IO-Warrior100

Generic universal I/O Controller for USB

Code Mercenaries

1.0 Features

- USB interface
- USB V2.0 compliant full speed device
- USB HID 1.1 compliant
- 82 general purpose I/O pins
- 1000 Hz polling rate for I/O
- IIC master up to 1 MHz
- 4 channel 12 bit A/D converter up to 30 kHz sample rate
- Internal bandgap reference for absolute voltage measurement
- Supports HD44780 compatible display modules and several graphic display modules
- Support for 8 x 64 LED matrix
- Supports up to 500 digital LEDs i.e. WS2812B
- Support for 8x8 or 8x16 switch matrix
- 4 channel 16 bit PWM
- 2 channel 12 bit DAC
- Single +3.3V power supply
- Available in LQFP100 package or as a module

IO-Warrior40 backward compatibility

IOW100 and IOW56 are the recommended replacements for IOW40. There is no direct backward compatibility as both chips offer more functions and higher performance than IOW40 and they come in a different package.
But the basic concept behind the chips is the same. So modifications are necessary on the software side but they will be straight forward as the underlying logic is the same and similar functions are as close as possible to the old chip.

1.1 Custom variants

Custom adaptions are available on request. If necessary complex functions can be added to the standard chips to directly control specific circuits.

2.0 Functional overview

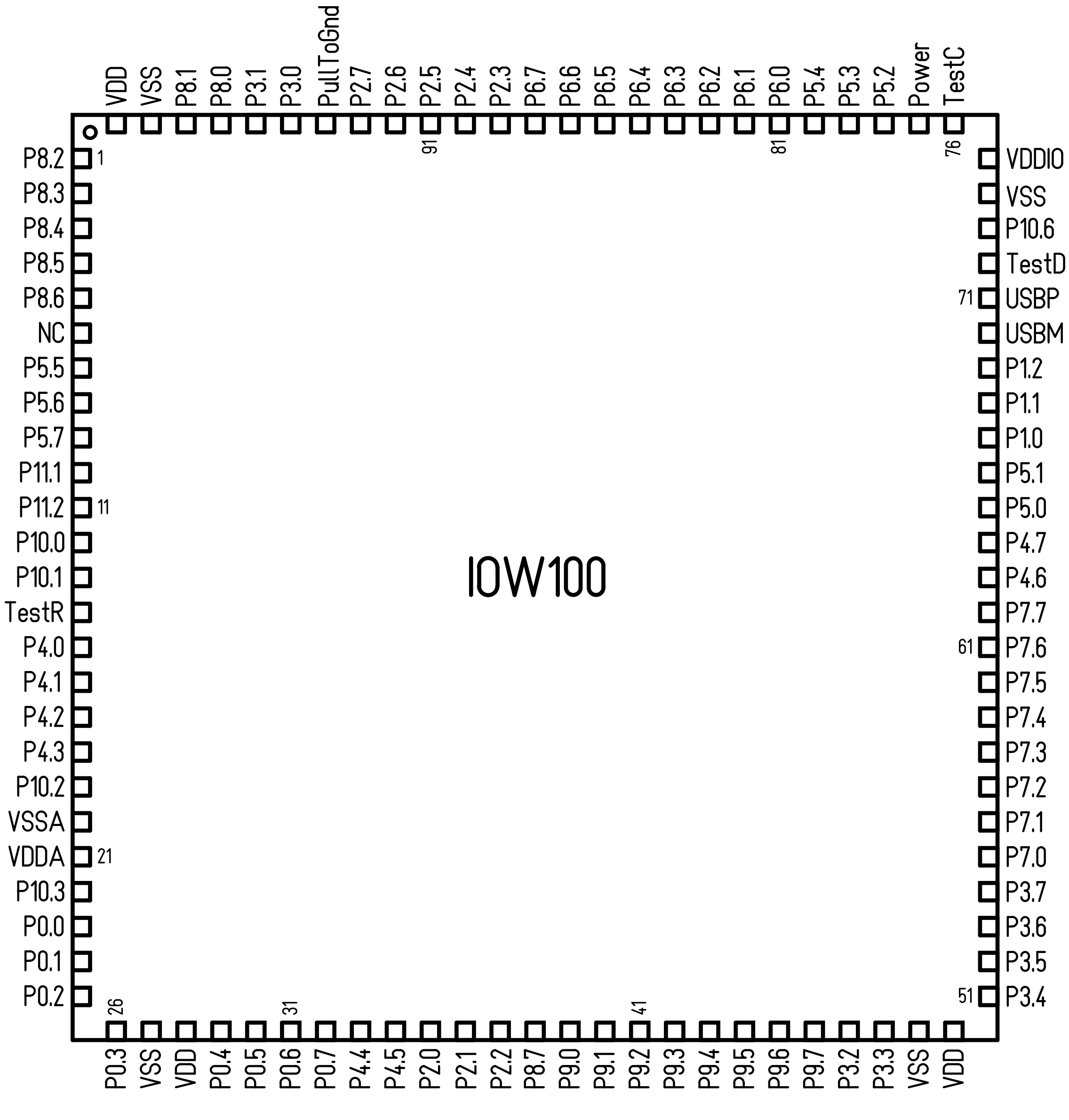
IO-Warrior brings simplicity to the USB. The protocol is encapsulated in the IO-Warrior Chip. You only have to care about the I/O pins and have to only write a few simple lines of code to access them.
IO-Warrior also supports a range of industrial standard interfaces to simplify interfacing to certain chips or modules. These interfaces are handled internally in IO-Warrior removing the bandwith wasting controlling of individual pins.

2.1 Product selection matrix

Type	I/O Pins	LCD	IIC	SPI	RC5	Keys	LEDs	Timer	ADC	DAC	WSLED	PWM	DIL	QFN	Starter Kit
IO-Warrior24 (obsolete)	16	√	√	√	√		√	√					√		√
IO-Warrior24PV (obsolete)	12		√	√									√		
IO-Warrior40 (obsolete)	32	√	√			√	√						√		√
IO-Warrior28L (obsolete)	17	√	√												
IO-Warrior28	19	√	√						√		√		√	√	√
IO-Warrior56	50	√	√	√		√	√		√			√			√
IO-Warrior100	82	√	√			√	√		√	√	√	√			

IO-Warrior100

3.0 Pin configuration
IO-Warrior100-LF100
LQFP100 Chip



All drawings: TOP VIEW!

4.0 Pin descriptions IO-Warrior100-LF100

Name	I/O	Type	Pins	Description
USBP, USBM	I/O	special	71, 70	USB differential data lines
P0.0, P0.1, P0.2, P0.3, P0.4, P0.5, P0.6, P0.7	I/O	I/O open drain, internal pullup	23, 24, 25, 26, 29, 30, 31, 32	First I/O port. Also ADC and DAC port
P1.0, P1.1, P1.2	I/O	I/O open drain, internal pullup	67, 68, 69	Second I/O port
P2.0, P2.1, P2.2, P2.3, P2.4, P2.5, P2.6, P2.7	I/O	I/O open drain, internal pullup	35, 36, 37, 89, 90, 91, 92, 93	Third I/O port. Also fast IIC port, and LED-Matrix outputs
P3.0, P3.1, P3.2, P3.3, P3.4, P3.5, P3.6, P3.7	I/O	I/O open drain, internal pullup	95, 96, 47, 48, 51, 52, 53, 54	Fourth I/O port Also switch matrix function X0…7 pins
P4.0, P4.1, P4.2, P4.3, P4.4, P4.5, P4.6, P4.7	I/O	I/O open drain, internal pullup	15, 16, 17, 18, 33, 34, 63, 64	Fifth I/O port Also LCD control lines
P5.0, P5.1, P5.2, P5.3, P5.4, P5.5, P5.6, P5.7	I/O	I/O open drain, internal pullup	65, 66, 78, 79, 80, 7, 8, 9	Sixth I/O port Also LED data lines
P6.0, P6.1, P6.2, P6.3, P6.4, P6.5, P6.6, P6.7	I/O	I/O open drain, internal pullup	81, 82, 83, 84, 85, 86, 87, 88	Seventh I/O port Also switch matrix function Y0…7 pins
P7.0, P7.1, P7.2, P7.3, P7.4, P7.5, P7.6, P7.7	I/O	I/O open drain, internal pullup	55, 56, 57, 58, 59, 60, 61, 62	Eight I/O port Also switch matrix Y8…15 pins
P8.0, P8.1, P8.2, P8.3, P8.4, P8.5, P8.6, P8.7	I/O	I/O open drain, internal pullup	97, 98, 1, 2, 3, 4, 5, 38	Nineth I/O port Also PWM outputs
P9.0, P9.1, P9.2, P9.3, P9.4, P9.5, P9.6, P9.7	I/O	I/O open drain, internal pullup	39, 40, 41, 42, 43, 44, 45, 46	Tenth I/O port
P10.0, P10.1, P10.2, P10.3, P10.6	I/O	I/O open drain, internal pullup	12, 13, 19, 22, 73	Eleventh I/O port
P11.1, P11.2	I/O	I/O open drain, internal pullup	10, 11	Twelfth I/O port
Power	I	input, internal pullup	77	Checked at power up to select 100 mA or 500 mA
Vss, Vssa		power supply	20, 27, 49, 74, 99	Ground
Vdd, Vdda, Vddi		power supply	21, 28, 50, 75, 100	Supply voltage, connect to 3.3 V
NC		special	6	Not used
PullToGND		special	94	Used during manufacturing, connect to GND
TestR, TestD, TestC		special	14, 72, 76	Used during manufacturing, do not connect

4.1 Pin descriptions

USBP, USBM / D+, D-
Differential data lines of USB. Connect these signals direct to a USB cable. ESD protection may be added.

P0.0...P0.7
First I/O port of the chip. These pins correspond with the lowest 8 bits of the input or output reports (bits 0..7, first data byte).

P1.0...P1.2
Second I/O port. Corresponds to the bits 8..10 (second data byte).

P2.0...P2.7
Third I/O port. Corresponds to the bits 16..23.

P3.0...P3.7
Fourth I/O port. Corresponds to the bits 24..31.

P4.0...P4.7
Fifth I/O port. Corresponds to the bits 32..39.

P5.0...P5.7
Sixth I/O port. Corresponds to the bits 40..47.

P6.0...P6.7
Seventh I/O port. Corresponds to the bits 48..55.

P7.0...P7.7
Eighth I/O port. Corresponds to the bits 56..63.

P8.0...P8.7
Ninth I/O port. Corresponds to the bits 64..71.

P9.0...P9.7
Tenth I/O port. Corresponds to the bits 72..79.

P10.0...P10.3, P10.6
Eleventh I/O port. Corresponds to the bits 80..83, 86.

P11.1, P11.2
Third I/O port. Corresponds to the bits 89, 90.

Power
The Power pin is used to determine the power setting at power up. Pull the pin high to set high power (500 mA) mode and pull it low for low power mode (100 mA). This determines how much power the IOW100 will demand from the host. Checking the power setting happens during reset while shortly enabling an internal pull down resistor.

Vss
Power supply ground.

Vdd, Vddi, Vdda
Supply voltage. Requires 3.3 V.
Vdda is the supply for the ADC and also used as the reference for A/D conversion. When using the ADC it is recommended that the power to this pins is additionally filtered and screened from interference.

TestC, TestD, TestR
These pins are used during production of the IO–Warrior chips, do not connect.

4.2 Special mode pin functions

IO-Warrior supports driving IIC compatible chips and HD44780 compatible display modules and some other display modules direct. Handling IIC via the normal generic I/O would be very slow as each edge of data and clock would have to be transmitted separately. At a rate of 1000 such transactions per second (which is the maximum IO-Warrior is allowed by USB specifications) the maximum bit rate would be around 500 bits/sec.
To make IIC and other devices usable, IO-Warrior implements the special mode functions. By handling the IIC inside IO-Warrior the actual data rate is increased by some orders of magnitude.
When any of the special mode functions is activated a couple pins will no longer respond as generic I/O pins but are under control of the activated special mode function.

4.2.1 IIC mode pins

IO-Warrior100 can act as an IIC master. It offers speeds from 10 kBit/s to 1000 kBit/s. The IIC port has been optimized for maximum performance, allowing transfers of > 60 kByte/s without interfering with other functions of the IOW100. This allows to get close to the theoretical maximum of IIC data transfer except for the fastest speed at 1 MBit/s, where about 60% of the theoretical throughput is possible.
The following pins get reassigned when the IIC function is enabled:

Function	Port pin
SCL	P2.6
SDA	P2.7

These pins will no longer be affected by the data sent via the normal port setting command. Both pins have internal pull up resistors and can be connected direct to IIC compatible chips.
For IO-Warrior100 external 1 kΩ pull-up resistors to 3.3 V are recommended. Without them the IIC performance may be significantly reduced.
IO-Warrior100 monitors its IIC lines and can adapt to relatively high parasitic capacitance on the bus, but this will reduce the effective data rate by slowing down SCL.

4.2.2 LCD mode pins

IO-Warrior100 has support for controlling alpha-numeric display modules based on or compatible with HD44780 as well as some graphic displays which use the same interface.
For simplicity the function is refered to as the LCD function, but there are also OLED and vacuum fluorescence displays that are compatible with this interface.
The following pins get reassigned when the LCD function is enabled:

Function	Port pin
/On	P4.0
RS	P4.1
R/W	P4.2
E	P4.3
Data0	P5.0
Data1	P5.1
Data2	P5.2
Data3	P5.3
Data4	P5.4
Data5	P5.5
Data6	P5.6
Data7	P5.7

When the LCD function is enabled these pins will no longer be affected by the normal port setting command.
/On should be used to enable power supply to LCD modules that have high current demand or backlighting. The /On signal is low when the LCD function is enabled, it does go high when IO-Warrior enters suspend mode or when the LCD function is disabled.
The LCD function requires displays that work with 3.3 V power supply at least for the logic part.

4.2.3 ADC mode pins

Up to four analog input channels are available on IO-Warrior100. The internal ADC has a resolution of 12 bits and a calibrated internal voltage reference. This allows absolute voltage measurement.
It also has an internal temperature sensor with good linearity, but a rather poor absolute accuracy. It may be calibrated for absolute measurements but is fine as it is for measuring temperature difference.
The following pins get reassigned when the ADC function is enabled:

Function	Port pin
AD0	P0.0
AD1	P0.1
AD2	P0.2
AD3	P0.3

Either 0, 1, 2, or 4 channels may be selected. Zero channels is for using the internal temperature sensor, or measuring the Vdda voltage only.
The ADC works relative to the voltage on the Vdda power pin.

4.2.4 Switch Matrix Mode Pins

IO-Warrior100 supports scanning of a 8x8 or 8x16 matrix of keys or switches. When this function is enabled P6.0..7 will turn off their internal pull up resistors and will be used as the Y lines that are periodically driven to Gnd voltage level. P3.0..7 will serve as the X0..7 matrix inputs, they will keep their internal pull up resistors active so a closed switch in the matrix will pull down the X line when the corresponding Y line is driven low.
If the 8x16 mode is enabled P7.0..7 will be used as additional Y lines, serving as Y8..15.
To allow more than two switches to be closed at the same time and still be able to faultlessly detect which of the matrix points are closed it is necessary to insert a diode in series with every key or switch in the matrix. The kathodes of the diodes have to be connected to the Y lines.
The following pins get reassigned when the key mode is enabled:

Function	Port pin
X0	P3.0
X1	P3.1
X2	P3.2
X3	P3.3
X4	P3.4
X5	P3.5
X6	P3.6
X7	P3.7
Y0	P6.0
Y1	P6.1
Y2	P6.2
Y3	P6.3
Y4	P6.4
Y5	P6.5
Y6	P6.6
Y7	P6.7
Y8	P7.0
Y9	P7.1
Y10	P7.2
Y11	P7.3
Y12	P7.4
Y13	P7.5
Y14	P7.6
Y15	P7.7

When IO-Warrior100 enters the suspend mode the X and Y lines will be pulled high by internal pull up resistors. Closing a switch/key does not wake the IO-Warrior.

4.2.5 PWM Outputs

IO-Warrior100 has 4 PWM outputs with 48MHz master clock, 16 bits resolution, common 16 bit phase length, and a common 16 bit prescaler. One to four of the channels can be selected.
The following pins are reassigned in PWM mode:

Function	Port pin
PWM0	P8.3
PWM1	P8.4
PWM2	P8.5
PWM3	P8.6

4.2.6 DAC outputs

IO-Warrior100 supports one or two DAC outputs with 12 bit resolution each. Output ranges from Vss to Vdda. Either one or both channels may be enabled.
The following pins are reassigned for DAC mode:

Function	Port pin
DAC0	P0.4
DAC1	P0.5

4.2.7 LED matrix mode pins

IO-Warrior100 supports driving a LED matrix with up to 8x64 LEDs.

Function	Port pin
Strb	P2.2
Clk	P2.3
/OE	P2.4
Data	P2.5

When the LED Matrix function is enabled these pins will no longer be affected by the normal port setting command.
/OE is driven high when IO-Warrior100 enters the suspend mode. The external driver should then disable the LEDs to stay within the USB power limits for suspend mode.
For more details on how to control a LED matrix please refer to the separate application note.

4.2.8 Digital LED mode pins

IO-Warrior100 supports driving up to 500 digital LEDs with a serial communication like WS2812/ WS2812B.
The timing is programmable to accomodate many different variants of the digital LEDs

Function	Port pin
Dout	P2.1

When the digital LED function is enabled this pin will no longer be affected by the normal port setting command.

5.0 Device operation

Due to the fact that all current operating systems offer an especially easy access to devices in the HID class, IO-Warrior was designed as a generic HID device. While this does not exactly fit the device it makes using it a lot easier.
By identifying as a generic HID class device IO-Warrior avoids being controlled by any of the higher level system drivers, which makes it possible to access IO-Warrior from application level.

5.1 Accessing IO-Warrior

A common misconception with people new to the USB is that they think they can "talk to the USB port". The truth is that you do that as likely as you are going to directly talk to your Ethernet port.
Communication on the USB is always with a specific device attached to the USB. The USB itself is only the medium through which you communicate.
To get access to a certain device you have to look for the VendorID and ProductID of that device. The specific mechanisms for doing so depend on the individual operating system.
For details refer to our sample code.

5.2 IO-Warrior communication

IO-Warrior100 has 8 USB endpoints for the I/O communication. Endpoints are like virtual communication ports into or out of the device.
An endpoint can be assigned to an interface. Interfaces are like virtual devices or subsystems within a device. IO-Warrior100 uses interface 0 to talk to the pins directly, interface 1 to talk to the special mode functions, interface 2 for the fast IIC function, and interface 3 for the ADC.
A SetReport request sending a twelve byte output report to interface 0 sets the port pins. SetReport requests to interface 1, 2, and 3 with a length of 64 bytes are used to control the special mode functions.
For input data IO-Warrior is using endpoint 1 as an interrupt-in endpoint. Interrupt in this case is a bit misleading. For USB interrupt means that data is sent when there is new data available, the host computer is periodically asking the device for new data, the device itself can not initiate the data transfer. IO-Warrior sends a new report any time it detects a change to the input pin status.
Reactions to commands to the special mode functions are sent via endpoint 2, 3, or 4, also in interrupt transfer mode.

5.3 IO-Warrior input behaviour

IO-Warrior scans the status of all pins once every millisecond. If it detects a change from the last status a new report via endpoint 1 is issued. Pins which are currently used by a special mode function are not checked and are always reported as being high.

5.4 IO-Warrior output behaviour

Upon receiving a SetReport request IO-Warrior writes the new data to the output pins. There is a small timing offset between the ports getting their new status, which is about 1/4 μs.

5.5 Using pins as inputs or outputs

All I/O pins on IO-Warrior can be used as input or output pins.
Basically all pins act as inputs all of the time. When receiving an input report from IO-Warrior you always get the current status on the pins.
Writing a 0 as output value to any pin causes it to drive the pin low with an open drain driver. Usually this will result in this pin being read as a zero input as well, unless so much current has to be drained by the pin that the voltage at it gets above the threshold level.
Writing a 1 to a pin causes the open drain driver to be turned off. The pin will be pulled high by an internal pull up resistor. Now the pin acts either as an output with a high level, or can be used as an input.

5.6 Power supply

USB does allow a device to be "Bus Powered". This means the device does get its power off the USB port. To avoid overloading on the USB ports devices need to advertise their power requirements.
There are two power classes for devices: Low power and high power. Low power devices may draw up to 100 mA off the USB, high power devices up to 500 mA.
Likewise there are high power and low power ports. Usually high power ports are those on the motherboard and on hubs with external power supply or hubs in a monitor. Low power ports are typically on hubs that get their power off the USB, like hubs in keyboards.
If the system decides that there is not sufficient power to supply a high power decive that device does not get enabled.
IO-Warrior can operate either as a high power or low power device. The power setting is determined

by checking the status of the Power pin on powering up the IO-Warrior.

5.7 Suspend

All devices on the USB port need to support the suspended state. When the host computer stops to periodically access the USB, like when it goes to sleep, all devices need to enter the suspended state and drop their power draw to less than 500 μA for low power devices or less than 2.5 mA for high power devices.
When entering suspended state IO-Warrior pulls all pins high. Care must be taken in designing external circuits so that they will draw no more than the allowed suspend power rating while all pins of IO-Warrior are high.

5.8 Remote Wakeup

IO-Warrior100 does not support remote wakeup.

5.9 Special mode I/O

To enable IO-Warrior to talk to devices that have more complex demands we have added the special mode functions. When any of these functions is enabled a couple pins of IO-Warrior turn into special function pins.
Talking to the special mode functions is handeled via the USB interface 1, which is also configured as generic HID compliant.
Commands to the functions are sent via endpoint 2 to interface 1. Replies from the functions are returned as interrupt in reports via endpoint 2.
To talk to these functions and to handle different requests to them ReportIDs are used which enable multiple functions to use the same endpoint. All reports to and from special mode interfaces are always eight bytes long, including the ReportID.
The following chapters describe the individual special mode functions.

5.9.1 Fast IIC special mode function

The fast IIC function of IO-Warrior100 offers high performance, close to theoretical maximum transfer rates. It uses a separate interface, so the bandwith of the two endpoints is not shared with any other functions.
Fast mode and fast mode plus are supported, allowing data rates up to 1 MBit/s and a throughput of > 60.000 bytes/s.
The native IIC function is enabled and disabled by sending a report with the following structure with ReportID=1 to interface 2:

ReportID	1	2	3	4	…	62	63
$01 out	enable	flags	$00	$00	$00	$00	$00

enable=$01 enables the IIC function, $00 disables it. Other values are reserved for future use.
Upon enabling IIC the SDA and SCL pins are pulled high and are no longer under control of SetReport requests to interface 0. Disabling IIC does return the pins under control of interface 0 and pulls them high initially.
flags selects the speed of the IIC interface and sets the drivers on the SCL/SDA pins accordingly.
The values for flags are as follows:
0 - 100 kbit/s
1 - 400 kbit/s
2 - 50 kbit/s
3 - 10 kbit/s
4 - 1000 kbit/s

A write request to the IIC is sent with ReportID=2 and has the following format:

ReportID	1	2	3	4	…	62	63
$02 out	flags	data	data	data	data	data	data

flags contains the following bits:
7 - Generate Start
6 - Generate Stop
5 - data count MSB
4 - data count
3 - data count
2 - data count
1 - data count
0 - data count LSB

If bit 7 - "Generate Start" is set a start signal (SDA falling edge while SCL is high) is generated on the IIC prior to sending out the first data byte.
Bit 6 - "Generate Stop" causes a stop signal (SDA goes high while SCL is high) to be generated after sending the last valid data byte of this report.
"data count" gives the number of valid data bytes in the report. The number may range from one to 62, other values cause an error to be returned.
To do write transactions that are longer than 62 bytes, send the first report with just the "Generate Start" bit set, then send additional reports with neither bit 6 or 7 set until the report with the last bytes is send which has the "Generate Stop" bit set.

Any write transactions are acknowledged by a report via interrupt-in endpoint 3:

ReportID	1	2	3	4	…	62	63
$02 in	flags	code	$00	$00	$00	$00	$00

flags contains the following bits:
7 - Error bit, 1 = error
6 - Arb Loss, 1 = lost arbitration
5 - data count MSB
4 - data count
3 - data count
2 - data count
1 - data count
0 - data count LSB

"data count" indicates the last byte that was successfully transfered and acknowledged by the slave (if any). An error is indicated when the slave does not acknowledge a transfer.
"code" contains an additional error code to indicate what happend. The following values are defined:
0 - no special code

1 - wrong number of bytes requested
2 - transaction without Start requested
3 - NACK received
4 - Bus Error

Reading data off the IIC is initiated with a ReportID=3. The initiating report has the following format:

ReportID	1	2	3	4	…	62	63
$03 out	count	addr	Mcount	$00	$00	$00	$00

"addr" holds the slave address to be send to the IIC.
"count" is the number of bytes that should be read off the IIC after sending the command byte, "Mcount" is the most significant byte for the count. The maximum number of bytes that may be requested at once is 65535.
A start signal is automatically generated before sending the command byte and a stop is generated after the last data byte is received.
Data is returned in input reports with ReportID=3 (which is different from the output ReportID=3 used to initiate the read transaction) via endpoint 3. If more than 62 bytes are requested the data is returned in multiple reports.

ReportID	1	2	3	4	…	62	63
$03 in	flags	data	data	data	data	data	data

flags contains the following bits:
7 - error, set if slave does not ack command byte
6 - unused, zero
5 - data count MSB
4 - data count
3 - data count
2 - data count
1 - data count
0 - data count LSB

Should the IIC slave fail to acknowledge the command byte the error flag will be set and the transaction aborted. IIC does not have an error condition during the actual reading of data after the command byte was sent.

5.9.2 LCD special mode function

The LCD special mode function supports display modules that are compatible with the HD44780 controller and several graphic display controllers that use a compatible interface. This controller is made by Hitachi and has set the de-facto standard for alphanumeric LCD modules. There are also OLED and vacuum fluorescence displays that use this type of interface.
The display modules come in various configurations with up to 80 characters total in any kind of arrangement from single line to four lines. Displays with more than 80 characters typically use more than one HD44780. IO-Warrior100 does not directly support modules with more than a single HD44780, some additional hardware may be required for this.
It is recommended to read the HD44780 manual for using the LCD function. Also have a look at our application note AN5: Driving Display Modules with IO-Warrior.

The LCD function is enabled by sending an output report with ID 4 to the USB interface 1:

ReportID	1	2	3	4	…	62	63
$04 out	enable	$00	$00	$00	$00	$00	$00

enable = $00 disables the LCD function. enable = $01 enables the LCD function, other values are reserved.
Upon enabling the LCD function the pins are put under control of the LCD function and can no longer be controlled by interface 0.
The /On pin is pulled low when the LCD function is enabled, it will go high when the IO-Warrior enters suspend state.
To write data to the connected LCD module an output report with ReportID=5 is written with the following format:

ReportID	1	2	3	4	…	62	63
$05 out	flags	data	data	data	data	data	data

"flags" contains the following bits:
7 - RS, Register Select bit
6 - unused, zero
5 - data count MSB
4 - data count
3 - data count
2 - data count
1 - data count
0 - data count LSB

The status of the RS bit is used to set the RS line to the LCD module. This allows access to the Instruction register (RS=0) or Data Register (RS= 1) of the LCD module.
With "data count" the number of bytes to be written is specified. IO-Warrior will write up to 62 data bytes to the register specified by the RS bit. The Busy bit of the LCD module is automatically checked and data written only when the LCD module is ready to accept it. There is a 16 ms timeout when waiting for the Busy bit, if this time has passed the transaction is cancled, no special feedback is provided.

To read data from the LCD module an output report with ReportID=6 is sent to interface 1:

ReportID	1	2	3	4	…	62	63
$06 out	flags	$00	$00	$00	$00	$00	$00

"flags" contains the following bits:
7 - RS, Register Select bit
6 - unused, zero
5 - data count MSB
4 - data count
3 - data count
2 - data count
1 - data count
0 - data count LSB

"RS" specifies which register is to be accessed. Data count sets the number of bytes to be read off the LCD (will be ignored if RS=0, only a single read will be done then).
Up to 62 bytes can be read with one request. The data read from the LCD module is returned in input reports with ID 6:

ReportID	1	2	3	4	…	62	63
$06 in	count	data	data	data	data	data	data

"count" specifies the number of bytes returned in this report.

5.9.3 Driving a LED matrix

IO-Warrior100 has the capability to drive a matrix of up to 8x64 LEDs with the aid of a few simple external driver chips.
To enable the LED matrix function a report with ID $14 is sent to interface 1:

ReportID	1	2	3	4	...	62	63
$14 out	enable	$00	$00	$00	$00	$00	$00

enable = $01 enables the LED function, enable = $00 disables it again.

Data to be displayed in the matrix is written in two blocks of 32 bytes:

ReportID	1	2	...	33	34	...	63
$15 out	block	data0	datan	data31	$00	$00	$00

"block" = 0 writes to the first four lines, "block" = 1 to the second four lines.

5.9.4 Switch Matrix Mode

IO-Warrior100 can handle a matrix of 8x8 or 16x8 switches or keys. Other than with a keyboard controller it is possible to read out all possible combinations of closed switches. Since the switch status is reported as a bitmap there is no limitation to the number of switches that can be closed at the same time (diodes are required in the matrix though if more than two keys can be pressed at the same time).

To enable the switch matrix function a report with ID $18 is sent to interface 1:

ReportID	1	2	3	4	…	62	63
$18 out	enable	$00	$00	$00	$00	$00	$00

enable = $00 disables the function
enable = $01 enables the 8x8 switch matrix
enable = $02 enables the 16x8 switch matrix

The status of the matrix is returned when ever there is a change of status or it can be requested immediately by sending a report with ID $19:

ReportID	1	2	3	4	…	62	63
$19 out	$00	$00	$00	$00	$00	$00	$00

The status of the matrix is returned in an input report with ID $19. A set bit denotes a closed switch:

ReportID	1	2	3	…	15	16	…	63
$19 in	Y0	Y1	Y2	Yn	Y15	Y16	$00	$00

The first 8 bytes contain the status for 8x8 matrix size.

5.9.5 ADC special mode

IO-Warrior100 has an internal 12 bit ADC with up to 4 input channels, a calibrated band gap reference, and a temperature sensor.
The ADC function has its own interface and endpoints to allow a high sampling rate without interfering with the other functions. Controlling the ADC and receiving data is done via interface 3.
Enabling and disabling the ADC is done by sending a report with ReportID=$1C to interface 3:

ReportID	1	2	3	4	5	6	…	63
$1C out	en	chan	$00	$00	mod	samp	$00	$00

"en" = $01 enables the ADC function
"en" = $00 disables it again.
"chan" is the number of channels to use, i.e. chan = 1 enables just AD0, chan = 2 enables AD0 and AD1, chan = 4 enables AD0…3, chan = 0 enables the temperature sensor only. All other values are invalid.
"mod" selects between single shot and continuos mode, 0 = single shot, 1 = continuous
"samp" specifies the sample rate:
0 - 1 kHz
1 - 2 kHz
2 - 3 kHz
3 - 4 kHz
4 - 6 kHz
5 - 8 kHz
6 - 10 kHz
7 - 12 kHz
8 - 15 kHz
9 - 16 kHz
10 - 20 kHz
11 - 24 kHz
13 - 30 kHz

For four channels the sample rate is limited to a maximum of 6000 Hz, for two channels at 15000 Hz, and for one channel at 30000 Hz. The 15000Hz for two channels and 30000 Hz for a single channel may not work on all computer setups as they generate the maximum number of reports allowed for this type of USB data transfer.

In continuous mode IO-Warrior100 continuously samples the selected ADC channels and sends reports with ReportID=$1D via interface 3 with the data.
The first byte of a data report is a sequence number that is counted up for each report transmitted. This allows to detect if there is a packet lost.

The sampling rate depends on the number of channels selected:
1 channel - 30 kHz
2 channels - 15 kHz
4 channels - 6 kHz
Data is reported with the lower byte first. For multiple channels the individual channels are sampled with an offset of about 1 μs.
For a single channel each report contains 30 samples from AD0:

ReportID	1	2, 3	…	60, 61	62	63
$1D in	seq	data0	…	data29	$00	$00

If two channels are selected each report contains 15 pairs fo AD0 and AD1 data:

ReportID	1	2, 3	4,5	…	60, 61	62	63
$1D in	seq	AD0	AD1	…	AD1	$00	$00

For four channels each report contains 7 sets of data for the four channels:

ReportID	1	2, 3	4,5	6,7	8,9	…	54.55	…
$1D in	seq	AD0	AD1	AD2	AD3	…	AD3	0

ADC data is 12 bit unsigned. The ADC input range is GND to Vdda. Vdda serves as the high reference for the data conversion. If Vdda is known with sufficient precision it may be used to calculate the absolute voltage. Though if Vdda is not known or can vary, the actual voltage can be calculated using the internal bandgap reference.
IO-Warrior100 has a bandgap reference for which the ADC value at Vdda = 3.3 V ±10 mV is stored internally.
The calibration data as well as the temperature sensor data can be read in single shot mode. If continuous mode is enabled all single shot commands will be ignored.

ReportID	1	2	3	…	63
$1E out	command	$00	$00	$00	$00

"command" = 0 requests the calibration and temperature data.
"command" = 1 reads the analog inputs.
For command = 0 the following data is returned:

RepID	1	2,3	4,5	6,7	8,9	10,11	…
$1E in	0	VrefCal	TCal1	TCal2	Vref	Temp	$00

To calculate the actual input voltage of an ADC channel the calibration data is used with the following formula:
Vin = 3.3 V * VrefCal * ADn / Vref /4095

The chip temperature can be calculated as follows:
t = 80°C / (TempCal2 - TempCal1) * (Temp * VrefCal / Vref - TempCal1) + 30

Absolute precision of the temperature sensor is limited since the TempCal1 and TempCal2 values

are measured with just ±5°C accuracy. For more precise results the calibration may be done externally.
"command" = 1 returns the data from the enabled ADC channels:

RepID	1	2,3	4,5	6,7	8,9	…	63
$1E in	$01	AD0	AD1	AD2	AD3	$00	$00

Not active channels return zero. If no channel is selected there will be no reply report.

5.9.6 PWM outputs

IO-Warrior100 offers a PWM generator with four channels. This means it can generate up to four synchornized PWM signals with identical period and individual pulse length.
The PWM function is enabled and disabled with an out report with ReportID=$20:

ReportID	1	2	3	4	...	62	63
$20 out	enable	$00	$00	$00	$00	$00	$00

"enable" = 0 disables the PWM function and returns all pins back to normal mode. Values 1 to 4 enable the respective number of PWM channels.
The number of channels can only be changed by first disabling the PWM function and then enabling it with the new number of channels.
Setting the parameters is done with a ReportID= $21. The new values are used immediately.

ReportID	1	2/3	4/5	6/7	8/9	10/11	12/13
$21 out	0	Presc	Cycle	Ch0	Ch1	Ch2	Ch3

The values are 16 bits each, low byte first.
"Presc" is the prescaler for the 48 MHz master clock. The 48 MHz are divided by Presc+1 and then used to clock the period counter.
"Cycle" is the cycle length in clock cycles.
"Chx" defines the length of the active (low) output pulse in clock cycles. Chx = 0 produces a constant high (off), Chx > Period a constant low (on)

5.9.7 DAC outputs

IO-Warrior100 has two DAC outputs with 12 bit resolution. The voltage output ranges from Vss to Vdda and is linear to the output value.
Enabling and disabling the DAC function is done by sending ReportID=$24 to Interface 1:

ReportID	1	2	3	4	...	62	63
$24 out	enable	$00	$00	$00	$00	$00	$00

"enable" = 0 disables the DAC function and returns all pins back to normal mode. Values 1 and 2 enable the respective number of DAC channels.
The number of channels can only be changed by first disabling the DAC function and then enabling it with the new number of channels.

The output of the DACs is set by ReportID=$25:

ReportID	1	2/3	4/5	6	...	62	63
$25 out	0	DAC0	DAC1	$00	$00	$00	$00

The values are 16 bits each, low byte first.

5.9.8 Digital LED function

IO-Warrior100 has the capability to drive up to 500 digital LEDs of the type that uses a single digital data line, like the WS2812 or WS2812B.
To enable the digital LED function a report with ID $28 is sent to interface 1:

ReportID	1	2	3	4	5	…	63
$28 out	enable	0Cyc	0Pha	1Cyc	1Pha	$00	$00

"enable" = $01 enables the digital LED function, "enable" = $00 disables it again.
The LED matrix function can be disturbed by the digital LED function. Writing the data to the LEDs can interrupt the matrix update if many LEDs are connected.
ADC continuous mode should not be used at the same time as writing data out to the LEDs as this will disturb the data signal and cause flickering or wrong LED states. Make sure to disable the continuous mode of the ADC before sending the data to the LEDs. Writing data to the buffer is not affected.

To handle different variants of the digital LEDs IOW100 requires parameters for the timing of the serial data transfer.
The data signal is generated from a 48 MHz master clock, allowing a timing resolution of 20.83 ns steps. All timining parameters (X) will generate pulse or cycle times of t = (X+1)*20.83 ns.
"0Cyc" is the length of a bitcell signaling a zero bit.
"0Pha" is the length of the high phase for a zero bit.
"1Cyc" is the length of a bitcell signaling a one bit.
"1Pha" is the length of the high phase for a one bit.
Typical timings:

LED type	0Cyc	0Pha	1Cyc	1Pha
SK2812	$39	$0E	$39	$1C
LC8812	$2B	$0E	$48	$2B
WS2812B/SK2812	$3C	$13	$3C	$26
WS2812	$37	$10	$3E	$21
WS2813	$45	$10	$45	$34

Data to be send to the LEDs is transfered in up to 25 blocks containing data for 20 LEDs each. The data is buffered internally in the IOW100 and individual blocks can be overwritten to reduce USB traffic when updating.

ReportID	1	2, 3	4	…	63
$29 out	block	count	data 0	data n	data 59

"block" is the block number in the buffer to which the data is written, 0 to 24 are valid.
If the high bit of "block" is set the data is shipped out to the LEDs after the block has been copied to the buffer.
"count" is ignored until the high bit of "block" indicates that the data is to be transfered to the LEDs. Then it specifies the number of LEDs to write. 1 to 500 is the valid range, byte 2 is LSB.
The data is then shipped to the LEDs with the timing specified on enabling the function.

5.9.9 Reading special mode function status

To get the information which special modes are currently in use, send a report with ReportID=$FE to interface 1:

ReportID	1	2	3	4	...	62	63
$FE out	$00	$00	$00	$00	$00	$00	$00

This will immediately return an input report with ID $FE that contains flags for the special mode functions. Non zero means the function is active:

0 RepID = $FE
1 IIC
2 LCD
3 0
4 LED Matrix
5 Switch Matrix
6 ADC
7 PWM
8 DAC
9 WSLED

5.9.10 Getting current pin status

Due to the way Windows implements HID support IO-Warrior is unable to continuously send its status. HID class devices do have a function that allows the host to set the rate at which reports should be repeated if there is no change to the data. Windows does this rate to zero for IO-Warrior, which means IO-Warrior may send data only if there are changes.
To be able to get the current status from IO-Warrior it does support a special mode function that always returns the current status of all pins.

To get the port status just send a report with ID $FF to interface 1:

ReportID	1	2	3	4	5	…	63
$FF out	$00	$00	$00	$00	$00	$00	$00

This will result in the current pin status to be returned immediately in an input report with ID $FF with the following format:

ReportID	1	2	…	12	…	63
$FF in	Port0	Port1	Portn	Port11	$00	$00

6.0 Absolute maximum ratings

Storage Temperature ....-65°C to +150°C
Ambient Temperature with power applied..-40°C to +85°C
Supply voltage on Vdd, Vdda, Vddi relative to Vss ....-0.3 V to +4 V
Supply voltage on Vcc relative to Gnd (IOW28L) ....-0.3 V to +6.5 V
DC input voltage ....-0.3 V to +4 V
Maximum current into all ports..80 mA
Power Dissipation..max. 476 mW
Static discharge voltage..>2000 V
Latch-up current ....>200 mA

6.1 DC Characteristics

	Parameter	Min	Max	Units	Remarks
V_{dd}	Operating Voltage	2.0	3.6	V	typ. 3.3 V
I_{dd}	Operating Supply Current		30	mA	
I_{sb}	Suspend mode current		350	µA	internally active
I_{ol}	Sink current on port pins		25	mA	max. combined all pins 80 mA
V_{ol8}	Output low voltage		0.4	V	I = 8 mA
V_{oh8}	Output high voltage	V_{ddi}-0.4		V	I = 8 mA
V_{ol20}	Output low voltage		1.3	V	I =20 mA
V_{oh20}	Output high voltage	V_{ddi}-1.3		V	I =20 mA
R_{up}	Pull up/down resistors	25	55	kΩ	typ. 40 kΩ
V_{ith}	Input Threshold Voltage	0.7 x V_{ddi}		V	

6.2 ADC Characteristics

	Parameter	Min	Max	Units	Remarks
V_{dda}	Analog Operating Voltage	2.4	3.6	V	typ. 3.3 V
I_{cca}	Analog Operating Supply Current		1	mA	typ. < 0.9 mA
V_{ain}	Input Conversion Range	0	V_{dda}	V	
V_{ref}	Internal Reference Voltage	1.2	1.25	V	typ. 1.23 V
V_{refCal}	Accuracy of Calibration Value		± 10	mV	bandgap calibrated at 3.3 Vdda
$V_{refTemp}$	Reference Voltage Temperature Offset	-100	100	ppm	over full operating temperature
R_{ains}	Input Impedance, slow measurement		50	kΩ	for single shot mode
R_{ainf}	Input Impedance, fast measurement		5.9	kΩ	for continous mode
AD_{et}	ADC Total Unadjusted Error		±2	LSB	typ. ±1.3
AD_{eo}	ADC Offset Error		±1.5	LSB	typ. ±1
AD_{eg}	ADC Gain Error		±1.5	LSB	typ. ±0.5
AD_{et}	ADC Differential Linearity Error		±1	LSB	typ. ±0.7
AD_{el}	ADC Integral Linearity Error		±1.5	LSB	typ. ±0.8
T_{cal1}	Temperature Calibration Value 1	25	35	°C	tested value
T_{cal2}	Temperature Calibration Value 2	105	115	°C	tested value
T_{lin}	Temperature Sensor Linearity Error		±2	°C	typ. ±1°C

7.0 Ordering information

Partname	Order Code	Description	Package
IO-Warrior100	IOW100-LF100	IO-Warrior 82 I/O	LQFP100
IO-Warrior100-MOD	IOW100-MOD	IO-Warrior 82 I/O, PCB module	Module

The chips listed here are standard products. Customized chips are available on request.

7.1 Packaging info

LQFP100 chips come in trays - TBD
Modules come in antistatic bags, packed individually or bulk.

7.2 USB VendorID and ProductID

By default all IO-Warrior chips are shipped with the USB VendorID of Code Mercenaries ($7C0 or decimal 1984) and a fixed ProductID.
On request chips can be equipped with the customers VendorID and ProductID. VendorIDs can be obtained from the USB Implementers Forum www.usb.org
Customized chips may be subject to minimum order quantities, contact sales@codemercs.com for details.

Following are the ProductIDs for the IO-Warrior controllers:
IO-Warrior100 $1506

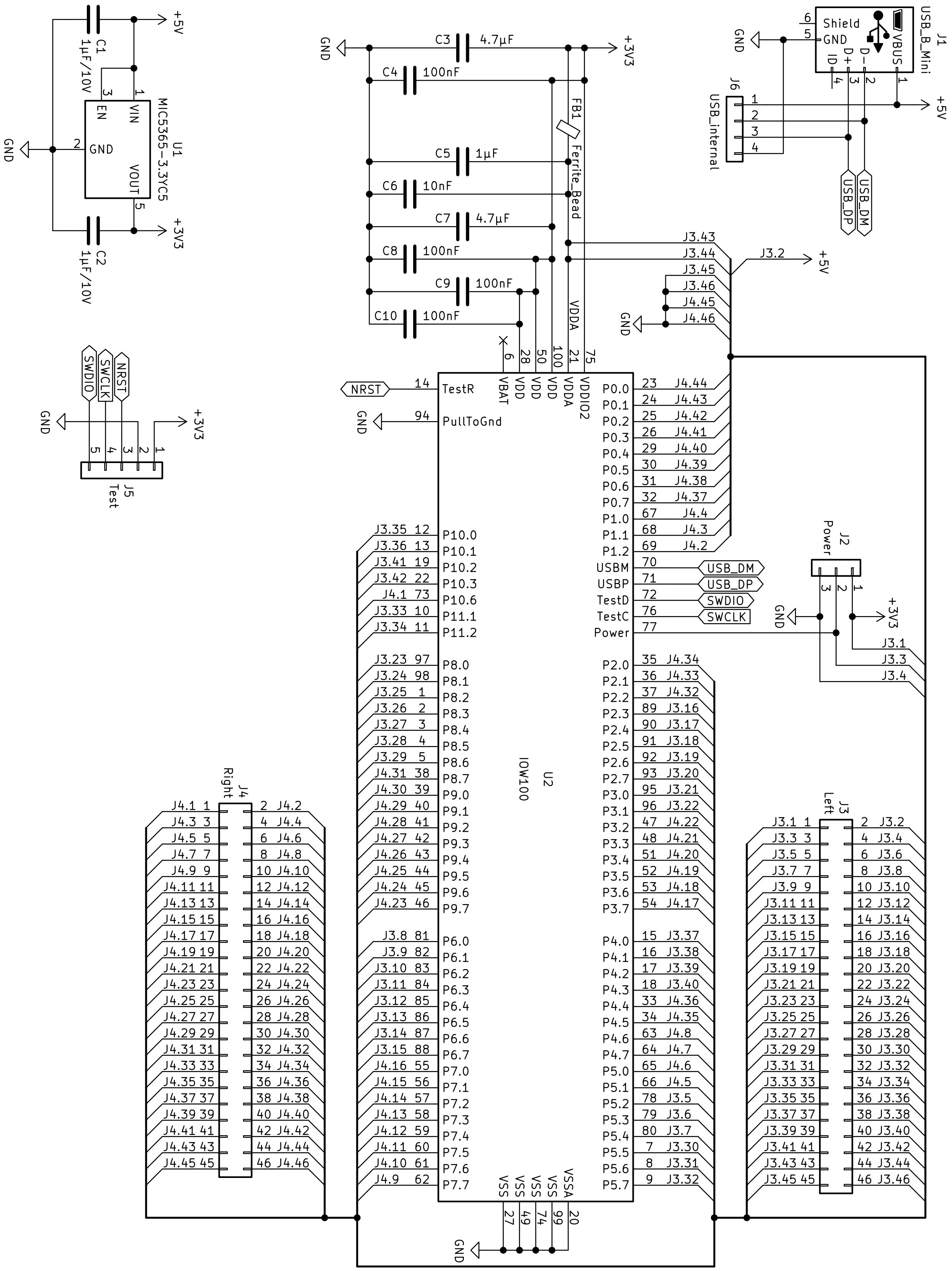
ProductIDs are independent of the package type.

7.3 Serial numbers

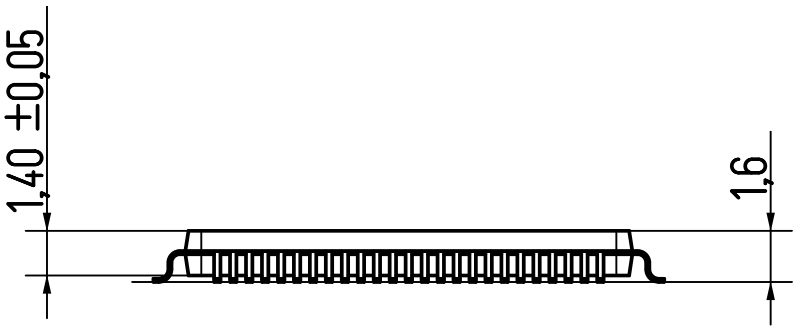
All IO-Warrior100 chips have unique serial numbers in their device descriptors. These serial numbers can be used to simplify programming for multiple IO-Warriors connected to a single computer.

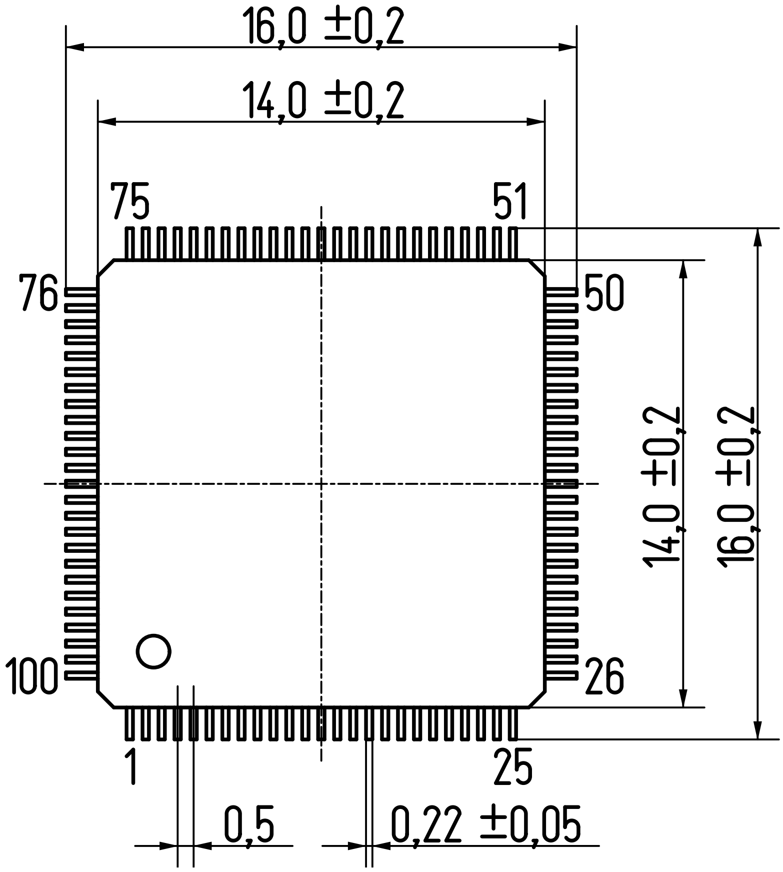
The serial numbers are factory programmed and can not be changed. Serial numbers are 8 digit hexadecimal numbers. No two chips of a type will be produced with identical serial numbers.
It is not possible to order chips with a specific serial number unless they are ordered as custom chips which are subject to minimum order volumes and setup charges.

8.0 IO-Warrior100 basic circuit (circuit of IOW100-MOD)



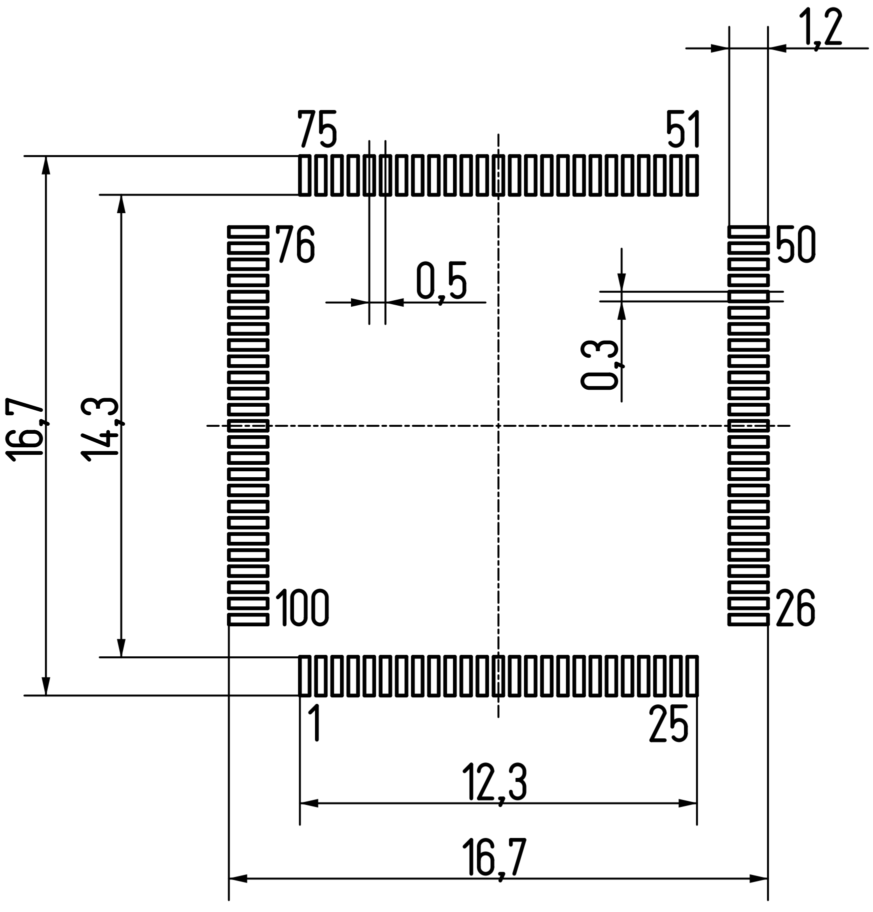
9.0 Package dimensions
100 Pin LQFP - 16 x 16 mm outer contour 0.5 mm pitch





Package thickness: 1.4 mm ±0.05 mm
All dimensions: mm

9.1 Recommended footprint



All dimensions: mm

10.0 ESD considerations

IO-Warrior has an internal ESD protection to withstand discharges of more than 2000 V without permanent damage. However ESD may disrupt normal operation of the chip and cause it to exhibit erratic behaviour.
For the typical office environment the 2000 V protection is normally sufficient. Though for industrial use additional measures may be necessary.
When adding ESD protection to the signals special care must be taken on the USB signal lines. The USB has very low tolerance for additional resistance or capacitance introduced on the USB differential signals.

10.1 EMC considerations

IO-Warrior uses relatively low power levels and so it causes few EMC problems.
To avoid any EMC problems the following rules should followed:

- Put the 100 nF ceramic capacitors right next to the power supply pins of the chip and make sure the PCB traces between the chips power pins and the capacitor are as short as possible.
- Run the power supply lines first to the capacitor, then to the chip.
- Make the matrix lines only as long as absolutly necessary.
- Keep the two USB signal lines close to each other, route no other signal between them. USB uses differential signalling so the best signal quality with lowest RF emission is achieved by putting these lines very close to each other.

11.0 Revision history

Current shipping version of IO-Warrior100 is V1.0.1.8, the initial release version of IOW100 is V1.0.1.8.

11.1 Document revision history

V1.0.0.1 - Fixed PID information in chapter 7.2. A wrong ProductID was specified there
V1.0.0.0 - initial release version

12. RoHS compatibility

IO-Warrior100 conforms to the requirements that are necessary to use it in a RoHS compliant device.

Legal stuff

The information contained herein is subject to change without notice. Code Mercenaries makes no claims as to the completeness or correctness of the information contained in this document.

Code Mercenaries assumes no responsibility for the use of any circuitry other than circuitry embodied in a Code Mercenaries product. Nor does it convey or imply any license under patent or other rights.

Code Mercenaries products may not be used in any medical apparatus or other technical products that are critical for the functioning of lifesaving or supporting systems. We define these systems as such that in the case of failure may lead to the death or injury of a person. Incorporation in such a system requires the explicit written permission of the CEO of Code Mercenaries.

Trademarks used in this document are properties of their respective owners.

Code Mercenaries
Hard- und Software GmbH
Karl-Marx-Str. 147a
12529 Schönefeld OT Grossziethen
Germany
Tel: +49-3379-20509-20
Mail: support@codemercs.com
Web: www.codemercs.com

HRB 9868 CB
Geschäftsführer: Guido Körber, Christian Lucht