EECS192 Lecture 3
Feb. 5, 2019

Topics
• Updated connector pin IO list
• Timing in Linux:
• Motor electrical model conclusion
  – Motor electromechanical behavior
• Driving MOSFETs and motor
  – Op amp TLC084-NE5532
  – MC33883 MOSFET driver
  – H bridge
• PWM issues for motor
• (Back EMF measurement)

Notes:
• Handouts: lab rules
• 2/8 project proposal – upload to bcourses by Fri 5 pm
• 2/15: Motor drive/stall, steering servo
• Quiz 1: motor behavior Tues 2/12 (See Motor Worksheet week 2)
• CalDay Sat April 4/13
Beaglebone Blue Connections
### Beaglebone Blue Connections

http://inst.eecs.berkeley.edu/~ee192/sp19/files/BeagleBone_Blue_Pin_Table.xlsx

#### SERVO 8X3 connections

<table>
<thead>
<tr>
<th>Use on Blue</th>
<th>MODE0</th>
<th>MODE1</th>
<th>MODE2</th>
<th>MODE3</th>
<th>MODE4</th>
<th>MODE5</th>
<th>MODE6</th>
<th>MODE7</th>
</tr>
</thead>
<tbody>
<tr>
<td>SERVO_1</td>
<td>lcd_vsync*</td>
<td>gpmc_a8</td>
<td></td>
<td></td>
<td>pr1_edio_data_out2</td>
<td>pr1_pru1_pru_r30_8</td>
<td>pr1_pru1_pru_r31_8</td>
<td>gpio2[22]</td>
</tr>
<tr>
<td>SERVO_2</td>
<td>lcd_pclk*</td>
<td>gpmc_a10</td>
<td></td>
<td></td>
<td>pr1_edio_data_out4</td>
<td>pr1_pru1_pru_r30_10</td>
<td>pr1_pru1_pru_r31_10</td>
<td>gpio2[24]</td>
</tr>
<tr>
<td>SERVO_3</td>
<td>lcd_hsync*</td>
<td>gpmc_a9</td>
<td></td>
<td></td>
<td>pr1_edio_data_out3</td>
<td>pr1_pru1_pru_r30_9</td>
<td>pr1_pru1_pru_r31_9</td>
<td>gpio2[23]</td>
</tr>
<tr>
<td>SERVO_4</td>
<td>lcd_ac_bias_e<em>n</em></td>
<td>gpmc_a11</td>
<td></td>
<td></td>
<td>pr1_edio_data_out5</td>
<td>pr1_pru1_pru_r30_11</td>
<td>pr1_pru1_pru_r31_11</td>
<td>gpio2[25]</td>
</tr>
<tr>
<td>SERVO_5</td>
<td>lcd_data6*</td>
<td>gpmc_a6</td>
<td></td>
<td></td>
<td>pr1_edio_data_out6</td>
<td>pr1_pru1_pru_r30_6</td>
<td>pr1_pru1_pru_r31_6</td>
<td>gpio2[12]</td>
</tr>
<tr>
<td>SERVO_6</td>
<td>lcd_data7*</td>
<td>gpmc_a7</td>
<td></td>
<td></td>
<td>pr1_edio_data_out7</td>
<td>pr1_pru1_pru_r30_7</td>
<td>pr1_pru1_pru_r31_7</td>
<td>gpio2[13]</td>
</tr>
<tr>
<td>SERVO_7</td>
<td>lcd_data4*</td>
<td>gpmc_a4</td>
<td></td>
<td></td>
<td>pr1_edio_data_out5</td>
<td>pr1_pru1_pru_r30_4</td>
<td>pr1_pru1_pru_r31_4</td>
<td>gpio2[10]</td>
</tr>
<tr>
<td>SERVO_8</td>
<td>lcd_data5*</td>
<td>gpmc_a5</td>
<td></td>
<td></td>
<td>pr1_edio_data_out5</td>
<td>pr1_pru1_pru_r30_5</td>
<td>pr1_pru1_pru_r31_5</td>
<td>gpio2[11]</td>
</tr>
<tr>
<td>SERVO_PWR (80)</td>
<td>lcd_data10*</td>
<td>gpmc_a14</td>
<td></td>
<td></td>
<td>ehrpwm1A</td>
<td>mmc0_axr0</td>
<td>pr1_mii0_rxd1</td>
<td>uart3_ctsn</td>
</tr>
</tbody>
</table>

#### Quad Encoder

<table>
<thead>
<tr>
<th>QEP_1A</th>
<th>Mcasp0_ackr</th>
<th>eQEP0A_in</th>
<th>Mcasp0_axr2</th>
<th>Mcasp1_ackx</th>
<th>pr1_pru0_pru_r30_4</th>
<th>pr1_pru0_pru_r31_4</th>
<th>gpio3[18]</th>
</tr>
</thead>
<tbody>
<tr>
<td>QEP_1B</td>
<td>mcasp0_fsr</td>
<td>eQEP0B_in</td>
<td>mcasp0_ackl</td>
<td>Mcasp1_fsx</td>
<td>EMU2_mux2</td>
<td>pr1_pru0_pru_r30_5</td>
<td>pr1_pru0_pru_r31_5</td>
</tr>
<tr>
<td>QEP_2A</td>
<td>lcd_data12*</td>
<td>gpmc_a16</td>
<td>eQEP1A_in</td>
<td>mcasp0_ackr</td>
<td>mcasp0_axr2</td>
<td>pr1_mii0_rxlink</td>
<td>uart4_ctsn</td>
</tr>
<tr>
<td>QEP_2B</td>
<td>lcd_data13*</td>
<td>gpmc_a17</td>
<td>eQEP1B_in</td>
<td>mcasp0_fsr</td>
<td>mcasp0_axr3</td>
<td>pr1_mii0_rxer</td>
<td>uart4_rtsn</td>
</tr>
<tr>
<td>QEP_3A</td>
<td>gpmc_ad12</td>
<td>lcd_data19</td>
<td>mmc1_dat4*</td>
<td>mmc2_dat0</td>
<td>EQEP2A_IN</td>
<td>pr1_mii0_txd2</td>
<td>pr1_pru0_pru_r30_14</td>
</tr>
<tr>
<td>QEP_3B</td>
<td>gpmc_ad13</td>
<td>lcd_data18</td>
<td>mmc1_dat5*</td>
<td>mmc2_dat1</td>
<td>EQEP2B_in</td>
<td>pr1_mii0_txd1</td>
<td>pr1_pru0_pru_r30_15</td>
</tr>
<tr>
<td>QEP_4B</td>
<td>gpmc_ad15</td>
<td>lcd_data16</td>
<td>mmc1_dat7*</td>
<td>mmc2_dat3</td>
<td>eQEP2_strobe</td>
<td>pr1_ecap0_ecap_cap</td>
<td>pr1_pru0_pru_r31_15</td>
</tr>
<tr>
<td>QEP_4A</td>
<td>gpmc_ad14</td>
<td>lcd_data17</td>
<td>mmc1_dat6*</td>
<td>mmc2_dat2</td>
<td>eQEP2_index</td>
<td>pr1_mii0_txd0</td>
<td>pr1_pru0_pru_r31_14</td>
</tr>
</tbody>
</table>
Delay timing using rc library

```c
start_time = rc_nanos_thread_time();
    for(j = 0; j< 1000; j++)
        {shared_mem_32bit_ptr[ENCODER_MEM_OFFSET+1] = 1;
// set flag to start conversion by PRU
        while(shared_mem_32bit_ptr[ENCODER_MEM_OFFSET+1] == 1);
// wait for PRU to zero word
        for(i = 0; i< 128; i++){
            linescan[i]=
                (int) shared_mem_32bit_ptr[ENCODER_MEM_OFFSET+2+i];
// copy data
        }
}
end_time = rc_nanos_thread_time();
run_time = end_time - start_time;
```
A to D using PRU

pru_firmware/src/main_pru0.c

#define PRU_SHAREDMEM 0x10000 //shared memory with Cortex A8
#define GPIO1 0x4804C000
#define GPIO_CLEARDATAOUT 0x190
#define GPIO_SETDATAOUT 0x194

// try to access shared memory from C
shared_mem_32bit_ptr = (volatile unsigned int *)PRU_SHAREDMEM;
init_adc();
// signal to Linux process that PRU is running/ready by clear []
shared_mem_32bit_ptr[ENCODER_MEM_OFFSET] = 0x0;...
for(i = 0; i< 128; i++)
{
    __R30 &= ~CLK; // reset CLK line
    shared_mem_32bit_ptr[ENCODER_MEM_OFFSET+2+i] =
    read_adc(0);
    __R30 |= CLK; // set CLK
    __delay_cycles(10); // allow hold time = 50 ns
}
DC Motor Physical Model

\[ \overrightarrow{F} = i \overrightarrow{l} \times \overrightarrow{B} \]

\[ \tau = \overrightarrow{r}_1 \times \overrightarrow{F}_1 + \overrightarrow{r}_2 \times \overrightarrow{F}_2 \]
Motor Electrical Model

Motor Electrical Model
Back EMF
Motor electromechanical behavior

Continued on board
Also- see motor worksheet……

Note: missing e-stop!
Motor model

For this problem, consider a DC permanent magnet motor (as used in your car). The car is on a carpet and moves in a straight line with no slip between the wheels and the carpet. The car is initially moving at a speed of 2 meters per second.

You can assume a motor model as shown below. The qualitative shape of the curves is more important than magnitudes.

Let peak speed = 5 m/sec
Accel = 5 m/s²
k_e = 1 v/(m/sec)
On board

(for answer see sp99 final solution)
Given: $R_m = 0.1$ ohms, $V_{batt} = 7.2$ V, $R_{bat} = 0$.

$V_{ds} =$ ? $\Rightarrow I_{ds} =$ ? amps

(LiPo 11 V!)
Driving MOSFETs and motor

\[ R_m = 0.1 \, \text{ohms}, \quad V_{\text{batt}} = 7.2 \, \text{V}, \quad R_{\text{bat}} = 0. \]

\[ V_{\text{ds}} = 3.6 \, \text{V} \Rightarrow I_{\text{ds}} = \frac{(7.2 - 3.6 \, \text{V})}{(0.1 \, \text{ohm})} = 36 \, \text{amps} \]

\[ V_{\text{gs}} = 20 \, \text{V} \]

\[ I_{\text{ds}} \approx 100 \, \text{amps}, \quad P_{\text{trans}} \approx 35 \, \text{W} \]

\[ V_{\text{gs}} = 6 \, \text{V} \]

\[ I_{\text{ds}} \approx 50 \, \text{amps}, \quad P_{\text{trans}} \approx 72 \, \text{W} \]

Key design points:
1) High \( V_{\text{gs}} \) better than low \( V_{\text{gs}} \)
2) Switch quickly
3) Make sure \( V_s = 0 \) (big ground)
Motor Drive Quick and Dirty w/ Op Amp

Details On board....

TLC084 (short circuit protection lacking)
or TLC074
NE5532 or LMC6482 op amp

100 ohm series resistance for current limiting

Checklist:
1) Emergency stop
2) Reset Protection
3) Snubbing
4) High current diode
   (or MOSFET body diode)
Driving MOSFETs and motor

How to choose PWM frequency?

IRFB 3006 Power MOSFET

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Parameter</th>
<th>Min.</th>
<th>Typ.</th>
<th>Max.</th>
<th>Units</th>
<th>Conditions</th>
</tr>
</thead>
<tbody>
<tr>
<td>gfs</td>
<td>Forward Transconductance</td>
<td>280</td>
<td>—</td>
<td>—</td>
<td>S</td>
<td>$V_{DS} = 25\text{V}$, $I_D = 170\text{A}$</td>
</tr>
<tr>
<td>$Q_g$</td>
<td>Total Gate Charge</td>
<td>—</td>
<td>200</td>
<td>300</td>
<td>nC</td>
<td>$I_D = 170\text{A}$</td>
</tr>
<tr>
<td>$Q_{gs}$</td>
<td>Gate-to-Source Charge</td>
<td>—</td>
<td>37</td>
<td>—</td>
<td>—</td>
<td>$V_{DS} = 30\text{V}$</td>
</tr>
<tr>
<td>$Q_{gd}$</td>
<td>Gate-to-Drain (&quot;Miller&quot;) Charge</td>
<td>—</td>
<td>60</td>
<td>—</td>
<td>—</td>
<td>$V_{GS} = 10\text{V}$</td>
</tr>
<tr>
<td>$Q_{sync}$</td>
<td>Total Gate Charge Sync. ($Q_g - Q_{gd}$)</td>
<td>140</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>$I_D = 170\text{A}$, $V_{DS} = 0\text{V}$, $V_{GS} = 10\text{V}$</td>
</tr>
<tr>
<td>$t_{d(on)}$</td>
<td>Turn-On Delay Time</td>
<td>16</td>
<td>—</td>
<td>—</td>
<td>ns</td>
<td>$V_{DS} = 39\text{V}$</td>
</tr>
<tr>
<td>$t_r$</td>
<td>Rise Time</td>
<td>182</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>$I_D = 170\text{A}$</td>
</tr>
<tr>
<td>$t_{d(\text{ref})}$</td>
<td>Turn-Off Delay Time</td>
<td>118</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>$R_G = 2.7\Omega$</td>
</tr>
<tr>
<td>$t_f$</td>
<td>Fall Time</td>
<td>180</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>$V_{GS} = 10\text{V}$</td>
</tr>
<tr>
<td>$C_{iss}$</td>
<td>Input Capacitance</td>
<td>8970</td>
<td>—</td>
<td>—</td>
<td>pF</td>
<td>$V_{GS} = 0\text{V}$</td>
</tr>
</tbody>
</table>

TLC084 Op amp (same pinout as TL074)

<table>
<thead>
<tr>
<th>$V_{OH}$</th>
<th>High-level output voltage</th>
<th>$V_{IC} = 2.5\text{V}$</th>
<th>$I_{OH} = -1\text{ mA}$</th>
<th>$I_{OH} = -20\text{ mA}$</th>
<th>$I_{OH} = -35\text{ mA}$</th>
<th>$I_{OH} = -50\text{ mA}$</th>
<th>$-40\degree\text{C}$ to $85\degree\text{C}$</th>
<th>V</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>$25\degree\text{C}$</td>
<td>4.1</td>
<td>4.3</td>
<td>3.9</td>
<td>4.4</td>
<td>3.6</td>
<td>$-0.25$</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Full range</td>
<td>3.9</td>
<td>4.3</td>
<td>3.9</td>
<td>4.4</td>
<td>3.6</td>
<td>$-0.25$</td>
</tr>
<tr>
<td>$V_{OL}$</td>
<td>Low-level output voltage</td>
<td>$V_{IC} = 2.5\text{V}$</td>
<td>$I_{OL} = 1\text{ mA}$</td>
<td>$I_{OL} = 20\text{ mA}$</td>
<td>$I_{OL} = 35\text{ mA}$</td>
<td>$I_{OL} = 50\text{ mA}$</td>
<td>$-40\degree\text{C}$ to $85\degree\text{C}$</td>
<td>V</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$25\degree\text{C}$</td>
<td>0.18</td>
<td>0.25</td>
<td>0.35</td>
<td>0.39</td>
<td>0.45</td>
<td>$0.25$</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Full range</td>
<td>0.35</td>
<td>0.25</td>
<td>0.35</td>
<td>0.39</td>
<td>0.45</td>
<td>$0.25$</td>
</tr>
<tr>
<td>$I_{OS}$</td>
<td>Short-circuit output current</td>
<td>Sourcing</td>
<td>25$^\circ\text{C}$</td>
<td>100</td>
<td>25$^\circ\text{C}$</td>
<td>100</td>
<td>25$^\circ\text{C}$</td>
<td>mA</td>
</tr>
<tr>
<td>$I_O$</td>
<td>Output current</td>
<td>$V_{OH} = 1.5\text{V}$ from positive rail</td>
<td>25$^\circ\text{C}$</td>
<td>57</td>
<td>25$^\circ\text{C}$</td>
<td>55</td>
<td>25$^\circ\text{C}$</td>
<td>mA</td>
</tr>
</tbody>
</table>
How to choose PWM frequency (if using op amp)?

(on board)

50 mA

9000 pF

Driving MOSFETs and motor
Motor Drive Quick and Dirty w/ Op Amp

- IRFB3006: $C_{GS} = 9000$ pF
- TL084: 50 mA
- $i = C \frac{dV}{dt}$
- $\frac{dV}{dt} = i/C \approx 5$ V/usec $\Rightarrow$ 2us on + 2 us off, PWM < 100 us period
- TL084 slew rate: 10 V/usec $\Rightarrow$ exceed current limit
- Series resistor on output of op amp of 200 ohms, limits current to 50 mA $(10V/200$ ohms $= 50$ mA)
PWM Issues for Motor

PWM for Main Motor control

\[ i_m = \left( \frac{T}{T_o} \right) i_{\text{max}} \]

Is \( i_{\text{max}} \) constant?
H-Bridge Gate Driver IC

The 33883 is an H-bridge gate driver (also known as a full-bridge pre-driver) IC with integrated charge pump and independent high and low side gate driver channels. The gate driver channels are independently controlled by four separate input pins, thus allowing the device to be optionally configured as two independent high side gate drivers and two independent low side gate drivers. The low side channels are referenced to ground. The high side channels are floating.

The gate driver outputs can source and sink up to 1.0 A peak current pulses, permitting large gate-charge MOSFETs to be driven and/or high pulse-width modulation (PWM) frequencies to be utilized. A linear regulator is incorporated, providing a 15 V typical gate supply to the low side gate drivers.

This device powered by SMARTMOS technology.

Features

- \( V_{CC} \) operating voltage range from 5.5 V up to 55 V
- \( V_{CC2} \) operating voltage range from 5.5 V up to 28 V
- CMOS/LSTTL compatible I/O
- 1.0 A peak gate driver current
- Built-in high side charge pump
- Under-voltage lockout (UVLO)
- Over-voltage lockout (OVLO)
- Global enable with <10 µA Sleep mode
- Supports PWM up to 100 kHz

ORDERING INFORMATION

<table>
<thead>
<tr>
<th>Device</th>
<th>Temperature Range ((T_A))</th>
<th>Package</th>
</tr>
</thead>
<tbody>
<tr>
<td>MC33883HEG</td>
<td>-40 °C to 125 °C</td>
<td>20 SOICW</td>
</tr>
</tbody>
</table>

![Figure 1. 33883 Simplified Application Diagram](image-url)
MC3383 + H bridge

!!!CAUTION!!!!
Software fries hardware....
Need protection logic- 74HCxxx
Estop?
Driving MOSFETs and motor

MC3383 protection

Figure 12. Gate Protection and Flyback Voltage Clamp

Caution: don’t run motor current through here
Wiring Notes: caution on Vgs

On board

Watch out for voltage drop in wires/PCB traces.
#22 wire: 50 mOhm/m
#12 wire: 5 mOhm/m
Extra Slides
PWM Issues for Motor

https://github.com/ucb-ee192/SkeletonMCUX/tree/master/frdmk64f_pwm

```c
FTM_UpdatePwmDutycycle
(BOARD_FTM_BASEADDR, BOARD_FTM_CHANNEL,
 kFTM_CenterAlignedPwm, updated_duty_cycle);
```

---

Figure 40-21. CPWM period and pulse width with ELSnB:ELSnA = 1:0