



# Rotary Racer Display

|           |                         |
|-----------|-------------------------|
| Project   | GpSpeedDisplay          |
| Reference | Version 1.0             |
| Date      | 2017/03/26              |
| Authors   | Terry, Ben, Dan, Louise |

## Introduction

This describes the GpSpeed to GpSpeedDisplay Interface. The GpSpeed is the motor speed controller designed and built by the RR team. The GpSpeedDisplay is a display and control module designed and built by the RR team

## Design

The GpSpeedDisplay module provides a simple 2 line 16 character LCD display along with 4 user push buttons and 4 coloured LED's. It is powered and connected to a GpSpeed motor speed controller using a 6 way cable that provides 5V and has an I2C communications bus.



The GpSpeed controller is the master device. It is responsible for:

1. Controlling the motor speed/power.
2. Writing to the LCD display and LED's the necessary driver information.
3. Reading the switch state from the driver and changing operation as needed.

The GpSpeedDisplay is a slave device and responds to commands from the GpSpeed controller. It is responsible for:

1. Displaying characters on the display as requested by the GpSpeed controller.
2. Setting the LED's on/of state as requested by the GpSpeed controller.
3. Reading the switches continually and returning their on/off state when requested by the GpSpeed controller.
4. Future: Setting an interrupt line whenever the switches state has changed



# Rotary Racer Display

## Display Software Interface

The GpSpeedDisplay is controlled over an I2C bus. It implements two slave devices on this I2C bus with two separate I2C addresses.

| Device             | Address | Description   |
|--------------------|---------|---|
| LCD Display        | \$7C    | The two line LCD display                            |
| Led's and Switches | \$10    | The Picaxe 20X2 implementing LED and switch control |

The LCD display operation is covered in the LCD modules data sheet. The LED and Switch operation is covered below.

| Register Address | Item     | Description  |
|------------------|----------|--|
| \$00             | Switches | The bits 0,1,2,3 of this 8 bit register indicate the state of the switches 0,1,2,3 |
| \$01             | LED's    | The bits 0,1,2,3 of this 8 bit register enables the LED's 0,1,2,3                  |

So to set LED 2 on you would need to send the following I2C message:

```
<address = $10><register = $01><data = $04>
```

In Picaxe basic this would be:

```
hi2cout [$10],$01,($04)
```

To read the switches you can use the following Picaxe command:

```
hi2cin [$10],$00,(b0)
```

The variable b0 will now contain the switches state in its bits 0 – 3.

```
if b0 bit 2 then
```

```
...
```

```
endif
```

In the future the GpSpeedDisplay could set an interrupt line(wire) high whenever a switches state is changed. The GpSpeed controller could then respond to this as an interrupt to its operation and read the switches state. This can speed up switch operation.

## Notes