Teensy Hardware for Linux Driver
Circuit Description

‒   Jim Larson   11/25/2010

The purpose of this document is to describe the circuitry used to demonstrate the capabilities of the Linux driver created for Teensy. The schematic shows circuitry that will support two DC motors controlled with Pulse Width Modulation (PWM) through a dual H-Bridge, and six ADC inputs on ports F0-F5. The H-Bridge shown is a low-current L293D that can handle up to 600mW. The advantage of this part is that anti-kickback diodes are built in. A photocell is shown as an example of an analog input.

Parts List:
IC1 – Teensy 2.0 from pjrc.com.
IC2 – L293D Dual H-Bridge
R5 – CdS Photocell
L1 – Tri-color LED
L2-L5 – LEDs (any color works)
C1 – 100uF decoupling Cap for motor power supply
R1-R4, R6-R8 – 220 Ohm
R9 – 3.9K (see details below)

The Teensy2 is the basis of this project. It extends hardware control to a Linux driver using the USB interface.

The L293D is a dual H-Bridge that is controlled by the Teensy2. Since the anti-kickback diodes are built in, the part is very easy to use. Current drive capability is limited to 600mW or less. The power for the motors must be provided by an external power supply or battery. Do not use the USB power from Teensy as that could damage the USB port on the host PC. Connect the battery positive voltage where shown and connect the battery ground to the same ground as the rest of the circuitry. Alternative H-Bridge circuitry is certainly possible. An L298 provides up to 3A capability, but requires external diodes. It is a nice alternative, but not suitable for a breadboard.

The Tri-color LED is used for signaling and debug. It is not strictly required and the software does not support control of it from the driver. It is purely optional.

LEDs L2-L5 are also optional. They light when the motor direction control lines are high and aid debug. They also look cool.

Capacitor C1 is a bulk decoupling cap for the motor power supply.

Resistors R1-R4 and R6-R8 are current limiting resistors for the LEDs. The value of these is somewhat arbitrary: if your LEDs seem to bright, use a higher value. Lower values are not recommended. If you seem to need them, look carefully at the Teensy2 drive capabilities to be sure that the ports can handle the current required.

The photocell is just an example of an analog input that can be attached to port F on the Teensy2. The photocell and resistor R9 form a voltage divider. The voltage is applied to pin F0 and may be read with the driver. The other pins on port F may be used in a similar manner.