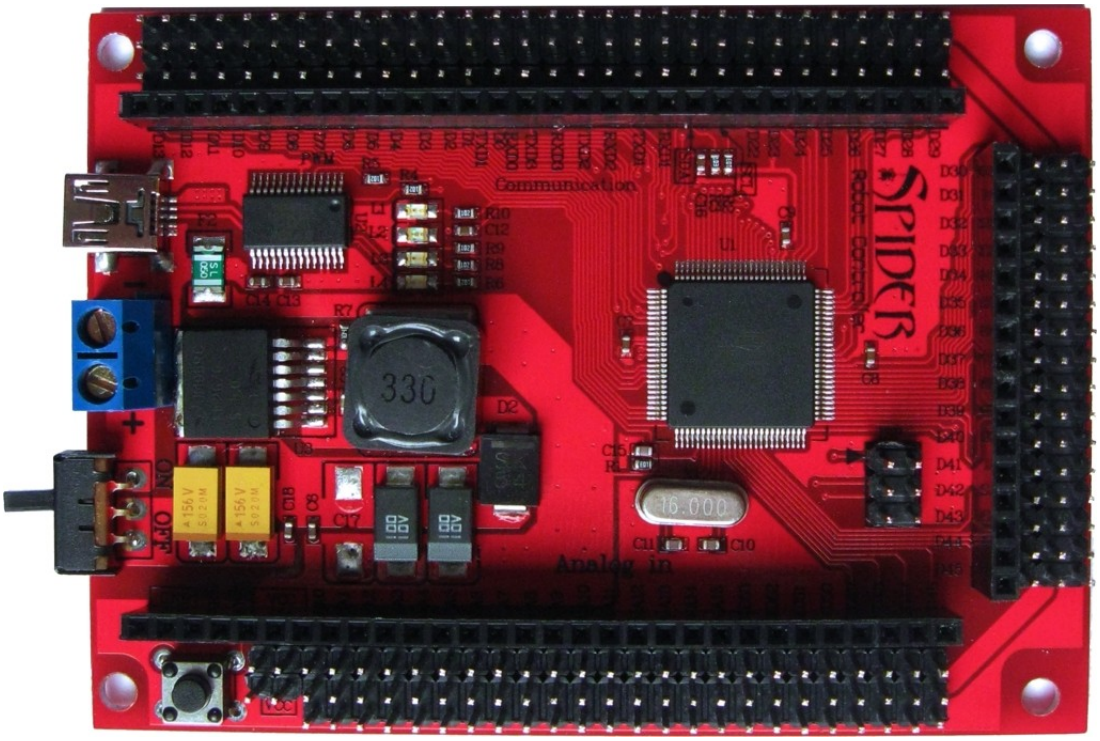
Red Back SPIDER Robot Controller

The Spider robot controllers were designed for robots requiring a large numbers of servos such as humanoid and hexapod style robots. Using an ATmega1280 CPU this controller is fully compatible with the Arduino Mega (1280).

This controller includes a 3A switch mode power supply and is capable of driving up to 48 micro servos directly from the PCB. All of the processors 70 I/O pins are terminated as both a female header and a servo compatible 3pin male header.



Features:

- ATmega1280 16MHz CPU
- 128K FLASH, 8K SRAM and 4K EEPROM
- USB interface and ISP socket
- Built in 3A switch mode power supply (7V – 30V input)
- 70 I/O pins with male and female headers
- Support for up to 48 servos
- 16x 10bit analog inputs
- 15x 8bit PWM outputs
- 1x I2C
- 6x External interrupt pins
- 4x Serial communication
- Comes with Arduino boot-loader installed

Programming the Spider controller:
The Spider controller has been designed to be 100% compatible with the Arduino Mega and comes with the Arduino boot loader installed. The boot loader allows programs written in the Arduino IDE to be uploaded via the USB interface. More information about Arduino can be found here: http://arduino.cc/en/Guide/Introduction

This manual assumes use of Arduino 0022 or later, which can be downloaded from here: http://arduino.cc/en/Main/Software.

A full list of commands can be found here: http://arduino.cc/en/Reference/HomePage

Communications:
The Spider controller has 4x serial ports and 1x I²C interface. Serial interface 0 is used by the USB interface. Serial ports 1,2 and 3 are available for use with devices such as serial interface LCD's, Xbee, Blue Tooth and WiFi modules or even other processors. Instructions on how to use these serial interfaces can be found here:
http://arduino.cc/en/Reference/Serial

The I²C interface (SDA pin 20, SCL pin 21) allows the controller to communicate with devices such as external memory, real time clock (RTC), DC motor controllers and more. For the I²C interface to function properly pullup resistors must be used. Some devices will include these resistors. If necessary you can enable the internal pullup resistors on pins 20 and 21. Instructions on how to use the I²C interface can be found here:
http://www.arduino.cc/en/Reference/Wire

Digital I/O pins:
All 70 of the Spiders I/O pins can be used for simple digital inputs or outputs. By default, all pins except digital pin 13 have their mode set to input. When a pin is in input mode it is in a high impedance state (effectively open circuit). Digital pin 13 is set to output by the boot loader and has an LED attached. Although D13 can be used as an input the LED may interfere.

Use the pinMode() command to change a pins mode between input and output. Use digitalRead() and digitalWrite() commands to read and write to these pins. **Note:** Analog pins A0 - A15 are digital pins D54 – D69.

Writing a "1" to a digital pin while it is configured as an output will connect that pin to Vcc(+5V). Writing a "0" to a digital pin while it is configured as an output will connect that pin to ground (0V). Each pin is capable of sinking or sourcing up to 40mA maximum but care must be taken to limit the total current to 200mA. If you wish to drive large numbers of LEDs or other devices then the output pins should be buffered.

Writing a "1" to a digital pin while it is set as an input will enable that pins internal pullup resistor. When enabled, an internal 20K resistor is connected between that pin and Vcc (+5V). Writing a "0" to that pin while configured as an input will disable the pullup resistor.

Analog pins:
Pins A0 to A15 are by default analog inputs. Each analog input has 10bit resolution and can measure the voltage on its pin. Input voltage should not exceed the analog reference voltage (default value is +5V).

The input voltage is measured using the analogRead() command. The reference voltage is Vcc(+5V) by default. The reference voltage can be changed using the analogReference() command.

PWM outputs:
The Spider controller is capable of generating 8bit resolution pulse width modulated outputs on digital pins D2 – D13 and D44 – D46. The output of a PWM pin is driven high and then low repeatedly with the duty cycle being adjusted to synthesize an analog output. By adding a simple RC filter to a PWM output a true analog output can be generated.

PWM outputs are generated using the analogWrite() command. As the internal timers of the processor are used to generate these outputs they may be disabled by other commands using the same timer. Pin assignments should be planned to avoid conflicts.

External interrupts:
The Spider has 6 interrupt pins. Interrupts allow functions to be called only when an external event occurs. These pins are useful for monitoring devices such as encoders. The interrupts and their pins are:

Interrupt 0 – D2
Interrupt 1 – D1
Interrupt 2 – D21
Interrupt 3 – D20
Interrupt 4 – D19
Interrupt 5 – D18

More information on the interrupt library can be found here:
http://arduino.cc/en/Reference/Interrupts

Using the EEPROM memory:
The Spider's Atmega1280 includes 4K of EEPROM memory that can be used to store information while the power is off. More information on using the EEPROM library can be found here: http://www.arduino.cc/en/Reference/EEPROM

Using servos:
All 70 I/O pins have a servo compatible 3 pin male header. The pin closest to the outer edge of the PCB is ground, the center pin is +5V and the inner most pin is the signal. This pin arrangement is also useful for powering sensors.

Most miniature and standard servos require between 4.8V and 6V and will work happily directly from the PCB. High-powered servos requiring 6V or more should be powered via an external power source or directly from the battery.

The Spider can drive up to 48 servos simultaneously using the Servo library:
http://www.arduino.cc/en/Reference/Servo

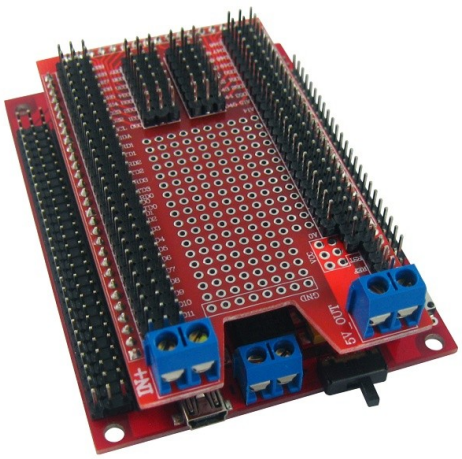
The Servo command uses a timer for each 12 servos used starting with Timer 5. As these timers are also used for commands like PWM you need to plan your pin assignments to avoid conflicts.

1 to 12 servos use timer 5 disabling PWM on pins 44,45 and 46.
13 to 24 servos use timers 1&5 disabling PWM on pins 11,12,44,45 and 46.
25 to 36 servos use timers 1,4&5 disabling PWM on pins 6,7,8,11,12,44,45 and 46.
37 to 48 servos use timers 1,3,4&5 disabling PWM on pins 2,3,5,6,7,8,11,12,44,45 and 46.

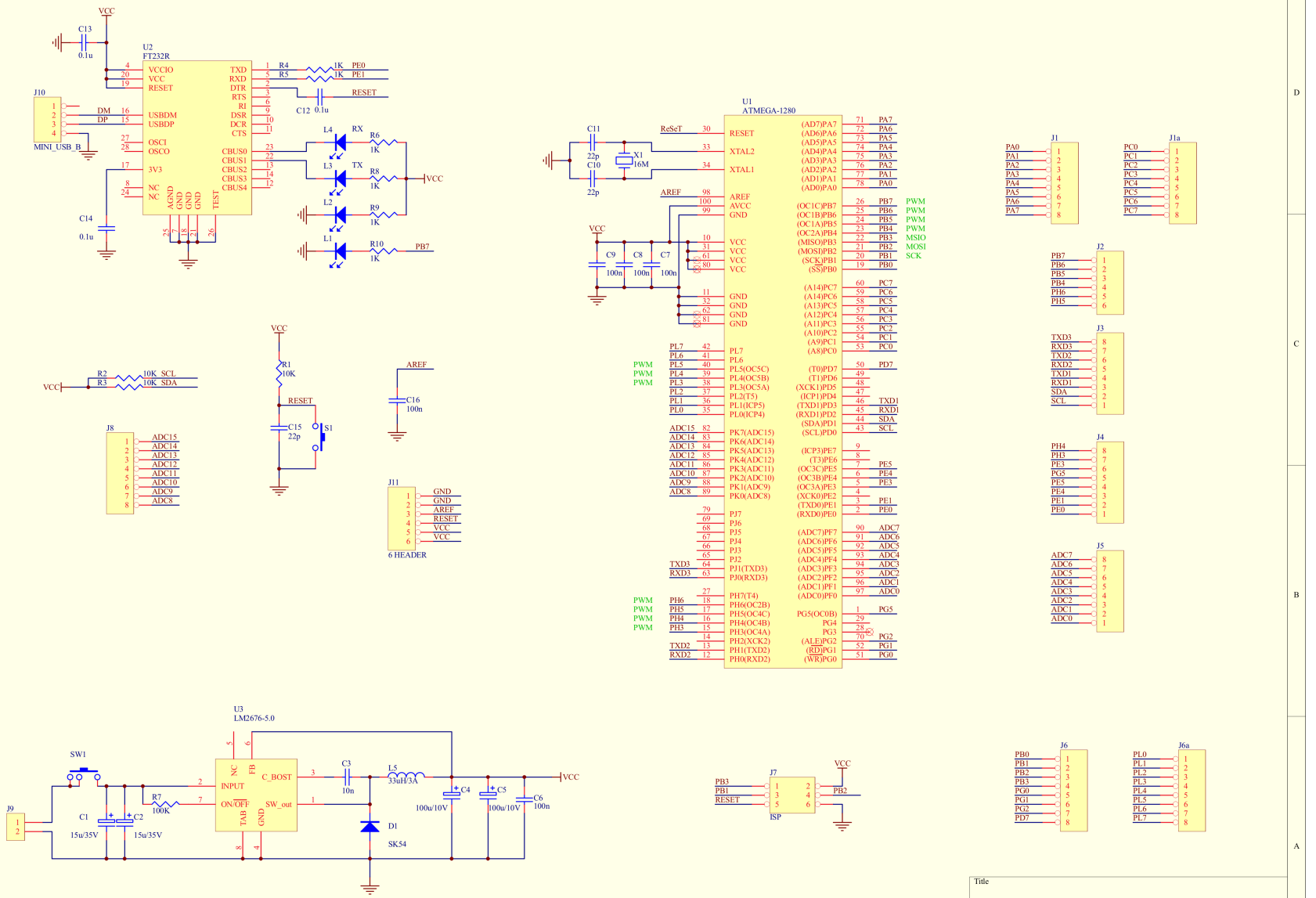
Servos can be assigned to any digital pin from D0 - D53.
Analog pins A0 - A9 (D54 – D63) can also be used if required.

A short tutorial including sample code for driving 48 servos can be found here:
http://letsmakerobots.com/node/25923

Shields:
The female headers on the Spider controller are spaced so that a shield can be easily made from standard prototype PCB. Shields allow you add custom circuitry to your controller. A simple shield for powering servos directly from a battery or external power supply is now available.



Alternative programming methods:
Experienced users may choose to use WinAVR or AVR studio to program their controller directly via the ISP socket. The bootloader can be removed which will fee up an additional 4K of memory.
WinAVR: http://winavr.sourceforge.net/
AVR Studio 5: http://www.atmel.com/microsite/avr_studio_5/


U2
FT232R
VCCIO
VCC
RESET
USBDM
USBDP
OSCI
OSCO
3V3
NC
NC
AGND
GND
GND
GND
TEST
TXD
RXD
DTR
RTS
RI
DSR
DCR
CTS
CBUS0
CBUS1
CBUS2
CBUS3
CBUS4
C13
0.1u
C14
0.1u
C12 0.1u
R4 1K PE0
R5 1K PE1
RESET
J10
MINI_USB_B
DM
DP
L4 RX
L3 TX
L2
L1
R6 1K
R8 1K
R9 1K
R10 1K PB7
VCC
U1
ATMEGA-1280
C11 22p
C10 22p
X1 16M
ReSeT
C9 100n
C8 100n
C7 100n
AREF
C16 100n
R1 10K
C15 22p
S1
RESET
R2 10K SCL
R3 10K SDA
J8
ADC15
ADC14
ADC13
ADC12
ADC11
ADC10
ADC9
ADC8
J11
6 HEADER
GND
AREF
RESET
VCC
J1
J1a
J2
J3
J4
J5
J6
J6a
PA0
PA1
PA2
PA3
PA4
PA5
PA6
PA7
PC0
PC1
PC2
PC3
PC4
PC5
PC6
PC7
PB7
PB6
PB5
PB4
PH6
PH5
TXD3
RXD3
TXD2
RXD2
TXD1
RXD1
SDA
SCL
PH4
PH3
PE3
PG5
PE5
PE4
PE1
PE0
ADC7
ADC6
ADC5
ADC4
ADC3
ADC2
ADC1
ADC0
PB0
PB1
PB2
PB3
PG0
PG1
PG2
PD7
PL0
PL1
PL2
PL3
PL4
PL5
PL6
PL7
U3
LM2676-5.0
SW1
J9
C1 15u/35V
C2 15u/35V
R7 100K
C3 10n
L5 33uH/3A
C4 100u/10V
C5 100u/10V
C6 100n
D1
SK54
J7
ISP
PB3
PB1
RESET
PB2
Title
Size
A3
Number
Revision
Date: 15-Mar-2011
Sheet of
File: E:\硬件原理\arduino1280\arduino1280.DdbDrawn By: