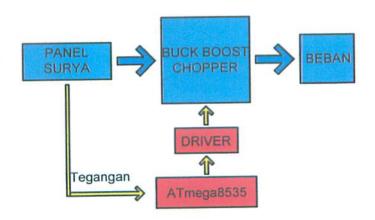
BAB III

PERANCANGAN BUCK BOOST CHOPPER SEBAGAI MPPT DENGAN KONTROL DIGITAL BERBASIS ATmega8535

3.1 Pendahuluan

Pada tugas akhir ini, penulis akan mengimplementasikan *Buck Boost Chopper* sebagai MPPT dengan kontrol digital berbasis ATmega8535. Rancangan sistem tersebut dapat dilihat dalam diagram block berikut ini:



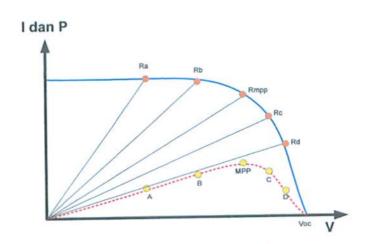
Gambar 3.1 Diagram Block Sistem yang akan dibuat

Mengacu pada diagram block di atas, panel surya akan terhubung ke *Buck Boost Chopper* yang difungsikan sebagai MPPT. *Buck Boost Chopper* akan terhubung dengan beban. Untuk kontrolnya, tegangan pada panel surya akan disensor kemudian hasil sensor tersebut akan dimasukkan ke dalam ATmega8535. Di dalam ATmega8535 inilah akan dilakukan proses kalkulasi dengan kontrol hysteresis secara digital. Setelah mikrokontrol menghasilkan suatu respon maka akan dibawa menuju driver yang berfungsi mengontrol saklar pada chopper sesuai

dengan respon pada mikrokontrol. Dengan menggunakan sistem *close-loop* ini diharapkan *Buck Boost Chopper* akan dapat berfungsi sebagai MPPT dengan baik.

3.2 **MPPT**

Berdasarkan karakterisik pada gambar 3.2, maka diperoleh analisa bahwa daya PV tidak akan maksimal jika diberikan beban yang berbeda-beda. Oleh karena itu, untuk memungkinkan didapat daya yang selalu maksimal dengan beban yang berbeda, maka diperlukan MPPT. Dengan alat ini, maka PV akan seolah-olah selalu bekerja pada titik dimana arus dan tegangan maksimal untuk untuk mendapat daya maksimal. Untuk mendapatkan daya maksimal, PV harus bekerja pada titik MPP-nya, dimana diperoleh nilai terbesar dengan tegangan terbesar dan arus terbesar.



Gambar 3.2 Kurva karakteristik PV terhadap pembebanan

Titik operasi A dan B merupakan titik operasi PV yang terbentuk akibat pembebanan dengan nominal beban lebih kecil dari Rmpp. Sedangkan titik operasi C dan D terjadi akibat PV dibebani dengan beban yang lebih besar daripada Rmpp.

Berdasarkan titik operasi tersebut maka MPPT harus berupaya untuk menjadikan setiap beban dari sistem ini terlihat atau dianggap sebagai Rmpp yang nantinya akan menghasilkan MPP. Untuk melakukan hal tersebut maka MPPT harus bisa menggeser titik operasi dari setiap pembebanan agar selalu mendekati MPP.

Pergeseran yang dapat dilakukan untuk menaikkan daya PV mendekati MPPT adalah pergeseran dari titik daya A ke B dan C ke D. Sedangkan untuk pergeseran dari B ke A dan C ke D akan mengakibatkan daya PV berkurang.

Untuk melakukan pergeseran daya PV dari titik A ke titik B(ke kanan) maka arus PV harus diturunkan atau tegangan PV harus dinaiikan. Sehingga menghasilkan Δ I akan bernilai negatif(-). Dan Δ V akan bernilai positif(+). Perubahan tersebut akan menyebabkan daya PV akan bertambah besar dengan $P_A < P_B$.

Untuk melakukan pergeseran daya PV dari titik D ke titik C(ke kiri) maka arus PV harus dinaikkan atau tegangan PV harus diturunkan. Sehingga menghasilkan Δ I akan bernilai positif(+). Dan Δ V akan bernilai negatif(-). Perubahan tersebut akan menyebabkan daya PV akan bertambah besar dengan $P_D < P_C$.

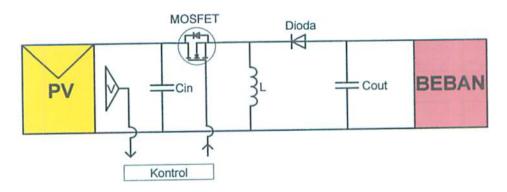
Pada tugas akhir ini,penulis akan mengaplikasikan MPPT dengan kontrol tegangan PV. Sehingga tegangan PV akan disensor dan hasil sensor tersebut akan diolah dan dibandingkan dengan nominal Vmpp yang telah ditentukan. Ketika titik operasi PV berada pada sisi kiri MPP,maka tegangan harus ditambahkan sampai daya PV mencapai MPPT. Sedangkan jika titik operasi PV berada pada

sisi kanan MPP tegangan harus diturunkan. Pengaturan tiggi rendahnya tegangan PV dapat dilakukan dengan cara merubah *duty cycle(D)* pada konverter.

Jika tegangan PV kurang dari Vmpp, maka D harus diturunkan, agar Arus PV semakin kecil dan tegagan PV semakin besar sehingga daya PV bergerak ke kanan mendekati MPP. Jika tegangan PV lebih besar dari Vmpp, maka D harus dinaikkan, agar Arus PV bertambah dan tegangan PV berkurang sehingga daya PV bergerak ke kiri mendekati MPP.

3.3 Rangkaian Daya Buck Boost Chopper

Rangkaian daya yang akan diterapkan dalam Tugas Akhir ini adalah sebagai berikut:



Gambar 3.3 Rangkaian Daya Buck Boost Chopper

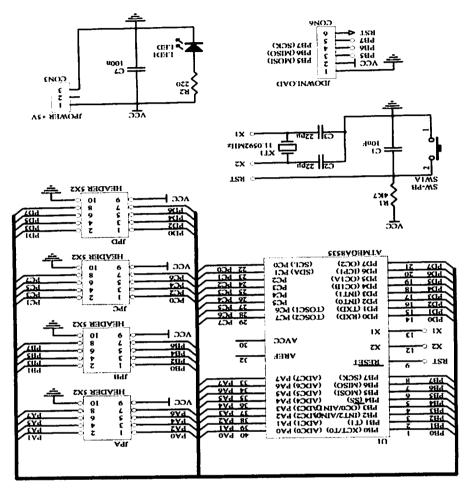
Prinsip kerja dari rangkaian ini adalah ketika saklar ON, arus dari PV akan akan mengisi kapasitor masukan. Setelah tegangan kapasitor input sama dengan tegangan PV maka kapasitor input akan mengalirkan arus ke L melalui saklar dan kembali ke kapasitor input. Disaat yang bersamaan, kapasitor keluaran akan membuang energi dengan mensuplai beban.

Ketika saklar OFF,PV akan mengisi kapasitor input dan L akan mengalirkan arus untuk mengisi kapasitor output serta mensuplai beban. Pada kapasitor inputlah nantinya proses pembentukan tegangan input akan terjadi. Dimana tegangan input akan selalu dibuat mendekati Vmpp.

SE28agamTA lontrol AImega8535

Untuk memfungsikan ATmega8535 dibutuhkan suatu rangkaian dasar atau

Sistem Minimum ATmega 8535. Berikut ini adalah rangkaiannya:

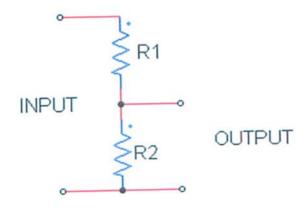


S£288gamTA muminiM naiaygna A. & Tadma D

Rangkaian Sistem minimum ATmega8535 ini terdiri dari rangkaian penghasil Osilator, rangkaian untuk koneksi dan rangkaian power. Rangkaian osilator terdiri dari komponen seperti Kristal, kapasitor dan juga tombol reset. Sedangkan untuk rangkaian koneksi inilah yang nantinya akan menghubungkan ATmega8535 dengan perangkat lain. Koneksi yang ada adantara lain, Port ISP untuk memasukka program dan PORT A-D untuk menghubungkan port ATmega8535 dengan perangkat lain.

3.5 Rangkaian Sensor Tegangan

Mengacu pada prinsip dasar sensor yaitu menghasilkan output yang berbentuk sama dengan input namun dengan level ukuran yang berbeda,maka penulis merancang sebuah sensor tegangan dengan menggunakan rangkaian pembagi tegangan. Rangkaian ini terdiri dari sejumlah resistor yang terhubung seri kemudian diparalel dengan input. Keluaran diambil dari titik pertemuan antar resistor. Nominal keluaran akan berbeda-beda tiap titik namun memiliki bentuk yang sama. Berikut merupakan ilustrasinya:



Gambar 3.5 Rangkaian pembagi tegangan

Untuk mendapatkan nominal output diperguanan rumus berikut ini:

$$V_{OUT} = \frac{R2}{R1 + R2} x V_{IN} \tag{3.1}$$

Oleh karena itu,dengan mengubah-ubah nominal resistor tersebut, akan menghasilkan tegangan output yang dapat diatur ukurannya. Misalnya jika kita menginginkan tegangan output 0,1 kali dari tegangan input, maka perbandingan resistor yang dipasang R1:R2=9:1.

Karena rangkaian pembagi tegangan ini difungsikan sebagai sensor maka resistor yang digunakan harus memiliki tahanan yang besar agar arus listrik dari rangkaian daya tidak drop atau terserap pada rangkaian sensor tegangan.

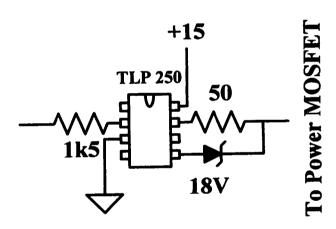
3.6 Rangkaian Driver

Saklar statis yang dimplementasikan dengan MOSFET memiliki tiga buah terminal, yaitu G (gate), D (drain) dan S (source). Pada implementasi MOSFET model NPN jenis enchasment arus mengalir dari D ke S jika tegangan G-S melebihi nilai ambangnya (threshold), jika tidak maka arus akan ditahan (blocking).

Saklar daya sejenis MOSFET atau IGBT bekerja berdasar pulsa pemicuan dari rangkaian kontrol pada gate-nya tetapi bekerja pada orde daya yang lebih tinggi sehingga untuk mengendalikan setiap saklar daya diperlukan rangkaian driver. Pada aplikasi rangkaian inverter ini isolasi dilakukan dengan menggunakan optocoupler jenis TLP 250 yang terpasang dengan rangkaian driver. Rangkaian driver berfungsi untuk memindahkan sinyal picu dari sistem kontrol ke sistem daya dengan memisahkan bagian ground daya dari ground kontrol, karena

keduanya bekerja pada catu tegangan yang berbeda. Untuk itu pada setiap rangkaian driver harus dicatu dengan catu daya tersendiri yang sesuai dengan tegangan yang dibutuhkan oleh terminal G (gate) MOSFET yang digunakan.

Kerusakan saklar statis MOSFET juga sering terjadi karena panas yang ditimbulkan dari gesekan pulsa yang melewatkan arus pada saklar tersebut, untuk itu demi keamanan saklar daya tersebut rangkaian driver juga dilengkapi dengan deadtime untuk mengatur perpindahan pulsa pemicuan pada setiap saklar dalam satu lengan.



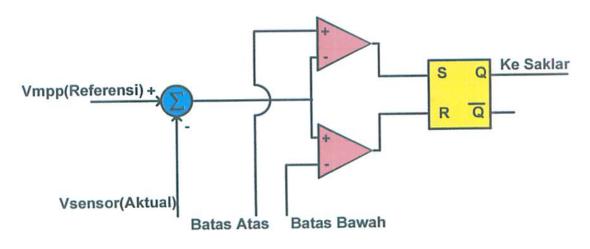
Gambar 3.6 Rangkaian driver

Selain rangkain driver pada setiap aplikasi saklar daya atara gate dan source diberikan sebuah dioda Zener, (dalam aplikasi ini menggunakan zener 18 volt). Dioda zener berfungsi untuk melindungi driver TLP 250 dari umpan balik tegangan daya apabila terjadi hubung singkat pada rangkaian daya. Dipilih nilai 18 volt karena tegangan kerja maksimal driver sebesar 22 volt sehingga apabila terjadi umpan balik tegangan, zener bisa menahannya pada nilai aman terhadap kerusakan TLP 250.

3.7 Perancangan Sistem Kontrol Digital

Sistem kontrol yang akan diimplementasikan merupakan sisem kontrol digital. Proses pemrograman dilakukan dengan bantuan software Code Vision AVR. Pada software inilah nantinya program kontrol digital akan dibuat dengan menggunaka bahasa pemrograman C. Program yang dibuat akan berpedoman pada karakteristis kontrol secara analog. Karakteristik ini meliputi prinsip kerja dan juga tabel kebenaran tiap komponen.

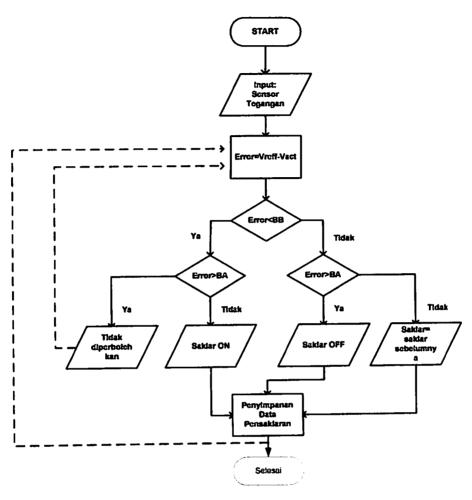
Berikut ini adalah sistem kontrol analog yang akan didigitalkan:



Gambar 3.7 Rangkaian Kontrol Analog

Seperti yang terlihat pada gambar di atas,tegangan aktual dari sensor akan dibandingkan dengan tegangan referensi Vmpp yang telah ditentukan. Hasilnya merupakan nominal selisih antara Vactual dan Vreferensi yang disebut *Error*. Error akan dimasukkan kedalam rangkaian hysteresis kontrol dan menghasilkan sinyal pensaklaran bagi MOSFET.

Sehingga alur pemrograman sistem kontrol digital ini adalah sebagai berikut:



Gambar 3.8 Flow Chart Sistem Kontrol

3.8 Pemrograman pada ATmega8535

Program kontrol dibuat menggunakan software Code Vision AVR.

Pemrograman menggunakan bahasa program C. Dalam melakukan pemrograman, sistem kontrol dibagi mejadi berbagai tahapan yaitu:

- 1. Pembacaan sensor
- 2. Pembuatan Error
- 3. Perbandingan antara Error dengan Pita Atas dan Pita Bawah

 *Hysteresis**
- 4. FLIP FLOP.

3.8.1 Pembacaan Sensor

Karena sinyal hasil sensor adalah analog dan mikrokontrol ATmega8535 mengolah data secara digital maka untuk pembacaan sensor tegangan pada rangkaian daya, penulis meggunakan fitur ADC(Analog to Digital Converter). Fitur ini mengubah besaran analog berupa tegangan menjadi besaran digital.

Proses pengubahan sinyal analog ke digital adalah tegangan analog dibandingkan dengan tegangan referensi, kemudian hasilknya dikalikan dengan nominal bilangan digital maksimal yang mampu diolah oleh mikrokontrol.

$$V_{dig} = \frac{Vin}{Vreff} x 2^n \tag{3.2}$$

Pada ATmega8535, fitur ADC dapat merubah tegangan Analog menjadi digital 8 bit dan 10 bit. Untuk mengaktifkan ADC diperlukan proses inisiasi atau pemanggilan pada register-register terkait. Inisiasi ADC tersebut dapat dilihat pada gambar dibawah:

```
#define ADC_VREF_TYPE 0x60
29
30
     // Read the 8 most significant bits
31
     // of the AD conversion result
32
33 Dunsigned char read_adc(unsigned char adc_input)
34 □ {
     ADMUX=adc_input | (ADC_VREF_TYPE & 0xff);
35
     // Delay needed for the stabilization of the ADC input voltage
36
     delay us(10);
37
      // Start the AD conversion
38
     ADCSRA | =0x40;
39
      // Wait for the AD conversion to complete
40
     while ((ADCSRA & 0x10)==0);
41
     ADCSRA | =0x10;
42
     return ADCH;
43
44
```

Gambar 3.9 Inisiasi ADC pada ATmega 8535

Setelah proses inisiasi, maka PORTA ATmega8535 akan aktif sebagai Pin Input ADC, sehingga tidak dapat digunakan untuk proses lain. Dengan demikian

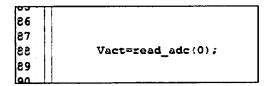
berarti jumlah chanel atau kanal inputan ADC adalah 8 buah dan kesemuanya dapat diaktifkan melalui proses pembacaan ADC. Selain inisiasi, ADC pada ATmega8535 memelukan pengaturan mengenai frekwensi dari pembacaan ADC, tegangan referensi dan juga berapa bit ADC yang diinginkan. Setingan tersebut dapat dilihat pada gambar berikut:

```
// ADC initialization
     // ADC Clock frequency: 691,200 kHz
70
     // ADC Voltage Reference: AVCC pin
     // ADC High Speed Mode: Off
72
     // ADC Auto Trigger Source: ADC Stopped
     // Only the 8 most significant bits of
74
     // the AD conversion result are used
75
     ADMUX=ADC_VREF_TYPE & 0xff;
     ADCSRA=0x84;
     SFIOR&=0xEF;
78
79
```

Gambar 3.10 Setting ADC pada ATmega 8535

Tegangan referensi yang akan dipakai menggunakan tegangan AVCC pin atau tegangan suplay ATmega8535 yaitu sebesar 5 Volt. Tegangan tersebut didapatkan dengan cara menghubungkan pin AREFF dengan AVCC. Jumlah bit yang dipakai adalah 8 bit. Dengan demikian jika tegangan Input ADC sebesar 5 Volt maka akan terkonversi menjadi 28 atau 256. Dan untuk tegangan lain yang berada dikisaran 0V-5V hasilnya berkisar dari 0-256.

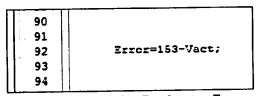
ADC yang akan digunakan untuk membaca sensor tegangan berada pada PORTA.0 atau bisa disebut ADC Chanel 0. Berikut ini merupakan perintah untuk membaca ADC pada chanel 0 dan kemudian disimpan ke dalam suatu variabel "Vact" guna proses yang lebih lanjut.



Gambar 3.11 Pembacaan ADC pada ATmega 8535

3.8.2 Pembuatan Error

Setelah data sensor didapatkan, maka proses kontrol dilanjutkan kepada tahap pembuatan Error. Error sendiri merupakan selisih antara tegangan referensi dengan tegangan aktual. Berarti pada kontrol ini, Error adalah selisih antara Vmpp(referensi) dengan Vpv(Aktual). Karena Vaktual yang didapatkan telah dikonversi menjadi bilangan digital, maka Vreferensinya pun juga harus didigitalkan juga. berikut merupakan listing programnya:

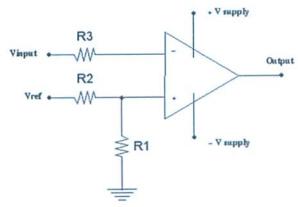


Gambar 3.12 Listing Pembuatan Error

Nominal tegangan referensi yang telah ditentukan dikurangi dengan Vact yang merupakan hasil pembacaan ADC dari sensor tegangan. Hasil dari proses tersebut disimpan dalam Variable "Error".

3.8.3 Perbandingan Error Dengan Pita Atas dan Pita Bawah Hysteresis.

Setelah nominal Error didapatkan maka dilanjutkan ke tahap dimana Error akan dibandingkan dengan pita atas serta pita bawah Hysteresis. Proses membandingkan tersebut mengambil dari prinsip kerja komparator yang dipakai pada rangkian Hysteresis analog. Berikut ini adalah penerapan komparator analog:



Gambar 3.13 Komparator Analog

Prinsip kerjanya adalah membandingkan Vinput (-) dengan Vreff (+). Jika masukan (-) lebih besar dari masukan (+) maka, keluaran op-amp akan menjadi sama dengan –Vsupply, apabila tegangan masukan (-) lebih kecil dari masukan (+) maka keluaran op-amp akan menjadi sama dengan + Vsupply.

Berdasarkan pada prinsip kerja komparator analog di atas, maka didapatkan listing program untuk membandingkan error dengan dengan pita atas serta pita bawah Hysteresis sesuai dengan rangkaian analognya. Listing nya adalah sebagai berikut:

```
//SET RISET
96
              if(Error>=BA)
97
98
99
              if (Error<BA)
100
               R=0;}
101
102
              if (Error<=BB)
103
               S=1;}
104
105
              if (Error>BB)
106
              { S=0;}
107
108
109
```

Gambar 3.14 Listing Program Komparator

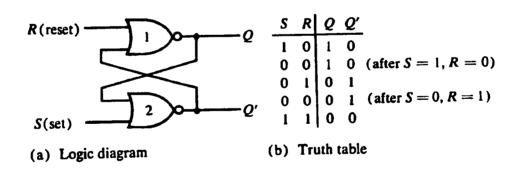
Seperti pada rangkaian analognya,hasil komparasi antara Batas Atas dan error akan dijadikan sebagai masukan R(Reset) pada Flip Flop. Sedangkan untuk hasil komparasi *error* terhadap batas bawah akan dijadikan nominal S(Set).

Nominal batas atas(BA) dan batas bawah (BB) sendiri telah dimasukkan melalui listing berikut:

Gambar 3.15 Listing pengisian nilai BA dan BB

3.8.4 FLIP FLOP

Tahap terakhir dalam sistem kontrol ini adalah membentuk pensaklaran dengan algoritma yang sesuai prinsik kerja Flip Flop. Flip Flop yang dipakai pada sistem analog adalah jenis S-R atau Set-Reset. Di bawah ini merupakan rangkaian ekuivalen SR Flip Flop dan tabel kebenarannya.



Gambar 3.16 Rangkaian SR Flip Flop dan tabel kebenarannya

Jika S=1 dan R=0 maka output(Q) akan bernilai 1. Jika S=0 dan R=1 maka output(Q) akan bernilai 0. Namun ketika S=0 dan R=0 maka output(Q) akan sama dengan nilai Q sebelumnya. Kondisi S=1 dan R=1 adalah kondisi Forbiden atau kondisi yang tidak diperbolehkan. Pada kontrol ini, nilai S dan R didaptkan dari

proses komparasi sebelumnya sehingga listing programnya adalah sebagai berikut ini:

```
//FLIP FLOP
117
            if((R==1)&&(S==0))
118
             { Switch=0;}
119
120
             if((R==0)&&(S==1))
121
             { Switch=1;}
122
123
             if((R==0)&&(S==0))
124
             { Switch=Lastout; }
125
126
             Lastout=Switch;
127
128
129
```

Gambar 3.17 Listing program Flip Flop

Pada listing di atas, Jika S=1 dan R=0 maka ouput(switch) akan bernilai 1. Sedangkan jika S=0 dan R=1 maka ouput(switch) akan bernilai 0. Namun jika S=0 dan R=0 maka ouput(switch) akan mengeluarkan nilai yang sama dengan output sebelumnya. Dalam hal ini adalah variable "Lastout" yang menyimpan data pensaklaran terakhir. Pada sistem kontrol ini, output akan dihubungkan ke rangkaian driver melalui PORTB.6.