|  |
| --- |
| C:\Users\h223669\AppData\Local\Microsoft\Windows\INetCacheContent.Word\Picture1.jpg |
| Digital Interface Board Reference Manual  Version 1.3 |
| |  |  |  | | --- | --- | --- | | Alakh.Sethi@Honeywell.com | 7/27/17 | HONEYWELL – CONFIDENTIAL | |

# Introduction

The digital interface board has been designed to interface with several types of I2C and SPI slave devices. The boards follow a modular approach and each can interface 32 slaves. To expand the number of slaves interfaced, up to four boards can be stacked together.

The boards accept simple commands through a RS-232 port via a PC or workstation. Each command is processed and a response is generated, leading to a simple communication strategy. As the command and response formats are character-based, it is easy to create control software on the PC or workstation through simple scripting languages.

There are two component revisions of the board. The difference between them is outlined in the later sections. They will be referred to as **CREV1** and **CREV2**.

This manual describes the hardware and firmware features of the digital interface boards. This can be used to understand the theory of operation of these boards, or to manufacture and assemble these boards, or to develop interfacing software for these boards. Please refer to the embedded design documents at the end of this manual for additional details.

# Features

* The hardware has been designed to operate at both 3.3 V and 5 V logic levels.
* The **CREV1** hardware can operate at ambient temperatures between -40°C and +85°C, and the **CREV2** hardware can operate at ambient temperatures between -40°C and +125°C.
* Each board can interface up to 32 slave devices (I2C or SPI).
* The hardware design is modular; up to four boards can be stacked together.
* Firmware is slave-independent; slave-specific drivers are developed at the PC/workstation level.
* Simple character-based communication protocol via an RS-232 port.
  + **CREV1:** 19.2 kbaud serial communication.
  + **CREV2:** 115.2 kbaud serial communication.
* All four SPI modes of communication are supported.
* Supported SPI communication speeds:
  + **CREV1:** 4 MHz, 1 MHz, 250 kHz.
  + **CREV2:** 5.6448 MHz, 1.4112 MHz, 352.8 kHz.
* Supported I2C communication speeds: 100 kHz, 50 kHz, 25 kHz.

# Applications

* Monitoring digital outputs from many sensors.
* Calibration and characterization of sensors.
* In-chamber operation with slaves.
* I2C or SPI communication validation.

# System Diagram



# Protoc delete connected to this should connect to mosi there is nothing connected shipping is connected to most 16 is connected to 7th correct it is not working even though it is not working I love youol Structure

The Mass Evaluation Digital Interface board accepts commands from the PC via a RS-232 port. All commands and responses are encoded in ASCII and the C string termination character (NUL) is used to delimit individual command strings. All commands elicit a response from the board, and the front-end interface must wait for the response before sending further commands. The maximum command size is 200 characters (including the NUL termination character).

## RS-232 Packet Format

The digital interface board's RS-232 connection runs at 19200 baud for **CREV1** and 115200 baud for **CREV2**, with 8 data bits, 1 start bit, 1 stop bit and no parity bits. The command and response formats are described below:

Table 1: Command format

|  |  |
| --- | --- |
| CMD | NUL |

**CMD** - A string of characters corresponding to any I2C, SPI or MISC command. Includes any other data necessary for the command.  
**NUL** - The C string termination character ('\0').

Table 2: Response format

|  |  |  |  |
| --- | --- | --- | --- |
| "ER\_" | TYPE | DATA | NUL |

**TYPE** - Error type. All types listed in Table 3.  
**DATA** - The data of the response, if any was requested and there was no error. Data is returned as bytes in order with each byte being represented by two hexadecimal characters (0-9, A-F).  
**NUL** - The C string termination character ('\0').

Table 3: Error types

|  |  |
| --- | --- |
| **TYPE** | **Result** |
| "00" | No error. Command finished successfully. |
| "10" | I2C acknowledge error. The slave device failed to acknowledge its address or any data byte. |
| "20" | Timeout error. The specified timeout has elapsed without any response, or an I2C bus collision has occurred. |
| "40" | Internal error. An error occurred while processing the command. Inspect the sent commands to diagnose the error. |
| "80" | Command error. There was an error detected in the format of the sent command. |

## I2C Commands



## SPI Commands



C

## Miscellaneous Commands



# The ID Data

The ID fields and the ID read command has been introduced from firmware revision 2.1. This is a powerful system which can be used to track the boards across any deployment. This feature provides traceability for the boards in case of failure and various other benefits like software compatibility checks in case of major protocol changes in future revisions. The ID fields consist of six bytes which are described in the table below. By issuing the ID read command, one to six of these bytes can be retrieved. These fields provide scalability in case of future hardware and software expansion.

Table 4: ID fields description

|  |  |  |  |
| --- | --- | --- | --- |
| **ID Byte** | **Name** | **Description** | **Remarks** |
| ID [0] | FREV | Firmware revision | This byte reads back as the complete firmware revision which is represented as:  **HEX2DEC(**FREV >> 4**).HEX2DEC(**FREV & 0x0F**)** – **Major.Minor** |
| ID [1] | HREV | Hardware revision | This byte reads back as 0x01 for **CREV1** boards and 0x02 for **CREV2** boards. A value of 0x00 is defined to be reserved. |
| ID [2] | UIDH | Unique ID high byte | This is the high byte of the unique ID programmed on the boards. UIDH:UIDL is an unsigned 16-bit integer which is programmed in sequence. |
| ID [3] | UIDL | Unique ID low byte | This is the low byte of the unique ID programmed on the boards. UIDH:UIDL is an unsigned 16-bit integer which is programmed in sequence. |
| ID [4] | Reserved | Reserved for future use | Reserved. Reads result in a value of 0x00. |
| ID [5] | Reserved | Reserved for future use | Reserved. Reads result in a value of 0x00. |

# Jumper Configuration

The digital interface board is configured using five jumpers. It is important to ensure correct configuration before powering the boards and establishing communication. The jumper configurations are described in the table that follows:

* The MSEL jumper (**P33**) is used to select the state of the PIC18LF2523 MCU mounted on the board. **IMPORTANT:** There can be only one active MCU per stack.
* The P\_RS232 jumper (**P35**) is used to enable power delivery via pin 1 of the DE-9 connector.
* The DSEL jumper (**P39**) selects either the lower nibble or the higher nibble of the decoder outputs to be used for the multiplexer enables. It will need to be closed for boards connected to slaves 1 - 32, 65 - 96; and open for boards connected to slaves 33 - 64 and 97-128.
* The LSEL and HSEL jumpers (**P38**, **P37** respectively) are used to select whether the board is a part of the lower 64 slaves or the higher 64 slaves.

Table 5: Jumper configuration

|  |  |  |
| --- | --- | --- |
| **MSEL** | **Result** | |
| Closed | The MCU on the board is active and drives the address lines and the communication buses. | |
| Open | The MCU on the board is inactive and does not drive any lines (High-Z). | |
| **P\_RS232** | **Result** | |
| Closed | Power can be accepted through pin 1 of the DE-9 connector. | |
| Open | Power cannot be accepted through pin 1 of the DE-9 connector. | |
| **DSEL** | **Result** | |
| Closed | Selects the lower nibble of the decoder outputs to be used for the multiplexer enables. | |
| Open | Selects the higher nibble of the decoder outputs to be used for the multiplexer enables. | |
| **LSEL** | **HSEL** | **Result** |
| Closed | Closed | WARNING: Invalid configuration. Do not use. |
| Closed | Open | The current board services a section of the lower 64 slaves. |
| Open | Closed | The current board services a section of the higher 64 slaves. |
| Open | Open | WARNING: Invalid configuration. Do not use. |

**IMPORTANT:** Trying to communicate with a slave device outside the address range of the available hardware may cause erroneous data to be captured. This is due to the lines being released from all available channels and not being held at a defined level. An example of a situation where this may occur is when an address select command is issued for addressing slave #100 (situated on board #4 in the stack) and board #4 has not been mounted in the stack. Sending further communication commands like an SPI read will read back values which may not even match with a situation where board #4 has been mounted but slave #100 is not attached.

# Component Revisions

There are currently two component revisions (**CREV1** and **CREV2**) of the boards as mentioned earlier. They are essentially the same hardware board revision assembled with a different set of components. As of this manual revision, the **CREV1** boards are depreciated. Only a limited set of **CREV1** boards have been assembled. This section documents the core differences between them:

Table 6: **CREV1** vs. **CREV2** boards

|  |  |  |
| --- | --- | --- |
| **Parameter** | **CREV1** | **CREV2** |
| **MCU** | The **CREV1** boards use a PIC18LF2523 MCU. | The **CREV2** boards use a PIC18F25K40 MCU. |
| **XTAL** | The **CREV1** boards use a 16 MHz crystal oscillator. This limits the RS-232 communication to 19.2 kbaud. | The **CREV2** boards use a 22.5792 MHz crystal oscillator. This allows RS-232 communication at 115.2 kbaud. |
| **Series Resistors** | The **CREV1** boards use a 100 Ω resistor in series with all signal lines. | The **CREV2** boards use a 33 Ω resistor in series with all signal lines, improving the noise margin. |
| **Slave Pinouts** | The I2C SDAx line is muxed with the SPI MISOx line. | The I2C SDAx line is muxed with the SPI MOSIx line. |
| **UID Field** | The unique IDs assigned to these boards are from 0 to 3 as only four boards have been assembled. | The unique IDs assigned to these boards start at 4. |

The easiest way to distinguish between the two revisions is via the MCU soldered on the board. The **CREV2** boards also have a different pinout configuration from the **CREV1** boards on the slave connectors. This is described in the next section.

# Pinouts

## Power Connector

The power connector (**E1**) is a screw terminal block which can be used for supplying power to the board.

**IMPORTANT:** Care should be taken while screwing or unscrewing the power connector, as over-torqueing it can result in the solder joint shearing off.

Table 7: Power connector pinout

|  |  |  |
| --- | --- | --- |
| **Pin** | **Name** | **Function** |
| **1** | PVIN | The positive input supply point. Only apply voltages between 3V and 5V to this terminal (referenced to the common). |
| **2** | GND | The common reference connection for the power supply. |

## RS-232 Connector

The RS-232 connector (**P36**) is a DE-9 male connector which is used to interface to a DTE (PC).

Table 8: RS-232 connector pinout

|  |  |  |
| --- | --- | --- |
| **Pin** | **Name** | **Function** |
| **1** | P\_VIN | Connects to PVIN if the jumper P\_RS232 is closed. Not connected otherwise. |
| **2** | TxD | This is the transmitter output of the RS-232 transceiver. |
| **3** | RxD | This is the receiver input of the RS-232 transceiver. |
| **4** | N.C. | Not connected. |
| **5** | GND | The shared common reference voltage. |
| **6** | N.C. | Not connected. |
| **7** | N.C. | Not connected. |
| **8** | N.C. | Not connected. |
| **9** | N.C. | Not connected. |



Figure 1: The power and RS-232 connectors

## ICSP and ICD Connector

The ICSP connector (**P34**) offers the possibility of programming and debugging the MCU mounted on the board. Since the final application requires the use of PGM as an address pin, only high voltage programming may be used.

Table 9: ICSP connector pinout

|  |  |  |
| --- | --- | --- |
| **Pin** | **Name** | **Function** |
| **1** | MCLR | The master clear pin. Used to supply the programming voltage. |
| **2** | VCC | This pin connects to the supply voltage of the board. Can be used to power the board from the debugger. |
| **3** | GND | The shared common reference voltage. |
| **4** | PGD | Programming data. |
| **5** | PGC | Programming clock. |
| **6** | RB5 | Unused. Do not connect. |



Figure 2: The ICSP connector

## Slave Connectors

The slave connectors (**P1** – **P32**) on each board are responsible for interfacing the I2C and SPI slave devices.

Table 10: Slave connector pinout for **CREV1** boards

|  |  |  |
| --- | --- | --- |
| **Pin** | **Name** | **Function** |
| **1** | VCC\_t | Slave device supply voltage. |
| **2** | VCC\_t | Slave device supply voltage. |
| **3** | **I2C:** Pull-up  **SPI:** SSx | The SPI slave select line. |
| **4** | GND | The shared common reference voltage. |
| **5** | **I2C:** SDAx  **SPI:** MISOx | The I2C data line or SPI MISO line. |
| **6** | GND | The shared common reference voltage. |
| **7** | **I2C:** SCLx  **SPI:** SCLKx | The I2C and SPI clock line. |
| **8** | **I2C:** Pull-up  **SPI:** MOSIx | The SPI MOSI line. |
| **9** | N.C. | Not connected. |
| **10** | N.C. | Not connected. |



Figure 3: The slave connectors

Table 11: Slave connector pinout for **CREV2** boards

|  |  |  |
| --- | --- | --- |
| **Pin** | **Name** | **Function** |
| **1** | VCC\_t | Slave device supply voltage. |
| **2** | VCC\_t | Slave device supply voltage. |
| **3** | **I2C:** Pull-up  **SPI:** SSx | The SPI slave select line. |
| **4** | GND | The shared common reference voltage. |
| **5** | **I2C:** Pull-up  **SPI:** MISOx | The SPI MISO line. |
| **6** | GND | The shared common reference voltage. |
| **7** | **I2C:** SCLx  **SPI:** SCLKx | The I2C and SPI clock line. |
| **8** | **I2C:** SDAx  **SPI:** MOSIx | The I2C data line or SPI MOSI line. |
| **9** | N.C. | Not connected. |
| **10** | N.C. | Not connected. |

## Stack Connectors

The stack connectors (**J1**, **J2**) are stack-through connectors which enable the boards to mate by stacking them on each other. Power and various signals required for slave communication are shared via these connectors.

Table 12: J1 stack connector pinout

|  |  |  |
| --- | --- | --- |
| **Pin** | **Name** | **Function** |
| **2** | SCLK | The shared SCLK signal. |
| **6** | MISO | The shared MISO signal. |
| **10** | MOSI | The shared MOSI signal. |
| **14** | SS | The shared SS signal. |
| **2n, *n ≠ 1,3,5,7*** | VCC | The shared positive supply voltage. |
| **2n+1, *n ≥ 0*** | GND | The shared common reference voltage. |

Table 13: J2 stack connector pinout

|  |  |  |
| --- | --- | --- |
| **Pin** | **Name** | **Function** |
| **1** | A0 | The shared A0 signal. |
| **5** | A1 | The shared A1 signal. |
| **9** | A2 | The shared A2 signal. |
| **13** | A3 | The shared A3 signal. |
| **17** | A4 | The shared A4 signal. |
| **21** | A5 | The shared A5 signal. |
| **25** | A6 | The shared A6 signal. |
| **2n, *n ≥ 0*** | VCC | The shared positive supply voltage. |
| **2n+1, *n ≠ 0,2,4,6,8,10,12*** | GND | The shared common reference voltage. |



Figure 4: The stack connectors

# Revision History

|  |  |  |
| --- | --- | --- |
| **Revision** | **Date** | **Changes** |
| 1.0 | 4-Jul-17 | First release with v2.0 beta firmware. |
| 1.1 | 13-Jul-17 | Added release board firmware images, PCB artwork, and the truth table. |
| 1.2 | 21-Jul-17 | Updated firmware and documentation for a new ID read command. |
| **1.3** | 27-Jul-17 | Updated firmware and documentation; added ID tracking spreadsheet. |

# Design Documents

|  |  |  |  |
| --- | --- | --- | --- |
| **#** | **File Name** | **Description** | **Attachment** |
| **1** | 32329177\_e.0.1.pdf | Schematic document |  |
| **2** | 32329178\_e.artwork.1.pdf | PCB artwork |  |
| **3** | 32329178\_e.pcb.1.zip | PCB gerbers |  |
| **4** | 32329178\_e.stencil.1.zip | Stencil gerbers |  |
| **5** | 32329179\_e.1.1.pdf | Assembly drawing |  |
| **6** | 32329179\_Rev1\_BOM\_CREV1.xlsx | Bill of materials (**CREV1**) |  |
| **7** | 32329179\_Rev1\_BOM\_CREV2.xlsx | Bill of materials (**CREV2**) |  |
| **8** | Signal Switching Truth Table.xlsx | Signal switching truth table |  |
| **9** | XC8\_18LF2523.zip | Firmware package v2.2 (**CREV1**)  *Updated 27-Jul-17* |  |
| **10** | XC8\_18F25K40.zip | Firmware package v2.2 (**CREV2**)  *Updated 27-Jul-17* |  |
| **11** | DIB ID Tracking.xlsx | ID tracking of the currently assembled boards.  *Updated 27-Jul-17* |  |