



PROGRAM KOMPUTER

**Program Simulasi PSIM Pembangkitan Sinyal PWM
(Pulse Width Modulation) Berdasarkan STM32
dengan Memanfaatkan Simplified C-Block**

Pencipta

**Muhammad Rizani Rusli, Mochamad Ari Bagus Nugroho, Ahmad Firyal Adila,
Muhammad Nizar Habibi, Luki Septya Mahendra**

Uraian Singkat

Pulse Width Modulation (PWM) merupakan sebuah teknik yang populer dan berguna untuk mengendalikan rangkaian analog dengan output digital mikrokontroler. PWM dapat digunakan pada banyak aplikasi, mulai dari komunikasi hingga kontrol konversi daya. Misalnya, PWM biasa digunakan untuk mengontrol kecepatan motor listrik, mengatur tegangan luaran rangkaian elektronika daya, dan banyak lagi.

PWM pada dasarnya merupakan sinyal gelombang persegi *unipolar* digital di mana durasi waktu ON dapat disesuaikan (atau dimodulasi) sesuai keinginan. Dengan cara ini daya yang dikirim ke beban dapat dikontrol dari pembangkitan PWM-nya yang dalam hal ini lumrahnya menggunakan mikrokontroler.

Dalam STM32, PWM terintegrasi dengan *timers* (TIMx). Aplikasi *timer/ counter* secara luas dapat menggunakan TIMx. *Timers* dalam STM32 rata-rata berkapasitas 16-bit. Setiap *timer* menyematkan sebuah *linear clock prescaler* yang memungkinkan untuk membagi *timer* dengan bilangan bulat apa pun antara 1 dan 65536. Hal ini memungkinkan kecepatan penghitungan disesuaikan dengan tepat. *Auto reload register* (ARR) menentukan periode penghitungan. Dalam mode *counting down*, *counter* secara otomatis dimuat ulang dengan nilai periode saat *underflow*. Dalam mode *counting up*, *counter* beroperasi dan diatur ulang ketika melebihi nilai ARR.

Salah satu cara mensimulasikan PWM yaitu dengan menggunakan *software* simulasi PSIM. Akan tetapi semua *tools* PWM yang ada di PSIM tidak menganut konsep dan tatacara pembangkitan berdasarkan mikrokontroller STM32. Dalam hak cipta ini, telah diusulkan simulasi pembangkitan sinyal PWM berdasarkan konsep STM32 dengan memanfaatkan *simplified C-Block*. Isi dari *code* blok dijabarkan dengan detail berikut dengan perhitungan dan caranya. Rangkaian dan program ini bisa berguna untuk pengaplikasian PWM yang digunakan untuk konverter daya, pengemudian motor listrik, rangkaian elektronika daya, maupun rangkaian elektronika lain yang membutuhkan PWM sebagai *trigger*-nya.

Deskripsi Program

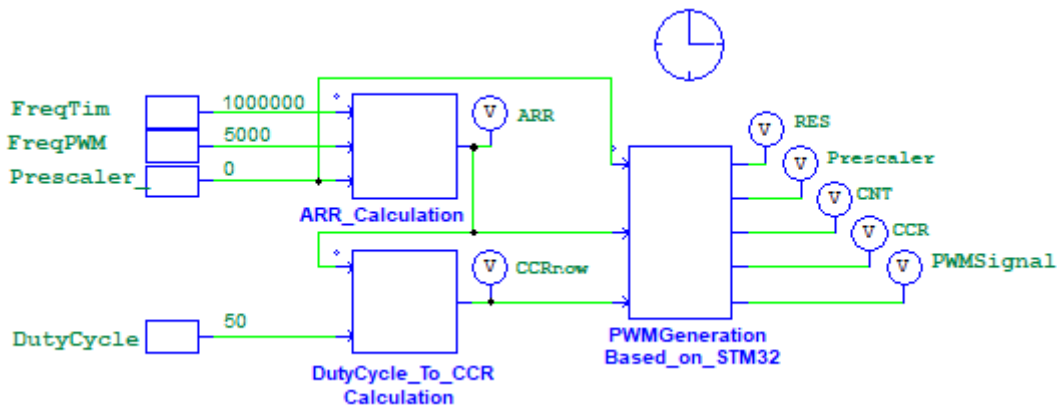
Konsep dasar pembangkitan sinyal PWM adalah dengan membandingkan dua sinyal kontrol, yaitu sebuah sinyal *carrier* dan sebuah sinyal modulasi. Sinyal *carrier* merupakan gelombang *triangular* yang dibangkitkan dari *counter* (CNT). Sinyal modulasi merupakan sebuah nilai DC yang dibangkitkan oleh CCR. Luaran sinyal PWM (PWMSignal) akan menghasilkan sinyal ON saat *counter* mulai menghitung dari null. Ketika *counter* cocok dengan nilai CCRx, output PWMSignal dimatikan. Ketika *counter* nilainya sama dengan ARR, *counter* dihapus ke null dan output dihidupkan dan *counter* mulai menghitung lagi.

Dalam membangkitkan sinyal PWM berdasarkan konsep STM32, ada beberapa tahapan dan formulasi. Langkah-langkahnya yaitu menentukan frekuensi dari PWM, frekuensi *timer*, dan *prescaler*. Kemudian ARR dapat dihitung melalui Persamaan (1).

$$ARR = \left(\frac{Freq_{Tim}}{Freq_{PWM} \cdot (Prescaler + 1)} \right) - 1 \quad (1)$$

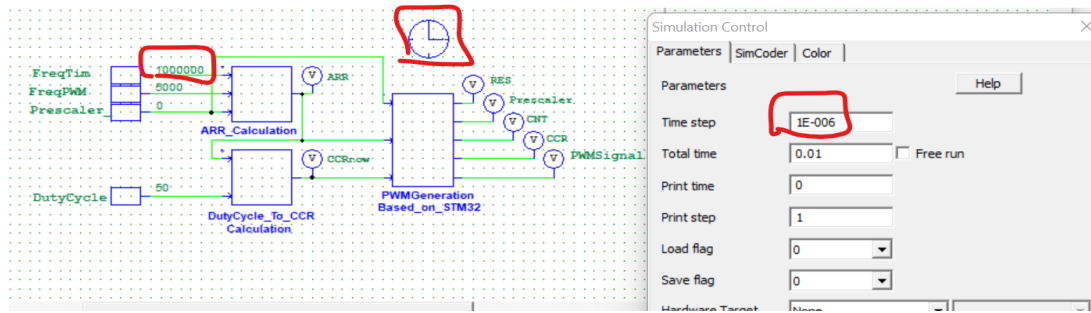
Perhitungan *duty cycle* PWM didasarkan dari presentase dari nilai ARR dan menghasilkan nilai CCRx melalui persamaan (2).

$$CCR = \frac{Duty}{100} \cdot ARR \quad (2)$$



Gambar 1. Rangkaian simulasi PSIM pembangkitan sinyal PWM berdasarkan STM32 dengan memanfaatkan *simplified C-Block*

Detail program rangkaian simulasi PSIM pembangkitan sinyal PWM berdasarkan STM32 dengan memanfaatkan *simplified C-Block* terlihat pada Gambar 1. Terlihat diperlukan beberapa nilai masukan yang berupa frekuensi *timer* (FreqTim) dalam Hz, frekuensi PWM (FreqPWM) dalam Hz, *Prescaler*, dan *duty cycle*. FreqTim dapat diisi sesuai dengan frekuensi *timer* STM32 yang bernilai 168 MHz atau 84 MHz tergantung *Advanced Peripheral Bus* (APB) *signal support*, pada contoh ini diberikan nilai 1 MHz untuk mempermudah dan mempercepat kerja simulasi dari software PSIM. Nilai frekuensi *timer* sebesar 1 Mhz ini dikonversi ke dalam satuan *second* dan dimasukkan ke nilai *time step* pada *simulation control* software PSIM seperti pada Gambar 2.



Gambar 2. Cara pengisian Simulation Control

Terdapat tiga blok *simplified C-Block* yaitu *ARR_Calculation*, *DutyCycle_To_CCR_Calculation*, dan *PWMGeneration_Based_on_STM32*. Isi *code* dari blok *ARR_Calculation* seperti berikut:

```
static double FreqTim, FreqPWM, Prescaler, ARR;
FreqTim=x1;           //in Hz unit
FreqPWM=x2;           //in Hz unit
Prescaler=x3;
ARR= (FreqTim/ (FreqPWM*(Prescaler+1)))-1;
y1=ARR;
```

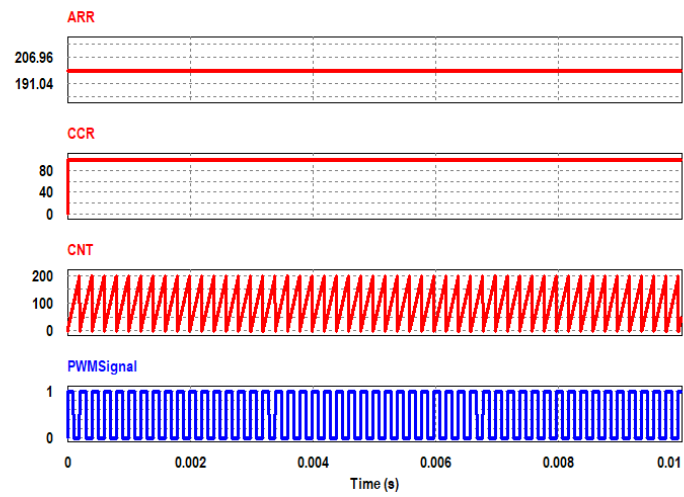
Sementara isi *code* blok *DutyCycle_To_CCR_Calculation* seperti berikut:

```
static double Duty, ARR,CCRvalue;
ARR=x1;
Duty=x2;              //in %unit
CCRvalue= (Duty/100)*ARR;
y1=CCRvalue;
```

Isi *code* blok *PWMGeneration_Based_on_STM32* seperti berikut:

```
static double Res, Prescaler, CNT, ARR, CCR, PWM_Ch1;
Prescaler=x1;
ARR=x2;
CCR=x3;
//---**Timer Resolution Calculation / prescaler output value**-----
Res++;
if(Res>Prescaler)
{
    CNT++;
    Res=0;
}
//---**Triangular generation**-----
if (CNT==ARR)
{
    CNT=0;
}
//---**PWM generation 1 channel : Compare triangular and CCR value**--
if (CNT<CCR)
{
    PWM_Ch1=1;
}
else
{
    PWM_Ch1=0;
}
y1=Res;
y2=Prescaler;
y3=CNT;
y4=CCR;
y5=PWM_Ch1;
```

Hasil *running* keseluruhan rangkaian simulasi PSIM pembangkitan sinyal PWM berdasarkan STM32 dengan memanfaatkan *simplified C-Block* terlihat pada Gambar 3 berikut ini.



Gambar 3. Hasil running keseluruhan program simulasi.