Remote I/O Protocol

Philip Munts
Munts Technologies

Revision 7
19 December 2018

Contents

Revision History

Revision 1, 21 April 2017	Initial release.
Revision 2, 17 May 2017	Added device number and delay fields to the SPI transaction request message. Changed the maximum number of SPI devices from 256 to 128.
Revision 3, 30 May 2017	Added missing SPI message definitions. Added ADC message definitions.
Revision 4, 7 August 2017	Corrected some typographical errors.
Revision 5, 27 January 2018	Return the ADC input resolution in the configuration response.
Revision 6, 1 September 2018	Added delay field to the I^2C transaction request message.
Revision 7, 19 December 2018	Added message definitions for DAC outputs, PWM outputs, and abstract devices.

Introduction

This document specifies a lightweight message protocol for performing remote I/O operations. The protocol is implemented using a request/reply pattern, where the ***master*** device (*e.g.* a Linux computer) transmits an I/O request in a 64-byte message to the ***slave*** device (*e.g.* a single chip microcontroller). The slave device performs the requested I/O operation and returns an I/O response in a 64-byte message back to the master device.

The protocol is kept as simple as possible (exactly one 64-byte request message and one 64-byte response message) to allow using low end single chip microcontrollers such as the **`PIC16F1455`** for the slave device. Although particularly suited for USB raw HID devices, this protocol can use any transport mechanism that can reliably transmit and receive 64-byte messages, such as UDP, ONC/RPC or ZeroMQ.

Identifiers

Message Numbers

LOOPBACK_REQUEST	0
LOOPBACK_RESPONSE	1
VERSION_REQUEST	2
VERSION_RESPONSE	3
CAPABILITY_REQUEST	4
CAPABILITY_RESPONSE	5
GPIO_PRESENT_REQUEST	6
GPIO_PRESENT_RESPONSE	7
GPIO_CONFIGURE_REQUEST	8
GPIO_CONFIGURE_RESPONSE	9
GPIO_READ_REQUEST	10
GPIO_READ_RESPONSE	11
GPIO_WRITE_REQUEST	12
GPIO_WRITE_RESPONSE	13
I2C_PRESENT_REQUEST	14
I2C_PRESENT_RESPONSE	15
I2C_CONFIGURE_REQUEST	16
I2C_CONFIGURE_RESPONSE	17
I2C_TRANSACTION_REQUEST	18
I2C_TRANSACTION_RESPONSE	19
SPI_PRESENT_REQUEST	20
SPI_PRESENT_RESPONSE	21
SPI_CONFIGURE_REQUEST	22
SPI_CONFIGURE_RESPONSE	23
SPI_TRANSACTION_REQUEST	24
SPI_TRANSACTION_RESPONSE	25
ADC_PRESENT_REQUEST	26
ADC_PRESENT_RESPONSE	27
ADC_CONFIGURE_REQUEST	28
ADC_CONFIGURE_RESPONSE	29
ADC_READ_REQUEST	30
ADC_READ_RESPONSE	31
DAC_PRESENT_REQUEST	32
DAC_PRESENT_RESPONSE	33
DAC_CONFIGURE_REQUEST	34
DAC_CONFIGURE_RESPONSE	35
DAC_WRITE_REQUEST	36
DAC_WRITE_RESPONSE	37

Common Message Definitions

All remote I/O devices must implement the following request and response messages.

Common Message Header

Every message shall begin with the following common message header.

Byte 0	Message type
Byte 1	Message number

The ***message type*** determines the contents of the rest of the message.

The ***message number*** is initialized by the master device. The slave device will use the same message number in the response message.

Loopback Request

Byte 0	0
Byte 1	Message number
Bytes 2-62	Arbitrary data

Loopback Response

Byte 0	1
Byte 1	Message number
Byte 2	Error code
Bytes 3-63	Arbitrary data

Version Request

Byte 0	2
Byte 1	Message number

Version Response

Byte 0	3
Byte 1	Message number
Byte 2	Error code
Bytes 3-63	Version string

The version string is free format text and must be terminated with a **NUL** (zero) byte.

Capability Request

Byte 0	4
Byte 1	Message number

Capability Response

Byte 0	5
Byte 1	Message number
Byte 2	Error code
Bytes 3-63	Capability string

The capability string shall contain capability tokens separated by a single space and must be terminated with a **NUL** (zero) byte. Tokens may be in any order.

The following capability tokens are defined:

ADC
DAC
DEVICE
GPIO
I2C
PWM
SPI

An example of a valid capability string from a remote I/O device capable of both GPIO and I^2C services would be:

“GPIO I2C”

GPIO Message Definitions

All of the following request and response messages must be implemented by the remote I/O device if it reports **GPIO** in the capability string.

GPIO pins are numbered 0 through 127 inclusive, and are named **GPIO0** to **GPIO127**.

GPIO Pins Present Request

Byte 0	6
Byte 1	Message number

GPIO Pins Present Response

Byte 0	7
Byte 1	Message number
Byte 2	Error code
Bytes 3-18	GPIO's present

The GPIO present bits are numbered left to right: Byte 3 bit 7 indicates **GPIO0** is present, byte 3 bit 0 indicates **GPIO7** is present, and byte 18 bit 0 indicates **GPIO127** is present.

GPIO Configure Request

Byte 0	8
Byte 1	Message number
Bytes 2-17	GPIO's selected
Bytes 18-33	Data direction bits

The GPIO select bits are numbered left to right: Byte 2 bit 7 corresponds to **GPIO0**, byte 2 bit 0 corresponds to **GPIO7**, and byte 17 bit 0 corresponds to **GPIO127**.

The GPIO data direction bits are also numbered left to right: Byte 18 bit 7 corresponds to **GPIO0**, byte 18 bit 0 corresponds to **GPIO7**, and byte 33 bit 0 corresponds to **GPIO127**.

A data direction bit with a value of **0** indicates the GPIO pin shall be configured as an input. A value of **1** indicates the GPIO pin shall be configured as an output.

*Note: The data direction values (**0**=input, **1**=output) follow the convention of most GPIO devices, **except** Microchip PIC microcontrollers which use the **opposite** convention..*

The slave device must silently ignore any GPIO pin that is not selected, not present, or not configurable.

GPIO Configure Response

Byte 0	9
Byte 1	Message number
Byte 2	Error code

GPIO Read Request

Byte 0	10
Byte 1	Message number
Bytes 2-17	GPIO's selected

The GPIO select bits are numbered left to right: Byte 2 bit 7 corresponds to **`GPIO0`**, byte 2 bit 0 corresponds to **`GPIO7`**, and byte 17 bit 0 corresponds to **`GPIO127`**.

GPIO Read Response

Byte 0	11
Byte 1	Message number
Byte 2	Error code
Bytes 3-18	GPIO state bits

The GPIO state bits are numbered left to right: Byte 3 bit 7 corresponds to **`GPIO0`**, byte 3 bit 0 corresponds to **`GPIO7`**, and byte 18 bit 0 corresponds to **`GPIO127`**.

The slave device must clear the state bit for any GPIO pin that was not selected in the request message or that it cannot read from (either because the pin does not exist or because it is write only).

GPIO Write Request

Byte 0	12
Byte 1	Message number
Bytes 2-17	GPIO's selected
Bytes 18-33	GPIO state bits

The GPIO select bits are numbered left to right: Byte 2 bit 7 corresponds to **GPIO0**, byte 2 bit 0 corresponds to **GPIO7**, and byte 17 bit 0 corresponds to **GPIO127**.'

The GPIO state bits are also numbered left to right: Byte 18 bit 7 corresponds to **GPIO0**, byte 18 bit 0 corresponds to **GPIO7**, and byte 33 bit 0 corresponds to **GPIO127**.

The slave device must silently ignore any GPIO pin is not selected or that it cannot write to (either because the pin does not exist or because it is read only).

GPIO Write Response

Byte 0	13
Byte 1	Message number
Byte 2	Error code

I²C Message Definitions

All of the following request and response messages must be implemented by the remote I/O device if it reports **`I2C`** in the capability string.

I²C buses are numbered 0 through 127 inclusive, and are named **`I2C0`** to **`I2C127`**.

I²C Buses Present Request

Byte 0	14
Byte 1	Message number

I²C Buses Present Response

Byte 0	15
Byte 1	Message number
Byte 2	Error code
Bytes 3-18	Buses present

The I²C bus present bits are numbered left to right: Byte 3 bit 7 indicates **`I2C0`** is present, byte 3 bit 0 indicates **`I2C7`** is present, and byte 18 bit 0 indicates **`I2C127`** is present.

I²C Bus Configuration Request

Byte 0	16
Byte 1	Message number
Byte 2	I²C bus number
Byte 3	Freq bits 31:24
Byte 4	Freq bits 23:16
Byte 5	Freq bits 15:8
Byte 6	Freq bits 7:0

The most common I²C clock frequencies are 100 kHz (100,000) and 400 kHz (400,000). Other frequencies may or may not be supported by the particular remote I/O device.

Note: The maximum usable I²C clock frequency will be limited by the slowest device on the I²C bus.

I²C Bus Configuration Response

Byte 0	17
Byte 1	Message number
Byte 2	Error code

I²C Bus Transaction Request

Byte 0	18
Byte 1	Message number
Byte 2	I²C bus number
Byte 3	I²C device address
Byte 4	Write length, bytes
Byte 5	Read length, bytes
Byte 6-7	Delay, µs
Bytes 8-63	Write data

Either the write length or the read length fields may be zero, indicating a read-only or write-only transaction respectively. The maximum write length is 56 bytes, limited by the 64-byte message size. The maximum read length is 60 bytes, also limited by the 64-byte message size.

I²C Bus Transaction Response

Byte 0	19
Byte 1	Message number
Byte 2	Error code
Byte 3	Read length, bytes
Bytes 4-63	Read data

SPI Message Definitions

All of the following request and response messages must be implemented by the remote I/O device if it reports **SPI** in the capability string.

SPI devices are numbered 0 through 127 inclusive, and are named **SPI0** to **SPI127**.

Note: The SPI bus organization (i.e. which devices are attached to which buses) is private to the remote I/O device.

SPI Devices Present Request

Byte 0	20
Byte 1	Message number

SPI Devices Present Response

Byte 0	21
Byte 1	Message number
Byte 2	Error code
Bytes 3-18	Devices present

The SPI device present bits are numbered left to right: Byte 3 bit 7 indicates **SPI0** is present, byte 3 bit 0 indicates **SPI7** is present, and byte 18 bit 0 indicates **SPI127** is present.

SPI Device Configure Request

Byte 0	22
Byte 1	Message number
Byte 2	Device 0-127
Byte 3	Mode 0-3
Byte 4	Word size in bits
Bytes 5-8	Speed in Hz

The allowed values for the SPI device number, mode, word size, and speed fields depend on the remote I/O device implementation. A word size of 0 implies 8 bits.

SPI Device Configure Response

Byte 0	23
Byte 1	Message number
Byte 2	Error code

SPI Transaction Request

Byte 0	24
Byte 1	Message number
Byte 2	Device 0-127
Byte 3	Write length, bytes
Byte 4	Read length, bytes
Bytes 5-6	Delay, µs
Bytes 7-63	Write data

Either the write length or the read length fields may be zero, indicating a read-only or write-only transaction respectively. The maximum write length is 57 bytes, limited by the 64-byte message size. The maximum read length is 60 bytes, also limited by the 64-byte message size.

SPI Transaction Response

Byte 0	25
Byte 1	Message number
Byte 2	Error code
Byte 3	Read length, bytes
Bytes 4-63	Read data

ADC (Analog to Digital Converter) Message Definitions

All of the following request and response messages must be implemented by the remote I/O device if it reports **ADC** in the capability string.

ADC channels are numbered 0 through 127 inclusive, and are named **ADC0** to **ADC127**.

ADC Channels Present Request

Byte 0	26
Byte 1	Message number

ADC Channels Present Response

Byte 0	27
Byte 1	Message number
Byte 2	Error code
Bytes 3-18	Channels present

The ADC channel present bits are numbered left to right: Byte 3 bit 7 indicates **ADC0** is present, byte 3 bit 0 indicates **ADC7** is present, and byte 18 bit 0 indicates **ADC127** is present.

ADC Channel Configure Request

Byte 0	28
Byte 1	Message number
Byte 2	Channel 0-127

ADC Channel Configure Response

Byte 0	29
Byte 1	Message number
Byte 2	Error code
Byte 3	Bits of resolution

ADC Read Request

Byte 0	30
Byte 1	Message number
Byte 2	Channel 0-127

ADC Read Response

Byte 0	31
Byte 1	Message number
Byte 2	Error code
Bytes 3-6	Data sample

The analog data sample is a 32-bit unsigned integer. Response message byte 3 is the most significant byte and byte 6 is the least significant byte. Allowed values are **0** to $2^{Resolution}-1$.

Note: The actual ADC subsystem organization (devices, channels, resolutions, signal conditioning, etc.) is private to the remote I/O device.

DAC (Digital to Analog Converter) Message Definitions

All of the following request and response messages must be implemented by the remote I/O device if it reports **DAC** in the capability string.

DAC channels are numbered 0 through 127 inclusive, and are named **DAC0** to **DAC127**.

DAC Channels Present Request

Byte 0	32
Byte 1	Message number

DAC Channels Present Response

Byte 0	33
Byte 1	Message number
Byte 2	Error code
Bytes 3-18	Channels present

The DAC channel present bits are numbered left to right: Byte 3 bit 7 indicates **DAC0** is present, byte 3 bit 0 indicates **DAC7** is present, and byte 18 bit 0 indicates **DAC127** is present.

DAC Channel Configure Request

Byte 0	34
Byte 1	Message number
Byte 2	Channel 0-127

DAC Channel Configure Response

Byte 0	35
Byte 1	Message number
Byte 2	Error code
Byte 3	Bits of resolution

DAC Write Request

Byte 0	36
Byte 1	Message number
Byte 2	Channel 0-127
Bytes 3-6	Data sample

The data sample field is a 32-bit unsigned integer. Byte 3 is the most significant byte and byte 6 is the least significant byte. Allowed values are **0** to $\mathbf{2^{Resolution}-1}$.

DAC Write Response

Byte 0	37
Byte 1	Message number
Byte 2	Error code

Note: The actual DAC subsystem organization (devices, channels, resolutions, signal conditioning, etc.) is private to the remote I/O device.

PWM (Pulse Width Modulated) Output Message Definitions

All of the following request and response messages must be implemented by the remote I/O device if it reports **PWM** in the capability string.

PWM channels are numbered 0 through 127 inclusive, and are named **PWM0** to **PWM127**.

PWM Channels Present Request

Byte 0	38
Byte 1	Message number

PWM Channels Present Response

Byte 0	39
Byte 1	Message number
Byte 2	Error code
Bytes 3-18	Channels present

The PWM channel present bits are numbered left to right: Byte 3 bit 7 indicates **PWM0** is present, byte 3 bit 0 indicates **PWM7** is present, and byte 18 bit 0 indicates **PWM127** is present.

PWM Channel Configure Request

Byte 0	40
Byte 1	Message number
Byte 2	Channel 0-127
Bytes 3-6	Frequency

The PWM pulse frequency field is a 32-bit unsigned integer in Hz. Byte 3 is the most significant byte and byte 6 is the least significant byte.

Note: Two or more PWM outputs can often share a common clock generator, which means they must be configured with the same PWM pulse frequency. If such groups of PWM outputs are configured with different PWM pulse frequencies, the frequency of the last output configured will typically be used for all outputs in the group.

PWM Channel Configure Response

Byte 0	41
Byte 1	Message number
Byte 2	Error code
Byte 3	Bits of resolution

PWM Write Request

Byte 0	42
Byte 1	Message number
Byte 2	Channel 0-127
Bytes 3-6	Duty Cycle

The duty cycle field is a 32-bit unsigned integer. Byte 3 is the most significant byte and byte 6 is the least significant byte. Allowed values are **0** to $2^{Resolution}-1$.

The PWM output duty cycle, in percent, will be:

$$100.0*DutyCycle/(2^{Resolution}-1)$$

PWM Write Response

Byte 0	43
Byte 1	Message number
Byte 2	Error code

Abstract Device Message Definitions

All of the following request and response messages must be implemented by the remote I/O device if it reports **DEVICE** in the capability string. An abstract device can read and/or write a single 32-bit data item. GPIO pins and ADC inputs can also be implemented and reported as abstract device channels, as can other kinds of sensors (e.g. temperature sensors) and actuators (e.g. PWM outputs).

Abstract device channels are numbered 0 through 127 inclusive, and are named **DEV0** to **DEV127**.

Device Channels Present Request

Byte 0	44
Byte 1	Message number

Device Channels Present Response

Byte 0	45
Byte 1	Message number
Byte 2	Error code
Bytes 3-18	Channels present

The abstract channel present bits are numbered left to right: Byte 3 bit 7 indicates **DEV0** is present, byte 3 bit 0 indicates **DEV7** is present, and byte 18 bit 0 indicates **DEV127** is present.

Device Channel Configure Request

Byte 0	46
Byte 1	Message number
Byte 2	Channel 0-127
Bytes 3-6	Parameter 1
Bytes 7-10	Parameter 2

The meaning (if any) of the parameter fields will vary according to the kind of abstract device. Example meanings might include gain for ADC inputs, 0 (input) or 1 (output) for GPIO pins, or the pulse frequency in Hz for PWM outputs.

Device Channel Configure Response

Byte 0	47
Byte 1	Message number
Byte 2	Error code
Byte 3	Bits of resolution

Device Read Request

Byte 0	48
Byte 1	Message number
Byte 2	Channel 0-127
Bytes 3-6	Parameter 1
Bytes 7-10	Parameter 2

The meaning (if any) of the parameter fields will vary according to the kind of abstract device.

Device Read Response

Byte 0	49
Byte 1	Message number
Byte 2	Error code
Bytes 3-6	Data item

EIO will be returned in byte 2 and all zeroes in bytes 3 to 6 if an attempt is made to read from a write-only device.

Allow values for the data item field are **0** to $\mathbf{2^{Resolution}-1}$.

Device Write Request

Byte 0	50
Byte 1	Message number
Byte 2	Channel 0-127
Bytes 3-6	Parameter 1
Bytes 7-10	Parameter 2
Bytes 11-14	Data item

Allow values for the data item field are **0** to $\mathbf{2^{Resolution}-1}$.

Device Write Response

Byte 0	51
Byte 1	Message number
Byte 2	Error code

EIO will be returned in byte 2 if an attempt is made to write to a read-only device.