una delle modalità ad anello chiuso. Nella modalità ad anello chiuso, ampi o finale, la pompa di carica fornisce corrente al filtro di anello a seconda della frequenza richiesta dal PFD. La modalità di rilevamento di frequenza è destinata a portare il sistema ad un livello di frequenza da cui il sistema a ciclo chiuso è in grado di raggiungere rapidamente il blocco di fase.

Il livello di precisione della frequenza auspicata è generalmente indicato come il range di aggancio. Quando la frequenza è all'interno del range di aggancio, il tempo richiesto per ottenere il blocco di fase può essere determinato mediante metodi di analisi standard dei PLL a transitori. C'è da notare che le sorgenti di pompa di carica di corrente associati alla modalità di rilevamento di frequenza sono collegati al pin 64, mentre le sorgenti di corrente ad anello chiuso sono collegati ai pin 65 e 66. Il pin 64 è collegato direttamente al filtro di loop tramite un condensatore di compensazione come illustrato in Figura 4.3. Questo collegamento permette la transizione più agevole dalla modalità di rilevamento di frequenza alle modalità ad anello chiuso e permette tempi di commutazione complessivamente molto veloci. I Pin 65 e 66 sono collegati al filtro di loop in modo convenzionale. Il

blocco di rilevamento della frequenza funziona come segue. La logica di confronto nella circuiteria di rivelazione di frequenza opera ad un ottavo del clock di sistema del DDS. Viene effettuato un confronto delle frequenze presenti all'ingresso PD e l'ingresso DIV su 19 cicli di clock DDS. Al fine di garantire il rilevamento di frequenza o e che la differenza di frequenza è all'interno del range di aggancio PLL, lo slew rate della ingresso VCO dovrebbe essere limitato in modo che il range di aggancio non può superare i 152 cicli macchina. La relazione che lega lo slew rate del segnale di ingresso al VCO, il condensatore di compensazione e la corrente dipendente dal livello programmato di frequenza da rilevare è:

$$\frac{dV}{dt} = \frac{I_{f_{det}}}{C_z}$$

5.2.7 Mixer analogico

Il mixer analogico viene utilizzato per cicli di traslazione, noti anche come anelli di offset. Gli ingressi a radio frequenza (RF) e l'oscillatore locale (LO) sono progettati per funzionare a frequenze fino a 2 GHz. Entrambi gli ingressi sono stadi di ingresso analogici differenziali. Entrambe le fasi di ingresso sono internamente DC devono essere collegato attraverso un meccanismo di accoppiamento AC esterno. Il livello di ingresso atteso è nell'intervallo di 800 mV_{pp} (differenziale). La IF (frequenza intermedia) di uscita è uno stadio differenziale con uscita analogica progettato per funzionare a frequenze inferiori a 400 MHz. Tale mixer è basato sull'architettura cella gilbert.

5.3 Modalità di Funzionamento

L'AD9858 ha due modalità di funzionamento e sono:

- single tone
- frequency sweeping

per accedere ad una di queste modalità è necessario accedere a dei particolari registri che verranno descritti nel prossimo paragrafo.

5.3.1 Single Tone

In modalità singolo tono, l' AD9858 genera un segnale, o tono, alla singola frequenza desiderata. Questa frequenza è impostata dal valore caricato dall'utente nel chip FTW registro. Questa frequenza può variare tra 0 Hz e il valore leggermente inferiore alla metà della frequenza di campionamento DAC (SYSCLK). Il limite superiore pratico della gamma di frequenza fondamentale di un DDS è determinata dalla caratteristica del filtro esterno passa-basso, noto come il filtro di ricostruzione. Questo filtro ricostruisce il segnale analogico sinusoidale desiderato a partire dai valori in uscita del DAC. La fig. 5.4 mostra graficamente la transizione dalla condizione di default (0 Hz) ad una condizione User-defined (F1 Hz).

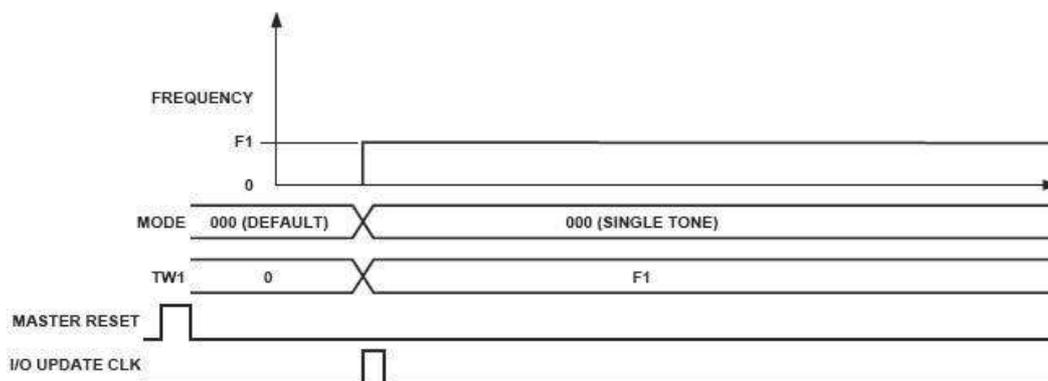


Tabella 5.4 - Transizione dalla condizione di default ad una condizione user-defined

Dopo aver inviato un impulso di Master Reset quindi si provvede a memorizzare il valore della Frequency Tuning Word 1 che, come tutti i dispositivi prodotti dalla Analog con tecnologia DDS, corrisponde al valore reso intero e convertito in numero binario da 32 bit risultante dalla seguente formula:

$$F_{TW1} = \frac{\text{frequenza desidera} \cdot 2^N}{\text{System Clock}}$$

dove N è pari a 32.

Il range di frequenza va da dc a ½ della frequenza di clock. E' bene tener presente che qualora vengano effettuate delle modifiche della frequenza mentre il dispositivo è in funzione tali cambiamenti saranno continui in fase. Cioè la “prima fase” del segnale alla nuova frequenza è riferita temporalmente all'ultimo valore della fase del segnale alla frequenza precedente.

5.3.2 *Frequency sweeping*

L'AD9858 ha la capacità di fornire uno sweep in frequenza automatizzato. Questo consente all'AD9858 di generare segnali per applicazioni radar o altro. La funzione sweep di frequenza viene implementata attraverso l'uso di un accumulatore di frequenza (da non confondere con l'accumulatore di fase). L'accumulatore di frequenza aggiunge ripetutamente una quantità incrementale alla FTW corrente, creando così nuovi valori istantanei di *frequency tuning words*, cambiando nel tempo la frequenza generata dal DDS. L'incremento di frequenza, o *step size*, viene caricato nella delta frequency tuning word (DFTW) register. La velocità con cui avviene l'incrementato di

frequenza è impostato dal *delta frequency ramp rate word* (DFRRW) *register*. Insieme, questi registri consentono all'AD9858 lo sweep da una frequenza iniziale fissata dalla FTW, verso l'alto o verso il basso, a una velocità desiderata e alla dimensione del passo di frequenza. Il risultato è uno sweep di frequenza o chirp lineare.

La funzione DFRRW effettua un conteggio alla rovescia, in cui viene decrementato il valore della DFRRW al tasso di $\text{SYSCLK} / 8$. Ciò significa che il più rapido aggiornamento della FTW si verifica quando il valore 1 viene caricato nel DFRRW e risultando quindi un incremento di frequenza a $1/8$ del tasso SYSCLK. Con un SYSCLK di 1 GHz, la frequenza può essere aumentata a una velocità massima di 125 MHz (DFRRW = 1). Il DFTW deve specificare se lo sweep di frequenza deve procedere verso l'alto o verso il basso dalla frequenza di partenza (FTW). Pertanto, il DFTW è espressa come un valore binario in complemento a due, in cui positivo indica sopra e negativo indica giù.

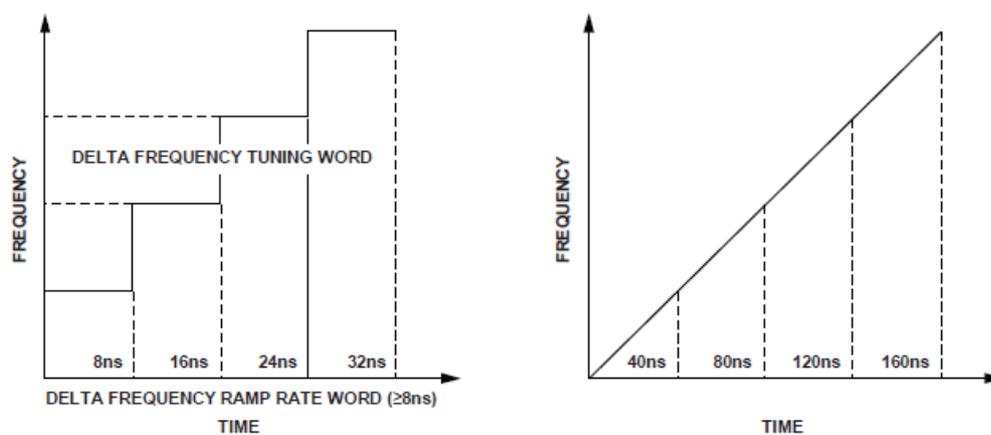


Tabella 5.5 - Grafici di frequenza in funzione del tempo per un dato profilo Sweep.

Per lo sweep in frequenza non esiste una funzione che lo blocchi automaticamente ma è necessario porre a zero il DFRRW. Per realizzare quanto detto l'utente deve calcolare l'intervallo di tempo necessario per raggiungere la frequenza finale e quindi emettere un comando per scrivere 0 nel registro in cui è contenuto il DFRRW. Il tempo necessario per uno sweep di frequenza è calcolato secondo la seguente relazione:

$$t = \frac{|f_F - f_s| \times 2^{34}}{SYSCLK^2} \times \frac{DFRRW}{DFTW}$$

dove:

T è la durata dello sweep in secondi.

f_s è la frequenza iniziale e vale:

$$f_s = \frac{FTW}{2^{32}} \times SYSCLK$$

f_F è la frequenza finale.

La dimensione del delta frequency step è data da:

$$\Delta f = \frac{DFTW \times SYSCLK}{2^{31}}$$

ricordando che il valore di DFTW è a complemento a due.

Il tempo tra ogni step frequency è dato da:

$$\Delta t = \frac{8 \times DFRRW}{SYSCLK}$$

Il valore della frequenza finale è determinato da:

$$f_F = f_s + t \times \frac{\Delta f}{\Delta t}$$

5.4 Sincronizzazione

La temporizzazione per l'AD9858 è fornita dall'utente tramite l'ingresso REFCLK. L'ingresso REFCLK è bufferizzato ed è la fonte per la generazione interna del SYSCLK. La frequenza di SYSCLK può essere uguale al clock fornito (REFCLK) o anche la metà (CFR [6]). L'ingresso REFCLK è in grado di gestire frequenze di ingresso fino a 2 GHz. Tuttavia, il dispositivo è progettato per una frequenza SYSCLK massima di 1 GHz. Pertanto, è obbligatorio attivare la funzione divisione per 2 del SYSCLK quando la frequenza di REFCLK è maggiore di 1 GHz.

SYSCLK è il clock per il DAC il quale passa attraverso un divisore per otto di frequenza per produrre SYNCLK disponibile sul pin SYNCLK, consentendo la sincronizzazione di hardware esterni con il clock interno del DDS AD9858. L'Hardware esterno che è sincronizzato con il segnale SYNCLK può quindi essere utilizzato per fornire la frequenza di update (FUD) del segnale all'AD9858.

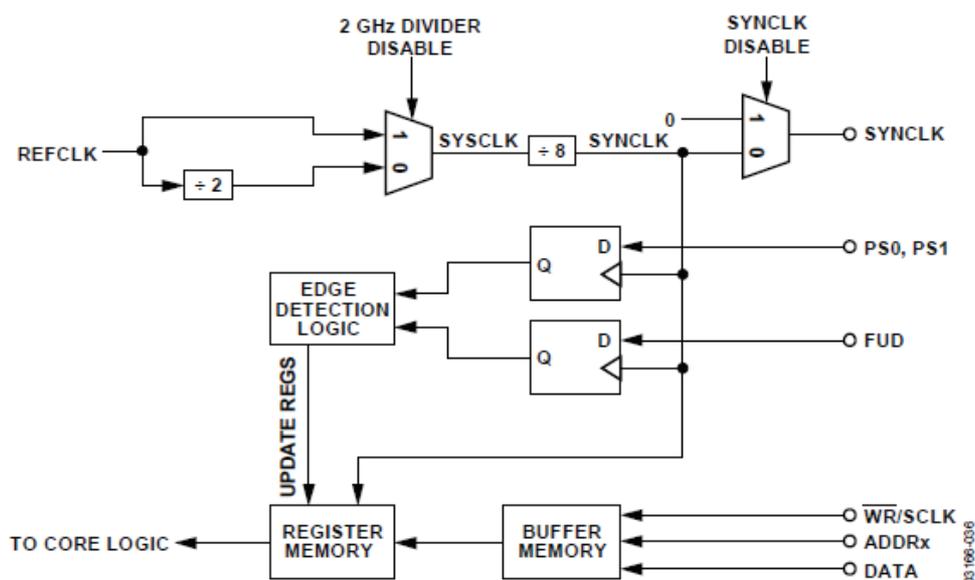


Figura 5.5 - I/O Synchronization Block Diagram

I segnali FUD e SYNCLK sono utilizzati per trasferire il contenuto dei registri interni a registri di memoria del dispositivo. La figura 5.5 mostra lo schema a blocchi della metodologia di sincronizzazione, mentre la Figura 5.6 mostra un diagramma di temporizzazione I / O di sincronizzazione.

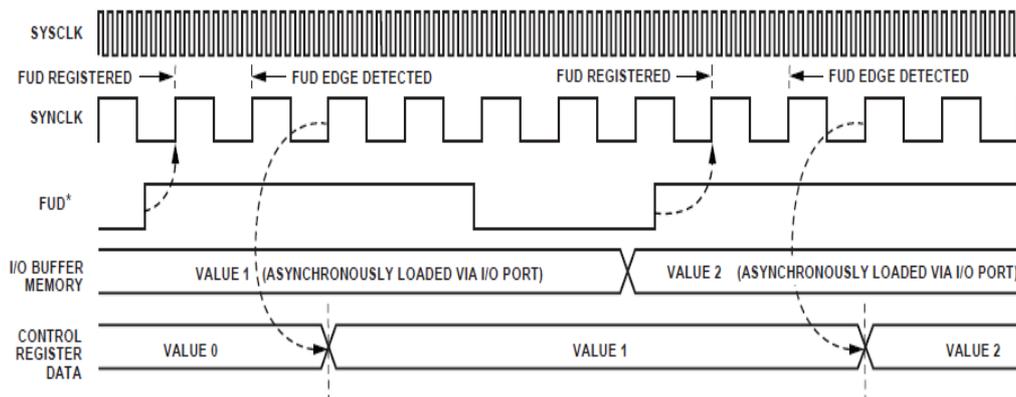


Figura 5.6 - I/O Synchronization Timing Diagram

5.5 I registri dell'AD9858

L'AD9858 dispone di 36 registri di controllo da 8 bit che permettono all'utente di programmare il dispositivo. A tale registri si può accedere sia in modo parallelo, mantenendo alto il pin 91 SP select, sia in modo seriale mantenendo invece tale pin basso. In tabella 5.3 è mostrato il layout dei registri.

Di seguito vengono descritti i registri utilizzati per la nostra applicazione.

Register Name	Address		(MSB)							(LSB)	Default Value	Profile	
	Ser	Par	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0			
Control function register (CFR)	0x00	0x00 [7:0]	Not used	2 GHz divider disable	SYNCLK disable	Mixer power-down	Phase detect power-down	Power-down	SDIO input only	LSB first	0x18	N/A	
		0x01 [15:8]	Freq. sweep enable	Enable sine output	Charge pump offset	Phase detector divider ratio (N) (see Table 10)	Charge pump polarity	Phase detector divider ratio (M) (see Table 11)			0x00	N/A	
		0x02 [23:16]	Auto Clr freq. accum	Auto Clr phase accum	Load delta freq. timer	Clear freq. accum	Clear phase accum	Not used	Fast lock enable	FTW for fast lock		0x00	N/A
		0x03 [31:24]	Frequency detect mode charge pump current (see Table 7)		Final closed-loop mode charge pump current (see Table 8)			Wide closed-loop mode charge pump current (see Table 9)				0x00	N/A
Delta freq. tuning word (DFTW)	0x01	0x04	Delta Frequency Word[7:0]									N/A	
		0x05	Delta Frequency Word[15:8]									N/A	
		0x06	Delta Frequency Word[23:16]									N/A	
		0x07	Delta Frequency Word[31:24]									N/A	
Delta frequency ramp rate (DFRRW)	0x02	0x08	Delta Frequency Ramp Rate Word[7:0]									N/A	
		0x09	Delta Frequency Ramp Rate Word[15:8]									N/A	
Frequency Tuning Word 0 (FTW0)	0x03	0x0A	Frequency Tuning Word 0[7:0]						0x00	0			
		0x0B	Frequency Tuning Word 0[15:8]						0x00	0			
		0x0C	Frequency Tuning Word 0[23:16]						0x00	0			
		0x0D	Frequency Tuning Word 0[31:24]						0x00	0			
Phase Offset Word 0 (POW0)	0x04	0x0E	Phase Offset Word 0[7:0]						0x00	0			
		0x0F	Not used	Phase Offset Word 0[13:8]					0x00	0			
Frequency Tuning Word 1 (FTW1)	0x05	0x10	Frequency Tuning Word 1[7:0]							1			
		0x11	Frequency Tuning Word 1[15:8]							1			
		0x12	Frequency Tuning Word 1[23:16]							1			
		0x13	Frequency Tuning Word 1[31:24]							1			
Phase Offset Word 1 (POW1)	0x06	0x14	Phase Offset Word 1[7:0]							1			
		0x15	Not used	Phase Offset Word 1[13:8]						1			
Frequency Tuning Word 2 (FTW2)	0x07	0x16	Frequency Tuning Word 2[7:0]							2			
		0x17	Frequency Tuning Word 2[15:8]							2			
		0x18	Frequency Tuning Word 2[23:16]							2			
		0x19	Frequency Tuning Word 2[31:24]							2			
Phase Offset Word 2 (POW2)	0x08	0x1A	Phase Offset Word 2[7:0]							2			
		0x1B	Not used	Phase Offset Word 2[13:8]						2			
Frequency Tuning Word 3 (FTW3)	0x09	0x1C	Frequency Tuning Word 3[7:0]							3			
		0x1D	Frequency Tuning Word 3[15:8]							3			
		0x1E	Frequency Tuning Word 3[23:16]							3			
		0x1F	Frequency Tuning Word 3[31:24]							3			
Phase Offset Word 3 (POW3)	0x0A	0x20	Phase Offset Word 3[7:0]							3			
		0x21	Not used	Phase Offset Word 3[13:8]						3			
Reserved	0x0B	0x22	Reserved, do not write, leave at 0xFF						0xFE	N/A			
		0x23	Reserved, do not write, leave at 0xFF						0xFF	N/A			

Tabella 5.3 - Layout registri AD9858

5.5.1 Frequency Tuning Word1

E' composto 32 bit ed è presente agli indirizzi paralleli 0A-0D Hex nel quale è memorizzata l'informazione relativa alla frequenza del

segnale da generare nel caso di Single Tone o alla frequenza di start nel caso di Chirp, mentre all'indirizzo seriale 03 Hex.

Il numero binario contenuto in questa word corrisponde al valore risultante dalla seguente formula:

$$F_{TW1} = \frac{F_{OUT} \cdot 2^N}{SYNCLK}$$

dove N è il numero di bit della Frequency Tuning Word 1 cioè 32.

5.5.2 Delta Frequency Word

E' anch'esso composto da 32 bit ed è presente agli indirizzi paralleli 04-07 Hex, mentre all'indirizzo seriale 01 Hex. In tale registro è memorizzata l'informazione relativa all'intervallo di frequenza corrispondente ad ogni "salto" del segnale chirp.

Il numero binario da memorizzare in questo registro corrisponde al valore risultante dalla seguente formula:

$$DFTW = \frac{F_{OUT} \cdot 2^N}{SYNCLK}$$

dove N è il numero di bit della Delta Frequency Word meno uno (quindi vale 31) perché il bit più significativo viene utilizzato per indicare il segno (si utilizza quindi il complemento a due).

5.5.3 Ramp Rate Clock

E' costituito da 24 bit agli indirizzi paralleli 08-09 Hex, nel caso seriale 02 Hex, nel quale vi è memorizzato un numero legato al periodo di tempo (periodo di step) che intercorre dal passaggio da una frequenza alla successiva nel segnale chirp.

$$Periodo\ di\ step = \frac{(N + 1) \cdot 2}{SYSCLK}$$

dove N è il valore memorizzato in tale registro. Il suo valore minimo è 1.

5.5.4 User Profile Registers

I registri user profile si compongono di quattro frequency tuning words e quattro phase offset words. Ogni coppia di registri di frequenza e fase forma un profilo configurabile dall'utente selezionando i pin user profile PS0 e PS1 (tabella 5.4). Ogni profilo ha la propria FTW consentendo all'utente di caricare diverse FTW. Questo rende possibile generare le diverse frequenze a velocità fino a 1/16 del SYSCLK in modalità single-tone.

PS1	PS0	Profile
0	0	0
0	1	1
1	0	2
1	1	3

Tabella 5.4 - User Profile

L'AD9858 fornisce anche 14-bit word offset phase (POW) per ogni profilo. Il valore di questo registro è un unsigned a 14-bit che rappresenta il proporzionale ($PO/2^{14}$) sfasamento da aggiungere al valore di fase istantanea. Ciò consente di regolare la fase del segnale di uscita con incrementi di fase di circa $0,022^\circ$. È possibile aggiornare il FTW e POW di qualsiasi profilo, mentre l' AD9858 funziona alla frequenza specificata da un altro profilo per poi passare al profilo contenente la frequenza appena caricato.

È anche possibile scrivere ripetutamente una nuova frequenza nel registro FTW di un profilo selezionato e passare alla nuova frequenza tramite il pin frequency update (FUD). Ciò consente di saltare a

frequenze arbitrarie, ma è limitato nella velocità con cui questo può essere ottenuto dovuto alla velocità della porta I / O (100 MHz in parallelo) e la necessità di trasferire diversi byte di dati per ogni nuova FTW.

6. GESTIONE DEL CHIP AD9858 MEDIANTE MICROCONTROLORE MICROCHIP

6.1 Introduzione

Il microcontrollore utilizzato per la gestione del dispositivo oggetto di questa tesi appartiene alla famiglia dei PIC, acronimo di Peripheral Interface Controller.

Questi particolari dispositivi integrano tutti i circuiti necessari a realizzare un sistema digitale programmabile, dispongono infatti di:

- una CPU (Central Processor Unit), che ha il compito di interpretare le istruzioni del programma;
- una memoria FLASH in cui vengono memorizzate permanentemente le istruzioni del programma da eseguire;
- una memoria RAM utilizzata per la gestione delle variabili del programma;
- una serie di linee di ingresso/uscita per permettere la comunicazione con altri dispositivi;
- una serie di dispositivi ausiliari quali clock, bus e contatori.

Molte applicazioni di controllo si avvalgono dell'utilizzo di questi dispositivi, perché consentono l'utilizzo di tutti gli apparati prima elencati a fronte di un costo non eccessivo oltre che un'occupazione di spazio molto ridotta.

Esistono svariati modelli di questo tipo di dispositivo, che si differenziano fra loro per il numero di linee ingresso/uscita, dimensioni di memoria, numero di registri interni, presenza in essi di

particolari periferiche, quali convertitori A/D oppure moduli per l'interfacciamento seriale/parallelo. Consentono quindi al progettista un'ampia scelta in base alle specifiche di progetto da seguire.

Tutti questi dispositivi hanno una frequenza di lavoro abbastanza ridotta, compresa solitamente fra i 4 MHz e i 20 MHz, ma sufficiente per gli scopi per cui sono impiegati. Inoltre garantiscono un consumo ridotto, che come è noto aumenta con l'aumentare della frequenza di lavoro del dispositivo stesso. Sono alimentati solitamente a 5 V ed hanno un assorbimento di corrente che varia da qualche decina di μA a qualche mA a seconda del modello.

Il PIC utilizzato in questo dispositivo è un PIC 16F877A il cui schema a blocchi dell'architettura interna è visibile in figura 6.1.

Il PIC 16F877A ha molte caratteristiche che lo accomunano ad un normale microprocessore, ma si differenzia da esso per una caratteristica fondamentale, adotta infatti un'architettura di tipo Harvard. Adottando questo tipo di architettura, la memoria in cui sono contenuti i dati e la memoria in cui sono contenuti i programmi sono separate, e si accede ad esse mediante bus separati. L'utilizzo di un'architettura Von Neumann, che contraddistingue i microprocessori, prevede l'utilizzo di un unico banco di memoria in cui sono presenti dati e programmi, a cui si accede mediante un unico bus. L'adozione di un'architettura Harvard consente quindi un incremento della banda rispetto ad un'architettura Von Neumann.

Il PIC 16F877A è dotato di una ALU (unità aritmetico-logica) che opera su dati a 8 bit, ed è in grado di effettuare operazioni quali somme, sottrazioni, shiftamento e operazioni logiche, mentre non è in grado, come invece lo sono altri dispositivi della famiglia PIC, di effettuare divisioni e moltiplicazioni. La ALU è inoltre direttamente connessa con il registro W, che ha la funzione di accumulatore.

Le linee di ingresso/uscita a disposizione del PIC 16F877A sono in totale 33, organizzate su cinque PORT, indicati con POTR A, PORT B, PORT C, PORT D e PORT E.

Il PORT A dispone di 6 linee configurabili sia come ingresso che come uscita. Il PORT B come il PORT C ed il PORT D dispongono ciascuno di 8 linee configurabili sia come ingressi che come uscite. Il PORT E dispone di 3 linee configurabili sia come ingressi che come uscite. Le varie linee sono denominate come segue:

- PORT A : RA0, RA1, RA2, RA3, RA4, RA5;
- PORT B : RB0, RB1, RB2, RB3, RB4, RB5, RB6, RB7;
- PORT C : RC0, RC1, RC2, RC3, RC4, RC5, RC6, RC7;
- PORT D : RD0, RD1, RD2, RD3, RD4, RD5, RD6, RD7;
- PORT E : RE0, RE1, RE2;

Per gestire le linee di ingresso/uscita da programma è possibile utilizzare due special function register per ciascun port che sono denominati rispettivamente TRISA e PORTA per il port A, TRISB e PORTB per il port B, TRISC e PORTC per il port C, TRISD e PORTD per il port D e TRISE e PORTE per il port E.

I registri TRIS determinano il funzionamento della singola linea come ingresso o come uscita a seconda del valore che viene scritto in essi,

mentre i registri PORT hanno il compito di determinare lo stato della linee in uscita o di riportare lo stato della linee in ingresso.

Il PIC 16F877A possiede anche uno STACK a 8 livelli che è facilmente gestibile dal firmware, grazie al quale è possibile effettuare fino ad 8 chiamate a subroutine annidate salvando gli indirizzi di ritorno correttamente.

Il PROGRAM COUNTER a 13 bit è un particolare registro il cui contenuto viene incrementato a intervalli di tempo regolari ed ha la funzione di cadenzare l'esecuzione del programma. Tramite esso è possibile indirizzare 8 k locazioni di memoria, cioè tutte quelle a disposizione del programmatore.

I registri Timer 0, Timer 1 e Timer 2 sono contatori direttamente configurabili dall'hardware del PIC, è possibile infatti far incrementare il valore al loro interno con una frequenza direttamente proporzionale alla frequenza di clock applicata al chip. La frequenza di incremento di questi registri speciali può essere modificata andando a programmare in maniera adeguata il prescaler.

Il PIC possiede inoltre due moduli per la connessione seriale con il mondo esterno, tramite i quali è possibile comunicare con svariati dispositivi. I due moduli in questione sono USART, (Universal Synchronous-Asynchronous Receiver-Transmitter) e MSSP.

Per una trattazione più dettagliata del Timer 2 e del modulo USART rimando al prossimo capitolo, dove verranno affrontate le funzionalità implementate nel firmware, che utilizzano appunto questi due dispositivi.

Altre caratteristiche che un PIC possiede, perché utili per applicazioni in real-time, sono il Watch Dog Timer, il Power Down Mode e il meccanismo degli Interrupt.

Il Watch Dog Timer è un oscillatore interno al PIC e completamente indipendente dal resto dell' hardware ed ha il compito di rilevare eventuali blocchi della CPU per poi resettare il PIC e far così in modo che l'esecuzione del programma riprenda normalmente. Il reset ha luogo quando il timer va in overflow quindi per evitare questa situazione e necessario, all'interno del programma che gira sul nostro dispositivo, prevedere l'azzeramento periodico del timer stesso.

Il Power Down Mode, denominato anche Sleep Mode è un particolare stato di funzionamento del PIC che mi consente di diminuire l'assorbimento di corrente da parte del dispositivo quando questo non sta lavorando, perché per esempio è in attesa di ricevere un segnale a seguito di un evento esterno. Far lavorare il dispositivo in questa modalità, può consentirmi un assorbimento di corrente fino a mille volte inferiore rispetto alle normali condizioni di funzionamento.

Il meccanismo degli Interrupt dà la possibilità di gestire via software tutte quelle situazioni impreviste che si vengono a creare in maniera asincrona rispetto all'esecuzione del programma principale. Ogni possibile interruzione, fra l'ampia gamma di cui è dotato il PIC 16F877A, è associata a due flag di cui uno ne identifica l'origine e l'altro dà la possibilità di abilitare/disabilitare tale interruzione.

6.2 MPLAB : ambiente di sviluppo integrato

MPLAB IDE è l'ambiente di sviluppo, fornito gratuitamente da Microchip, che è stato utilizzato per realizzare il firmware del dispositivo oggetto di questa tesi.

Questo ambiente di sviluppo comprende i seguenti tool:

- un editor per la scrittura del codice assembly;
- un linker per la generazione del file eseguibile utilizzato per programmare il PIC;
- un simulatore che permette il debugging del codice.

Tramite l'utilizzo di MPLAB è quindi possibile scrivere, compilare e linkare il codice assembler per tutti i PIC che vengono prodotti da Microchip.

Il simulatore è un software che gira su PC con il compito di simulare l'esecuzione del programma come avverrebbe nella realtà. Ovviamente non lavora in tempo reale perché la velocità di simulazione dipende dalla velocità del PC su cui gira, dalla complessità del codice e dal numero di processi in esecuzione. Comunque tramite l'utilizzo del simulatore è possibile controllare il flusso logico del programma da noi scritto, l'evoluzione delle variabili e dello stato dei registri mentre il programma stesso è in esecuzione.

Per poter testare nelle normali condizioni di utilizzo ed in tempo reale il dispositivo è possibile utilizzare i tool hardware, anch'essi forniti da Microchip, che sono la demo-board e la scheda di interfaccia con i loro driver. In questa ipotesi di verifica del dispositivo è necessario collegare lo stesso alla demo-board, per poi far comunicare quest'ultima con il PC tramite la scheda di interfaccia. I segnali di

comunicazione fra PC e PIC possono essere gestiti tramite programmi messi a disposizione da Microchip.

6.3 Programmazione del PIC

I microcontrollori della famiglia PIC adottano un'architettura di tipo RISC (Reduced Instruction Set Computer), che è formata da un set di istruzioni ridotto, tramite le quali è possibile eseguire operazioni semplici che hanno tempi di esecuzione simili.

Questo set di istruzioni è suddiviso in tre categorie di istruzioni:

- byte-oriented : dedicate a operazioni logiche-aritmetiche sui byte senza l'uso di costanti di tipo numerico (ad esempio copia da registro sorgente a registro destinazione);
- bit-oriented : operazioni di test e set eseguite su singoli bit;
- literal and control operations : operazioni logiche-aritmetiche tra registri e costanti di tipo numerico, chiamate di sottoprogrammi e ritorni da sottoprogrammi.

Ogni singola istruzione ha una lunghezza pari a 14 bit ed è divisa in un opcode, che fornisce indicazioni sul tipo di istruzione ed una parte contenente uno o più operandi.

In figura 6.3 si può vedere la struttura dei singoli tipi di istruzione.

Per rendere il codice il più leggibile possibile si possono utilizzare alcune direttive, che sono dirette al compilatore e non fanno parte del set di istruzioni del PIC.

Le più utili sono:

- EQU : consente di definire costanti numeriche simboliche, cioè consente di dare un nome ad una particolare locazione di memoria o ad una costante;

- **#DEFINE** : consente di assegnare nomi simbolici ad una particolare istruzione o ad una parte di essa;
- **MACRO** : consente di assegnare un nome simbolico ad un insieme di più istruzioni;
- **CODE** : fornisce indicazione al compilatore su dove vada scritta in memoria programma l'istruzione successiva alla direttiva stessa.

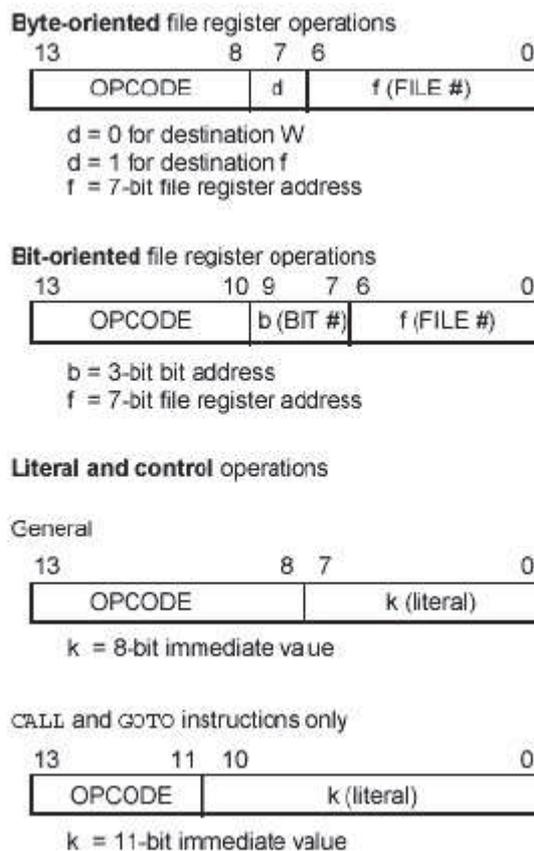


Figura 6.3- struttura istruzioni del PIC16F877A

Per ottenere il PIC programmato oltre alla conoscenza del set di istruzioni, di cui è stata fatta una sommaria descrizione in precedenza, bisogna disporre di un editor per realizzare il file sorgente, che avrà estensione .ASM. Una volta scritto il sorgente, si effettua la traduzione del sorgente stesso in opcode, tramite l'utilizzo del

compilatore fornito da Microchip. Il compilatore però richiede anche, come si può vedere in figura 4.2, un file con estensione `.INC`, che prende il nome di header. Questo file contiene alcune definizioni che dipendono dal chip adottato.

In fase di compilazione il compilatore genera file con nome uguale al modulo sorgente da cui derivano, ma che presentano estensioni diverse.

Viene generato un file con estensione `.HEX`, che è il file contenete gli opcode e che andrà caricato sulla memoria programma del PIC ed inoltre vengono generati altri due file, uno con estensione `.LST` ed uno con estensione `.ERR`. Il file con estensione `.LST` contiene il sorgente assembler e la corrispondente traduzione in opcode. Ha una certa utilità, infatti tramite esso è possibile verificare i processi che sono avvenuti in fase di compilazione.

Il file `.ERR` contiene la lista degli errori di compilazione e la relativa riga del sorgente a cui sono riferiti.

Il file da utilizzare per programmare il nostro PIC è quindi il `.HEX`, che a differenza di quanto ci si potrebbe aspettare, non è un file in formato binario, ma è codificato in un particolare formato che trasforma un file binario in un formato ASCII.

A questo punto dobbiamo inserire il nostro file sulla memoria programma. Per far questo si utilizza un programma fornito da Microchip che è in grado di leggere e trasferire al PIC il file con estensione `.HEX`.

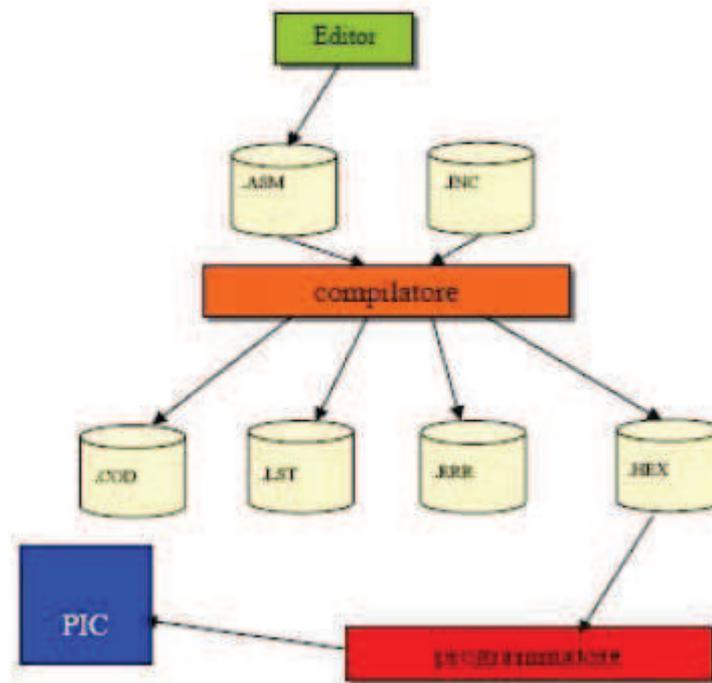


Figura 6.4 - Dall'editor al PIC programmato

6.4 Gestione del sincronismo

In questo paragrafo viene descritto come si è gestita la programmazione dell'AD9858. L'obiettivo era quello di ottenere un tono a 350 MHz, con un clock in ingresso di 800 MHz. Come descritto precedentemente è stato utilizzato un microcontrollore PIC16F877A della Microchip per il caricamento dei valori di funzionamento richiesti nei registri dell'AD9858.

In figura 6.5 è mostrato il diagramma di flusso secondo il quale lavora il firmware del microcontrollore.

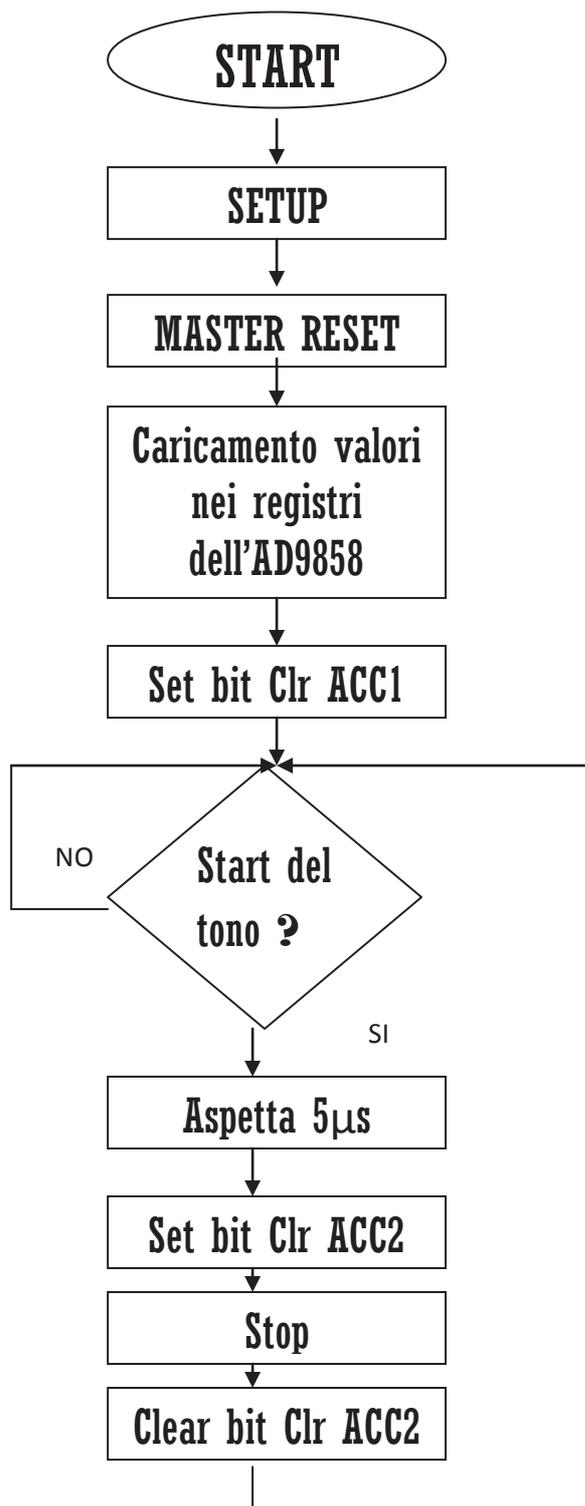


Figura 6.5 - Diagramma di flusso del firmware caricato sul PIC16F877A

Dopo aver alimentato il pic si provvede ad impostare, nelle fase di setup, la frequenza di clock di funzionamento e si stabilisce quali porte verranno utilizzate come ingresso e quali come uscite.

Dopo aver fatto ciò si invia un impulso di Master Reset che inizializza i registri dell'AD9858 a dei valori di default.

Si provvede quindi a caricare nei registri del sintetizzatore frequenziale i valori richiesti per aver un single tone di durata pari a 5 µs con una frequenza di 3500MHz.

Si setta alto il bit Clr ACC1 dell'AD9858 in modo che ad ogni impulso di I/O Update clock parta il single tone.

Un impulso di start esterno farà partire il single tone e al termine dei 5 µs si provvede a fermare tale segnale inviando un impulso di stop.

Il test del segnale di start viene eseguito sulla porta C4 mentre il segnale di stop viene dato dalla porta C7 .

Il segnale di stop deve pervenire solo dopo aver alzato il bit Clr ACC2 dell'AD9858. In questo modo al prossimo impulso di start il segnale ripartirà con fase nulla.

Viene di seguito riportata una tabella che mostra i ritardi introdotti dai componenti dall'AD9858 dichiarati dal produttore.

PIPELINE DELAYS (Temperature: 25°C)		
	Ritardo min(System Clock cycles)	Ritardo max(System Clock cycles)
FTW/POW to DAC Output	83	83
DFTW to DAC Output	99	99

Tabella 6.1- Ritardi introdotti dall'AD9858

Dalle misure effettuate mediante oscilloscopio digitale considerando un Clock a 800 MHz si è osservato un ritardo tra l'impulso di I/O Update (di start) e la partenza effettiva del tono di 287 ± 3 ns.

Se si utilizza un segnale di start a onda quadra è importante che tale segnale sia alto per un tempo inferiore a 5 μ s per non compromettere il funzionamento del sintetizzatore frequenziale.

6.5 Collegamento dei dispositivi

In figura 6.6 è mostrato il collegamento tra il PIC16F877A e l'AD9858:

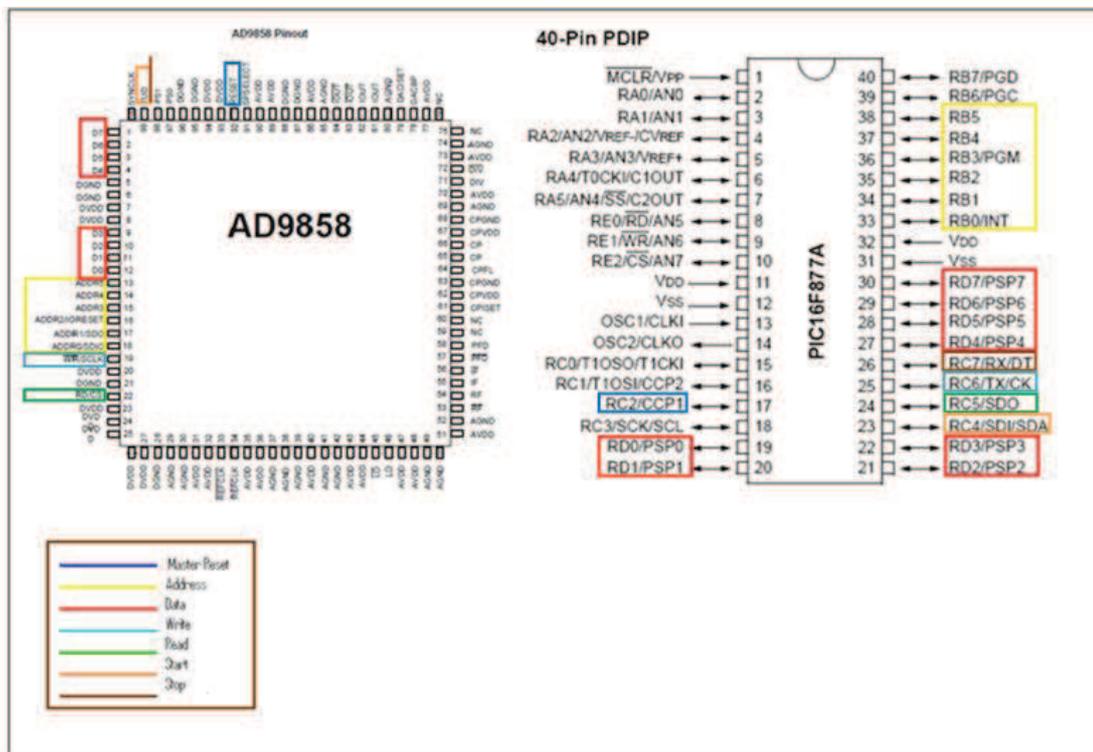


Figura 6.6 - Collegamento tra i dispositivi

In tabella 6.2, è mostrato il collegamento dei pin.

I segnali di Start e di Stop del single tone vengono collegati entrambi all'ingresso del FUD dell'AD9858 . Il segnale di start viene fornito dall'esterno ed è collegato sia alla porta C4 del microcontrollore, impostata come ingresso, sia al FUD. Il segnale di stop è invece fornito dalla porta C7 del microcontrollore anch'essa collegata FUD ma impostata come uscita. . E' quindi necessario utilizzare dei diodi seguendo il seguente schema di collegamento mostrato in figura 6.7.

Tabella dei collegamenti

PORTA	PIN	Descrizione
B	0-7	Address Pins
D	0-7	Data
C	2	Master Reset
C	4	Start del Tono
C	5	Read (0-attivo)
C	6	Write (0-attivo)
C	7	Stop del Tono

Tabella 6.2- Collegamento dei pin.

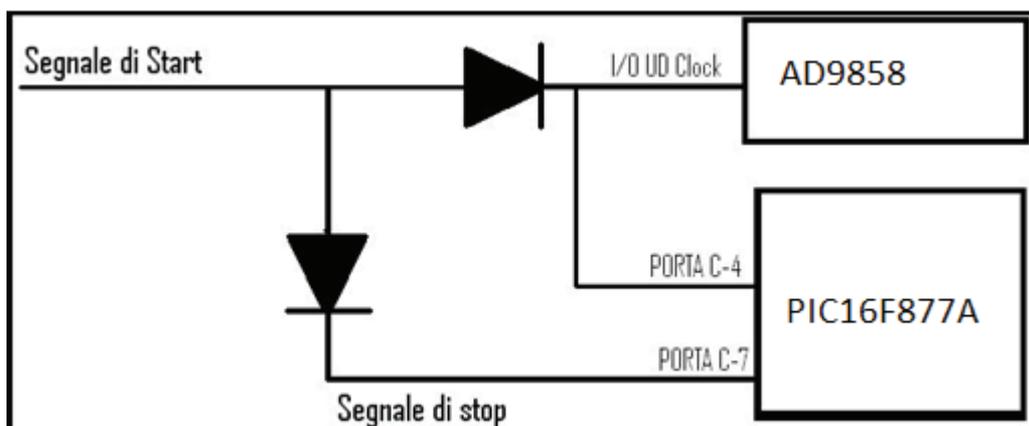


Figura 6.7 - Collegamento delle porte C4 e C7 all'ingresso del FUD.

I diodi utilizzati devono essere diodi veloci per non avere ritardi considerevoli tra l'invio del segnale di start e la generazione del single tone (è accettabile un ritardo massimo di 1 ns).

6.6 Firmware in Borland C++

Il software è sviluppato in linguaggio Borland C++ Builder 6.0 IDE. Il firmware è suddiviso in tre parti oltre al programma principale (main):

- Inizializzazione del dispositivo, dove vengono definita la piedinatura del dispositivo e il protocollo di comunicazione utilizzato(RS232);
- Settaggio dei registri del DDS;
- La definizione della funzione tono la quale richiede frequenza, fase e il profilo da selezionare che di default è quello appropriato per generare il single tone.

Il listato in Bordland C++ è il seguente:

```

/*
*****
*****
*
* Autore: Di Martino Luigi   Matricola A17/019
* Versione a 800MHz
*
*****
*****
*/

#include <test.h>

#define SYSCLK  800.00
#define F_OUT  350.00

unsigned int32 ftw;
int8 reg_0, reg_1, reg_2, reg_3;

#INT_EXT FAST
void fast_isr()

```

```

{
  disable_interrupts(GLOBAL);

  delay_cycles(5);    // Durata del single tone

  output_high(PICRST); // Resetta DDS
  delay_us(80);       // Mantiene Reset per eliminare rimbalzi
  output_low(PICRST);

  enable_interrupts(GLOBAL);
}

void main()
{
  printf("Tone Generator\n\r");
  printf("giugno 2012\n\r");
  delay_ms(100);
  init_hardware();

  /*******
  *****/

  //   Modalità Single Tone
  /*******
  *****/

  reg_0 = 0b00011000;
  reg_1 = 0b00000000;
  reg_2 = 0b00000000;
  reg_3 = 0b00000000;
  set_reg (reg_0, reg_1, reg_2, reg_3);

  printf("\n\r Registro_0 = %x",reg_0);
  printf("\n\r Registro_1 = %x",reg_1);
  printf("\n\r Registro_2 = %x",reg_2);
  printf("\n\r Registro_3 = %x",reg_3);

  ftw = (pow(2,32) * F_OUT / SYSCLK) ;

  tono(ftw,0,0);
  output_high(PICFUD);

```

```

output_low(PICFUD);

while (TRUE) {

}

}

/*
*****
*****
*
* init_hardware
*
* Inizializza le modalità operative
*
*****
*****
*/

void init_hardware(void)
{

    enable_interrupts(INT_EXT);
    disable_interrupts(GLOBAL);

    output_high(PICRST);
    output_low(PICRST);

    output_low(PS0);
    output_low(PS1);

    delay_us(20);
    // enable_interrupts(GLOBAL);

}

/*
*****
*****
*

```

```

* Tono
*
* Setta i DDS per la generazione di un tono
* accetta in ingresso un dato a 32 bit per la frequenza, uno a 16 per la
fase
*
*****
*****
*/

```

```

void tono (unsigned int32 frequenza, unsigned int16 fase, int8 profilo)
{
    unsigned int8 indirizzo,i,x;

    if (profilo == 0) { indirizzo = 0x0A;}
    else if (profilo == 1) { indirizzo = 0x10;}
    else if (profilo == 2) { indirizzo = 0x16;}
    else if (profilo == 3) { indirizzo = 0x1C;}
    else if (profilo >3 )
        { printf("Errore profilo \n\r");
          profilo=0;
          indirizzo = 0x0A;
        }

    for (i=0; i<4; i++) {
        x=((frequenza >> (i*8)) & 0xff); // Equivalente a Make8(dati,i)
        output_D(x);
        output_A(indirizzo);           // Scrive registro indirizzo
        output_low(WR);
        output_high(WR);                // Abilita segnale Write
        indirizzo++;
    }

    for (i=0; i<2; i++) {
        x=((fase >> (i*8)) & 0xff);    // Equivalente a Make8(dati,i)
        output_D(x);
        output_A(indirizzo);           // Scrive registro indirizzo
        output_low(WR);
        output_high(WR);                // Abilita segnale Write
        indirizzo++;
    }
}

```

```

}

/*
*****
*****
*
* Set_reg
*
* Setta i Control Registers del DDS
* accetta in ingresso 4 byte (reg_0, reg_1, reg_2, reg_3)
*
*****
*****
*/

void set_reg (int8 r0, int8 r1, int8 r2, int8 r3)
{
    output_A(0x00);    // Setta Control register 0
    output_D(r0);
    output_low(WR);
    output_high(WR);  // Scrive register 0

    output_A(0x01);
    output_D(r1);
    output_low(WR);
    output_high(WR);

    output_A(0x02);
    output_D(r2);
    output_low(WR);
    output_high(WR);

    output_A(0x03);
    output_D(r3);
    output_low(WR);
    output_high(WR);
}

```

```
*****
*****
*
* Test.h
*
*
*****
*****
*/

#include <16F877.h>

#define HIGH_INTS=TRUE

//device adc=10
#include <delay(clock=25000000)>
#include <fuses EC, NOPROTECT, NOWDT, NOLVP, NOBROWNOUT,
NOWRT, NOWRTD>
#include <rs232(baud=38400,xmit=PIN_C3,rcv=PIN_C7)>

#include <bootloader.h>
#include <ctype.h>
#include <float.h>
#include <math.h>
#include <input.c>
#include <stdlib.h>
#include <stdlibm.h>
#include <string.h>

#define PICRST PIN_C0
#define PICFUD PIN_C4
#define WR PIN_C6
```

6.7 Test sul dispositivo

La fase di test resta l'ultima operazione da realizzare per il completamento del progetto. Verificare il funzionamento del sistema, valutando se l'AD9858 risponde in modo appropriato ai comandi ricevuti dal PIC16F877A, è un passaggio fondamentale per la realizzazione di un dispositivo che possa garantire buoni margini di affidabilità.

Gli strumenti, utilizzati per l'implementare tale fase, sono:

- Analizzatore di spettro HP 8563E;
- Un generatore di clock HP 83620B, con il quale generiamo il clock a 800 MHz necessario al DDS;
- PC con sistema operativo Windows XP dotato di una porta seriale per comunicazione con protocollo RS232. Inoltre sul PC è installato un software fornito dall'HP[®] grazie al quale è possibile visualizzare a video e acquisire in formato jpeg le immagini dell'analizzatore di spettro.
- Alimentatore HP E3630A necessario per alimentare l'AD9858
- Una demo board contenente l'AD9858
- Il PIC16F877A anch'esso posizionato su una demo board.
- GPIB (*General Purpose Interface Bus*) per il collegamento PC-Analizzatore di spettro.

La figura seguente mostra lo schema a blocchi del sistema di test:

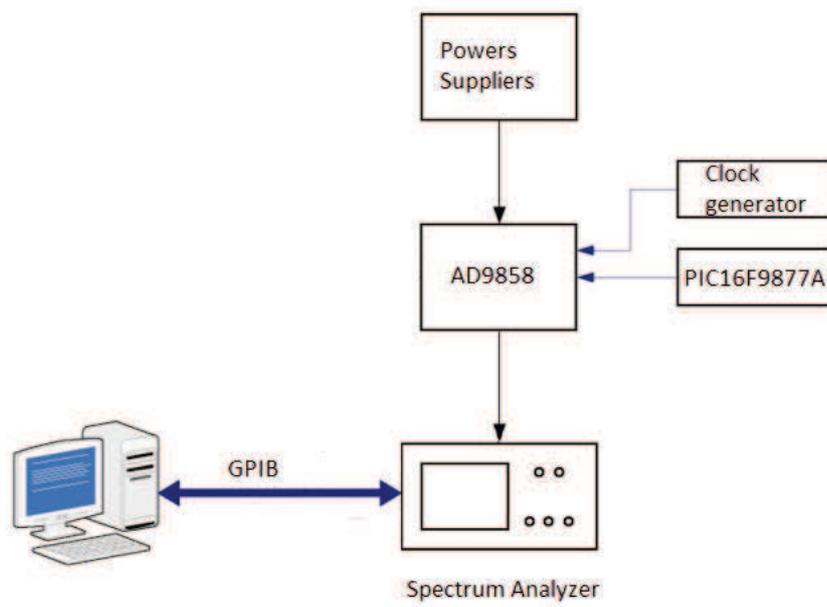


Figura 6.8 - Test Set-up

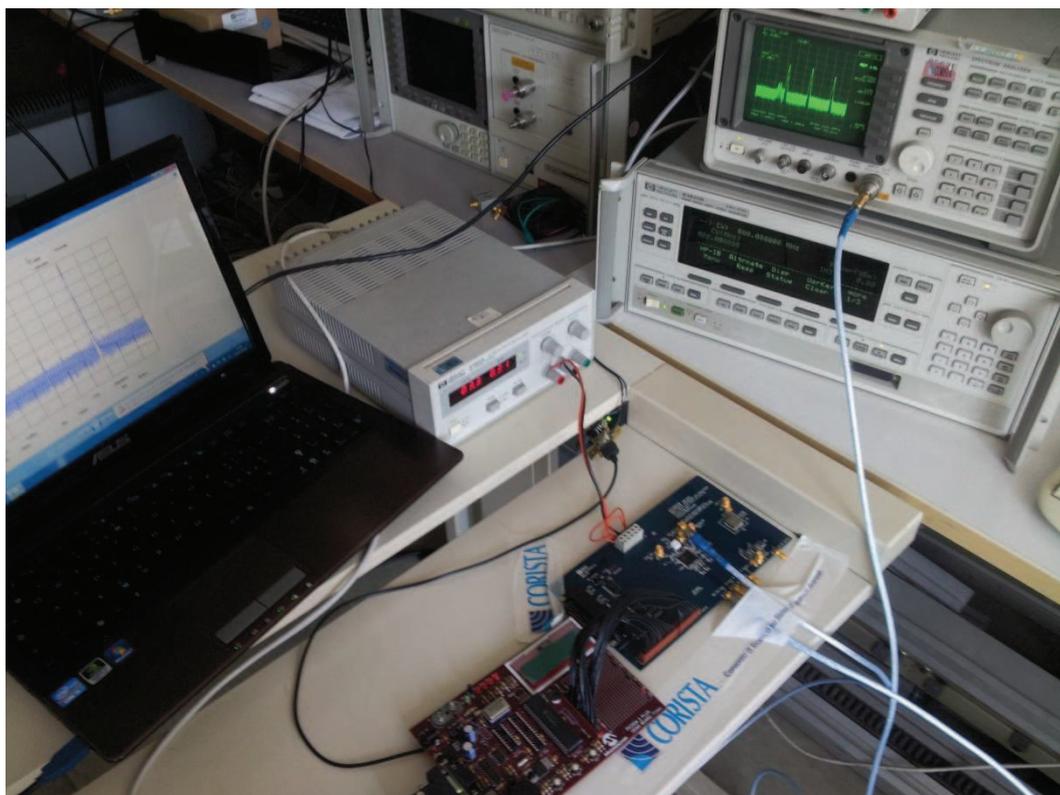


Figura 6.9 - Foto della strumentazione utilizzata in fase di test

6.7.1 Test report

Come descritto nel paragrafo 6.4, l'obiettivo è quello di ottenere un tono a 350 MHz, con un clock in ingresso di 800 MHz. Impostato il clock generator al valore di specifica il tono in uscita dal dispositivo si presenta come in figura:

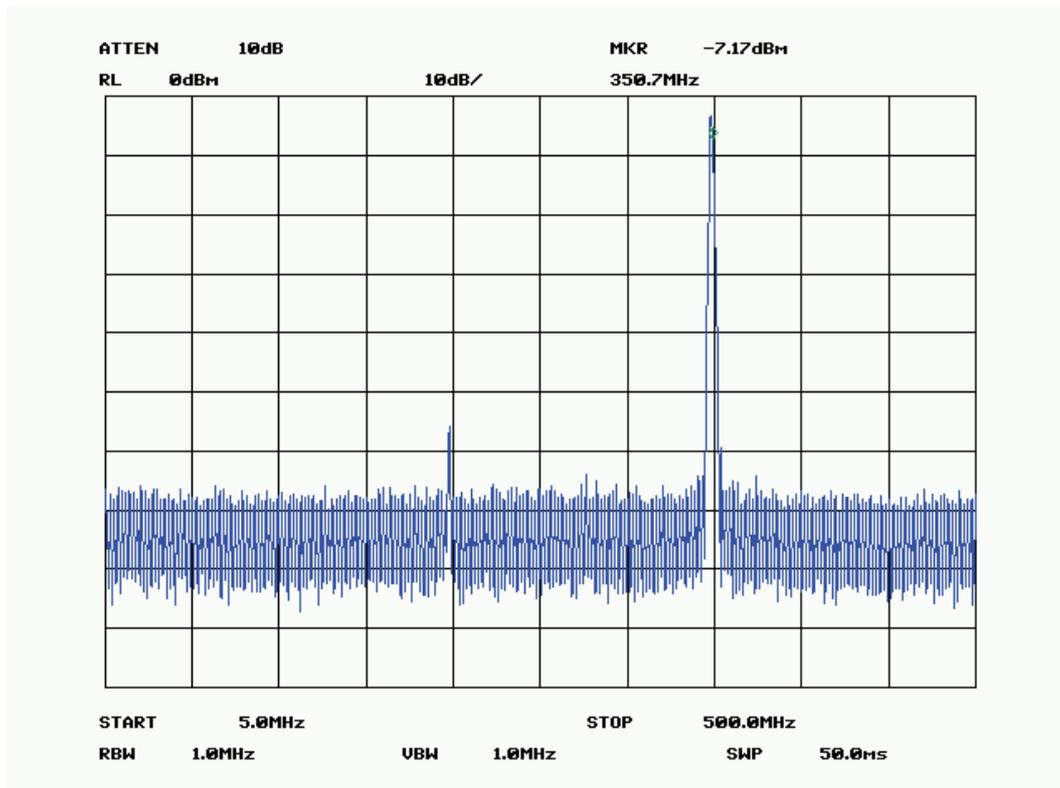


Figura 6.10 - Tone a 350MHz.

Nella figura seguente è invece possibile valutare lo SFDR(*spurious free dynamic range*) definito come il rapporto tra il valore efficace del segnale rispetto a quella maggiore componente di segnale spurio:

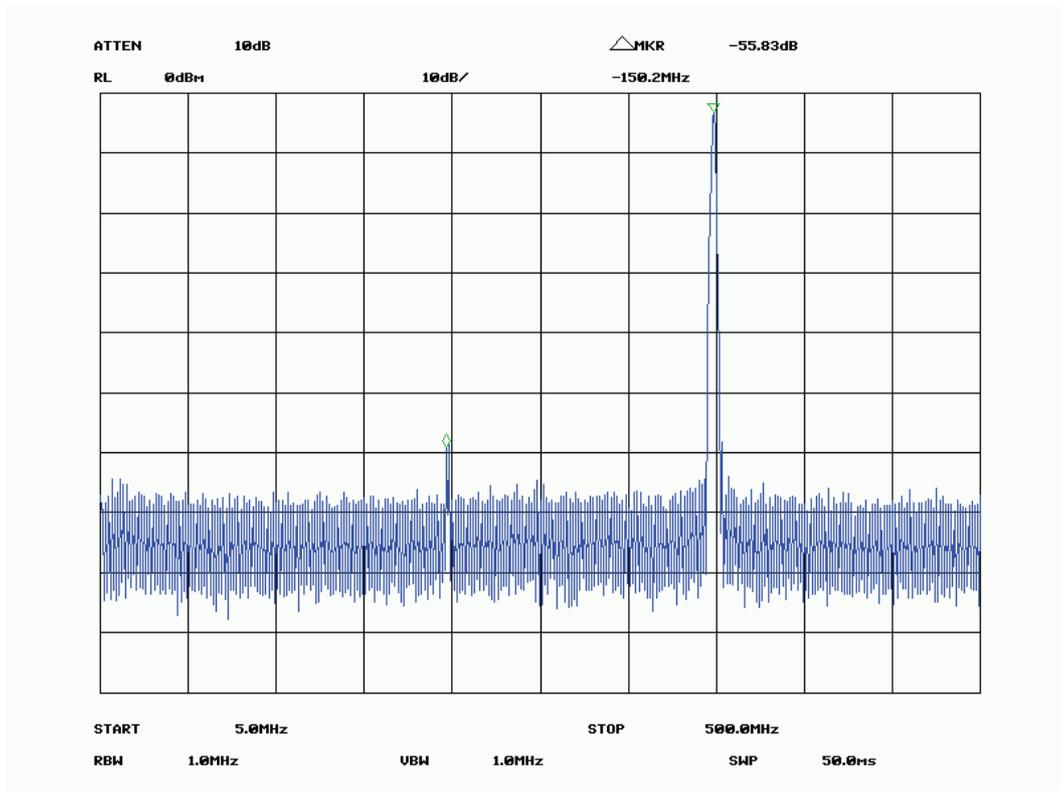


Figura 6.11 - SFDR con tono a 350 MHz.

Il valore dello SFDR con un clock in ingresso di 800MHz e un tone generatore a 350MHz risulta essere pari a -55.03dB, dove la maggiore componente di segnale spurio si trova a 200MHz.

La figura seguente mostra invece un tono a 150MHz:

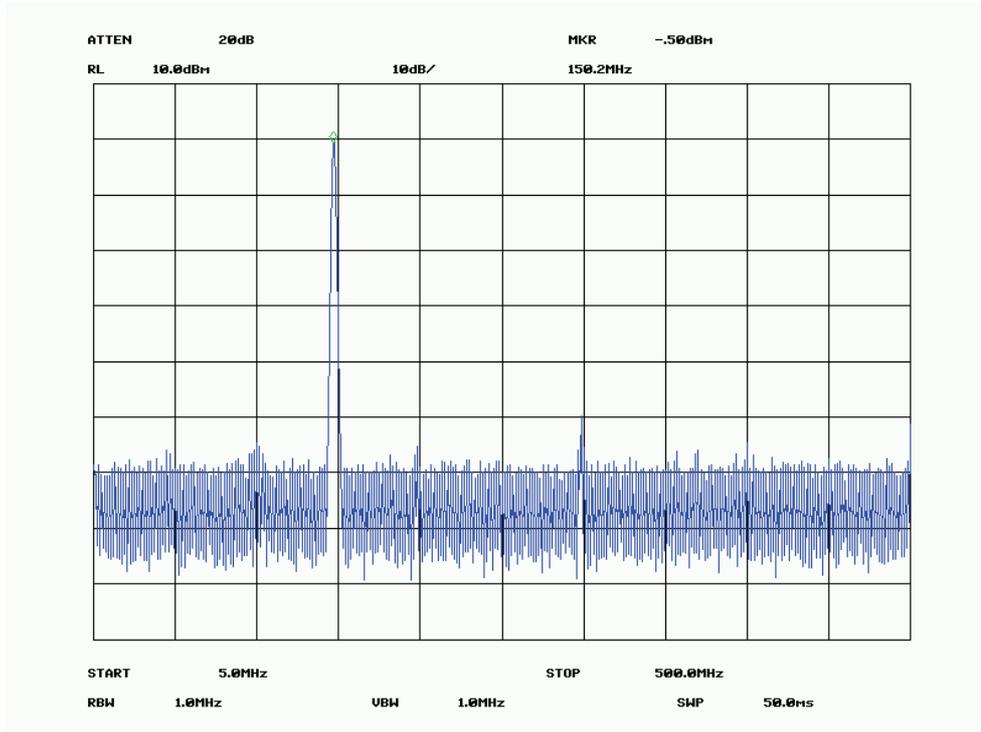


Figura 6.12 - tono a 150MHz.

Da tutte le prove di laboratorio effettuate si è potuto verificare che il chip genera le frequenze desiderate con le specifiche conformi a quanto dichiarato dal costruttore.

CONCLUSIONI

Il lavoro di tesi svolto presso il consorzio di ricerca Co.Ri.S.T.A è stato impostato per **progettare, realizzare e testare** un sintetizzatore di frequenze per applicazioni radar.

Il lavoro di tesi svolto è stato strutturato in due fasi:

- nella prima fase si è proceduto ad analizzare le tecniche di sintesi digitale e analogica, per poi passare ad un'analisi dettagliata della sintesi digitale diretta effettuando anche un'analisi di mercato valutando in termini di prestazioni e di costo i dispositivi prodotti dall'Analog Device leader in tale settore.
- nella seconda fase invece il lavoro è stato impostato con lo scopo di individuare, analizzare e studiare tutte le problematiche inerenti la gestione del sincronismo di un sintetizzatore frequenziale AD9858 mediante l'utilizzo di microcontrollori.

Lo studio effettuato, sulle tecniche di frequenza, ha evidenziato che la soluzione DDS presenta molti vantaggi rispetto ad un sintetizzatore che impiega un circuito PLL ed in particolare:

- permette di ottenere una risoluzione in frequenza dell'ordine del milli-Hertz.
- un'elevatissima velocità nella variazione della frequenza di uscita in quanto il sistema non richiede nessun tempo di assestamento nel cambiare la frequenza, infatti si va sull'ordine dei nanosecondi o poche decine di microsecondi.

- Tutti i cambiamenti di frequenza sono completati automaticamente con una variazione continua della fase, in questo modo la nuova frequenza generata riprende esattamente dall'ultimo valore di fase della frequenza precedente.
- elimina la necessità di una sintonizzazione manuale del sistema e di aggiustamenti dovuti all'età del componente e alle variazioni di temperatura come succede nelle soluzioni di sintesi analogica.
- l'interfaccia digitale di controllo dell'architettura DDS facilita un ambiente in cui i sistemi sono controllati da remoto e ottimizzati sotto il controllo di un processore.

Per quanto riguarda l'analisi di mercato, i dati raccolti hanno evidenziato che la scelta ottimale risultata essere l'AD9858 avendo come specifica ottimizzare i tempi di programmazione del dispositivo e nessun vincolo sul consumo di potenza e sulla risoluzione.

Per quanto concerne la gestione del sincronismo, nelle prove effettuate si è scelto di utilizzare il microcontrollore PIC16F877A in quanto questo permetteva di svolgere test su tutte le funzionalità del sintetizzatore.

Dalla fase di test realizzata è stato verificato il funzionamento del sistema; in particolare l'AD9858 risponde in modo appropriato ai comandi ricevuti dal PIC16F877A, garantendo buoni margini di affidabilità.

Possiamo quindi affermare che, al termine di tutte le prove di laboratorio effettuate, il chip genera le frequenze desiderate con le specifiche conformi a quanto dichiarato dal costruttore.

BIBLIOGRAFIA

- [I] Roger J. Sullivan, "Microwave Radar: Imaging and advanced concepts" Artech House Publishers*
- [II] G. Franceschetti, R. Lanari, "Synthetic aperture radar processing", CRCPress, 1999*
- [III] Skolnik, "Introduction to Radars Systems".*
- [IV] M. Massironi, "Il telerilevamento satellitare - Sensori attivi ed Envisat"*
- [V] J.C. Curlanderand, R.N. McDonough, "Synthetic Aperture Radar, Systems & Signal Processing", Wiley-IntersciencePub,1991.*
- [VI] Elachi, "Introduction to the physics and techniques of Remote Sensing"*
- [VII] John C. Curlander, Robert N. McDonough, "Synthetic aperture radar", Wiley Interscience*
- [IX] R. Bamler, B. Schattler, "SAR Data Acquisition and Image formation".*

[X] R. K. Raney, "Considerations for SAR image quantification unique to orbital system", *IEEE transaction on geoscience and remote sensing*.

[XI] Borriello, "Focalizzazione dati X-SAR con algoritmo Chirp Scaling in vista della simulazione della missione Cosmo-Skymed"

[XII] Vingione, "Analisi delle prestazioni e progetto preliminare del processing del radar di cassini in modalità altimetro"

[XIII] G. Fornaro, F. Serafino, F. Soldovieri, "Focalizzazione SAR 3-D con configurazione multipassaggio", Istituto per il Rilevamento Elettromagnetico dell'Ambiente (IREA), Consiglio Nazionale delle Ricerche.

[XIV] S. Ponte, "Sistemi Aerospaziali di Telerilevamento"

[XV] www.analog.com/dds

[XVI] www.corista.unina.it

[XVII] www.unipi.it

[XVIII] www.unisa.it

[XIX] www.sapere.it

[XX] www.microchip.com

[XXI] <http://www.aleniaspazio.com/index.htm>

[XXII] <http://www.nasa.gov/jpl>

[XXIII] <http://www.wikipedia.com>

Ringraziamenti

A conclusione di questo lavoro, voglio ringraziare il centro di ricerca CO.RI.S.T.A. per avermi consentito di realizzare la tesi di laurea in un ambiente stimolante e produttivo. Un sentito ringraziamento lo rivolgo, in particolare, al prof. Zeni, all'Ing. Ciofaniello per la preziosa guida umana, prima ancora che professionale, che mi hanno saputo dare nel corso di questi ultimi mesi.

Un ringraziamento va a Gianfranco Palmese e a Dario Califano, splendidi tutor da cui apprendere la loro preziosa esperienza.

Un ringraziamento particolare al direttore del Co.Ri.S.T.A. Doroteo Adirosi per l'opportunità che mi ha concesso.

Un primo speciale pensiero è destinato alla mia famiglia ed, in particolare, ai miei genitori: il loro continuo sostegno e la fiducia che hanno sempre riposto in me sono stati l'ingrediente fondamentale per raggiungere questo traguardo, senza mai perdere di vista le priorità fondamentali.

Un ringraziamento speciale ad Alessandra per tutto l'affetto ed il supporto che mi ha dato durante quest'ultimo periodo, non potrei immaginare di incontrare una persona migliore.