Generic UnivIO Device Specification

First Published:	2021-11-24
Publish site:	github.com/nvitya/univio
Original Author:	Viktor Nagy
License Type	MIT (Open Source)
Actual Version:	1.0, 2021-11-24

Generic IO Device Addresses

Configuration Objects

The pin configuration is writable only in CONFIG mode (OBJ#0010 = 0) and must be saved when changed to RUN mode.

Addr. (hex)	Type	R/W	Function
Device Information			
0100	u32	R	Configurable pin count
0101	u32	R	Maximal SPI Transaction Size, 0 = SPI is not available (buffer sizes at MEM#C000 and MEM#E000)
Configuration			
0200	u32	RW	PIN_0 configuration: bit0..7: PINTYPE, se table below bit8..15: n = target number (e.g. DIG_OUT_n) bit16..31: PINFLAGS, dependent on PINTYPE
0200 + n	u32	RW	PIN_n configuration
0700	u32	RW	PWM_0 Configuration The 32-bit value = PWM Base Frequency in Hz. Default Value = 1000 (1 kHz) This can be written in RUN mode too.
0700 +n	u32	RW	PWM_n Configuration

PINTYPE Values

PINTYPE	Description
0	Passive (default value), usually input with weak pull-up, which is the MCU default state for the unconfigured pins. The pin is not controllable, its state can not be read.

1	Digital Input (aka. DIG_IN or DIN), weak pull-up by default The digital input pin states can be read at OBJ#1100. PINFLAGS: 0x0001: weak pull-down 0x0002: floating (no pull-up or pull-down)
2	Digital Output (aka. DIG_OUT or DOUT), push-pull by default The digital input pin state can be set with OBJ#1000 and OBJ#1010 PINFLAGS: 0x0001: inverted 0x0002: open-drain
3	Analogue Input (aka. ANA_IN or AIN) The 16-bit (left-shifted) value can be read at OBJ#1200 or MEM#8000 16-bit value, left aligned if the device resolution is less. For example the device fills bit4..15 with the 12-bit ADC value, bit0..3 = 0
4	Analogue Output (aka. ANA_OUT or AOUT) The 16-bit value can be written at OBJ#1300 or MEM#8100
5	PWM Output (aka. PWM_OUT or PWM) The 16-bit Duty Cycle value can be written at OBJ#1400 or MEM#8200 PINFLAGS: 0x0001: invert output
6	LED Blink Pattern Output (aka. LEDBLP) The 32 bit blink pattern code controlled by OBJ#1500 PINFLAGS: 0x0001: inverted
7	SPI Pin The function should be predefined by the device configuration

IO Control

Addr. (hex)	**Type**	**R/W**	**Function**
Objects			
1000	u32	W	DIG_OUT 0-15 control: bit0..15: set GPOUTx bit16..31: clear GPOUTx
1001	u32	W	DIG_OUT_16..32 control
1010	u32	RW	DIG_OUT_0..31 values
1100	u32	R	DIG_IN 0-31 values
1200	u16	R	ANA_IN_0 value 16-bit value, 65536 = 100 % of measurement range. Usually the device has only 12 bit ADC, so the bits4..15 = 12-bit ADC value, bit0..3 = 0.

1200 + n	u16	R	ANA_IN_n value
1300	u16	RW	ANA_OUT_0 value 16-bit value, 65535 is always the maximal value. The device converts intern for the reduced native resolution by right-shifting
1300 + n	u16	RW	ANA_OUT_n value
1400	u16	RW	PWM_0 Duty Cycle The 16-bit value sets the PWM Duty Cycle: 65535 = 100 % 0 = 0 % The output (when not inverted) starts with high level and stays high during the duty cycle, then low level until period end. The period frequency are set at OBJ#0700 (in Hz), the default is 1 kHz.
1400 +n	u16	RW	PWM_n Duty Cycle
1500	u32	RW	LEDBLP_0 Blink Pattern The 32-bit output value controls the blinking pattern. Bit period: 1/16 s Every bit in the output control controls the high or low level for this period of time. The timing might be not as precise as by the PWM outputs.
1600	u32	RW	SPI Speed, bits / s in Hz
1601	u16	RW	SPI Transaction Length The value must be less or equal as OBJ#0101 (SPI buffer length)
1602	u8	RW	SPI Transaction Status and Control 0 = Idle, transaction finished 1 = Transaction Running / Start transaction
1603	u16	R	SPI Transaction Remaining (optional) Remaining bytes of the running transaction
Memory Range			
8000		R	ANA_IN_[0..n]: 16 Bit Analogue input values 8000..8001: ANA_IN_0 = OBJ#1200 8002..8003: ANA_IN_1 = OBJ#1201 ...
8100		RW	ANA_OUT_[0..n]: 16-bit Analogue Output Values 8100..8101: ANA_IN_0 = OBJ#1300 8102..8103: ANA_IN_1 = OBJ#1301 ...
8200		RW	PWM_OUT_[0..n]: 16-bit PWM Duty Cycles 8200..8201: PWM_OUT_0 = OBJ#1400 8202..8203: PWM_OUT_1 = OBJ#1401 ...

C000		RW	SPI Tx Buffer (8 kByte) The maximal size can be
E000		RW	SPI Rx Buffer (8 kByte)