Δ Expansion Ltd.

IO Expander Datashheet

Features

- Easy to use and easy to understand I/O Interfacing for any demands
- Powerful plattform for the simplest of requirements
- Giant footprint to improve weight distribution in YOUR product
- Simple UART interfacing for near instant IO operations

Applications

- Super secure systems in family businesses
- Highly specialized Linux systems doing ethernet stuff
- Usage in Hackathons because you can

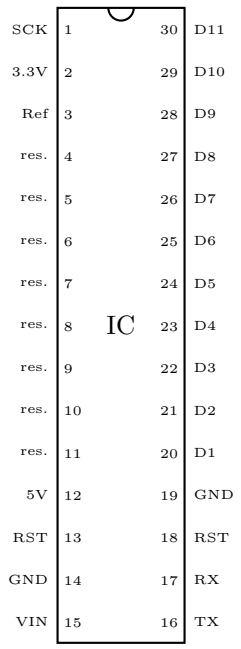

Figure 1: Pinout and internal circuit

General Description

This **datasheet** enables you to understand the basic concepts of this sophisticated and utterly overdimensioned I/O Expander.

Pin Description

Pin description is done using the table below. (Pinout should really be sufficient though...)

Table 1: Pin Description

Pin	Name	Function
1	SCK	Clock
2	3.3V	3.3 V Power supply
3	Ref	Reference
4 - 11	Res	Reserved - Nothing to see here
12	5V	5V Power supply
13	RST	Reset on high
14	GND	Ground
15	VIN	Voltage In
16	TX	UART Transmit - Connect to RX of given controller
17	RX	Uart Receive - Connect to TX of given controller
18	RST	Reset on high
19	GND	Ground
20	D1	Digital I/O Pin
21	D2	Digital I/O Pin
22	D3	Digital I/O Pin
23	D4	Digital I/O Pin
24	D5	Digital I/O Pin
25	D6	Digital I/O Pin
26	D7	Digital I/O Pin
27	D8	Digital I/O Pin
28	D9	Digital I/O Pin
29	D10	Digital I/O Pin
30	D11	Digital I/O Pin

[1] Errors in pin assignment are users fault.

Detailed Description

This I/O Expander is a general-purpose input/output (GPIO) peripheral that provides 11 I/O ports, D1 to D11, controlled through a high-speed serial interface. The 11 I/O ports can be used as inputs or open-drain outputs in any combination.

Communication

The communication with this device is done using 8 Bit messages. There are three distinct messages to control the I/O Expander and one response message which announces the levels of defined input pins. Those concepts are described in the next sections.

Setting Output Ports

The communication with this device is as simple as it gets. To set or reset I/O ports, one must supply 8 bit of date to the device. The protocol expects the application to identify the ports which shall change state to be identified with four LSB according to **??**. Using the most significant nibble of the message, one can set the specified Ports to high or low. The Mapping is shown in **??**

I/O direction definition

Identification if pins as Input / output is done with the "special" definition message. This message has all bits of the most significant nibble set to high. The least significant nibble specifies which pin shall be set as input (High) or as output (Low). This means that all I/O direction definition messages must start with 0xF.

Reading Inputs

Similar to the I/O direction definition, the read request message is a special case. For this request, the high nibble needs to be set to 0xE. In this special case, the least significant nibble is ignored.

When the read request message is received, the I/O Expander returns with a port status message. This Message identifies the direction of all ports in the most significant nibble. If a port is set as output with high level, the corresponding bit in the least significant nibble is set to high (1). If the port is defined as input, the corresponding Bit in the least significant nibble represents high state with a high level (1) or low state with a low level (0).

Table 2: Pin Address Map

Pin	bit8	bit7	bit6	bit5	bit4	bit3	bit2	bit1	Hex
D1	x	x	x	x	0	0	0	1	0xX1
D2	x	x	x	x	0	0	1	0	0xX2
D3	x	x	x	x	0	0	1	1	0xX3
D4	x	x	x	x	0	1	0	0	0xX4
D5	x	x	x	x	0	1	0	1	0xX5
D6	x	x	x	x	0	1	1	0	0xX6
D7	x	x	x	x	0	1	1	1	0xX7
D8	x	x	x	x	1	0	0	0	0xX8
D9	x	x	x	x	1	0	0	1	0xX9
D10	x	x	x	x	1	0	1	0	0xXA
D11	x	x	x	x	1	0	1	1	0xXB

[1] x means dont care. Those Bits are reserved to set high or low, see next table.

Table 3: Setting I/O's to high

Pin	**bit8**	**bit7**	**bit6**	**bit5**	**bit4**	**bit3**	**bit2**	**bit1**	**Hex**
D1	0	0	0	1	x	x	x	x	0x1X
D2	0	0	1	0	x	x	x	x	0x2X
D3	0	0	1	1	x	x	x	x	0x3X
D4	0	1	0	0	x	x	x	x	0x4X
D5	0	1	0	1	x	x	x	x	0x5X
D6	0	1	1	0	x	x	x	x	0x6X
D7	0	1	1	1	x	x	x	x	0x7X
D8	1	0	0	0	x	x	x	x	0x8X
D9	1	0	0	1	x	x	x	x	0x9X
D10	1	0	1	0	x	x	x	x	0xAX
D11	1	0	1	1	x	x	x	x	0xBX

[1] The high nibble sets addressed pins to high (1) or low (0)
[2] Multiple Pins can be set to high or low at the same time by addressing them using the low nibble