EE318 Electronic Design Lab Project Report, EE Dept, IIT Bombay, April 2007

Solar Cell Battery Charger Using Maximum Power Point Tracking Scheme

Group No: B02

Gaurav Sharma (04007006) <gauravsharma@ee.iitb.ac.in>
Priyadarshan Gupta (04007007) <pgupta@ee.iitb.ac.in>
Abhishek Jangir (04007010) <jangir@ee.iitb.ac.in>
Vivek Kumar Manglik (04010033) <vmanglik@ee.iitb.ac.in>

Supervisor: Prof. Dipankar

Abstract

The objective of this design project is to build a battery charging system for charging rechargeable NimH (Nickel metal hydride) batteries at maximum efficiency with input as the solar power. It contains implementation of a control system for tracking the maximum power point which provides feedback to the power extracting circuit to adapt accordingly. The power extracting circuit is a Buck converter (a DC to DC converter) and the control system is a MSP430f2012 microcontroller which regulates the duty cycle of the Buck converter.

1. Introduction

In today's world power is a very important necessity of our lives. However, reliable power is not easily available everywhere. Here we are concentrating to develop an alternative source of power for rural India where power supply is very erratic and unreliable. The basic concept is to store the energy in a rechargeable battery which can be used later on for small power applications such as torches. A number of viable ways do presently exist to charge a battery. One of them, on which we are going to concentrate, is to charge a battery by making the use of solar energy. A very crude and simple way to charge a battery is by connecting the solar panel directly to the battery terminal via diode. The light intensity should be such that the output voltage of the panel is higher than that of the battery for charging which puts a constraint on utilizing the solar energy fully. To overcome the above short-coming, we attempted to design a buck DC-DC converter that can convert the voltage in the input range to a particular voltage in the output range so that the charging process is completely under our control.

Specifications of DC to DC converter (Buck converter)
Input voltage – 3-10 V
Output Voltage – 2.4 – 3.5 V
Specifications of battery
NimH AA type rechargeable battery 1.2 V, 1300 mAh

Specifications of low power microprocessor used for control
MSP430F2012 (14 pin) Dual Inline Package
220 microA, 1.8- 3.6 V (rated)

2. Design and Implementation

The simplest way to reduce a DC voltage is to use a voltage divider circuit, but voltage dividers waste energy, since they operate by bleeding off excess voltage as heat. A Buck converter is a step-down DC to DC converter using reactive elements. A simple circuit diagram of a Buck converter is as follows:

Fig. 1: Circuit diagram of a Buck converter

The duty cycle of a buck converter corresponds to the duty cycle of the switch S. When switch S is closed, input side source pushes the current through inductor into the capacitor and the input current increases with rate proportional to the difference between source and load voltages. When switch is opened the inductor current decreases due to the voltage developed on the capacitor. When this operation of opening-closing is performed at high rate the voltage on capacitor is maintained almost constant and equal to V_o. The variation of Inductor current in a buck converter is shown below:

Fig. 2: Inductor current as a function of time in a Buck converter

The Buck converter has the output current almost constant (equal to inductor current), however the input stage current is variable. Input current is zero when switch is open, therefore source doesn't supply power all the times. To rectify this, a small LC filter can be included to ensure constant current in solar panel. The schematic modifies to the following.

Fig. 3: Power extracting circuit for the required system

3. The Solar Panel (3W)

The V-I characteristics of the 3 Watt polycrystalline solar panel is plotted using measured real data in bright sunlight around 2 pm.

S.No.	Voltage (volts)	Current (amperes)	Power (watts)
1	10.2 (open circuit)	0	0
2	9.11	0.177	1.62
3	8.98	0.198	1.78
4	8.78	0.216	1.90
5	8.55	0.240	2.05
6	8.31	0.242	2.013
7	8.16	0.254	2.074
8	8.12	0.269	2.19
9	7.88	0.270	2.133
10	7.82	0.276	2.16
11	7.57	0.287	2.178
12	7.47	0.292	2.188
13	7.17	0.298	2.142
14	6.81	0.304	2.07
15	6.27	0.310	1.946
16	5.14	0.315	1.62
17	3.96	0.314	1.24
18	3.19	0.312	0.997
19	2.49	0.311	0.775
20	1.84	0.306	0.564
21	1.07	0.260	0.279
22	0.71	0.236	0.168
23	0.56	0.254	0.142
24	0.03	0.042	0.0012

The plot for the V-I characteristics of the solar panel looks like

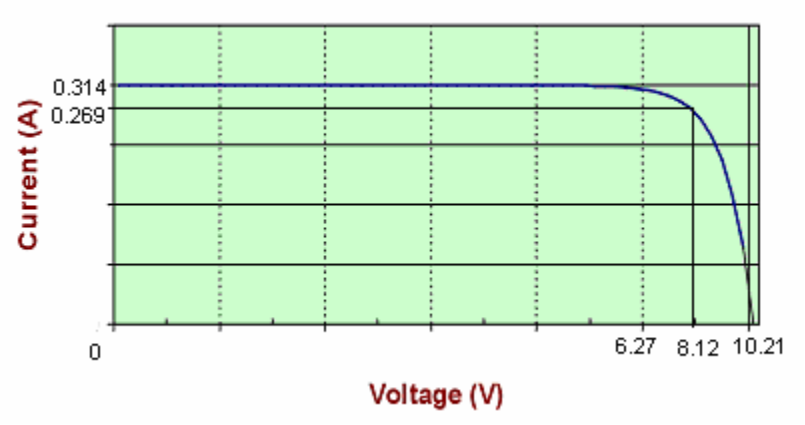


Fig. 4: Solar panel V-I characteristics

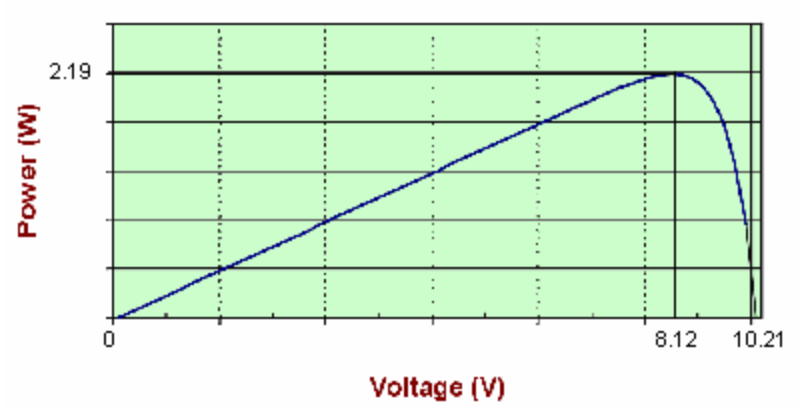


Fig. 5: Solar panel power vs output voltage characteristics

The implementation of the maximum power point tracking scheme ensures the power delivered to the load is maximum and hence with maximum efficiency for a given input power. As seen from Fig. 5 that the power delivered by the solar panel achieves a maximum value at a particular terminal voltage. This terminal voltage acts as input to the buck converter and can be adjusted through the duty cycle variation. The output side of the converter is attached to the 'nearly constant' voltage load i.e. the battery. The microcontroller which acts as a feedback control system is also powered through the output of the converter. Hence for a particular battery voltage $V_{o,}$ the duty cycle variation can adjust the input voltage V_{in} to the buck converter. The input current will then be decided according to the solar panel characteristics. The relation between V_{in}, V_o and the duty cycle D for a buck converter is given as

$$V_o = D\, V_{in}$$

The output current is monitored through the microcontroller by measuring the voltage against a small resistance (2.2 ohm) in series with the battery. The duty cycle is varied, through the use of low power microcontroller up to the point at which the output current is maximized which corresponds to the maximum power point of the solar panel. This is because of our almost true assumption that output voltage is nearly constant.

4. Microcontroller (TI MSP430F2012):

This is a low power microcontroller which has in built 10 bit Analog to digital converter. This ADC A1 (pin 3) is used to sense the analog current at a high sampling rate. This analog value is converted to digital 10 bit binary format and stored in ADC10MEM register for comparing operations. After each sample this value is assigned to a variable LASTVAL if it lies in tolerance range of the previous value. After a bunch of samples, nearly 2000, this value is assigned to another variable LASTSTABLE which keeps track of the last stable value of the output current. This corresponds to the new value of the stable output current after a variation in the duty cycle. If the new stable value is more than the last stable value then the duty cycle is continued to be varying in the same direction else the direction of variation of duty cycle is changed and continued until again at some point the LASTVAL has a value less than LASTSTABLE. The duty cycle stabilizes after some time to a value at which the output current is maximum.

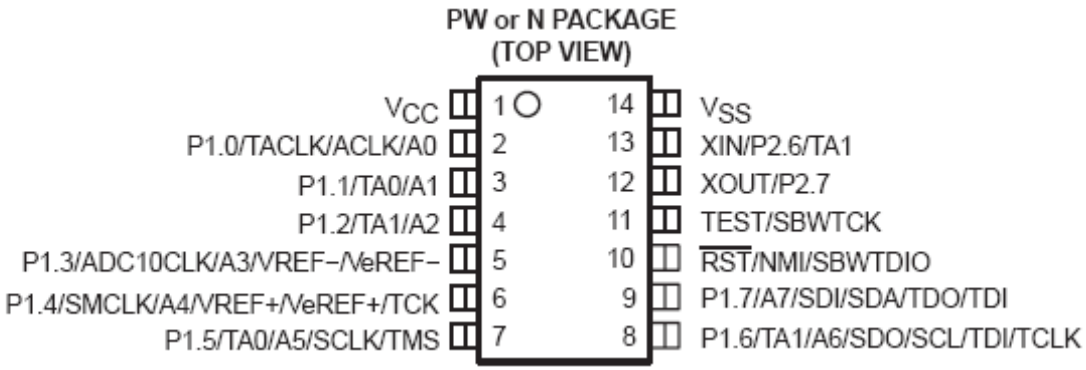


Fig. 6: Pin diagram of a low power MSP430F2012 microcontroller

To reduce the power consumption by the microcontroller it is set to Low Power Mode 0 whenever the ADC is not in use. The DCOCTL of the timer is set to 1 MHz. The duty cycle is generated through the Timer A1 (pin 4) of the microcontroller using the inbuilt PWM generator function in Up mode (mode 1) with output mode 6 that is the toggle/set mode. In this mode the time period of the PWM clock is set by the TACCR0 timer counter register and the duty cycle is decided by setting the TACCR1 timer counter register which generates the uptime of the PWM signal. Hence the duty cycle is the ratio of TACCR1 and TACCR0. The direction of variation of duty cycle is defined by a variable ‘dir’ which corresponds to increase in duty cycle when set and vice versa. The initial direction is set as 1 and LASTSTABLE is compared to the newly sampled value. If the new value is greater than the LASTSTABLE value then the direction is kept same and the duty cycle is increased smoothly by varying TACCR1 in steps of 1. The duty cycle is limited to the range of 1/20 - 19/20 for safety purposes. At some point the newly sampled value comes out to be less than LASTSTABLE value. At this point the direction variable ‘dir’ is cleared and duty cycle is decreased again in steps of 1 until again the LASTVAL (the newly sampled) is greater than the LASTSTABLE. After sometime the duty cycle is stabilized to a value which causes the maximum input power and hence the maximum output current. In other words, maximum efficiency condition is achieved.

5. Problem faced:

The mosfet IRFZ44N which is used as a switch for buck converter is working erratically and requires a mosfet driver circuit. This problem was realized when we fused all the different independently working parts to make a self sustainable circuit and looked into the working of each individual component.

References:

[1] http://en.wikipedia.org/wiki/Buck_converter

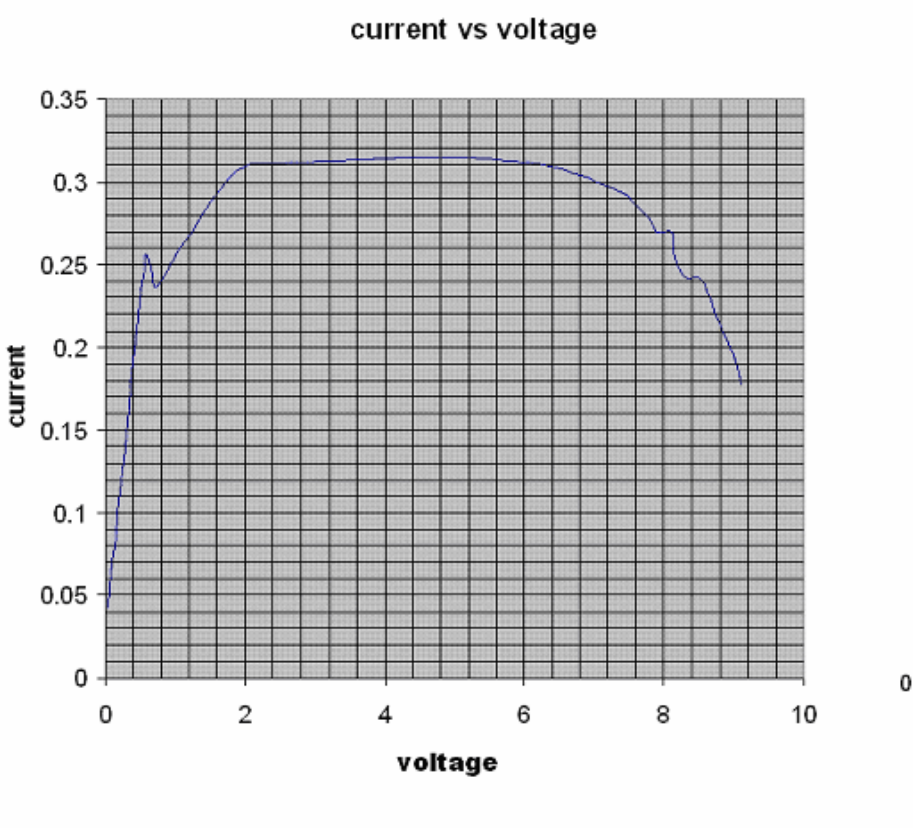
[2] http://www.powerdesigners.com/InfoWeb/design_center/articles/DC-DC/converter.shtm

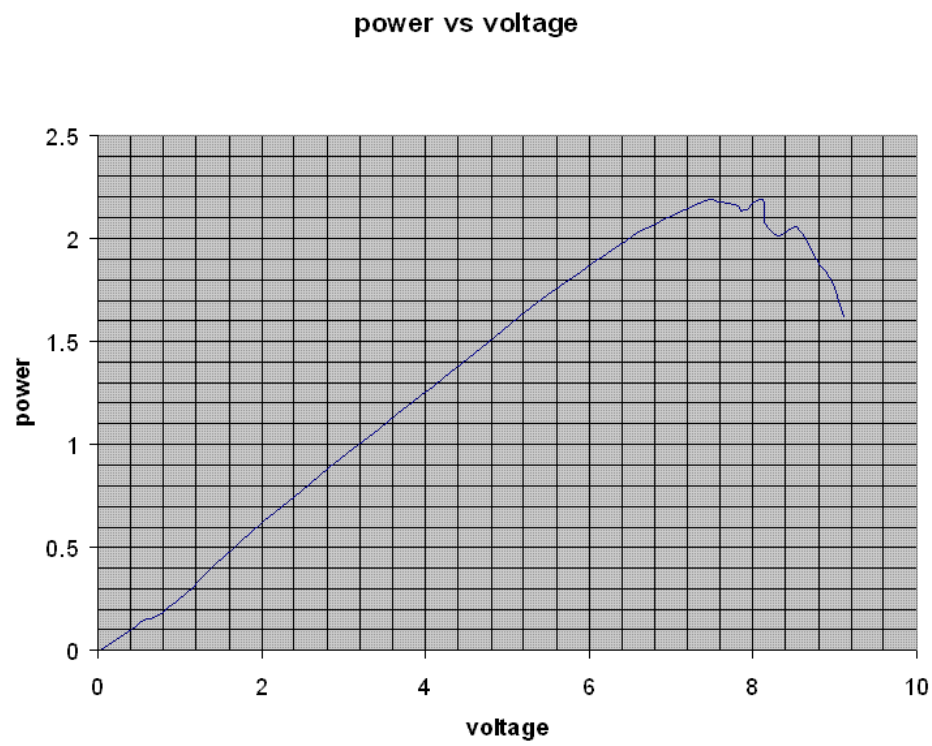
[3] MSP430F2012 datasheet http://www.ti.com/lit/gpn/msp430f2013
MSP430x2xx Family User's Guide http://www.ti.com/litv/pdf/slau144b

[4] IRFZ44 MOSFET datasheet
http://www.datasheetcatalog.com/datasheets_pdf/I/R/F/Z/IRFZ44.shtml

```
//******************************************************************************
//  MSP430F20xx Demo - Timer_A, PWM TA1, Up/Down Mode, DCO SMCLK
//
//  Description: This program generates one PWM output on P1.2 using
//  Timer_A configured for up/down mode. The value in CCR0, 128, defines the PWM
//  period/2 and the value in CCR1 the PWM duty cycles.
//  The desired duty cycle is on P1.2.
//  SMCLK = MCLK = TACLK = default DCO
//
//              MSP430F20xx
//           -----------------
//       /|\|              XIN|-
//        | |               |
//        --|RST         XOUT|-
//          |               |
//          |        P1.2/TA1|--> CCR1
//******************************************************************************

#include  <msp430x20x2.h>
#include "stdbool.h"
#define TOTAL (90)
#pragma location=0x200
__no_init int LASTVAL;
#pragma location=0x204
__no_init int LASTSTABLE;
bool flag=false,dir=true;
int count=0;

void main(void)
{
  WDTCTL = WDTPW + WDTHOLD;                 // Stop WDT
  LASTVAL=0;
 LASTSTABLE=0;
  P1DIR |= 0x04;                          // P1.2 and P1.3 output
  P1OUT &= ~ 0x04;
  P1SEL |= 0x04;                          // P1.2 and P1.3 TA1/2 options

  BCSCTL1= CALBC1_1MHZ;
   DCOCTL =CALDCO_1MHZ;
  CCR0 = TOTAL;                             // PWM Period/2
  CCTL1 = OUTMOD_6+CCIE;                          // CCR1 toggle/set
  CCR1 = TOTAL/2;                           // CCR1 PWM duty cycle
  TACTL = TASSEL_2 + MC_1;                  // SMCLK, up-down mode
  ADC10CTL0 = SREF_1  + REFON + ADC10ON + ADC10SHT_2;//+ ADC10IE ;
  ADC10CTL1 = INCH_1 +ADC10SSEL_3 ;
  ADC10AE0 |= 0x02;
 _BIS_SR(LPM0_bits+GIE);
```

```
  while (1){

   if(flag ){
      ADC10CTL0 |= ADC10ON;
        count=0;
    ADC10CTL0 |= ENC + ADC10SC;
   while (ADC10CTL1 & ADC10BUSY);
    if (ADC10MEM>LASTVAL-3 && ADC10MEM<LASTVAL+3 )
       {
        if(LASTSTABLE > (LASTVAL)) dir=!dir;
        if(dir){
           if(CCR1<=7*TOTAL/8 ) CCR1+=1;
           else CCR1=7*TOTAL/8 -1;}
        else if( CCR1>=TOTAL/8)  CCR1-=1;
         else CCR1=TOTAL/8+1;
        if(LASTVAL>3) LASTSTABLE= LASTVAL;
       }
    else LASTVAL=ADC10MEM;
    CCTL1 |= CCIE;
    ADC10CTL0 &=~ADC10ON;
    flag=false;
    _BIS_SR(LPM0_bits+GIE);
  }
 }
}


#pragma vector=TIMERA1_VECTOR
__interrupt void Timer_A (void)
{
  count++;
  if(count>=2000 ){
    _BIC_SR_IRQ(LPM0_bits+GIE);
     CCTL1 &= ~CCIE;
   flag=true;
  }
 }

#pragma vector=ADC10_VECTOR
__interrupt void ADC10_ISR(void)
{
  __bic_SR_register_on_exit(CPUOFF);
}
```


current vs voltage
0.35
0.3
0.25
0.2
0.15
0.1
0.05
0
current
0
2
4
6
8
10
voltage
0


power vs voltage
2.5
2
1.5
1
0.5
0
power
0
2
4
6
8
10
voltage