PARALLAX INC

Web Site: www.parallax.com
Forums: forums.parallax.com
Sales: sales@parallax.com
Technical: support@parallax.com

Office: (916) 624-8333
Fax: (916) 624-8003
Sales: (888) 512-1024
Tech Support: (888) 997-8267

Motor Control Board – Design Specifications

Parallax Inc. is looking to develop a motor control board compatible with current RepRap hardware. The design should feature the P8X32A multicore microcontroller from Parallax, which supports the current trend of RepRap hardware evolving into cheaper, easier to use and higher quality solutions. The design should be completed by March 1, 2012 and meet the requirements listed below.

- **Open Source.** This design will be released under the Creative Commons Attribution 3.0 license. For more information on Parallax Open Source products, visit http://www.parallax.com/go/opensource.

- **Full and Half H-Bridges.** This board should have a combination of twelve (12) half H-bridges and two (2) full H-bridges. This will require 14 I/O pins and will be able to run up to four brushless servo motors or up to four stepper motors.
  - Suggested circuit, Peter Jakacki's half H-bridge: http://www.pbjtech.com/downloads/halfbridge.png
    - Works as a transimpedance/transresistance amplifier with a positive gain
  - Create an inverted version for the second half of the full H-bridge
    - Should work as a transimpedance/transresistance amplifier with a negative gain
  - Include current sense resistors and one or more PTC fuse
    - They could be wired into the final product to replace a damaged H-bridge

- **Power Conversion Circuits**
  - Implement a 5 VDC switching regulator somewhere in the 0.5 to 1 A range
  - Use an LDO for the 3.3 VDC power supply
  - Determine a reasonable current output for all internal and most external current requirements

- **Analog Input Circuit**
  - Use sigma-delta ADC with an analog switch and over-voltage protection
  - It may be best to have an op-amp on the Propeller side of a multiplexer with an amplification that gives a small voltage range, such as 0 to 0.5 or 1 VDC. Sense resistors nodes could connect directly to the Op-amp through the multiplexer. Auxiliary inputs could connect directly or through a voltage divider for enhanced range.

- **Other Features**
  - Standard or micro SD-card socket
    - Use a standard socket if there is a cheap source, otherwise use a microSD socket
  - USB circuit
    - Jumper or switch to allow/prevent programming
    - Use newer example circuit that allows operation when the USB cable is not present
    - Allow for bus power to power the board

- **Other Design Considerations**
  - Use surface-mount parts wherever possible, but use parts that have through-hole equivalents
  - Determine when to use surface-mount or through-hole headers and connectors to optimize production costs

- **I/O pin assignments**
  - SD card P0 to P3
    - MISO P0
    - CLK P1
    - MOSI P2
    - CS P3
  - Full H-bridge x 4 P4 to P7
    - Each H-bridge needs an unpopulated header with access to ground and
  - Half H-bridge x 8 P8 to P15
  - Quadrature encoder x 3 P16 to P21
    - Encoder headers need power, (3.3 V) ground, and 2 I/O pins per encoder
    - Direct inputs with over-voltage protection
    - Perhaps a series resistor will be enough
    - Used as auxiliary inputs at in initial firmware
  - Shift registers P22 to P25 2 x data, 2 x latch, clock from I2C
      - MISO P22
      - MISO latch P23
      - MOSI P24
      - MOSI latch P25
    - 16 inputs, 16 outputs
    - ADC multiplexer (4 inputs)
    - Low-side switches (4 outputs)
      - N-channel FETs rated at many amps
    - Other inputs and outputs could run from 5 VDC
  - ADC P26 AND P27
      - Input P26
      - Feedback P27
    - 16-bit multiplexer, such as 74HC4067
    - 12 current-sense resistors inputs
    - 4 aux inputs
      - 2 for multi-volt readings (e.g. something like 0 to 5 or 12 VDC)
      - 2 for precision readings (e.g. something like 0 to 0.5 or 1 VDC)
      - Safe clamping when inputs are over voltage
        - The multiplexer may perform this if there is a resistor divider on the input
      - High input impedance (e.g. 1 megohm)
  - I2C P28 and P29
    - 64 KB EEPROM for calibration data
  - Serial/USB P30 and P31

- **Firmware requirements**
  - Stepper motor driver
    - Generates PWM on dira for microstepping
    - Generates inverted output for half H-bridges
    - Updates all outputs simultaneously
  - Reads motor position (signed long) from main memory
  - Reads output enable mask from main memory

- Can set arbitrary point as home (0)
- Observers a maximum step rate
- I/O driver
- Reads sigma-delta ADC
- Checks motor current and disables motors that are over current
- Repeated writes values to shift register
- Sets ADC channel using the multiplexer
- Repeatedly writes shift registers value to main memory
- Repeatedly writes ADC values to main memory
- Automatically cycles through ADC channels
- Uses a sample resolution set by a constant
- Waits for a propagation delay then a zero crossing after switching ADC channels
- The input changes states at a zero crossing
- Initial firmware will not use the SD card socket or Quadrature encoder inputs
- We should aim to use current RepRap code written in Processing.

- **Licensing**
  - This is a work made for hire; Parallax will be the original copyright holder.
  - Design files should include the text "Copyright 2010 Parallax Incorporated. This work is released under the Creative Commons Attribution 3.0 United States License. The full text of the license is available from: http://creativecommons.org/licenses/by/3.0/us/legalcode"
  - PCB design should include the "open source hardware" logo: http://oshwlogo.com/ (with or without text)