The uORC Manual

Edwin Olson
University of Michigan
ebolson@umich.edu

January 3, 2010

Contents

Chapter 1
Introduction

The uOrc (pronounced "mork") is a flexible I/O board designed to allow a standard computer to interface with the "real world." The uOrc provides data acquisition capabilities, motor drivers, as well as I2C, SPI, and CAN expansion ports. It is particularly well suited to robotics applications, but can also be used for a variety of automation and other interfacing tasks.
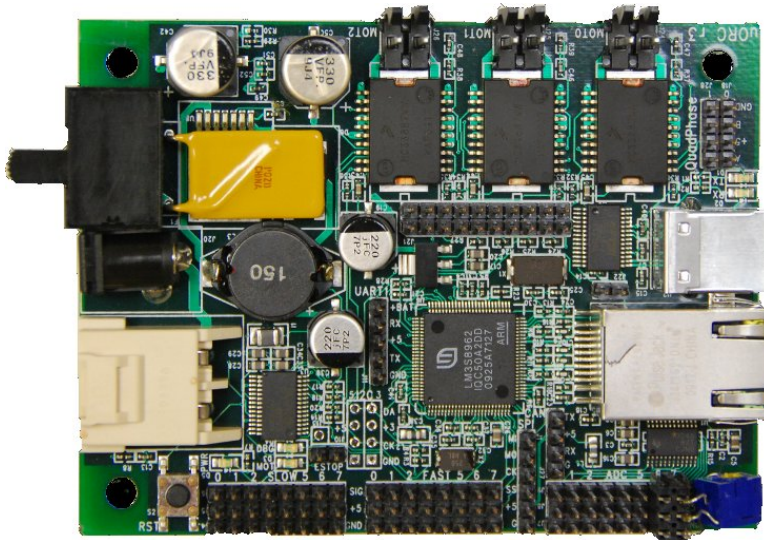
The uOrc is a smaller and cost-optimized version of an I/O controller known as the "OrcBoard". This controller (and its predecessors) have been used by a variety of research and educational organizations, including MIT and the University of Michigan. The controller is completely open source, including everything the controller board schematics, firmware, and user-land utilities.

1.1   Overview

This section provides a brief overview of the uOrc, including its capabilities and basic usage information.

> **Warning!** The uOrc can be easily damaged by incorrect use.
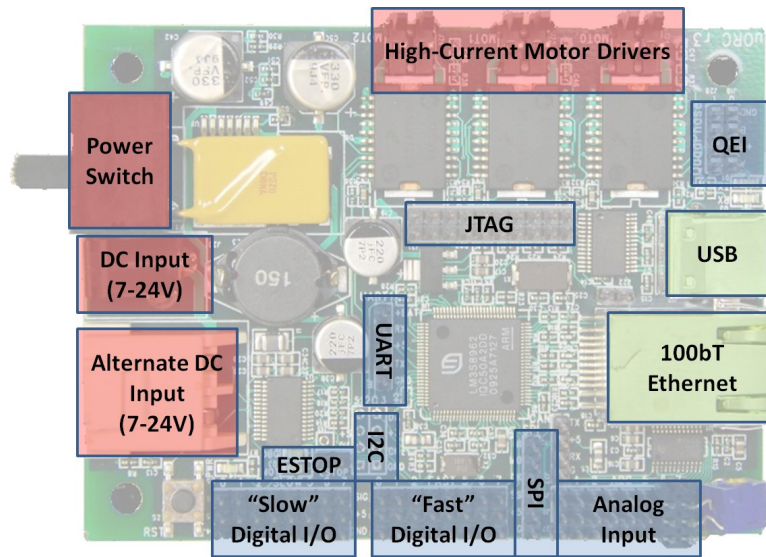> Watch for these warning sections to avoid common mistakes.

Figure 1.1: uOrc Floorplan. The uOrc provides a wide variety of digital, analog, power, and communication facilities.

The uOrc provides 16 digital I/O pins: eight "slow" ports suitable for simple devices like switches and/or LEDs, and eight "fast" ports that can generate/decode more complex signals (such as the PWM generation needed for RC servos). There are an additional two dedicated quadrature phase decoders, and a dedicated emergency stop input.

In addition to digital I/O, the uOrc provides eight dedicated analog inputs, which are sampled at 14bit resolution.

A variety of communications capabilities are available as well, including SPI and I2C expansion buses, and a UART (which can also be used to drive Dynamixel servos).

The uOrc features three high-current motor drivers, capable of up to 3 A of current. These motors are driven by full H-bridges, allowing bi-directional control. The current consumed by the motors is measured, providing additional means of motor feedback.

The uOrc's built-in CPU, a 32 bit ARM microcontroller running at 50 MHz, can be programmed by the user. This provides an all-in-one solution.

However, the uOrc is commonly used as a "slave" device which receives commands from a more powerful computer (like a laptop). For this communication, an ethernet and USB port are provided.

Chapter 2
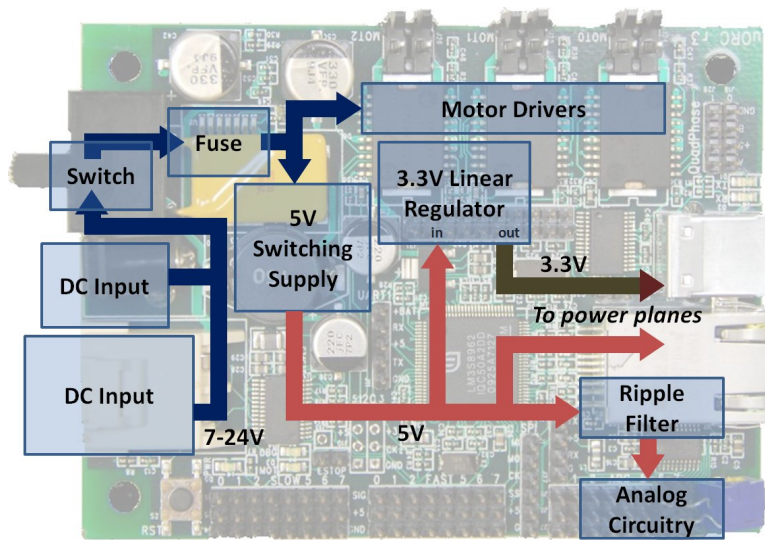# Hardware Reference

2.1   Power

Figure 2.1: Power Diagram.

2.1.1   Input Requirements

The uOrc can operate from input voltages of about 7 V to 28 V. Note that it is much easier to damage the motor drivers at higher voltages. Care should be taken to ensure that the power is connected with the correct polarity, or the uOrc will be damaged.

2.1.2   In-system recharging

Two input jacks are provided (see Fig.2.1); these are internally shorted together *prior* to the switch. The intended use of these two connectors is to allow a battery recharger to plug in to the battery via the uOrc.

In particular, using a 12 V lead-acid battery along with a 13.8 V DC power supply provides a convenient means of powering and recharging the system.

2.1.3   Analog Ripple Filter

In order to improve the quality of the analog-to-digital conversion circuit, a ripple filter is provided. However, if high-current devices are plugged into the analog ports, this ripple filter will need to be bypassed. See Section 2.4 for more information.
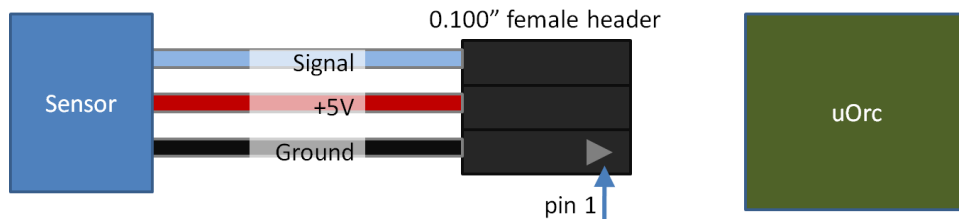
2.2   "Slow" Digital IO

Figure 2.2: Slow Digital IO pinout.

> ***Warning!*** Configuring a pin as an output and externally driving that pin will damage the uOrc board.

2.3   "Fast" Digital IO

2.3.1   Basic digital in and out

> ***Warning!*** Configuring a pin as an output and externally driving that pin will damage the uOrc board.

2.3.2   RC Servo control

2.4   Analog Input

> ***Note:*** The popular Sharp IR range finders consume a great deal of power, which is more than the uOrc's ripple filter can properly filter. When using these sensors, you must add a jumper from a fast digital I/O power pin to the analog I/O power pin. This effectively disables the ripple filter. Adding additional bypass capacitance (e.g., a 100uF electrolytic capacitor) can help reduce the noise in the measurements.

2.4.1 Gyro integration mode

Analog input ports 5-7 are automatically integrated for use with gyroscopes.

2.5 Motor control

2.5.1 Current Feedback

2.6 Quadrature Phase Decoders

2.7 I2C Expansion Bus

> ***Warning!*** Two I2C headers are provided, one which provides 5V power, and another which provides 3.3V power. Be sure to connect your device to the appropriate header, or you may damage your sensor.

2.8 SPI Expansion Bus

> ***Note:*** The SPI expansion bus is used on the uOrc to implment the "slow" digital IO and analog inputs. Thus, the full bandwidth of the SPI bus is not available.

Chapter 3
Communications

3.0.1   Configuring the Ethernet Port

3.0.2   DHCP

> **_Note:_** When the DHCP server is enabled, do not plug the uOrc into another network with its own DHCP server (i.e., a home or office network). Computers which receive their IP address from the uOrc, instead of from the intended DHCP server, may temporarily lose their ability to connect to other computers and the Internet.

Chapter 4
Software API