

Implementasi Filter IIR secara Real Time pada TMS 32C5402

Oleh:

Tri Budi Santoso, Hary Octavianto, Titon Dutono

E-mail: tribudi@eepis-its.edu

Laboratorium Sinyal, Politeknik Elektronika Negeri Surabaya

Abstrak:

Telah dikembangkan modul praktikum Pengolah Sinyal Digital yang bersifat aplikatif praktis dengan memanfaatkan DSP Card TMS 32C5400. Satu implementasi Filter Digital dengan teknik perancangan IIR sengaja dipilih sebagai topik pada paper ini karena merupakan materi kunci dalam materi Pengolah Sinyal Digital.

Berawal dari perancangan IIR Filter menggunakan alat bantu perangkat lunak Matlab kemudian koefisien-koefisien respon impulse yang telah diperoleh dimasukkan ke perangkat lunak DSP Starter Kit untuk dilakukan kompilasi. Hasil kompilasi berupa kode hexa selanjutnya di download ke DSP Card untuk melihat implementasi realnya pada sinyal yang sebenarnya. Hasilnya menunjukkan karakteristik filter IIR yang mirip dengan teori.

Dari perancangan IIR Filter dikembangkan dalam bentuk preset equalizer pada TMS32C5402 dan diujikan pada sinyal audio secara real time.

Kata Kunci: DSP Card, filter IIR, koefisien filter, preset equalizer.

1. Pendahuluan

Permasalahan umum dalam penyampaian materi kuliah Pengolahan Sinyal Digital adalah keseimbangan antara penyampaian materi di kelas dan penyajian praktikum menggunakan perangkat keras DSP di laboratorium [1]. Pada penelitian sebelumnya telah dilakukan penyusunan modul berbasis Matlab dan DSP Card TMS32C5400, tetapi belum mampu menyajikan modul yang mendukung penyampaian teori di kelas [2].

Satu contoh model perancangan IIR Filter telah dibahas secara mendetail oleh Hyeoko Chio [3]. Disini disajikan tutorial yang menggambarkan aspek praktis beragam prosedur perancangan FIR filter dan diberi perbandingan kinerjanya satu dengan yang lain.

Dengan berkembangnya teknologi *Sistem on Chip* (SoC), batasan antara device dan sistem semakin sulit dijelaskan. Konsentrasi kita sebagai pengajar lebih baik untuk memfokuskan pada bagaimana mengintegrasikan sebanyak mungkin (fungsi) komponen elektronik ke dalam satu chip [4]. Langkah ini akan memberi kemudahan dan fleksibilitas dalam penyampaian materi untuk sistem perkuliahan berbasis kompetensi.

Tomorakos dan Ledger [5] dalam papernya menjabarkan bagaimana membangun sebuah modul DSP dengan menggunakan DSP Card produk dari Analog Devices.

Satu hal yang baru dalam paper ini adalah bagaimana sebuah filter IIR dirancang dengan menggunakan perangkat lunak Matlab, didownload ke DSP Card TMS32C5402, dan diujikan dengan sinyal audio secara real time.

2. Konsep IIR Filter

Filter IIR dari (*Infinite Impulse Response*) adalah salah satu tipe dari filter digital yang dipakai pada aplikasi *Digital Signal Processing* (DSP). Keuntungan filter IIR antara lain adalah membutuhkan koefisien yang lebih sedikit untuk respon frekuensi yang curam sehingga dapat mengurangi jumlah waktu komputasi.

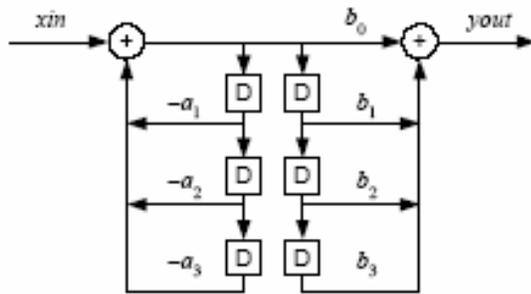
Fungsi transfer filter IIR adalah:

$$H(z) = \frac{b_0 + b_1 z^{-1} + \dots + b_N z^{-M}}{1 + a_1 z^{-1} + \dots + a_N z^{-N}} \quad (1)$$

dimana:

- $H(z)$ merupakan fungsi transfer dari filter IIR
- a_1, a_2, \dots, a_N merupakan koefisien feed back dari filter IIR
- b_0, b_1, \dots, b_N merupakan koefisien feed forward dari filter IIR

Diagram blok untuk sebuah filter IIR dalam bentuk *direct form II* dapat digambarkan seperti berikut:



Gambar 1. Diagram blok IIR direct form II

Proses pemfilteran pada sebuah sinyal akan mengikuti bentuk persamaan beda berikut ini.

$$y[n] = \sum_{m=0}^q b_m x[n-m] - \sum_{m=1}^p a_m y[n-m] \quad (2)$$

Karena adanya proses feedback ini filter IIR juga dikenal sebagai recursive filter

3. Implementasi Filter IIR pada TMS320C5402

Untuk mengimplementasikan IIR filter secara real time pada sebuah DSP Card yang dalam hal ini kita gunakan sebuah produk dari Texas Instrument, TMS32C5402 yang dilengkapi dengan Code Composer Studio (CCS) versi 2, kita dapat mengikuti langkah seperti Gambar 2.

Langkah pertama adalah bagaimana merancang filter dengan menggunakan perangkat Lunak Matlab. Berikut contoh disain program Matlab untuk menghasilkan koefisien filter IIR low-pass dengan frekuensi cut-off 2KHz pada frekuensi sampling sebesar 16KHz. Nilai koefisien yang dihasilkan disimpan dalam file teks. Pada langkah ini akan diperoleh respon frekuensi dan koefisien filter seperti yang diilustrasikan pada Gambar 2 tersebut.

Sebagai ilustrasi adalah perancangan low pass filter dengan spesifikasi berikut ini:

- Frekuensi sampling filter = 16KHz
- Setengah frekuensi sampling filter = 8KHz
- Frekuensi cut-off filter = 2KHz

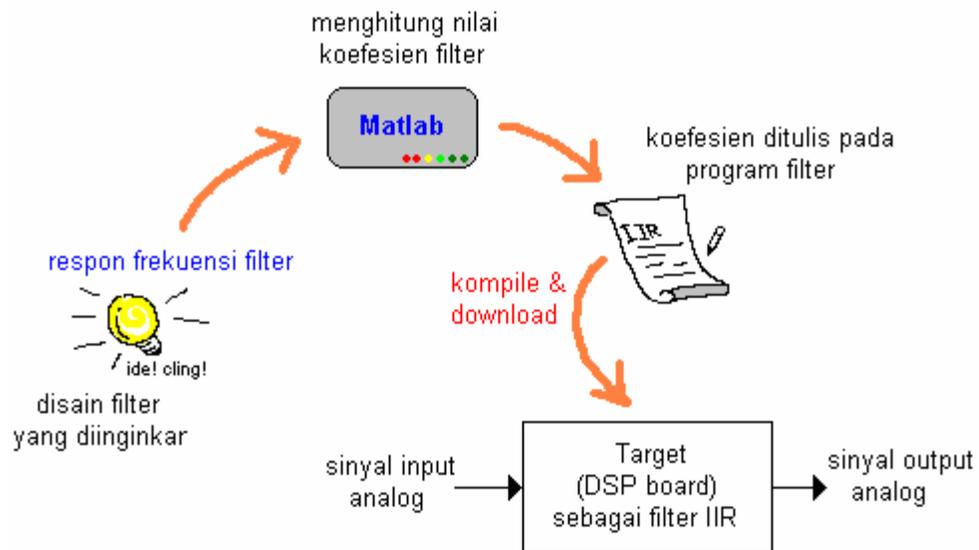
Nilai W1 sebagai *cut-off* pada kondisi ternormalisasi terhadap setengah dari frekuensi sampling adalah:

$$W1 \rightarrow \frac{\text{frek.cuoff}}{\frac{1}{2}\text{frek.samplng}} = \frac{2\text{KHz}}{8\text{KHz}} = 0.25$$

Program Matlab untuk mendapatkan koefisien filter low-pass IIR orde 4 adalah:

```
[b,a] = butter(4, 0.25);
```

Program secara lengkap dalam Matlab untuk mendapatkan koefisien filter low-pass IIR dengan frekuensi cut-off 2KHz pada frekuensi sampling 16KHz sebagai berikut.



Gambar 2. Ilustrasi alur implementasi filter IIR

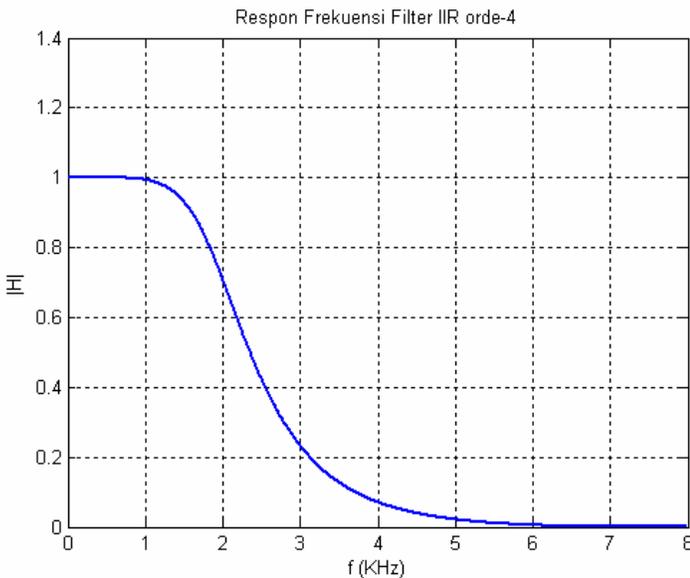
```
f=8;      %frekuensi sampling
n=3;      %orde filter
wl=0.25;  %frek. cut-off, dinormalisasi
          %terhadap 1/2 fs
[b,a]=butter(n,wl); % hitung koefisien IIR

[h,w]=freqz(b,a,1024);
plot((w*f)/3.14, (abs(h)), 'linewidth', 2);
xlabel ('f (KHz)');
ylabel ('|H|');
title('Respon Frek. Filter IIR orde-4');
grid;
```

Ini akan menghasilkan nilai-nilai koefisien filter IIR orde 4:

```
Nilai koefisien b
0.031689
0.095068
0.095068
0.031689
Nilai koefisien a
1.000000
-1.459029
0.910369
-0.197825
```

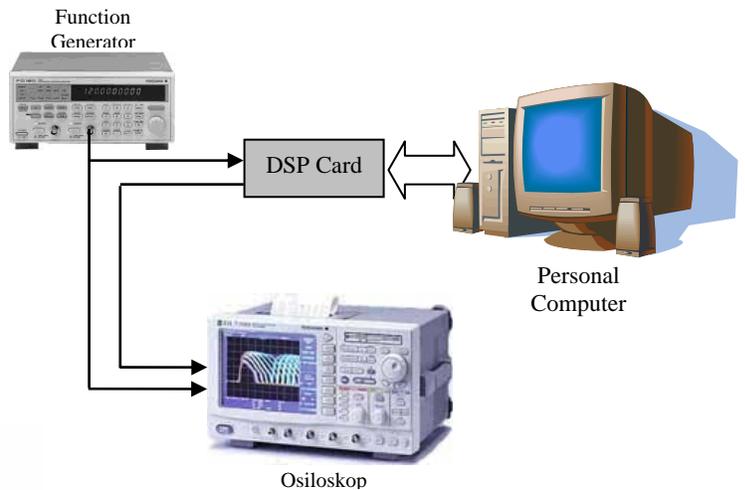
Dan sebuah respon frekuensi seperti Gambar 3.



Gambar 3. Respon frekuensi IIR filter hasil perancangan

Proses berikutnya adalah mengcopy koefisien filter hasil perancangan ke sebuah program CCS 2 yang merupakan perangkat lunak pendukung DSP Card TMS 32C5402. Program iir filter dikompilasi ulang ulang pada CCS 2 dan hasil kompilasi berupa kode hexa di downloadkan ke TMS32C5402.

Sebagai langkah awal pengujian filter IIR hasil perancangan digunakan function generator yang mampu bekerja pada frekuensi suara (300 ~ 4000 Hz), dalam hal ini kita gunakan yang mampu membangkitkan sinyal sinus dan persegi, dengan frekuensi kerja dari DC sampai 2 MHz. Untuk menguji hasilnya kita gunakan sebuah osiloskop yang dapat dilengkapi dengan fasilitas storage system. Lebih jelasnya bisa dilihat seperti pada Gambar 4.



Gambar 4. Gambaran sistem pengujian Filter IIR

Untuk menguji apakah sistem yang dirancang telah mampu bekerja dengan benar, masukkan sinyal sinus dari *function generator* ke DSP Card, dan amati hasil keluaran dari DSP Card. Bandingkan dengan sinyal asli yang berasal dari *function generator*. Selanjutnya nilai frekuensi sinyal input dinaikkan sedikit demi sedikit dari 0 Hz sampai dengan 6KHz. Apabila anda melihat perubahan nilai level sinyal output sebagai pengaruh kenaikan frekuensi sinyal input, hal ini menunjukkan sistem LPF yang anda rancang telah menunjukkan kinerja yang benar.

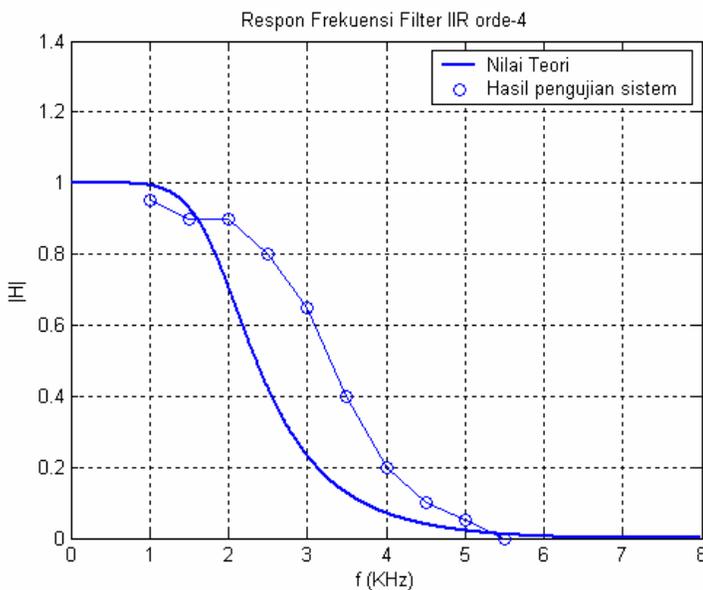
4. Karakteristik IIR Filter Hasil Perancangan

Sesuai dengan perancangan IIR untuk *low pass filter* (LPF), analisa yang disajikan disini juga untuk kasus LPF. Hasil pengujian terhadap sinyal input dan output menunjukkan kinerja seperti yang ditunjukkan pada Tabel 1 berikut ini.

Tabel 1. Hasil pengukuran pada LPF

No	V in = 50m Vpp	
	Freq (kHz)	V out mVpp
1	1.0	0.95
2	1.5	0.90
3	2.0	0.90
4	2.5	0.80
5	3.0	0.65
6	3.5	0.40
7	4.0	0.20
8	4.5	0.10
9	5.0	0.05
10	5.5	0.00

Dari hasil yang terdapat pada Tabel 1 diatas, dapat dibentuk grafik seperti pada Gambar 5 berikut ini.



Gambar 5. Perbandingan Teori dan Hasil Pengujian IIR

Walaupun ada penyimpangan dari teori, tetapi pola yang dihasilkan oleh sistem yang dirancang telah menunjukkan adanya kemiripan dengan teori. Hal ini wajar terjadi sebab dalam realisasi sistem secara real time kendala fisik seringkali ditemui, sehingga tidak mungkin mencapai kondisi ideal. Pada saat nilai frekuensi $f = 2\text{kHz}$, sistem ini mulai memberikan redaman. Hal ini sesuai dengan karakteristik sistem IIR low pass yang dirancang dengan nilai frkekuensi cut-off sebesar $f = 0,25$ dari 8KHz atau sebesar 2 KHz.

5. Aplikasi pada Preset Equalizer

Untuk lebih yakin dengan sistem yang telah dirancang, filter IIR dikembangkan menjadi sebuah preset equalizer 4 kanal. Dengan pengaturan kontrol melalui subrutin getDipSwStatus(), pengambila nilai posisi DipSwitch, ditetapkan nilai yang dikembalikan adalah 0,1,2 dan 3. Nilai-nilai ini akan menentukan seting Dip Switch yang ada di DSP Card.

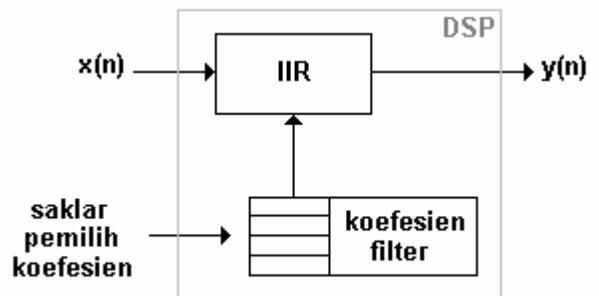
Proses perancangan filter tetap mengacu pada langkah sebelumnya dan nilai koefisien-koefieisnya disesuaikan dengan Tabel 2 .

Tabel 2. Saklar 4-preset equalizer

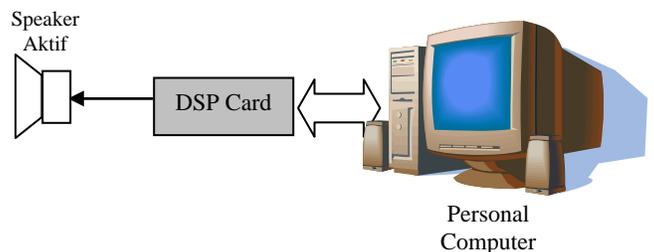
DipSW-7	DipSW-8	Filter
OFF	OFF	APF
OFF	ON	LPF 1 kHz
ON	OFF	LPF 4 kHz
ON	ON	HPF 4 kHz

*) posisi ON = saklar kebawah = logika 0

Proses bekerjanya preset equalizer yang dirancang dapat disederhanakan seperti Gambar 6. Disini masing-masing switch akan menentukan jenis filter yang digunakan seperti pada Tabel 2, dengan mengambil file audio yang ada di PC, dan input $x(n)$ diambil dari output Sound Card, sedangkan output $y(n)$ disambungkan ke speaker aktif. Ujud realnya seperti pada Gambar 7.



Gambar 6. Diagram Preset Equalizer



Gambar 7. Pengujian Sistem Preset Equalizer

Dengan merubah DipSwitch pada DSP Card akan merubah karakteristik suara yang dihasilkan. Misalnya pada saat di pilih SW7 Off dan SW 8 on akan menyebabkan suara yang muncul hanya nada rendah. Cara yang cukup sederhana ini ternyata cukup efektif dalam membuat siswa lebih kerasan melakukan percobaan. Dengan cara ini mereka merasa bisa menikmati dan tidak sadar kalau *Mata Kuliah Pengolahan Sinyal* berisi rumusan matematik yang membosankan. Model ini telah dikembangkan dan diujudkan dalam praktikum Pengolahan Sinyal Digital di Laboratorium Sinyal, Politeknik Elektronika Negeri Surabaya dan berlaku untuk semua Jurusan yang ada yaitu Elektronika, Telekomunikasi, Elektro Industri dan Teknologi Informasi.

6. Kesimpulan

Dari hasil yang diperoleh dalam pengujian sistem diatas menunjukkan sistem IIR untuk *low pass filter* (LPF) cukup mewakili sistem ideal dan telah menunjukkan kinerja yang bagus.

Dari rancangan yang dibuat juga telah diimplementasikan dalam aplikasi preset equalizer untuk pengolahan sinyal audio. Model percobaan ini telah digunakan di laboratorium PENS dan telah mampu menarik minat siswa pada materi Pengolah Sinyal yang biasanya membosankan.

Referensi:

- [1.] Thompson Michel, Burleigh Jojan, “*Matlab/DSK GUI for FIR Filter Design Using a TMS320C3X DSK*”, Baylor University, Texas, USA 2000.
- [2.] Tri Budi Santoso, Hary Oktavianto, “*Pengembangan Materi Praktikum Pengolah Sinyal Digital berbasis Matlab dan TMS 32C5400 DSK*”, Proceedings SNTE 2003, UGM, 25 September 2003.
- [3.] Hyeikho Chio, “*IIR Filter*”, Digital Signal Processing Laboratory, ELE C434, Rice University, 2000.
- [4.] Joseph Walsh and Ian Grouf, “An Investigation Into The Requirement of a PC-based Learning Environment for The Education of Microelectronic Test Engineering”, www.itta.net/EdTech2003/papers/washgrouf.doc
- [5.] Dan Ledger and John Tomarakos, “Using The Low Cost, High Performance ADSP 21065 Digital Signal Processor for Digital Audio Applications”, DSP Application Group, Analog Devices, USA 1998.

Bio Data Penulis



Tri Budi Santoso

Pada tahun 1994 menyelesaikan program S1 di Jurusan Teknik Fisika-Program Studi Energi dan Teknik Kondisi Lingkungan. Pada tahun 1999 menyelesaikan program S2 di Teknik Elektro-Telekomunikasi. Aktifitas riset yang sedang digelutinya adalah *Speech Processing*.

Hary Octavianto

Menyelesaikan pendidikan S1 Jurusan Teknik Elektro, program Studi Elektronika pada tahun 2001. Bergabung dengan Politeknik Elektronika Negeri Surabaya, sejak tahun 2001. Saat ini bekerja sebagai staf pengajar di Jurusan Elektronika dan riset yang digelutinya adalah *Embedded System for Digital Signal Processing*.

Titon Dutono

Menyelesaikan Pendidikan Jurusan Elektro ITS pada tahun 1985. Tahun 1996 berhasil menyelesaikan pendidikan S2 di Kumamoto University, Japan. Tahun 1999 berhasil menyelesaikan pendidikan S3 di tempat yang sama, dengan bidang spesialisasi *Speech Processing*. Saat ini merupakan advisor Laboratorium Sinyal PENS.