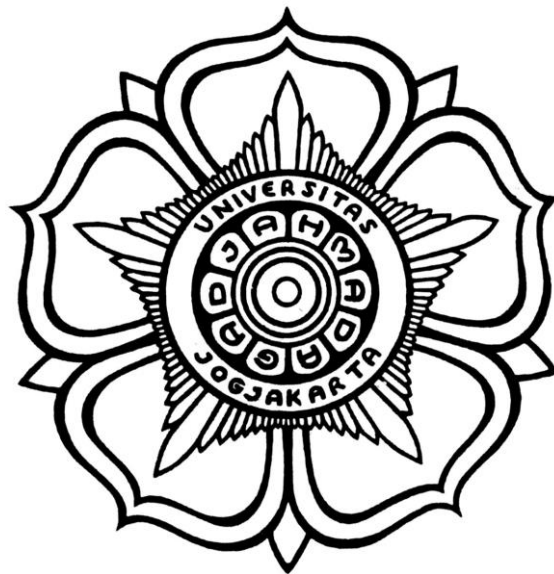


MODUL 08

PRAKTIKUM SISTEM DIGITAL

TKF213112 – 1 SKS



Disusun oleh:

Prof. Ir. Sunarno, M.Eng., Ph.D.

dan

Tim Asisten Praktikum Sistem Digital

**LABORATORIUM SENSOR DAN SISTEM TELEKONTROL
DEPARTEMEN TEKNIK NUKLIR DAN TEKNIK FISIKA
FAKULTAS TEKNIK
UNIVERSITAS GADJAH MADA
2024**

Daftar Isi

1.1	Tujuan	3
1.2	Materi.....	3
1.3	Teori.....	3
1.3.1	Jenis-jenis IC ADC da DAC	4
1.3.2	Karakteristik ADC maupun DAC	4
1.3.3	PWM	8
1.4	Alat da Bahan.....	10
1.5	Langkah-langkah Percobaan	10
1.5.1	ADC	10
1.5.2	DAC	14
1.5.3	PWM.....	16
1.6	Tabel Percobaan.....	18
1.6.1	DAC	18
1.6.2	ADC	18
1.6.3	PWM.....	19
DAFTAR PUSTAKA		20

Chapter 8

ADC, DAC, & PWM

1.1 Tujuan

Mahasiswa memahami cara kerja ADC (*Analog to Digital Converter*), cara kerja DAC (*Digital to Analog Converter*), dan cara kerja PWM (*Pulse Width Modulation*).

1.2 Materi

- a. DAC (Digital to Analog Converter)
- b. ADC (Analog to Digital Converter)
- c. PWM (Pulse Width Modulation)

1.3 Teori

Rangkaian yang pertama dikenal dengan ADC (*Analog to Digital Converter*). Rangkaian ini digunakan untuk mengubah suatu sinyal *input* yang analog menjadi *output* digital. *Output* ini biasanya berupa perintah yang akan digunakan untuk menjalankan aktuator. Rangkaian jenis kedua adalah DAC (*Digital to Analog Converter*). DAC digunakan untuk mengonversi sinyal dari *input* yang berupa sinyal digital untuk diubah menjadi *output* berupa sinyal analog. *Output* dari DAC ini biasanya berupa sinyal analog yang dapat di plot dalam bentuk grafik atau gambar.

Penggunaan sinyal *output* baik berupa sinyal analog maupun digital mempunyai banyak kelemahan dan kerugian masing-masing. Dengan menggunakan sinyal analog, data *output* akan dapat digunakan dalam jangka waktu yang lama, tetapi membutuhkan waktu yang lama pula dalam proses pengolahan sinyalnya. Sedangkan sinyal digital, mempunyai keuntungan pemrosesan data lebih cepat, tetapi tidak dapat digunakan dalam jangka waktu lama.

ADC dan DAC mempunyai ketelitian yang sangat akurat, tetapi hal ini juga bergantung pada jumlah dari bit-bit sinyal digital yang digunakan. Bit-bit sinyal yang dapat digunakan adalah 8 bit, 16 bit, atau 32 bit sinyal digital. Akan tetapi, yang paling sering digunakan adalah 8 bit sinyal-sinyal digital. Dengan menggunakan 8 bit, sinyal digital ini akan dapat membangkitkan tegangan sebesar 28 atau mencapai ketelitian sebesar 256 tingkatan tegangan. Jika tegangan maksimum yang dibangkitkan sebesar 2,55 V, maka DAC ini dapat

membangkitkan tegangan mulai dari 0 sampai 2,55 V dengan tingkatan 0,01 V. Artinya *input* biner bernilai 00000001 pada *input* DAC akan membangkitkan tegangan *output* 0,01 V, biner 00000010 membangkitkan 0,02 V dan seterusnya hingga mencapai biner 11111111 yang akan menghasilkan tegangan 2,55 V. Sebaliknya, tegangan 0 sampai 2,55 V ini akan pada rentang oleh ADC menjadi besaran digital dalam bentuk biner dari 00000000 sampai 11111111.

1.3.1 Jenis-jenis IC ADC dan DAC

Dewasa ini, pabrik-pabrik pembuat IC ADC dan DAC telah bersaing untuk memasarkan produk buatannya. Masing-masing perusahaan tentu melakukan segala cara agar produk IC yang dijual di pasaran dapat terjual laku. IC yang ada pun telah banyak yang menggunakan 16 bit, meskipun masih ada yang menjual IC 4 bit dan 8 bit.

IC ADC yang sering digunakan di Indonesia adalah ADC0804 atau ADC0808. Namun, ada juga ADC lain yang sering digunakan seperti ADC0801 / ADC0802 / ADC0803 / ADC0805 8-Bit Microprocessor Compatible A/D converters. Disamping itu, IC DAC yang sering kita jumpai di pasaran adalah DAC080 / DAC0801 / DAC0802 8-Bit Digital to Analog Converters. Kedua IC, baik ADC maupun DAC, yang telah disebut di atas merupakan buatan National Semiconductor.

Disamping itu, ada IC ADC yang lain seperti IC ADC yang dikenal dengan rancangan khusus dan sudah dilengkapi untuk keperluan membangun Digital *Voltmeter*, misalnya ICL7106 / ICL7107 3 1/2 -Digit A/D Converter produksi Maxim. IC ini mempunyai ketelitian yang lebih dibandingkan dengan ADC 8 bit, biasa digunakan untuk mengukur tegangan 0 sampai 1,999 atau hampir 11 bit. IC ADC ini juga dilengkapi dengan rangkaian yang dapat menampilkan tegangan yang diukur.

1.3.2 Karakteristik ADC maupun DAC

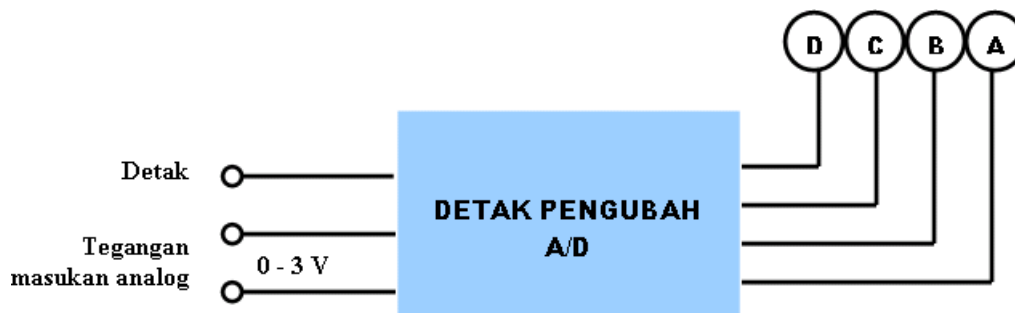
Operasi penting yang berhubungan dengan sinyal analog dan digital adalah konversi digital ke analog yang dilakukan oleh pengubah digital ke analog (DAC) dan konversi analog ke digital yang dilakukan oleh pengubah analog ke digital (ADC).

Apabila yang akan kita proses merupakan besaran analog, baik sebagai masukan ataupun keluaran analog, sedangkan unit pemroses yang kita pakai berbasis digital, maka harus dipakai konverter analog-ke-digital. Namun, apabila masukan adalah digital dan jika keluaran yang dikehendaki adalah analog, maka dibutuhkan konverter digital-ke-analog. Alat pengubah ini sangat banyak ditemukan, antara lain:

1. Pengubah analog ke digital penghitung,
2. Pengubah (A/D) pendekatan berurutan,
3. Pengubah (A/D) komparator-paralel,
4. Pengubah (A/D) 2-kemiringan (*dual-slope*) atau *ratio-metric*.

Analog to Digital Converter (ADC)

Pengubah analog ke digital (pengubah A/D) membalik proses dari pengubah D/A. Tegangan analog yang tak diketahui dimasukkan ke dalam pengubah A/D dan akan muncul keluaran biner yang bersangkutan. Keluaran biner tersebut akan berbanding lurus dengan masukan analog. Diagram blok dari suatu pengubah A/D diperlihatkan pada Gambar 8.1. Pengubah A/D ini mempunyai tegangan masukan analog yang berkisar dari 0 sampai 3 V. Kemudian, keluaran biner tersebut akan terbaca dalam bentuk biner dari 0000 sampai 1111. Perhatikan bahwa pengubah A/D tersebut juga mempunyai masukan detak. Ada 3 karakteristik yang perlu diperhatikan dalam pemilihan komponen ADC, antara lain:



Gambar 8. 1 Diagram blok dari suatu pengubah A/D 4-bit

1. Resolusi

Resolusi adalah perubahan masukan terkecil yang menyebabkan berubahnya *output*. Resolusi pada sebuah ADC ini dapat dianggap sebagai kesalahan kuantisasi. Semakin banyak jumlah bit yang mengkode sinyal, maka kesalahan kuantisasi ini dapat direduksi. Kesalahan kuantisasi secara umum adalah sebesar $\pm 1/2$ LSB. Perumusan resolusi secara umum adalah sebagai berikut.

$$\%Resolusi = \frac{1}{\text{jumlah step total}} \times 100\% = \frac{1}{2^n} \times 100$$

dimana n = jumlah bit.

2. Akurasi

Akurasi adalah jumlah dari semua kesalahan, misalnya kesalahan nonlinieritas, skala penuh, skala nol, dan lain-lain. Akurasi dapat juga menyatakan perbedaan antara tegangan *input* analog secara teoritis yang dibutuhkan untuk menghasilkan suatu kode

biner tertentu terhadap tegangan *input* nyata yang menghasilkan tegangan kode biner tersebut. Keakuratan sebuah ADC sama sekali tidak bergantung pada resolusi, tetapi tergantung pada keakuratan komponen-komponen penyusunnya, seperti komparator, resistor, dan level *amplifier*-nya. Spesifikasi kesalahan 0,01% F.S mempunyai arti hasil konversi berbeda sebesar 0,01% dari F.S dari yang seharusnya.

3. Waktu Konversi

Waktu konversi adalah waktu yang dibutuhkan untuk mendigitalkan setiap sampel atau data yang diperlukan untuk menyelesaikan suatu konversi. Spesifikasi ini sangat penting jika *bandwidth* sinyal *input* merupakan hal yang sangat signifikan.

Digital to Analog Converter (DAC)

DAC adalah alat pengkonversi kebalikan dari ADC. Alat ini mampu mengkonversi sinyal/data dari besaran digital menjadi besaran analog. DAC biasanya berhubungan dengan aktuator. Keluaran dari DAC berupa besaran analog akan digunakan sebagai perintah yang akan dijalankan pada aktuator. Namun, ada pula keluaran DAC yang hanya dituliskan ke dalam bentuk grafik saja. Hal ini tergantung dari pemakaian akan digunakan dalam bentuk grafik atau dalam suatu sistem kontrol.



Gambar 8. 2 Bagian fungsional dari suatu pengubah D/A

Secara umum, DAC terdiri dari dua bagian fungsional. Gambar 8.2 mengilustrasikan diagram blok dari sebuah DAC, yaitu terdiri dari jaringan resistor dan penguat penjumlahan. Jaringan resistor tersebut akan membebani masukan 1, 2, 3 dan 4 dengan tepat. Penguat penjumlahan akan memberikan skala tegangan keluaran yang sesuai dengan tabel kebenaran. IC penguat penjumlahan yang digunakan dalam DAC adalah jenis penguat operasional (*Op-Amp*). Ada tiga karakteristik yang penting dari DAC, yaitu:

1. Resolusi

Resolusi adalah perubahan terkecil pada *output* analog. Resolusi selalu sama dengan bobot dari LSB yang disebut besar langkah (*step size*). Harganya akan lebih kecil bila digunakan jumlah bit yang lebih banyak. Dengan menambah jumlah bit maka

akan menambah jumlah *step* untuk skala penuh, karena hanya jumlah bit yang menentukan persentasi resolusi.

2. Akurasi

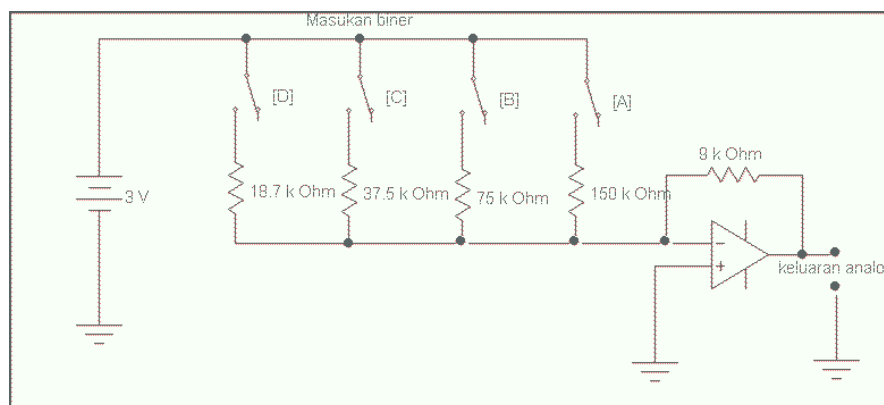
Akurasi (kecermatan) menghubungkan keluaran analog yang diperoleh sebenarnya dengan keluaran yang diharapkan, biasanya dinyatakan dalam persentasi dari skala penuh keluaran. Makin kecil persentasi harga kecermatan, akan semakin akurat dan tentu saja semakin mahal harganya. Kadang-kadang kecermatan DAC dilihat dari linieritasnya. Kecermatan dan resolusi dari DAC haruslah sebanding.

3. Settling Time

Apabila *input-input* digital suatu DAC berubah, bagi level *amplifier* dan rangkaian *internal* lainnya memerlukan waktu untuk memberikan respon menghasilkan suatu harga keluaran analog yang baru. Waktu yang diperlukan keluaran tersebut untuk menstabilkan sampai 99,95 % dari harga barunya disebut *settling time*.

Prinsip Kerja DAC

Sebuah DAC secara umum dapat diilustrasikan seperti pada Gambar 8.3. Seperti yang disebutkan sebelumnya, DAC terdiri dari rangkaian jaringan resistor dan penguat penjumlahan. Selain itu, juga terdapat *switch* (A, B, C, dan D) di mana kondisi *open* dan *close* tergantung pada data biner yang ada. Untuk data biner “1”, maka *switch* akan dalam kondisi *close* (*on*). Begitu sebaliknya, jika data biner “0”, maka kondisi *switch* akan *open* (*off*). *Switch-switch* tersebut dihubungkan dengan tegangan masukan (V_{in}).



Gambar 8. 3 Rangkaian pengubah D/A

Dari Gambar 8.3 di atas, resistor R4 adalah resistor MSB (*Most Significant Bit*). Terlihat bahwa R3 nilainya dua kali dari R4, R2 nilainya empat kalinya dan seterusnya. DAC akan mengambil harga yang dinyatakan dalam kode digital dan mengubahnya menjadi arus atau tegangan yang sebanding dengan harga digitalnya. Tegangan atau arus ini merupakan

besaran analog dan pada setiap jangkauan tertentu mempunyai harga yang berbeda. DAC akan mengubah bilangan biner yang direpresentasikan oleh tingkat logika digital ke suatu tegangan *output* yang sebanding dengan bobot bilangan biner tersebut. Harga tegangan atau arus keluarannya tergantung dari harga referensi yang dipilih untuk skala penuhnya. Harga tegangan *output* (V_o) tersebut adalah hasil kali antara sinyal digital (D) dengan tegangan referensi analog (V_{ref}), dan dapat dirumuskan seperti yang ada di bawah ini.

$$V_o = V_{in} \times \frac{R_f}{R_1} \left(\frac{D_0}{256} + \frac{D_1}{128} + \frac{D_2}{64} + \frac{D_3}{32} + \frac{D_4}{16} + \frac{D_5}{8} + \frac{D_6}{4} + \frac{D_7}{2} \right)$$

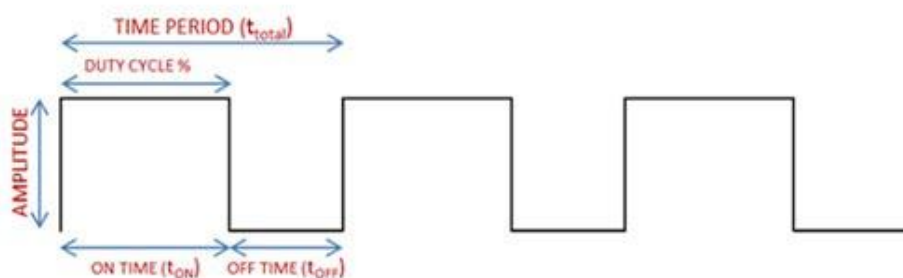
$$V_o = V_{in} \times D \times \frac{R_f}{R_1}$$

DAC Jenis Tangga

Pada DAC jenis tangga, resistor disusun seperti yang terlihat pada Gambar 8.3. Fungsi dari jaringan ini adalah sama seperti jaringan resistor yang digunakan sebelumnya. Hanya saja, besar resistansinya terdiri dari dua nilai yaitu R dan $2R$. Hal ini menyebabkan rangkaian ini lebih sederhana dan mudah dalam penentuan nilai dari resistor.

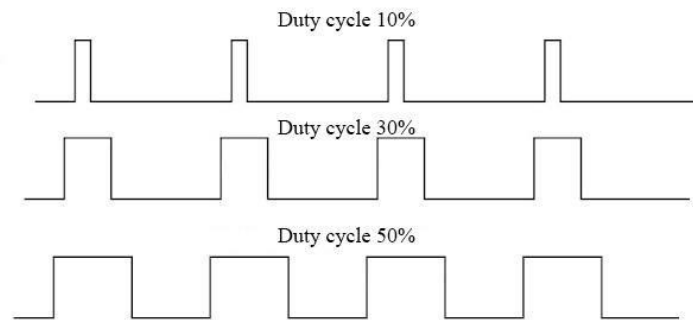
1.3.3 PWM

Pulse Width Modulation (PWM) adalah salah satu teknik yang digunakan untuk mengendalikan kekuatan (*power*) dengan mengatur berapa besar tegangan yang akan digunakan melalui pengiriman isyarat atau pulsa dalam bentuk sinyal. PWM bekerja dengan memanipulasi lebar sinyal yang dinyatakan dengan pulsa dalam suatu periode, untuk mendapatkan tegangan rata-rata yang berbeda. Teknik ini umum digunakan dalam berbagai aplikasi, seperti pemodulasian data untuk telekomunikasi, pengontrolan daya atau tegangan yang masuk ke beban, regulator tegangan, *audio effect* dan penguatan, serta berbagai aplikasi lainnya.



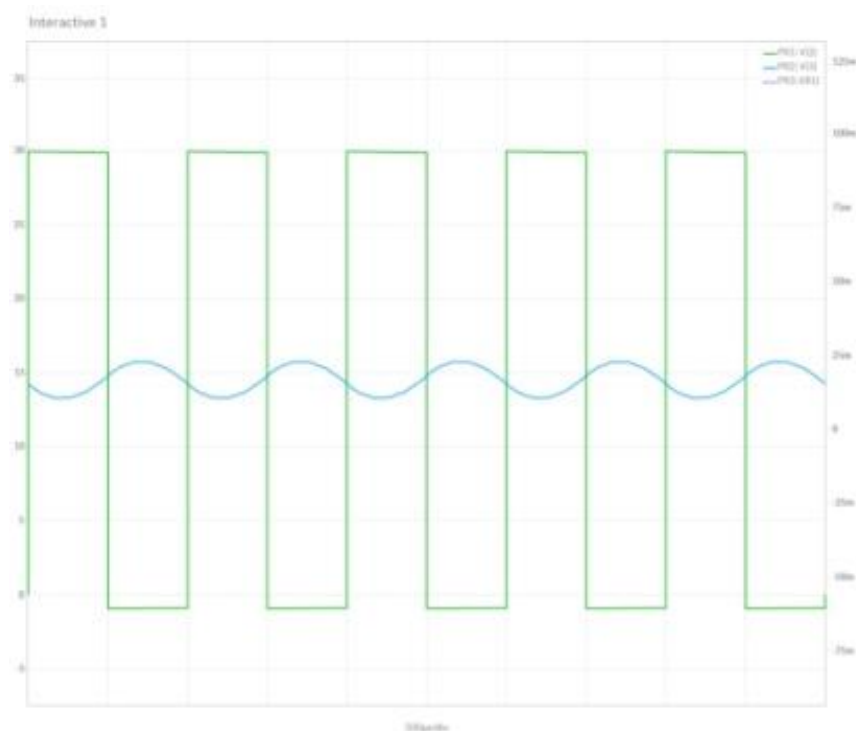
Gambar 8. 4 Bentuk sinyal PWM

Duty cycle dalam PWM adalah persentase dari satu periode sinyal PWM di mana sinyal tersebut dalam kondisi HIGH dibandingkan dengan total periode waktu tersebut. Dengan kata lain, duty cycle menentukan berapa lama sinyal aktif dalam satu siklus sebelum kembali ke kondisi LOW. Semakin tinggi duty cycle, semakin lama sinyal akan tetap HIGH dalam satu periode. Duty cycle umumnya dinyatakan dalam persen, misalnya 10%, 30%, 50%, hingga 100%.



Gambar 8. 5 Duty Cycle pada PWM

Frekuensi merupakan jumlah gelombang yang terbentuk dalam satu detik. Jadi bisa disimpulkan bahwa frekuensi gelombang merupakan jumlah gelombang dibagi waktu yang dibutuhkan. Berikut merupakan contoh sinyal PWM dengan variasi frekuensi (a) 10kHz dan (b) 50kHz



Gambar 8. 6 Contoh variasi frekuensi 10kHz pada PWM (a)



Gambar 8. 7 Contoh variasi Frekuensi 50kHz pada PWM (b)

Dalam percobaan modul ini, PWM digunakan untuk mengendalikan *duty cycle* pada sinyal yang akan diamati pada osiloskop. Aplikasi PWM berbasis mikrokontroler biasanya meliputi pengendalian kecepatan motor DC, pengendalian motor servo, pengaturan nyala terang LED, dan lain sebagainya

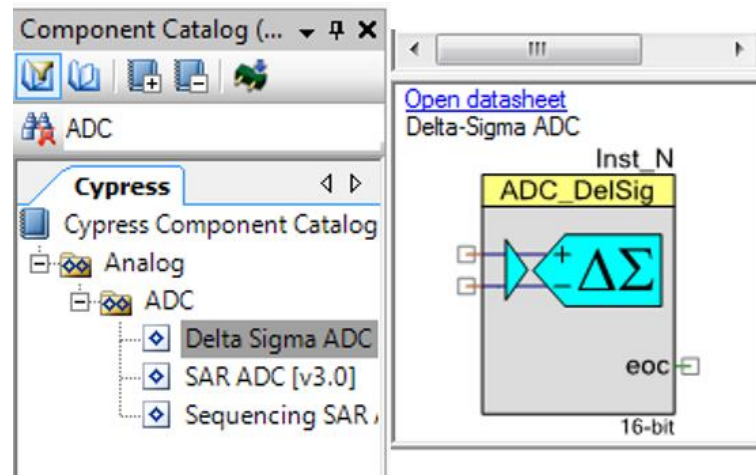
1.4 Alat dan Bahan

- Buku praktikum
- Modul PSoC
- Kabel jumper
- Laptop
- PSoC Creator

1.5 Langkah-langkah Percobaan

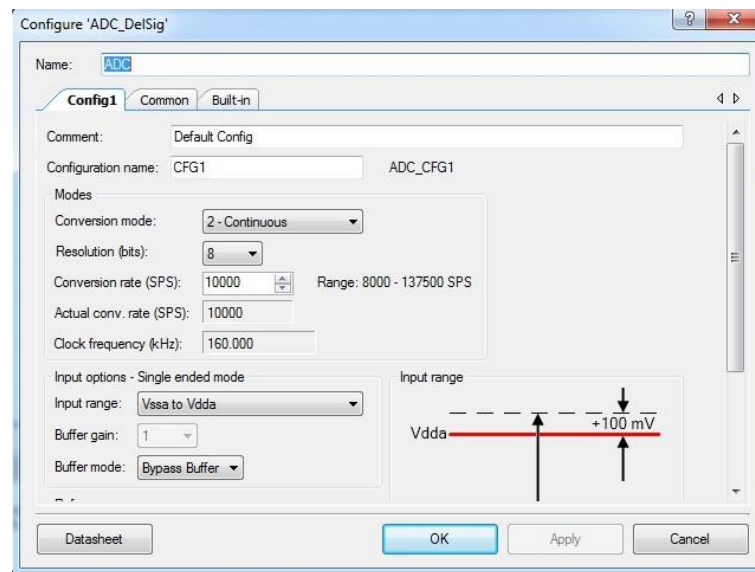
1.5.1 ADC

- Buka PSoC Creator
- Pada Workspace TopDesign.cysch masukkan komponen Delta-Sigma ADC yang didapat dari *Component Catalog* seperti Gambar 8.8.

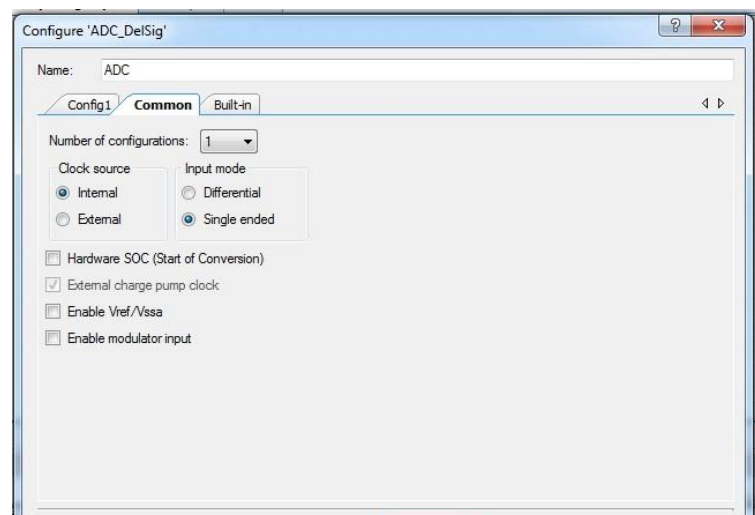


Gambar 8. 8 Component Catalog

3. Atur ADC dengan klik 2 kali pada komponen, gunakan pengaturan seperti pada Gambar 8.9 Pemberian nama komponen akan mempengaruhi nama pada program. Pengaturan *conversion mode* dipilih *continuous* dimaksudkan agar ADC melakukan pembacaan data/*sampling* secara terus menerus. Resolusi diatur dalam 8 bit atau sesuai kegunaan. Semakin tinggi resolusi yang dipilih maka ketelitian dari hasil pembacaan makin besar, namun akan berdampak pada kecepatan Delta-Sigma ADC dalam melakukan *sampling* atau biasa disebut SPS (*Sampling per Second*).
4. Selanjutnya pilih sub menu *common* kemudian dilakukan pengaturan seperti Gambar 8.9 Jumlah konfigurasi yang digunakan menggunakan jumlah 1 saja dan *Clock Source* dimaksudkan sumber waktu yang digunakan untuk mengetahui waktu dalam pengambilan data tersebut. *Input Mode* mengkonfigurasi ADC untuk *input Differential* atau *Single ended* pilihan *default*-nya adalah *Differential*. Dalam mode ini, *input* negatif dan positif ditunjukkan pada simbol. Bila mode *Single ended* dipilih, *input* negatif ke ADC terhubung ke *Vssa*. Jumlah konfigurasi diatur menjadi 1 apabila hanya menggunakan 1 ADC dalam penggunaannya.

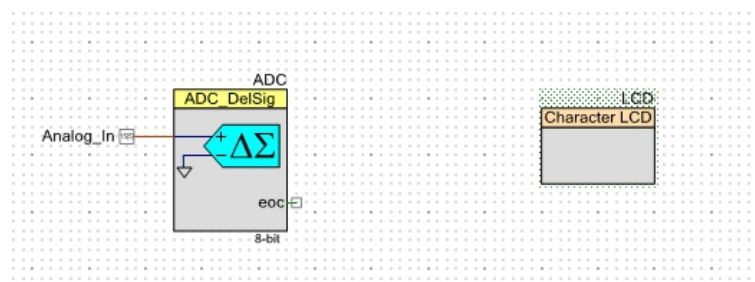


Gambar 8. 9 ADC konfigurasi



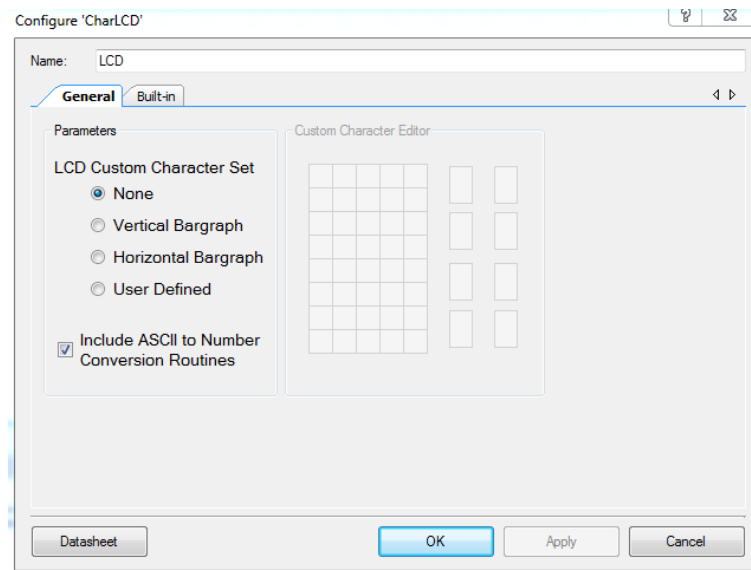
Gambar 8. 10 Common konfigurasi

5. Selanjutnya kembali ke sub menu *Config1* dan ubah *Input range* menjadi *Vssa to Vdda*. Nilai absolut maksimum dari ADC input range selalu didikte oleh nilai absolut maksimum dan minimum *Buffer mode*.
6. Tambahkan *pin Analog Input* dan LCD, hingga sesuai dengan Gambar 8.11.



Gambar 8. 11 Skema rangkaian

7. Atur konfigurasi pada LCD sesuai dengan Gambar 8.12.



Gambar 8. 12 LCD konfigurasi

8. Klik menu *Pins* pada *Workspace Explorer*, kemudian lakukan konfigurasi seperti Gambar 8.13.

	Name	Port	Pin
	\LCD:LCDPort[6:0]\	P2[6:0]	1, 68, 66...62
	Analog	P3[1]	30

Gambar 8. 13 Konfigurasi semua pin yang digunakan

9. Konfigurasi pin PSoc ke LCD seperti Gambar 8.14.

PSoc Pin	LCD Module Pin	Keterangan
2.0	DB4	Data Bit 0
2.1	DB5	Data Bit 1
2.2	DB6	Data Bit 2
2.3	DB7	Data Bit 3
2.4	E	LCD Enable
2.5	RS	Register Select
2.6	R/W	Read/Not

Gambar 8. 14 Tabel hubungan port LCD

10. Masukkan program *code* pada *main.c*, seperti pada Gambar 8.14.

```
#include "project.h"
uint16 data;
uint16 v;
int main(void)
{
    CyGlobalIntEnable; /* Enable global interrupts. */
    LCD_Start();
    ADC_Start();
    ADC_StartConvert();
    for(;;)
    {
        if(ADC_IsEndConversion(ADC_RETURN_STATUS)) {
            data = ADC_GetResult16();
            v = ADC_CountsTo_mVolts(data);

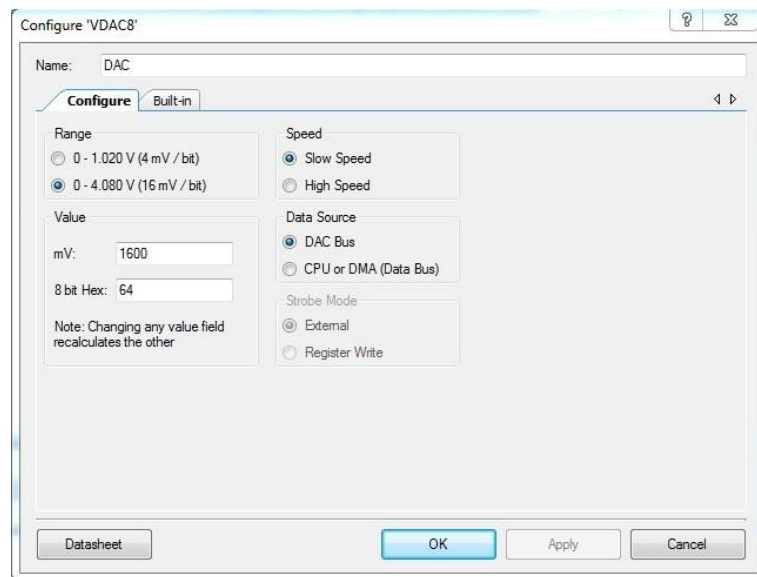
            LCD_ClearDisplay();
            LCD_Position(0,0);
            LCD_PrintString("tegangan:");
            LCD_Position(0,10);
            LCD_PrintNumber(v);
            LCD_Position(1,0);
            LCD_PrintString("desimal:");
            LCD_Position(1,9);
            LCD_PrintNumber(data);
            CyDelay(50);
        }
    }
}
```

Gambar 8. 15 Code di main.c

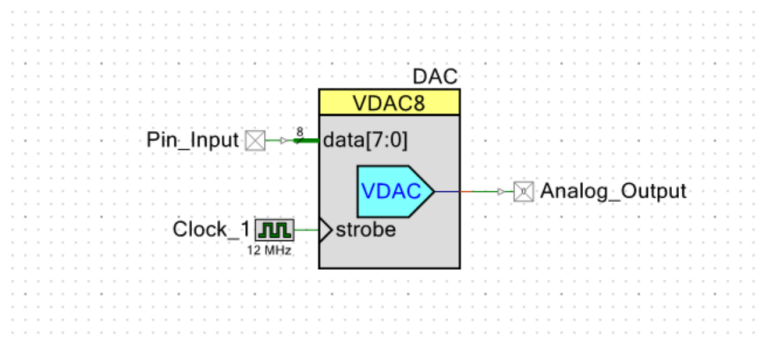
11. Lakukan *build* dan unduh program ke modul PSoC.

1.5.2 DAC

1. Lakukan prosedur yang telah disebutkan pada percobaan ADC untuk melakukan percobaan DAC. Gunakan blok VDAC. Lakukan konfigurasi seperti Gambar 8.16.
2. Tambahkan komponen-komponen lain hingga membentuk rangkaian seperti Gambar 8.17.
3. Konfigurasi Pin seperti Gambar 8.18.
4. Masukkan program sesuai kebutuhan seperti Gambar 8.19.
5. Unduh program ke dalam modul PSoC dan catat hasil percobaan. Ulangi percobaan dengan memvariasikan nilai resolusi seperti pada Tabel Hasil Pengamatan.



Gambar 8. 16 Konfigurasi VDAC



Gambar 8. 17 Skema Rangkaian

	Name /	Port	Pin	Lock
<input checked="" type="checkbox"/>	\Pin_Input[7:0]\	P0[7:0]	56_53, 51_48	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/>	Analog_Output	P15[5]	61	<input checked="" type="checkbox"/>

Gambar 8. 18 Konfigurasi pin yang digunakan

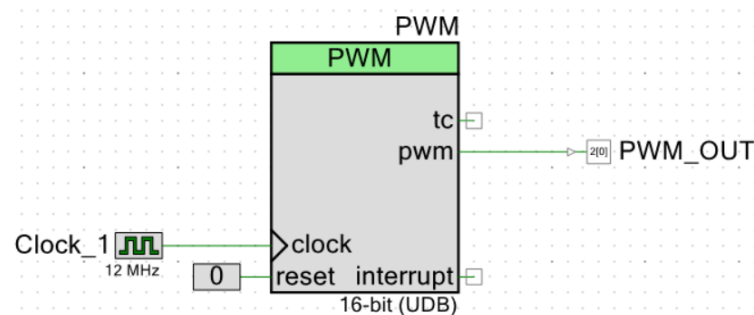
```
#include "project.h"

int main(void)
{
    CyGlobalIntEnable; /* Enable global interrupts. */
    DAC_Start();
    uint8 output_analog;
    for(;;)
    {
        output_analog = DAC_Data;
        CyDelay(100);
    }
}
```

Gambar 8. 19 Code di main.c

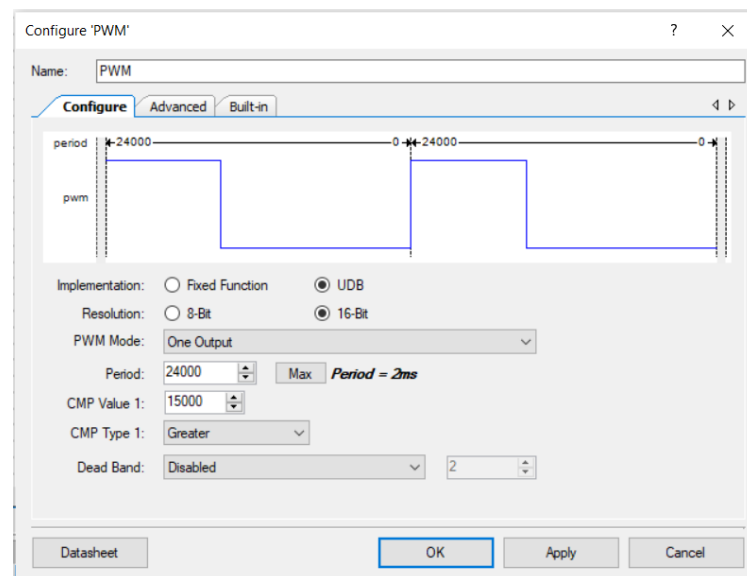
1.5.3 PWM

1. Buka PSoC Creator
2. Pada Workspace TopDesign.cysch masukkan komponen PWM yang didapat dari *Component Catalog*, lalu beri komponen tambahan seperti pada Gambar 8.20. Atur nilai *clock* menjadi 12 MHz.



Gambar 8. 20 Skema rangkaian

3. Atur PWM dengan klik 2 kali pada komponen, lalu atur blok PWM seperti pada Gambar 8.21. Atur *Implementation* menjadi **UDB**, *Resolution* menjadi **16-Bit**, dan *PWM Mode* menjadi **One Output**. Pemberian nama komponen akan mempengaruhi nama pada program.



Gambar 8. 21 Konfigurasi PWM

4. Pada pengaturan PWM, variasikan nilai **Period** untuk mendapatkan periode dari PWM, serta **CMP Value 1** untuk mengatur *duty cycle* dari *output* PWM. Nilai **Period** dan **CMP Value 1** diatur sedemikian rupa sehingga mendapatkan target frekuensi dan *duty cycle* seperti dalam tabel percobaan.

5. Beri tambahan program pada main.c untuk mengaktifasi blok PWM seperti pada Gambar 8.22.

```
#include "project.h"

int main(void)
{
    CyGlobalIntEnable; /* Enable global interrupts. */
    PWM_Start();

    /* Place your initialization/startup code here (e.g. MyInst_Start()) */

    for(;;)
    {
        /* Place your application code here. */
    }
}
```

Gambar 8. 22 Code di main.c

6. Klik menu **Pins** pada *Workspace Explorer*, kemudian lakukan konfigurasi pin.
7. Lakukan *build* dan unggah program ke modul PSoC.
8. Hubungkan pin *output* PWM ke *oscilloscope* lalu hitung berapa besar periode, frekuensi, serta *duty cycle* dari sinyal PWM yang terukur pada *oscilloscope*. Catat hasilnya pada tabel percobaan.

1.6 Tabel Percobaan

1.6.1 DAC

INPUT								OUTPUT
D7	D6	D5	D4	D3	D2	D1	D0	Tegangan pada Display multimeter (Vdc)
0	0	0	0	0	0	0	1	
0	0	0	0	0	0	1	0	
0	0	0	0	0	1	0	0	
0	0	0	0	1	0	0	0	
0	0	0	1	0	0	0	0	
0	0	1	0	0	0	0	0	
0	1	0	0	0	0	0	0	
1	0	0	0	0	0	0	0	
								1
								2
								3
								4

Variasikan hasil dengan menggunakan semua nilai *input range* yang tersedia pada sub menu config1, apa yang dapat Anda simpulkan?

1.6.2 ADC

NO	INPUT	OUTPUT		
	Vin (V)	Resolusi 8bit	Resolusi 12bit	Resolusi 16bit
1				
2				
3				
4				
5				
6				
7				
8				
9				
10				
11				

1.6.3 PWM

Frekuensi Clock: 12 MHz						
Target Frekuensi PWM (Hz)	Target <i>Duty Cycle</i> PWM (%)	<i>Period</i>	CMP Value 1	Periode Terukur	Frekuensi Terukur	<i>Duty Cycle</i> Terukur
500	20					
	50					
	70					
	90					
10k	20					
	50					
	70					
	90					
100k	20					
	50					
	70					
	90					
1M	20					
	50					
	70					
	90					

DAFTAR PUSTAKA

- David Bucchlah, Wayne McLahan, “Applied Electronic Instrumentation And Measurement”, MacMilian Publishing Company, 1992.
- Eggebrecht, Lewis C., Interfacing to The IBM PC, Howard W. Sams & Co., Indianapolis, 1987.
- Hall, Douglas V., Microprocessor and Interfacing : Programming and Hardware, McGraw Hill Book Company, Singapore, 1987.
- Hodges D., Jacson, Nasution S. “Analisa dan Desain Rangkaian Terpadu Digital”, Erlangga, Jakarta, 1987.
- Ian Robertson Sinclair, Suryawan, “Panduan Belajar Elektronik Digital”, ElexMedia Komputindo, Jakarta, 1993.
- K.F. Ibrahim, “Teknik Digital”, Andi Offset , Jakarta, 1996.
- Sendra, Smith, Keneth C., “ Rangkaian Mikroelektronika”, Penerbit Erlangga, Jakarta, 1989.
- Singh, Avtar & Walter A. Triebel, The 8088 Microprocessor : Programming, Interfacing, Software, Hardware and Applications, Prentice-Hall International Inc., New Jersey, 1987.
- Sofyan H. Nasution, “Analisa dan Desain Rangkaian Terpadu Digital”, Penerbit Erlangga, Jakarta, 1987.
- Sutrisno, “Rangkaian Digital dan Rancangan Logika”, Erlangga, Jakarta, 1990.
- Tokheim. R., “Elektronika Digital”, Edisi Kedua, Erlangga, Jakarta, 1995.
- Wijaya Widjanarka N., “Teknik Digital”, Erlangga, Jakarta, 2006.
- R. Erickson, D. Maksimovic, "Fundamentals of Power Electronics", Kluwer Academic Publishers, 2001.
- F. Farhan and B. Sujanarko, "Pengaruh Frekuensi dan Duty Cycle pada Ripple Tegangan Buck Converter," Dielektrika – Department of Electrical Engineering University of Mataram, vol. 9, no. 1, pp. 51-61, Feb. 2022.