

VII. PRAKTIKUM 6— Rafinasi offset Pewaktu (timing) dan Offset Fekkuensi

VII.1 PENDAHULUAN

Parameter kasar alignment pewaktu dan alignment frekuensi telah dicari dalam rangkaian proses Praktikum yang lalu (PRAKTIKUM 5). Proses korelasi yang sama dengan PRAKTIKUM 5 dilakukan untuk memperbaiki parameter tersebut dengan menggunakan langkah pergeseran vernier, namun pada proses yang lebih vernier/fine (teliti) ini mempergunakan deretan sinyal di domain waktu saja. Pada PRAKTIKUM 5 ditunjukkan bahwa metrik kinerja terhadap pengaturan pewaktu dan frekuensi berupa koefisien **korelasi-paket** dari spektrum frekuensi tiap blok simbol. Selanjutnya dalam PRAKTIKUM 6 ini, metrik kinerja masih berdasarkan koefisien **korelasi-paket** namun perhitungan **korelasi-simbol** dilaksanakan didomain waktu.

Hasil dari rangkaian PRAKTIKUM 5 adalah beberapa kandidat frekuensi pusat (CFC) yang akan diuji satu persatu mana diantara CFC itu yang valid dipakai sebagai acuan frekuensi saat dilaksanakan proses dekoding. Sebelum CFC diuji keabsahannya, maka terlebih dahulu dilaksanakan alignment frekuensi dan alignment pewaktu hingga level vernier/fine/teliti.

VII.2 Optimum Filter, Matched Filter dan Correlation Receiver

Pada sub-bab ini akan dibahas perihal tentang bagaimana melaksanakan **korelasi-simbol** di domain waktu. Untuk membahas perihal deteksi sinyal biner pada lingkungan Nois Gaussian Putih Aditif (AWGN), asumsi pertama adalah bahwa sinyal digital yang akan dideteksi merambat pada kanal komunikasi, masuk ke sistem penerima dengan gerbang suatu filter yang mempunyai respon impuls idial, yakni suatu unit impuls $\delta(t)$.

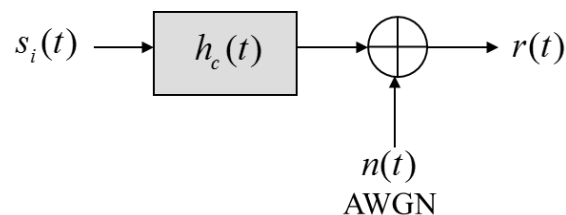
Sinyal transmisi biner sepanjang interval satu simbol (0,T) dapat ditulis sbb,

$$s_i(t) = \begin{cases} s_0(t) & 0 \leq t \leq T, \text{ untuk biner 0} \\ s_1(t) & 0 \leq t \leq T, \text{ untuk biner 1} \end{cases}$$

Namun dalam kenyataannya sinyal yang diterima $r(t)$ akan terdegradasi kualitasnya oleh keberadaan nois dan "filter-propagasi" yang mempunyai respon impuls tidak idial (bukan unit impuls $\delta(t)$) yakni $h_c(t)$, dapat ditulis sbb,

$$r(t) = [s_i(t) * h_c(t)] + n(t), \quad \text{untuk } i = 0 \text{ atau } 1$$

Blok diagram dari persamaan ini dapat dilihat pada gambar VII-1 dibawah ini,

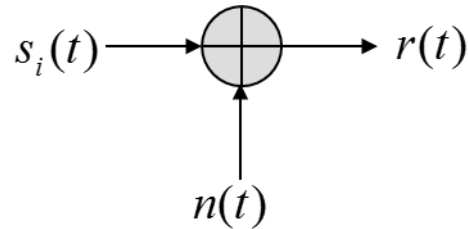


Gambar VII-1: Pemodelan kanal digital dalam lingkungan nois AWGN

Dimana $n(t)$ diasumsikan sebagai AWGN (Additive White Gaussian Noise). Jika $h_c(t)$ adalah idial (unit impuls $\delta(t)$), maka $r(t)$ dapat ditulis dengan lebih sederhana,

$$r(t) = s_i(t) + n(t) \quad i = 1,2 \quad 0 \leq t \leq T$$

Penyederhanaan ini dapat dilihat pada Gambar VII-2 dibawah ini,



Gambar VII-2: Pemodelan sederhana suatu kanal bernois aditif

Tujuan suatu Filter Optimum adalah untuk mendapatkan harga SNR maksimal dari satu perioda simbol $(S/N)_T$. Selanjutnya adalah mencari bentuk respon impuls filter optimum $h(t)$ yang membuat $(SNR)_T$ mempunyai harga maksimum.

Diasumsikan bahwa $a_i(t)$ adalah sinyal output dari filter disisi penerima yang mempunyai transfer function $H(f)$,

$$a_i(t) = \int_{-\infty}^{+\infty} H(f)S(f)e^{j2\pi ft} df$$

Dimana $H(f)$ adalah transfer function filter untuk deteksi sinyal $s_i(t)$, dan $S(f)$ adalah transformasi Fourier dari sinyal yang sedang diprediksi/dideteksi keberadaannya $s_i(t)$, namun untuk selanjutnya disederhanaan hanya ditulis $s(t)$.

Jika Power Spectrum Density (PSD) dari nois untuk 1 sisi spektrum adalah $N_0/2$, maka power nois dapat ditulis sbb,

$$\sigma_0^2 = \frac{N_0}{2} \int_{-\infty}^{+\infty} |H(f)|^2 df$$

Sehingga SNR pada saat $t=T$ dapat ditulis sbb,

$$\left(\frac{S}{N}\right)_T = \frac{\left| \int_{-\infty}^{+\infty} H(f)S(f)e^{j2\pi fT} df \right|^2}{\frac{N_0}{2} \int_{-\infty}^{+\infty} |H(f)|^2 df} \quad (***)$$

Jika bisa didapat $H(f)$ optimum maka harga $(S/N)_T$ adalah maksimum.

Ingat pada pertidaksamaan Schwarz (Schwarz's Inequality) yang menyatakan bahwa untuk suatu,

$$\left| \int_{-\infty}^{+\infty} f_1(x)f_2(x)dx \right|^2 \leq \int_{-\infty}^{+\infty} |f_1(x)|^2 dx \int_{-\infty}^{+\infty} |f_2(x)|^2 dx$$

Ruas kiri akan mempunyai harga sama dengan ruas kanan jika,

$$f_1(x) = f_2^*(x),$$

dimana, $f_2^*(x)$ adalah konjugat kompleks dari $f_2(x)$.

Jika $f_1(x) = H(f)$ dan $f_2(x) = S(f)e^{j2\pi fT}$, kembali pada persamaan (***) ,

$$\left| \int_{-\infty}^{+\infty} H(f)S(f)e^{j2\pi fT} df \right|^2 \leq \int_{-\infty}^{+\infty} |H(f)|^2 dx \int_{-\infty}^{+\infty} |S(f)e^{j2\pi fT}|^2 dx$$

Kedua ruas dibagi dengan power nois, maka didapat nilai (S/N) untuk waktu T, sbb,

$$\left(\frac{S}{N}\right)_T = \frac{\left| \int_{-\infty}^{+\infty} H(f)S(f)e^{j2\pi fT} df \right|^2}{\frac{N_0}{2} \int_{-\infty}^{+\infty} |H(f)|^2 df} \leq \frac{\int_{-\infty}^{+\infty} |H(f)|^2 df \int_{-\infty}^{+\infty} |S(f)e^{j2\pi fT}|^2 df}{\frac{N_0}{2} \int_{-\infty}^{+\infty} |H(f)|^2 df}$$

Karena $|e^{j2\pi fT}| = 1$, maka

$$\left(\frac{S}{N}\right)_T \leq \frac{2}{N_0} \int_{-\infty}^{+\infty} |S(f)|^2 df$$

Selanjutnya diasumsikan bahwa $s(t)$ adalah riil, maka $s(t) = s^*(t)$

Agar supaya $h(t)$ match terhadap sinyal input $s(t)$ maka diperlukan suatu kondisi dimana,

$$H(f) = f_1(x) = f_2^*(x) = S^*(f)e^{-j2\pi fT}$$

$$h(t) = \int_{-\infty}^{+\infty} S^*(f)e^{-j2\pi fT} e^{j2\pi ft} df$$

dengan prinsip konjugasi sederhana dapat dengan mudah dibuktikan bahwa,

$$[S^*(f)e^{-j2\pi fT} e^{j2\pi ft}] = [S(f)e^{j2\pi fT} e^{-j2\pi ft}]^*,$$

maka $h(t)$ dapat ditulis lagi sbb,

$$h(t) = \left[\int_{-\infty}^{+\infty} S(f)e^{j2\pi fT} e^{-j2\pi ft} df \right]^*$$

$$h(t) = \left[\int_{-\infty}^{+\infty} S(f)e^{j2\pi f(T-t)} df \right]^*$$

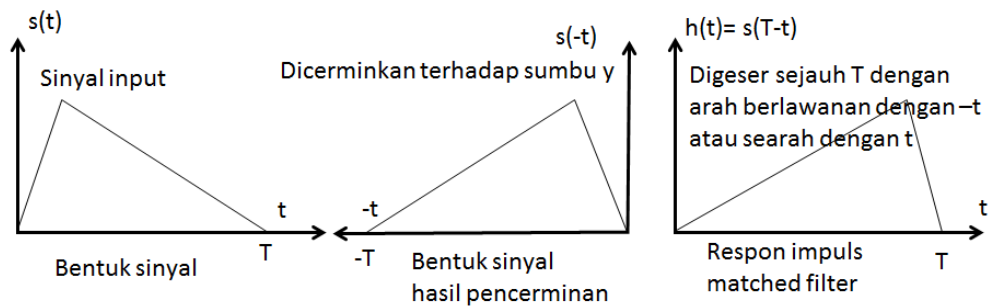
Berdasarkan properti simetri dari transformasi Fourier (lihat tabel transformasi Fourier dari buku referensi), dapat diperoleh $h(t)$ yang merupakan $s(t)$ versi konjugat, terbalik-waktu dan tertunda selama waktu T, atau dapat ditulis, sbb

$$h(t) = \begin{cases} s^*(T - t) & 0 \leq t \leq T \\ 0 & \text{tempat lain} \end{cases}$$

Untuk suatu harga riil maka $s^*(t) = s(t)$, maka $h(t)$ dapat ditulis menjadi,

$$h(t) = \begin{cases} s(T - t) & 0 \leq t \leq T \\ 0 & \text{tempat lain} \end{cases}$$

Merealisasikan matched filter dapat dengan cara *correlator* sbb,



Gambar VII-3: Respon impuls dari suatu matched filter

Gambar diatas menunjukkan respon impuls $h(t)$ yang merupakan bentuk sinyal tunda sejauh T dari pencerminan sinyal input $s(-t)$.

Jika dalam sistem yang kausal, diketahui suatu sinyal yang akan dideteksi adalah $r(t)$, maka output dari matched filter adalah,

$$z(t) = r(t) * h(t) = \int_0^t r(\tau)h(t - \tau)d\tau$$

Jika diketahui bahwa $h(t) = s(T - t)$ maka,

$$\begin{aligned} z(t) &= \int_0^t r(\tau)s[T - (t - \tau)]d\tau \\ &= \int_0^t r(\tau)s(T - t + \tau)d\tau \end{aligned}$$

Kondisi ini hanya untuk 1 perioda simbol, maka $t=T$, sehingga

$$z(t) = \int_0^T r(\tau)s(\tau)d\tau$$

Ini adalah suatu perkalian integrasi antara sinyal datang $r(\tau)$ dengan sinyal replika dari pemancar atau sinyal acuan $s(t)$, dan lazim disebut **correlation**. Dalam sistem diskrit dapat ditulis dalam bentuk penjumlahan dari perkalian sbb,

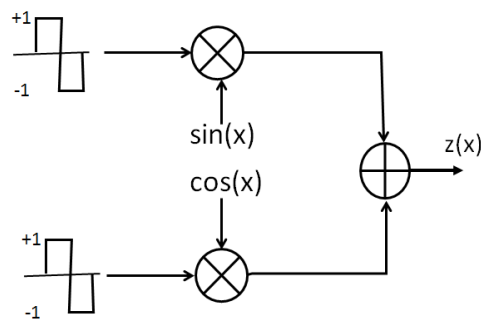
$$z(k) = \sum_{n=0}^{N-1} r(n)s(n)$$

Dimana, $z(k)$ adalah faktor/koefisien korelasi untuk simbol ke k , n adalah sampel, sedangkan N adalah jumlah sampel dalam satu simbol, r adalah sinyal datang, dan s adalah sinyal acuan atau sinyal hasil pembangkitan osilator lokal.

VII.3 Sinyal IQ Untuk Mempermudah Pembangkitan Modulasi Digital.

Pembangkitan modulasi digital seperti BPSK, QPSK, QAM akan menjadi mudah jika dilaksanakan di ranah baseband dan dilaksanakan dengan perangkat lunak. Dengan semakin majunya teknologi Software Defined Radio (SDR) maka pemanfaatan struktur sinyal IQ menjadi keniscayaan.

Sebagai contoh betapa sederhananya proses pembangkitan sinyal QPSK dengan modulator IQ dapat dilihat pada Gambar VII-6 dibawah ini,

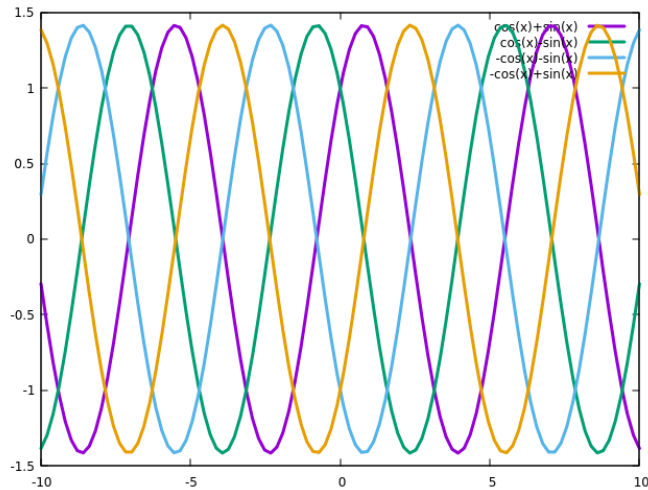


Gambar VII-4: Proses modulasi QPSK dengan memanfaatkan sinyal IQ

Pada Gambar VII-6 dapat dilihat bahwa jika masing-masing pengali dikenakan sinyal antipodal (+1, -1), maka akan didapat output $z(x)$ yang merupakan sinyal IQ dengan 4 sudut berbeda 90° tergantung dari kombinasi input antipodal. Sinyal output IQ $z(x)$ dapat dilihat pada Tabel VII-1 dibawah ini, sedangkan gambaran sinyal IQ dapat dilihat pada Gambar VII-7.

Tabel VII-1: Konstelasi QPSK

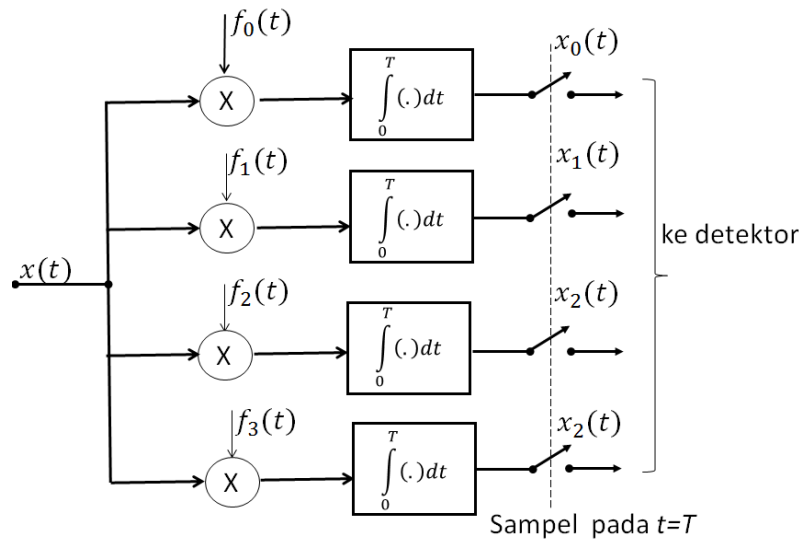
A(I)	A(Q)	$z(x)$
1	1	$\cos(x)+\sin(x)$
1	-1	$\cos(x)-\sin(x)$
-1	-1	$-\cos(x)-\sin(x)$
-1	+1	$-\cos(x)+\sin(x)$



Gambar VII-5: Gambaran 4 kemungkinan sinyal QPSK yang dibangkitkan oleh sinyal IQ

VII.4 Korelator FSK-4

Korelator FSK-4 pada WSPR dapat dilihat pada Gambar VII-8. Tampak pada gambar bahwa setiap saat WSPR menyediakan 4 osilator lokal f_0 , f_1 , f_2 dan f_3 , untuk deteksi 1 periode simbol. Selanjutnya keputusan dilakukan dengan cara mencari harga koefisien korelasi yang paling besar diantara 4 korelator tersebut.



Gambar VII-6: Korelator FSK-4 yang dipakai oleh WSPR untuk deteksi 1 simbol

Dari PRAKTIKUM-IV telah didapat deretan kandidat simbol 2×180 , dimana setiap kandidat simbol mempunyai jumlah sampel 256, dengan frekuensi sampling 387 Hz. Sehingga total sampel yang akan diuji adalah 2×46080 sampel (komponan riil dan imajiner). Selanjutnya didefinisikan bahwa $id[k]$ dan $qd[k]$ dimana $k=0,1,2,3,\dots,46079$ adalah sinyal input IQ yang akan diuji untuk dikenali dengan proses korelasi, yang kemudian di konversi menjadi biner, dan selanjutnya didekode dan di-unpacking untuk mereplika bit-bit data/berita/message yang telah dikirim dari sisi pemancar.

Dibawah ini adalah sintaksis korelator yang dipakai oleh WSPR,

```

i0[i]=0.0; q0[i]=0.0;
i1[i]=0.0; q1[i]=0.0;
i2[i]=0.0; q2[i]=0.0;
i3[i]=0.0; q3[i]=0.0;

for (j=0; j<256; j++) {
    k=lag+i*256+j;
    if( (k>0) && (k<np) ) { /* np = 46080 */
        i0[i]=i0[i] + id[k]*c0[j] + qd[k]*s0[j];
        q0[i]=q0[i] - id[k]*s0[j] + qd[k]*c0[j];
        i1[i]=i1[i] + id[k]*c1[j] + qd[k]*s1[j];
        q1[i]=q1[i] - id[k]*s1[j] + qd[k]*c1[j];
        i2[i]=i2[i] + id[k]*c2[j] + qd[k]*s2[j];
        q2[i]=q2[i] - id[k]*s2[j] + qd[k]*c2[j];
        i3[i]=i3[i] + id[k]*c3[j] + qd[k]*s3[j];
        q3[i]=q3[i] - id[k]*s3[j] + qd[k]*c3[j];
    }
}

p0=i0[i]*i0[i] + q0[i]*q0[i];
p1=i1[i]*i1[i] + q1[i]*q1[i];
p2=i2[i]*i2[i] + q2[i]*q2[i];
p3=i3[i]*i3[i] + q3[i]*q3[i];

p0=sqrt(p0);
p1=sqrt(p1);

```

```
p2=sqrt (p2) ;
p3=sqrt (p3) ;
```

dimana, k adalah urutan sampel, j adalah urutan relatif sampel, i adalah urutan blok simbol, sedangkan lag adalah timing offset untuk *alignment* pewaktu agar lebih sinkron. $i0[]$, $q0[]$, $i1[]$, $q1[]$, $i2[]$, $q2[]$, $i3[]$, $q3[]$ adalah faktor korelasi antara sinyal input dengan sinyal referensi dari osilator lokal. Sedangkan $p0$, $p1$, $p2$, $p3$ adalah nilai absolut dari harga IQ.

Selanjutnya adalah mencari nilai koefisien **korelasi-paket** yang merupakan penjumlahan dari masing-masing **korelasi-simbol**, perhatikan sintaksis pemrograman berikut ini,

```
totp=totp+p0+p1+p2+p3;
cmet=(p1+p3)-(p0+p2);
ss = (pr3[i] == 1) ? ss+cmet : ss-cmet;
}
ss=ss/totp;
if( ss > syncmax ) { //simpan paramater terbaik
    syncmax=ss;
    best_shift=lag;
    fbest=f0;
```

Tabel VII-2: Tabel kebenaran yang menggambarkan hasil deteksi idial

Jika Terdeteksi				Jika Diketahui	Maka
$P1$	$P3$	$P0$	$P2$	Bit Sinkronisasi	$cmet=(P1 + P3) - (P0 + P2)$
besar	kecil	kecil	kecil	+1	Harus bernilai +
kecil	besar	kecil	kecil	+1	Harus bernilai +
kecil	kecil	besar	kecil	-1	Harus bernilai -
kecil	kecil	kecil	besar	-1	Harus bernilai -

Tabel VII-1 dapat juga disebut tabel **korelasi-simbol** antara bit-sinkronisasi sesaat dengan deretan power sesaat hasil proses penerimaan radio $p0$, $p1$, $p2$ dan $p3$.

Hingga tahap ini sistem masih dalam upaya mencari nilai CFC yang paling baik, yang ditandai dengan dengan harga koefisien korelasi ss yang maksimum yang kemudian ditandai oleh timing offset **lag**.

Selanjutnya adalah proses rafinasi dari estimasi CFC atau $f1$, timing offset (jitter) atau $shift1$, dengan mamakai variabel ss (atau **sync**) sebagai ukuran/metrik ketelitian proses rafinasi dengan rentang harga mengambang [0,0 s/d 1,0].

Proses rafinasi dan sinkronisasi ini dilaksanakan oleh fungsi:

```
sync_and_demodulate(idat, qdat, npoints, symbols, &f1, ifmin,
ifmax, fstep, &shift1, lagmin, lagmax, lagstep, &drift1, symfac,
&sync1, mode)
```


Fungsi ini memiliki 2 moda operasi yang dipasang pada argumen terakhir yakni:

mode = 0 = tanpa pencarian frekuensi atau drift dan hanya mencari ingsut waktu (lag).
 mode = 1 = tanpa pencarian ingsut waktu (lag) atau drift frekuensi, dan hanya mencari frekuensi f1 terbaik.

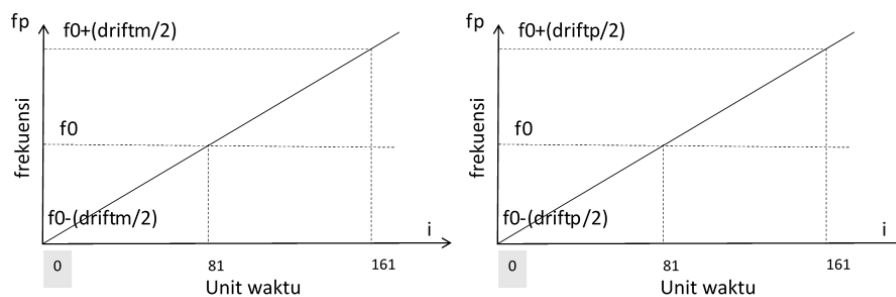
Argumen yang lain adalah sbb:

- **idat, qdat** adalah sampel riil dan imajiner yang akan diuji.
- **npoints** adalah jumlah sampel yang akan diuji yakni 46080
- **symbols** adalah angka representasi simbol antara 0 s/d 255
- **f1** adalah CFC yang terus di rafinasi harganya.
- **ifmin** adalah estimasi indeks ingsut frekuensi minimum karena efek dopler.
- **ifmax** adalah estimasi indeks ingsut frekuensi maksimum karena efek dopler.
- **fstep** adalah step indeks frekuensi saat mengompensasi dopler.
- **shift1** adalah angka timing offset (angka yang harus dikalikan 128).
- **lagmin** adalah offset/jitter nomor sampel minimum.
- **lagmax** adalah offset/jitter nomor sampel maksimum.
- **drift1** adalah besaran frekuensi yang bergeser karena kualitas osilator lokal.
- **symfac** adalah soft symbol normalizing factor = 50 (angka dari desainernya).
- **sync1** adalah nilai koefisien **korelasi-paket** yang ternormalisasi dan setiap saat diupdate.
- **mode** adalah penunjuk untuk rafinasi timing offset atau rafinasi frekuensi offset.

Setiap kandidat frekuensi pusat (CFC) menjalani proses akan menjalani proses rafinasi memakai ukuran metrik sync1,

langkah ke:

1. mode=0; timing offset/jitter --> lagmin=shift1-128; lagmax=shift1+128; lagstep=64; metrik sync1;
2. mode=1; kompensasi Doppler --> drift1=drift1; imin=-2; imax=2; fstep=0,25; sync1;



Gambar VII-9: Merubah slope persamaan linear pemodelan drift osilator lokal.

3. mode=1; frekuensi drift --> driftp=drift1+0,5; ifmin=0; ifmax=0; fstep=0; syncp;
4. mode=1; frekuensi drift --> driftm=drift1-0,5; ifmin=0; ifmax=0; fstep=0; syncm;
5. Jika syncp > sync1 --> drift1=driftp; sync1=syncp;

6. Jika $\text{syncm} > \text{sync1} \rightarrow \text{drift1}=\text{driftm}; \text{sync1}=\text{syncm};$
7. Cek apakah $\text{sync1} > \text{minsync1}(0,10)$
8. Jika YA lakukan langkah ke 9 dan ke 10; jika tidak maka lakukan langkah ke 11.
9. $\text{mode}=0;$ timing offset $\rightarrow \text{lagmin}=\text{shift1}-32; \text{lagmax}=\text{shift2}+32; \text{lagstep}=16;$ metrik $\text{sync1};$
10. $\text{mode}=1;$ kompensasi doppler $\rightarrow \text{ifmin}=-2; \text{ifmax}=2; \text{fstep}=0,05; \text{sync1};$
11. Jika tidak maka lanjut ke fungsi berikutnya (ada pada PRAKTIKUM 7)

VII.5 LANGKAH PERCOBAAN

1. Siapkan modul Raspberry Pi, nyalakan dan buka satu atau beberapa terminal window.
2. Masuk ke direktori "PRAKTIKUM-CORR"
3. Siapkan beberapa file WSPR hasil perekaman yang berasal dari beacon laboratorium dan yang berasal dari tempat yang jauh, dan letakkan pada direktori ini.
4. Kompilasi program `./wsprd` dengan menjalankan "Makefile"
`%> make -k`
5. Jalankan program `./wsprd`
`%> ./wsprd xxxxxx.WAV \r` atau
`%> ./wsprd xxxxxx.raw \r`
6. Amatilah tabel update parameter.

VII.6 LAPORAN PRAKTIKUM

1. Mengapa harga syncmax diberi initial harga $-1e30$.
 2. Jelaskan arti dari semua argumen yang ada pada fungsi `sync_and_demodulate()`.
 3. Tunjukkan sintaksis korelator antara sinyal input dan sinyal osilator lokal.
 4. Jelaskan semua parameter pada sintaksis $k=\text{lag}+i*256+j$ didalam sintaksis korelator.
 5. Tunjukkan sintaksis korelator antara bit-sinkronisasi dgn sinyal yang diterima oleh receiver.
 6. Laporan praktikum dikumpulkan sebelum PRAKTIKUM 7 dimulai.
-