eSi-GPIO

1 Contents

2 Overview

The eSi-GPIO core provides an interface to general purpose I/O ports. It supports the following features:

- Configurable number of I/Os.
- Programmable direction.
- Programmable interrupt generation on I/O events.
- Programmable pullups and pulldowns (technology dependant).
- Output set and clear without read-modify-write.
- AMBA 3 APB slave interface.

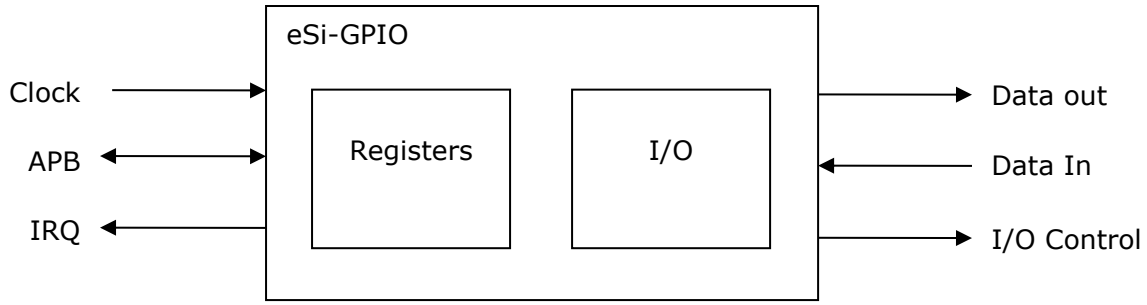

Figure 1: eSi-GPIO

3 Hardware Interface

Module Name	cpu_apb_gpio
HDL	Verilog
Technology	Generic
Source Files	cpu_apb_gpio.v

Port	Type	Description
pins	Integer	Specifies the number of GPIO pins
data_output_reset	Integer	Specifies the reset value for the data output register
data_direction_reset	Integer	Specifies the reset value for the data direction register
pull_up_reset	Integer	Specifies the reset value for the pullup register
pull_down_reset	Integer	Specifies the reset value for the pulldown register

Table 1: Parameters

Port	Direction	Width	Description
clk	Input	1	Clock used for event detection. This clock must be enabled when cactive is asserted.
pclk	Input	1	APB clock
presetn	Input	1	APB reset, active-low
paddr	Input	8	APB address
psel	Input	1	APB slave select
penable	Input	1	APB enable
pwrite	Input	1	APB write
pwdata	Input	BITS	APB write data
data_in	Input	pins	Input data
cactive	Output	1	Clock active
pready	Output	1	APB ready
prdata	Output	BITS	APB read data
pslverr	Output	1	APB slave error
data_out	Output	pins	Transmit data
data_out_enable	Output	pins	Ready to send, active-low
pull_up	Output	pins	Pull-up resistor enable
pull_down	Output	pins	Pull-down resistor enable
interrupt_n	Output	1	Interrupt request, active-low

Table 2: I/O Ports

For complete details of the APB signals, please refer to the AMBA 3 APB Protocol v1.0 Specification available at http://www.arm.com/products/solutions/AMBAHomePage.html

The GPIO does not include internal synchronizing flip-flops. These should be implemented externally for the data_in ports if the transmitting clock domain is asynchronous to clk.

4 Software Interface

4.1 Register Map

Register	Address offset	Access	Description
data_out	0x00	R/W	Data output register
data_in	0x04	R	Data input register
data_direction	0x08	R/W	Data direction register
interrupt_enable	0x0c	R/W	Interrupt enable
sense	0x10	R/W	Event sense
trigger	0x14	R/W	Event trigger
pending	0x18	R/W	Interrupt pending
ack	0x1c	W	Interrupt acknowledge
pullup	0x20	R/W	Pullup resistor enable
pulldown	0x24	R/W	Pulldown resistor enable
control	0x28	R/W	Control register
data_out_set	0x2c	W	Set data output register
data_out_clear	0x30	W	Clear data output register
edge	0x34	R/W	Event edge

Table 3: Register Map

4.1.1 Data Output Register

The data output register holds the value that is to be output on the port `data_out`. The data output register is reset to the value of the `data_output_reset` parameter.

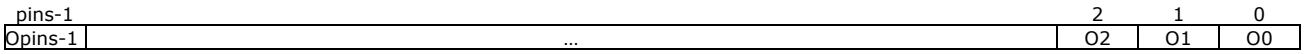

Figure 2: Format of the `data_out` register

Register	Values	Description
On	0 - Low 1 - High	Value to output `data_out[n]`

Table 4: Fields of the `data_output` register

4.1.2 Data Input Register

The data input register holds the current value on the `data_in` port.

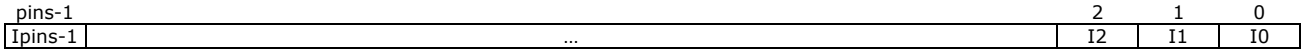

Figure 3: Format of the `data_in` register

4.1.3 Data Direction Register

The data direction register holds a direction flag for GPIO port, that drives the `data_out_enable` port. The data direction register is reset to the value of the `data_direction_reset` parameter.

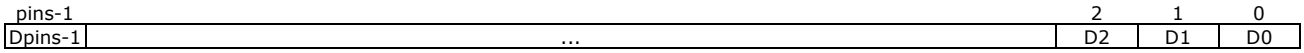

Figure 4: Format of the `data_direction` register

Register	Values	Description
Dn	0 - Input 1 - Output	Direction

Table 5: Fields of the `data_direction` register

4.1.4 Interrupt Enable Register

The interrupt enable register holds an interrupt enable flag for GPIO port. The interrupt enable register is reset to 0.

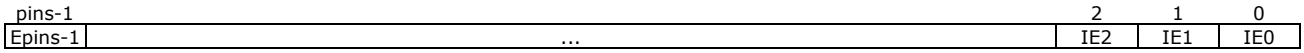

Figure 5: Format of the `interrupt_enable` register

Register	Values	Description
IEn	0 - Disabled 1 - Enabled	Interrupt enable

Table 6: Fields of the `interrupt_enable` register

4.1.5 Sense Register

The sense holds a sense flag for each GPIO port. The sense flag determines whether interrupts on the corresponding point are sensitive to the positive or negative edge or level.

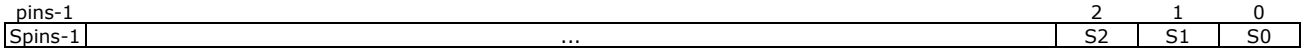

Figure 6: Format of the `sense` register

Register	Values	Description
Sn	0 - Negative 1 - Positive	Sense

Table 7: Fields of the `sense` register

4.1.6 Trigger Register

The trigger holds a trigger flag for each GPIO port. The trigger flag determines whether interrupts on the corresponding port are level-sensitive or edge-sensitive.

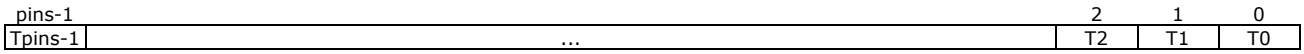

Figure 7: Format of the `trigger` register

Register	Values	Description
Tn	0 - Level	Trigger

	1 - Edge	

Table 8: Fields of the `trigger` register

4.1.7 Interrupt Pending Register

The interrupt pending register holds an interrupt pending flag for each GPIO port.

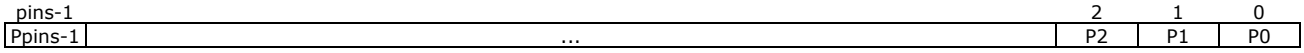

Figure 8: Format of the `pending` register

Register	Values	Description
Pn	0 - No interrupt 1 - Interrupt pending	Indicates if an interrupt is pending

Table 9: Fields of the `pending` register

4.1.8 Interrupt Acknowledge Register

The interrupt acknowledge register is a write only register, that when written with a 1, will clear the corresponding bit in the interrupt pending register.

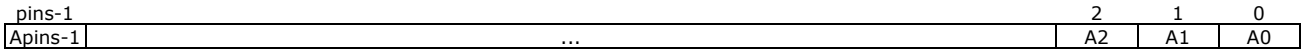

Figure 9: Format of the `ack` register

Register	Values	Description
An	0 - No ack 1 - Ack interrupt	Acknowledge corresponding interrupt

Table 10: Fields of the `ack` register

4.1.9 Pullup Register

The `pullup` register determines whether a pull-up resistor is enabled for each GPIO port. The `pullup` register directly drives the `pull_up` port. When a bit is set in the `pullup` register, the corresponding bit is cleared in the `pulldown` register. The pullup register is reset to the value of the `pull_up_reset` parameter.

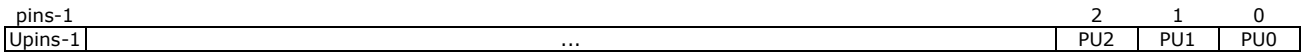

Figure 10: Format of the `pullup` register

Register	Values	Description
PUn	0 - No pull-up 1 - Pull-up	Enable pull-up

Table 11: Fields of the `pullup` register

4.1.10 Pulldown Register

The `pulldown` register determines whether a pull-down resistor is enabled for each GPIO port. The `pulldown` register directly drives the `pull_down` port. When a bit is set in the `pulldown` register, the corresponding bit is cleared in the `pullup` register. The pulldown register is reset to the value of the `pull_down_reset` parameter.

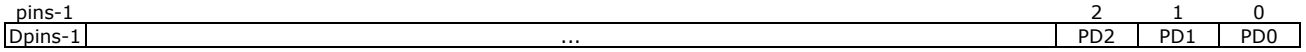

Figure 11: Format of the `pulldown` register

Register	Values	Description
PDn	0 - No pull-down 1 - Pull-down	Enable pull-down

Table 12: Fields of the `pulldown` register

4.1.11 Control Register

The `control` register has an enable flag that can be used to assert the `cactive` output. Without `EN` being set, `cactive` will only be asserted when one of the interrupt enable bits is set. In some SoCs, `cactive` will be used to gate the clock to external synchronizers, so setting this flag may be necessary in order to read data if no interrupt enables are set. The control register is reset to 0.

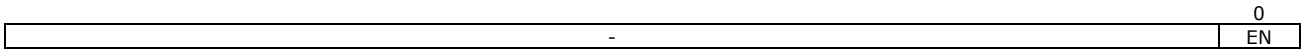

Figure 12: Format of the `control` register

Register	Values	Description
EN	0 – Disabled 1 – Enabled	Enable `cactive` output.

Table 13: Fields of the `control` register

4.1.12 Data Output Set Register

The data output set register, is a write-only register, that when written with a 1, causes the corresponding bits in the `data_out` register to be set (Set to 1). Bits written as 0 have no effect. This allows bits to be set in the `data_out` register without the need to perform a read-modify-write sequence.

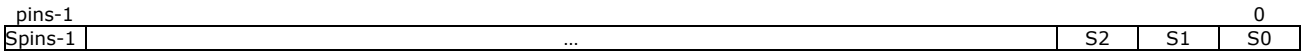

Figure 13: Format of the `data_out_set` register

Register	Values	Description
Sn	0 – No effect 1 – Set bit	Bits to set in `data_out` register

Table 14: Fields of the `data_out_set` register

4.1.13 Data Output Clear Register

The data output clear register is a write-only register, that when written with a 1, causes the corresponding bits in the `data_out` register to be cleared (Set to 0). Bits written as 0 have no effect. This allows bits to be cleared in the `data_out` register without the need to perform a read-modify-write sequence.

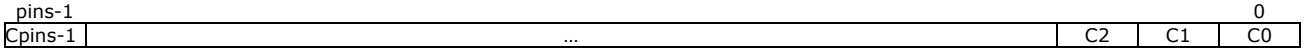

Figure 14: Format of the `data_out_clear` register

Register	Values	Description
Cn	0 – No effect 1 – Clear bit	Bits to clear in `data_out` register

Table 15: Fields of the `data_out_clear` register

4.1.14 Edge Register

The edge register holds an edge flag for each GPIO port. The edge flag determines whether edge-sensitive interrupts on the corresponding port occur on a single edge, as specified by the sense register, or either edge.

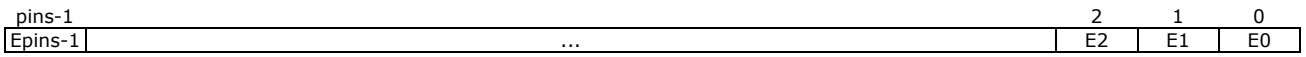

Figure 15: Format of the `edge` register

Register	Values	Description
En	0 – Single edge 1 – Either edge	Edge

Table 16: Fields of the `edge` register

4.2 Interrupts

The GPIO is able to raise an interrupt when an event is detected on any of the GPIO input ports. The event that is to be detected is set by the `sense`, `trigger` and `edge` registers. These registers allow for detecting a low-level, high-level, rising-edge, falling-edge or either-edge.

5 Revision History

Hardware Revision	Software Release	Description
1	1.0.0	Initial release
2	2.9.3	Add `control` register
3	3.2.14	Add `data_out_set` register Add `data_out_clear` register Writing a bit in `pullup` or `pulldown` clears the corresponding bit
4	3.3.10	Added reset parameters
5	7.0.7	Added `edge` register

Table 17: Revision History

