CMRS\_2560 Theory of Operation

Introduction

The CMRS\_2560 system has been designed to be a modular, integrated switch control, train detection, and signaling system for the Carquinez Model Railroad Society. The CMRS\_2560 uses Ethernet enabled microcontrollers for detecting switches and trains and controlling signal indications and switch positions. The system also provides feedback to JMRI via its SimpleServer interface so that microcontrollers can share state information across Ethernet and also send feedback to JMRI to keep dispatcher controls in sync.

JMRI’s SimpleServer is a critical component of the overall system; all CMRS\_2560 devices must be able to connect to the SimpleServer to not only send state information on devices that each microcontroller controls, but also to register to receive updates on state variables of interest. JMRI’s SimpleServer provides filtering to minimize the amount of data that is retransmitted to clients.

Hardware Description

Each CMRS\_2560 “station” consists of an Arduino microcontroller and a set of configurable devices that are attached to the Arduino. The Arduino selected for this application is the Mega 2560, an uprated version of the Arduino Uno R3. The Mega 2560 was selected because of the software complexity to run the “station.” The Mega 2560 has 256k of program memory, which is Read Only Memory (ROM), and 8k of Random Access Memory (RAM) for the dynamic variables. There is also 2k of system programmable ROM where configuration and persistent state data for the “station” is kept.

Arduino hardware architecture allows for the addition of “shields” to the system. Shields are daughter boards that provide additional I/O or communications functionality to the Arduino. Most CMRS\_2560 “stations” have two shields added to the Mega 2560 controller board. The first shield is a “Grove” shield that breaks out certain I/O pins from the Arduino with a modular 4 pin Grove connector (see below). The second shield is a W5100 Ethernet shield, which provides both physical Ethernet connectivity to the station, but also a microSD card interface that is used as part of the “station” configuration process.

The Grove shield provides a connectorized breakout of the Inter-IC Communications (I2C) serial bus from the Arduino that is used to communicate to all of the modular boards. This breakout is much more cost effective than using a screw terminal breakout shield (there are more than 50 digital and analog I/O pins on a Mega 2560) and far more reliable than simply sticking a wire into edge sockets of the Mega 2560.

I2C Hardware Bus

The I2C bus is a bidirectional data bus that was originally designed for serial communications between IC’s across a printed circuit board. Arduino and other microcontroller manufacturers have provided I2C as a short distance serial bus that provides access to a plethora of I/O, sensor, and control IC’s that are integrated with other hardware to provide small sensor and controls for robotics and other motion automation.

The I2C bus for the CMRS\_2560 application consists of a clock line (SCL) and a data line (SDA), along with 5V power and a common return. Rather than run the I2C clock at the default 1MHz, for the CMRS\_2560 the clock is software selected to be 100kHz. Running the clock at a slower speed should provide more reliable operation across a longer I2C bus. Using shielded wiring, bus lengths over 6 feet long have been tested on the bench; however, these bus lengths have not been tested installed at the club to date.

Ethernet Controller

The W5100 Ethernet board is a relatively low-end Ethernet controller, but the communications protocol for all of the devices is simple Telnet connection that sends data on state updates and an occasional sync message and receives a subset of those messages back from the JMRI SimpleServer.

A physical Ethernet board was selected over a WiFi enabled Arduino or shield for connection reliability and to minimize interference and load to the club WiFi network. Though not installed at this time, there is a plan to use three or four Cisco switches as concentrators around the Layout Room that will be exclusively for the CMRS\_2560 system and will be an unrouteable network space so as to minimize the possibility of the network being attacked.

To date, the CMRS\_2560 systems that have been deployed do not require any feedback from other parts of the network and are running without the network connected. The software has been designed to allow the CMRS\_2560 “station” to continue to run and allow for local operation of switches even if the network connection is missing or the connection to the JMRI Simple Server is unavailable. In these cases, missing feedback may cause unexpected signal aspects, but shouldn’t impact switch control, except for the rare situation where a “station” may have a remote panel mirroring the station behavior via the network connection.[[1]](#footnote-1)

Interface Boards

Connected to the I2C bus are the CMRS Interface Boards. These are a combination of custom built boards and commercially available I2C boards. Four types of I2C interface chips are used: the Texas Instruments PCF8574[[2]](#footnote-2) and PCF8574A[[3]](#footnote-3) Remote 8-Bit I/O Expander for I2C Bus, the Microchip Technologies MCP23017 16-Bit I/O Expander with Serial Interface[[4]](#footnote-4), and a Liquid Crystal display controller (TBD).

There are several boards currently in operation or development:

* 4 Channel Turnout Controller (MCP23017 controller)
* Toggle Switch Board – 4 inputs (PCF8574 controller)
* Indicator Board – 8 outputs (PCF8574A controller)
* Detector Input Board – 8 inputs (PCF8574A (?) controller)
* Signal Driver Board – 4 signals (MCP23017 controller)
* 4x4 Keypad (PCF8574 controller commercial board)
* 16 channel Relay Board Controller (MCP23017 controller commercial board)
* Liquid Crystal – 20x4 display (TBD controller commercial board integrated)

The addressing scheme for I2C follows the specification in the follow diagram:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 7 (MSB) | 6 | 5 | 4 | 3 | 2 | 1 | 0 (LSB) |
| I2C add. | L | D | D | D | A2 | A1 | A0 | R/W |

The “address” specified in the Arduino software interface is actually the 6 bits, 6 to 1. Bits 6 through 4 are usually masked permanently by the I/O device. For the PCF8574 and MCP23017, DDD is masked to 100; for the PCF8574A, DDD is masked to 111. The three programmable bits, A2, A1, and A0, are pins on the device and are generally set by a DIP toggle switch on the board in question. Thus the accessible address space for the CMRS\_2560 devices is

|  |  |  |
| --- | --- | --- |
| Device | Hex Address (A6-A1) | Decimal Address (A6-A1) |
| PCF8574 | 20 – 27 | 32 – 39 |
| PCF8574A | 38 – 3F | 56 – 63 |
| MCP23017 | 20 – 27 | 32 – 39 |
| Liquid Crystal | 27 | 39 |

Thus, for the selection of devices in this project, there are only 16 available addresses on the I2C bus. The current allocation of addresses is thus:

|  |  |  |
| --- | --- | --- |
| Device Type | Max in Project | Address allocated in Software |
| Toggle Switch Board | 2 | 32 – 33 |
| Sensor Board | 2 | 34 – 35 |
| Quad Turnout Board | 4 | 36 – 39 |
| Indicator Board | 2 | 56 – 57 |
| Signal Board | 2 | 38 – 39 |
| Relay Board | 1 | 39 |
| Liquid Crystal | 1 | 39 |

1. This has not been implemented yet but is a design element for the Mococo Line stations 62 and 63. [↑](#footnote-ref-1)
2. PCF8574 Datasheet: <https://www.ti.com/lit/ds/symlink/pcf8574.pdf> [↑](#footnote-ref-2)
3. PCF8574A Datasheet: <https://www.ti.com/lit/ds/symlink/pcf8574a.pdf> [↑](#footnote-ref-3)
4. MCP23017 Datasheet: <https://ww1.microchip.com/downloads/en/devicedoc/20001952c.pdf> [↑](#footnote-ref-4)