Servo and Motor Controller

Date: August 10, 2004

Description:
The servo motor controller drives three R/C servomotors and one brushless DC motor. All four motors are controlled by PWM signals sent from a PIC 18F252 microcontroller. The PWM signal to the brushless motor is used to toggle on/off an Allegro A3936 three-phase PWM motor driver. The servo controller regulates the shaft velocity of the brushless motor with encoder feedback, which is decoded by an HCTL2000 quadrature decoder. An in-circuit serial programming connector is provided on the PCB to allow for in board reprogramming of the PIC micro-controller.

PCB Block Diagram:

![PCB Block Diagram](image)

Power Consumption:
Logic Supply: 5V (max current: 1.75 A)
Motor Supply: 24V (operates between 15V – 50V, max current: 3A)

Data Interface:
TTL-level 3-wire serial
Control System:
The DC motor sends out encoded position signals to the quadrature decoder chip. The microcontroller reads the position feedback information from the decoder and converts it into a velocity reading. This velocity data is then compared to the desired speed and an error signal is generated, which is used to calculate the PD compensation.

Circuit Schematics:

Schematic 1: Logic Section
Schematic 2: Power Section
Figure 1: PCB Layout – Top
Figure 2: PCB Layout – Bottom
### Bill of Materials:

<table>
<thead>
<tr>
<th>Name</th>
<th>Value</th>
<th>Footprint</th>
<th>Notes</th>
<th>Vendor</th>
<th>Part #</th>
<th># per Board</th>
</tr>
</thead>
<tbody>
<tr>
<td>C6</td>
<td>1uF, 100V</td>
<td>Cylinder</td>
<td>Electrolytic</td>
<td>Digi-Key</td>
<td>P5589-ND</td>
<td>1</td>
</tr>
<tr>
<td>C7, C9, C11, C20, C22</td>
<td>0.22uF, 50V</td>
<td>SM/C_1206</td>
<td>Ceramic X7R</td>
<td>Digi-Key</td>
<td>399-1251-1-ND</td>
<td>5</td>
</tr>
<tr>
<td>C8</td>
<td>33uF, 100V</td>
<td>Cylinder</td>
<td>Electrolytic</td>
<td>Digi-Key</td>
<td>P5595-ND</td>
<td>1</td>
</tr>
<tr>
<td>C120, C121</td>
<td>22pF, 50V</td>
<td>SM/C_402</td>
<td>Ceramic</td>
<td>Digi-Key</td>
<td>PCC220CQCT-ND</td>
<td>2</td>
</tr>
<tr>
<td>C300, C301, C400, C500</td>
<td>0.1uF, 50V</td>
<td>SM/C_805</td>
<td>Ceramic X7R</td>
<td>Digi-Key</td>
<td>399-1170-1-ND</td>
<td>4</td>
</tr>
<tr>
<td>D7, D8, D9, D10, D11, D12</td>
<td>Diode, 100V, 1A</td>
<td>SM/D</td>
<td>Fast Recovery</td>
<td>Digi-Key</td>
<td>FR1BDICT-ND</td>
<td>6</td>
</tr>
<tr>
<td>J1</td>
<td>Connector 5x2 2mm</td>
<td>ICSP connector</td>
<td></td>
<td>Digi-Key</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>J2, J6, J14, J15, J16</td>
<td>Connector 3x1 2mm</td>
<td></td>
<td>Brushless motor 3-phase power</td>
<td>Digi-Key</td>
<td>H2084-ND (Recepticle)</td>
<td>5</td>
</tr>
<tr>
<td>J4</td>
<td>Connector 8x1 2mm</td>
<td></td>
<td>Brushless motor logic and encoder feedback</td>
<td>Digi-Key</td>
<td>H2106-ND (Header)</td>
<td>1</td>
</tr>
<tr>
<td>J5, J7</td>
<td>Connector 2x1 2mm</td>
<td></td>
<td>24V motor power</td>
<td>Digi-Key</td>
<td>H2083-ND (Recepticle)</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>IC socket 8 pin</td>
<td></td>
<td>Holds half HCTL200</td>
<td>Digi-Key</td>
<td>AE8908-ND</td>
<td>2</td>
</tr>
<tr>
<td>R5, R6</td>
<td>2K SM/R 402</td>
<td>1% 1/16W</td>
<td></td>
<td>Digi-Key</td>
<td>P2.00KLCT-ND</td>
<td>2</td>
</tr>
<tr>
<td>R7, R8, R9</td>
<td>10K SM/R 402</td>
<td>1% 1/16W</td>
<td></td>
<td>Digi-Key</td>
<td>311-10.0KLCT-ND</td>
<td>3</td>
</tr>
<tr>
<td>R12</td>
<td>0.25Ohm Radial</td>
<td></td>
<td>Current-Sense Res</td>
<td>Digi-Key</td>
<td>15FR250-ND</td>
<td>1</td>
</tr>
<tr>
<td>R13</td>
<td>51K SM/R 402</td>
<td>1% 1/16W</td>
<td></td>
<td>Digi-Key</td>
<td>311-51.0KLCT-ND</td>
<td>1</td>
</tr>
<tr>
<td>U23</td>
<td>HCTL2000 Through hole</td>
<td></td>
<td>Quadrature Decoder</td>
<td></td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>U26</td>
<td>10MHz OSC SMD</td>
<td></td>
<td>HSM938H</td>
<td>Digi-Key</td>
<td>CW303-ND</td>
<td>1</td>
</tr>
<tr>
<td>U35</td>
<td>PIC 18F252 SOIC 28 pin</td>
<td>Microcontroller</td>
<td></td>
<td>Digi-Key</td>
<td>PIC18F252-I/ISO-ND</td>
<td>1</td>
</tr>
<tr>
<td>Y1</td>
<td>10MHz Crystal SMT Can</td>
<td></td>
<td>ATS100SM</td>
<td>Digi-Key</td>
<td>CTX508-ND</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>Brushless Motor 24V, 40W, Encoder</td>
<td>Shinano Kenshi LA052-040E-3NL1</td>
<td>Digi-Key</td>
<td></td>
<td>1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Resettable fuse</td>
<td>1.25A 15V SMD</td>
<td></td>
<td>MF-SM123CT-ND</td>
<td></td>
<td>1</td>
</tr>
</tbody>
</table>
Overview of Code:

The program for the PIC is broken up into the three sections: Main program, Low Priority Interrupt Service Routine, High Priority Interrupt Service Routine

INITIALIZATION
Configuration Bits: Oscillator set to HS (10MHz)
Disable all other bit word settings

Configuration Variables: Motor speed Max = -3200rpm to 3200rpm
Motor PWM period = FF (255 clock cycles)
Max motor duty cycle = 1000
Fin position = 3000 to 5000 (4000 is initialized position)
Baud Rate = 57.6k
Internal clock = 10MHz

MAIN PROGRAM
While loop overview: Wait for new command to be sent
Poll motor speed and correct if necessary
Send out PWM signals to three fins
Returns motor speed and PWM signals

Processing Command: Once command is received from DimmPC it is searched for each element
Format of command: aXXXXbXXXXcXXXXmXXXX
(All or portions of the command may be sent)
Each element is saved into a variable to be implemented in the ISR’s
If “x” is sent new speeds are not returned
If “e” is sent speeds are returned

No Command is sent: Until a new command is received the speed is continuously polled and updated
After each speed check, the speed and PWM signal is returned to the user.

HIGH PRIORITY INTERRUPT
Triggered by Fins Flag: 6 step cycle: Set fin a high, Set fin a low
Set fin b high, Set fin b low
Set fin c high, Set fin c low
One step is implement each ISR is called due to

Triggered by command: When a command is received, the receive buffer triggers this ISR
The ISR then saves and receives the data into a command buffer
The received command flag is set so the main while loop will process the command when ISR exits

LOW PRIORITY INTERRUPT
Triggered by Motor Flag: Reads the decoder chip and exits ISR (repeats 10 times)
Read decoder 10 times (multiple reads decreases error)
Convert data to rpm (freq of 10th read = 10.1Hz)
Compare with goal speed to get speed error
Calculate speed comp based on error
Update duty cycle and direction

Triggered by Communication Error: If no command has been received in the past two seconds, the motor speed is set to zero and the fins are set to 4000 (center). This is a fail safe to stop the AUV in the event that there is a communication problem between the DimmPC and the motor control card.