Experiment 1: Photovoltaic System

Characterize the SQ-85 PV panels, and find numerical values of model parameters for use now and later in semester

Test the inverter provided

Charge the battery from the panel, using the Direct Energy Transfer method

Hope for sun!

Experiment 1 to be performed this week, weather permitting

Final report for Exp. 1 due at beginning of lab on Jan. 26-28
Photovoltaic cell model

\[ I_{D} = I_0 - I_{pv} \]

\[ R_s I_{pv} + V_{pv} = 0 \]

\[ V_{pv} = I_{pv} R_p + V_{D} - I_{pv} I_0 \]

\[ I_{DSS} \] Exponential diode characteristic – related to \( V_{OC} \)

\( R_p \) Shunt resistance (typically large)

\( R_s \) Series resistance (typically small)

Find these parameters:

\( I_0 \) Photogenerated current, proportional to solar irradiation [estimate constant having units of A/(W/m\(^2\))]

\( I_{DSS} \) Exponential diode characteristic – related to \( V_{OC} \)

\( R_p \) Shunt resistance (typically large)

\( R_s \) Series resistance (typically small)
Shell Solar SQ-85 panel

36 series-connected PV cells
2 backplane diodes
85 W at 1000 W/m²
Apparent path of sun through sky

Baseline Rd. is 40°N

Times are not corrected for location of Boulder in Mountain Time Zone

Net panel irradiation depends on $\cos(\varphi)$ with

$\varphi =$ angle between panel direction and direction to sun

So take your data quickly
Laboratory facilities: mobile PV cart

- **PV panel**: 85 W\(_{pk}\), 17.2 V at 4.95 A, Shell SQ-85P
- **Battery**: 12 V, deep-discharge 56 A-hr
- **Battery charger**: Off cart: on stationary workbench
- **Inverter**: 60 Hz 300 W, 120 Vrms
  - 6 outlet ac power strip
  - Alarm
  - Battery low voltage
  - Voltmeter
  - Battery voltage
- **Isolated dc-dc converters**: +12V, -12V, +5V

Connectors:
- PV panel
- Battery
- Battery charger
- Inverter
- Isolated dc-dc converters
Deep-Discharge Lead-Acid Batteries

Theory and modeling of batteries
Don’t overcharge: this causes outgassing and can quickly ruin the battery
Don’t discharge below 50% SOC: this reduces battery life

Battery state of charge (SOC) vs. terminal voltage
100% SOC 12.80 volts or greater
75% SOC 12.55 volts
50% SOC 12.20 volts
25% SOC 11.75 volts
0% SOC 10.50 volts

Concorde AGM Battery Expected Life Cycles

% Depth of Discharge

Cycles

56 Ampere-hour
Lab reports

• One report per group. Include names of every group member on first page of report.
• Report all data from every step of procedure and calculations. Adequately document each step.
• Discuss every step of procedure and calculations
  – Interpret the data
  – It is your job to convince the grader that you understand what is going on with every step
  – Regurgitating the data, with no discussion or interpretation, will not yield very many points
  – Concise is good
Upcoming due dates

Experiment 1
   Final report due at beginning of lab session next week
     Jan. 26-28

Experiment 3
   Prelab assignment (dc-dc converter design for MPPT) due at
     beginning of lecture in two weeks
     Feb. 2
Upcoming weeks:
Design and build MPPT system

Exp. 2: introduction to MSP430 microcontroller

Exp. 3: DC-DC converter
Experiment 2
Introduction to MSP 430F2616 Microcontroller
Microcontroller Pinouts

ADC inputs

Timer B outputs
Microcontroller default settings

Upon power-on reset (POR), the MSP430F2616 comes up with the following conditions:

- Watchdog timer is enabled
- All pins are set to read state
- Processor internal clock is set to 1.2 MHz
Peripherals are controlled by registers in addressable memory

*Example:* Port P5, comprised of eight pins labeled P5.0 – P5.7

<table>
<thead>
<tr>
<th>Register name</th>
<th>C code variable name</th>
<th>Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>Port P5 resistor enable</td>
<td>P5REN</td>
<td>0x0012</td>
</tr>
<tr>
<td>Port P5 selection</td>
<td>P5SEL</td>
<td>0x0033</td>
</tr>
<tr>
<td>Port P5 direction</td>
<td>P5DIR</td>
<td>0x0032</td>
</tr>
<tr>
<td>Port P5 output</td>
<td>P5OUT</td>
<td>0x0031</td>
</tr>
<tr>
<td>Port P5 input</td>
<td>P5IN</td>
<td>0x0030</td>
</tr>
</tbody>
</table>

TI provides a header file that sets up all registers with C code variable names assigned to the correct addresses, so you don’t have to worry about it. Just add the following statement to the beginning of your C code:

```c
#include <msp430x26x.h>
```

This file also defines constants that are useful for setting peripheral functions.
Examples

Configure pin P3.0 to be a digital output, and toggle its value

P3DIR |= 0x01; // OR the contents of register P3DIR with hex 01,  
               // forcing the first bit high  
               // This configures pin P3.0 to be an output

P3OUT ^= 0x01;  // EXOR the contents of P3OUT with hex 01,  
                // toggling the first bit  
                // This changes the state of logic output P3.0

Turn off the watchdog timer

WDTCTL = WDTPW + WDTHOLD;  // Sets the WDT control register to disable the  
                            // watchdog timer function
C code to toggle pin P3.0

#include <msp430x26x.h>

void main(void)
{
    WDTCTL = WDTPW + WDTHOLD;  // Stop watchdog timer
    P3DIR |= 0x01;              // Configure pin P3.0 to output direction
    for (;;)
    {
        volatile unsigned int i;
        P3OUT ^= 0x01;           // Toggle P3.0 output
        i = 10000;
        do(i--);
        while (i != 0);         // Wait 10000 counts
    }
}
Setting the processor internal clock frequency

The processor contains a digitally controlled oscillator (DCO) whose frequency can be programmed. The power-on default frequency is 1.2 MHz. The following commands set the frequency to the maximum (16 – 20 MHz):

\[
\text{BCSCTL1} = \text{RSEL0} + \text{RSEL1} + \text{RSEL2} + \text{RSEL3}; \\
\text{DCOCTL} = \text{DCO0} + \text{DCO1} + \text{DCO2};
\]
Operation of Timer B as a PWM

See *MSP430x2xx Family User Guide*, Chapter 13

Timer B includes:
- One timer block with 16 bit counter
- Seven capture/compare registers

Use one CCR to set switching frequency: \( f_s = \frac{\text{DCO freq}}{\text{count}} \)

Use other CCR’s to set duty cycles
- Can control up to six independent output duty cycles

Need to configure Timer B, and write values to set \( f_s \) and duty cycle(s)
## Timer B control registers

<table>
<thead>
<tr>
<th>Register name</th>
<th>C code variable name</th>
<th>Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>Timer_B register</td>
<td>TBR</td>
<td>0x0190</td>
</tr>
<tr>
<td>Capture/compare control 6</td>
<td>TBCCTL6</td>
<td>0x018E</td>
</tr>
<tr>
<td>Capture/compare control 5</td>
<td>TBCCTL5</td>
<td>0x018C</td>
</tr>
<tr>
<td>Capture/compare control 4</td>
<td>TBCCTL4</td>
<td>0x018A</td>
</tr>
<tr>
<td>Capture/compare control 3</td>
<td>TBCCTL3</td>
<td>0x0188</td>
</tr>
<tr>
<td>Capture/compare control 2</td>
<td>TBCCTL2</td>
<td>0x0186</td>
</tr>
<tr>
<td>Capture/compare control 1</td>
<td>TBCCTL1</td>
<td>0x0184</td>
</tr>
<tr>
<td>Capture/compare control 0</td>
<td>TBCCTL0</td>
<td>0x0182</td>
</tr>
<tr>
<td>Timer_B control</td>
<td>TBCTL</td>
<td>0x0180</td>
</tr>
<tr>
<td>Timer_B interrupt vector</td>
<td>TBIIV</td>
<td>0x011E</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Register name</th>
<th>C code variable name</th>
<th>Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>Capture/compare register 6</td>
<td>TBCCCR6</td>
<td>0x019E</td>
</tr>
<tr>
<td>Capture/compare register 5</td>
<td>TBCCCR5</td>
<td>0x019C</td>
</tr>
<tr>
<td>Capture/compare register 4</td>
<td>TBCCCR4</td>
<td>0x019A</td>
</tr>
<tr>
<td>Capture/compare register 3</td>
<td>TBCCCR3</td>
<td>0x0198</td>
</tr>
<tr>
<td>Capture/compare register 2</td>
<td>TBCCCR2</td>
<td>0x0196</td>
</tr>
<tr>
<td>Capture/compare register 1</td>
<td>TBCCCR1</td>
<td>0x0194</td>
</tr>
<tr>
<td>Capture/compare register 0</td>
<td>TBCCCR0</td>
<td>0x0192</td>
</tr>
</tbody>
</table>
Timer B
Control Register TBCTL

See *MSP430x2xx Family User Guide*, Chapter 13, p. 13-21

C code:
TBCTL = TBSSEL2 + MC2;

This sets the Timer B clock source to SMCLK (derived from processor clock DCO), and sets the mode to Continuous mode. TBSSEL2 and MC2 are constants defined in the standard header file.

The timer is in Stop mode at POR, and starts counting when this statement is executed.
Example: Configuring Timer B as a PWM

```c
#include <msp430x26x.h>
void main(void)
{
    WDTCTL = WDTPW + WDTHOLD; // Stop WDT
    BCSCTL1 = RSEL0+RSEL1+RSEL2+RSEL3; // Set processor DCO to max
    DCOCTL = DCO0+DCO1+DCO2; //
    P4DIR |= 0x7e; // P4.1 - P4.6 output
    P4SEL |= 0x7e; // P4.1 - P4.6 TBx options
    TBCCR0 = 200; // PWM freq = 20 MHz/200
    TBCCTL1 = OUTMOD_7; // CCR1 reset/set mode
    TBCCCR1 = 150; // CCR1 duty cycle = 150/200
    TBCCTL2 = OUTMOD_3; // CCR2 set/reset mode
    TBCCCR2 = 150; // CCR2 duty cycle = 150/200
    TBCCTL3 = OUTMOD_3; // CCR3 set/reset mode
    TBCCCR3 = 30; // CCR3 duty cycle = 30/200
    TBCTL = TBSSEL_2 + MC_1; // use SMCLK, up mode
    for(;;)
    {
    }
}
```
The MSP 430 board in kits

MSP430F2616

3.3V voltage regulator

JTAG port

LED that can be jumpered to pin 2.5

All processor pins available, along with ground and uncommitted pads for prototyping
Driving a Power MOSFET Switch

MOSFET is off when $v_{gs} < V_{th} \approx 2.5 \text{ V}$

MOSFET fully on when $v_{gs}$ is sufficiently large (10-15 V)

Warning: MOSFET gate oxide breaks down and the device fails when $v_{gs} > 20 \text{ V}$.

Fast turn on or turn off (10’s of ns) requires a large spike (1-2 A) of gate current to charge or discharge the gate capacitance

MOSFET gate driver is a logic buffer that has high output current capability
Driving a Power MOSFET Switch

MOSFET gate driver is used as a logic buffer with high output current (1.5 A) capability.

The amplitude of the gate voltage equals the supply voltage $V_{CC}$.

Decoupling capacitors are necessary at all supply pins of UC3525A and TSC427.
Next Week’s Experiment 2

- Become familiar with MSP430
- Set up your MSP 430 to drive a MOSFET at a programmable duty cycle
- No prelab assignment