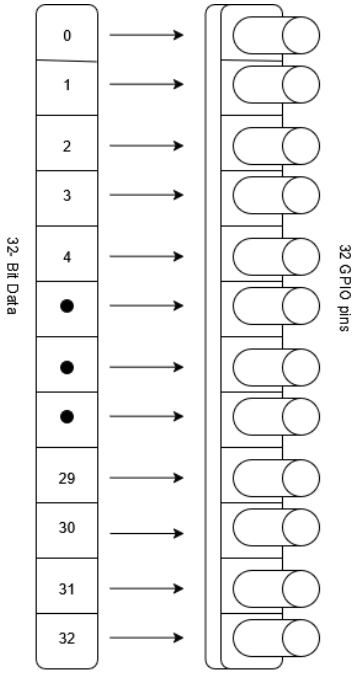
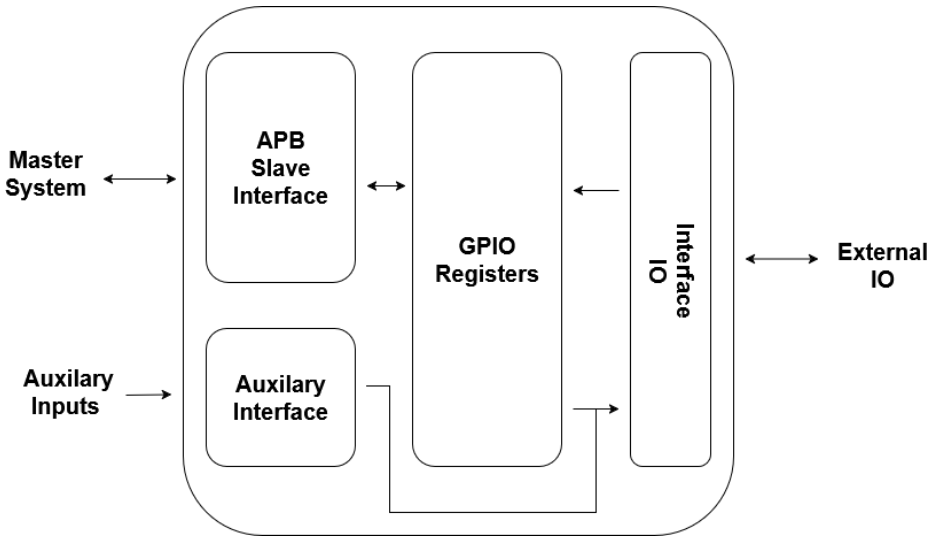
GPIO Core Documentation

By Gauthamaan M K

GPIO core:

The GPIO core is programmable interface between the processor and the external general input/output pins in an embedded system. The GPIO core allows the processor to retrieve data from the external GPIO pins and send data through the external pins. Each single bit of a data is mapped to an individual GPIO pin making possible to simultaneously control the 32 pins as inputs and outputs based on the instructions received from the processor.



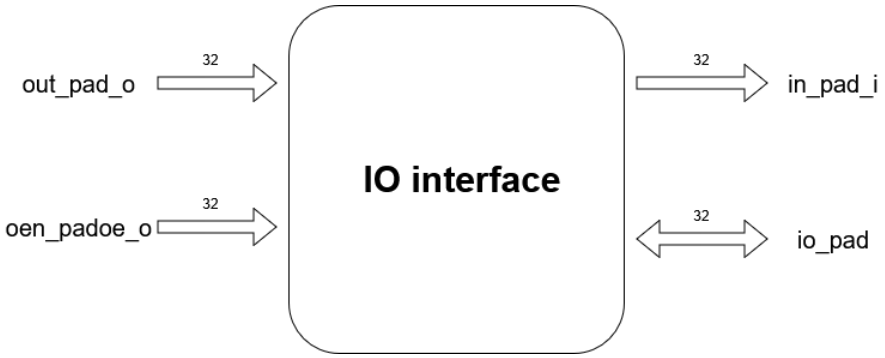


Sub modules present GPIO core:

- IO Interface
- Auxiliary Interface
- APB interface
- GPIO registers

IO Interface:

The IO interface acts as an intermediate layer between the GPIO core and the external GPIO pins. The GPIO pins are connected to the IO interface through a tri-state buffer that sends and receives the data. When the pin does not send or receive the data, the pin stays in high impedance, disconnected from the circuit.



Signals	Type	Function
out_pad_o	Input	32-bit data that is sent to GPIO pin
oen_padoe_o	Input	32-bit data that decides which pin sends and receives the data
io_pad_i	Output	32-bit data that is received from GPIO pin
Io_pad	Input/Output	Bi-directional physical wire to connect the core and GPIO pins

Working Principle:

- oen_padoe_o is used to determine the direction of transmission of data.
- When oen_padoe_o[i] = 1, io_pad[i] acts as an output and sends data from out_pad_o[i] to the GPIO pin.
- When oen_padoe_o[i] = 0, io_pad[i] acts as input and data is received by in_pad_i[i] from the GPIO pin.

Auxiliary interface:

The auxiliary interface syncs the external inputs to the system clock of the GPIO core. The external data is synchronized and the interface acts as a buffer for other modules to access the external data.

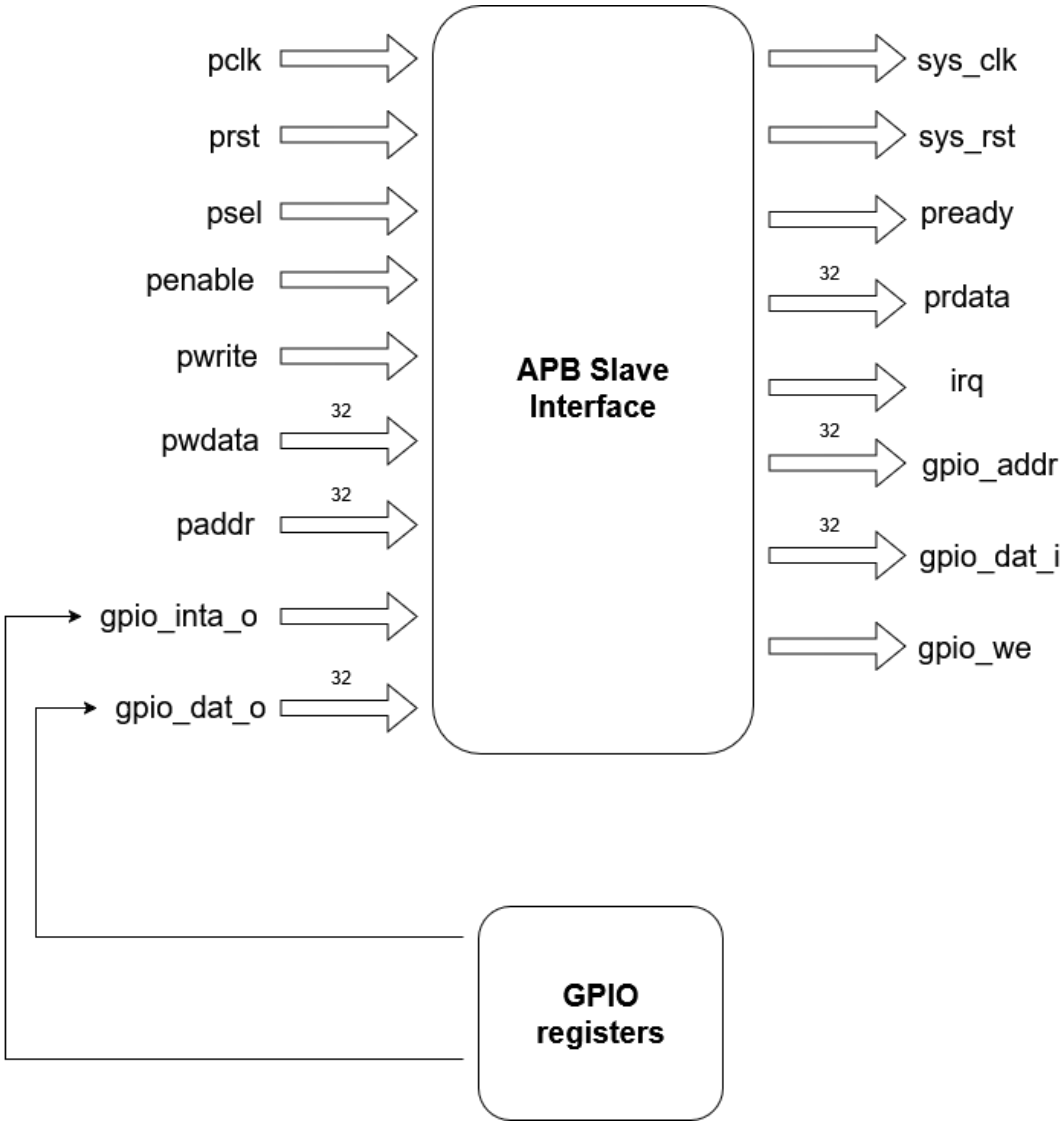
Signals	Type	Function
sys_clk	Input	System clock for the GPIO core
sys_rst	Input	System reset for the GPIO core
aux_in	Input	External 32-bit data from outside modules
aux_i	Output	Forwarded 32-bit data from external modules to APB slave interface

Working Principle:

- The module works with synchronous active low reset. When the sys_rst = 0, the system values are set to zero.
- When sys_rst = 1, the module begins to store and forward the data to the APB slave interface.
- During the positive edge of the sys_clk, the aux_in data is forwarded to aux_i which is used by the APB slave interface.

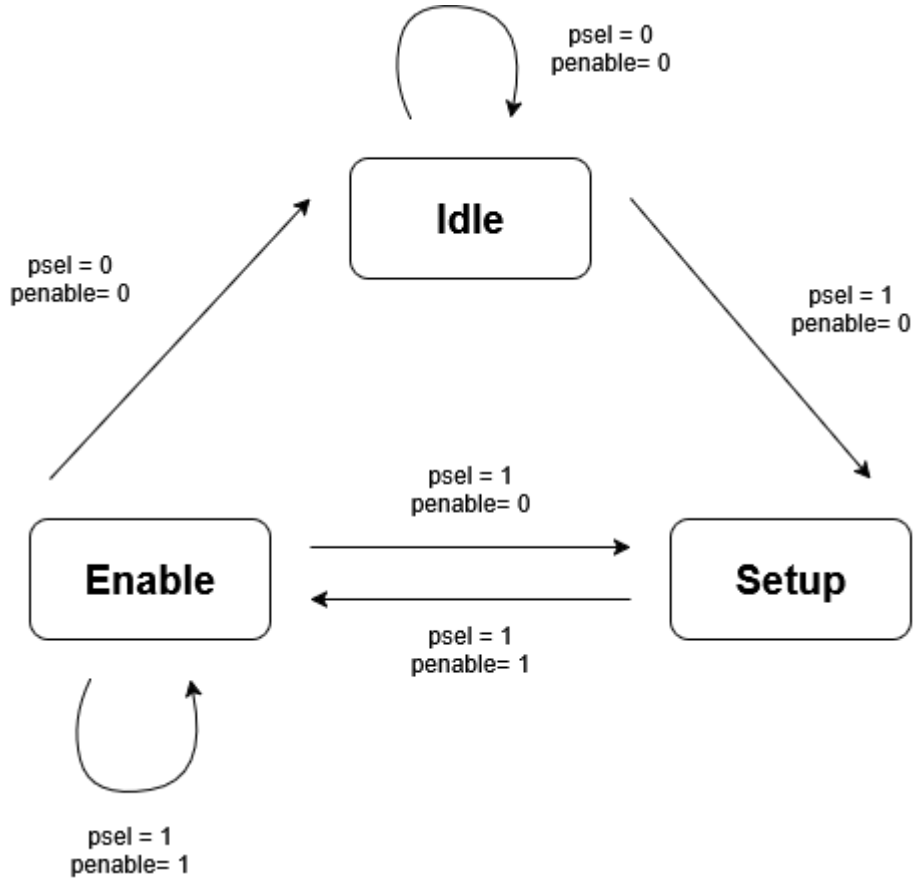
APB Slave Interface:

APB slave interface is the interpreter and between the system bus (AMBA protocol) and the GPIO core. It receives instructions from the master which is the central processing unit and performs read or write data to GPIO internal registers.



Signals	Type	Function
pclk	Input	Clock signal from the master system
prst	Input	Reset signal from the master system
psel	Input	Signal for selection of GPIO peripheral for transfer of read or write
penable	Input	Second stage signal to completely enable transfer of data
pwrite	Input	Signal for read or write operation
pwdata	Input	32-bit data to be written in GPIO register from master system
paddr	Input	32-bit address for the GPIO register from the master
gpio_dat_o	Input	32-bit data read from the GPIO register
gpio_inta_o	Input	Interrupt signal from the GPIO register
sys_clk	Output	Clock signal for the GPIO core
sys_rst	Output	Reset signal for the GPIO core
pready	Output	Signal for completion of transfer
prdata	Output	32-bit data from GPIO registers to the master system
irq	Output	Interrupt signal forwarded to the internal core for handling
gpio_addr	Output	32-bit address of the GPIO register to perform given instruction
gpio_dat_i	Output	32-bit data forwarded to GPIO register
gpio_we	Output	Enable signal for successful write operation

State Diagram:

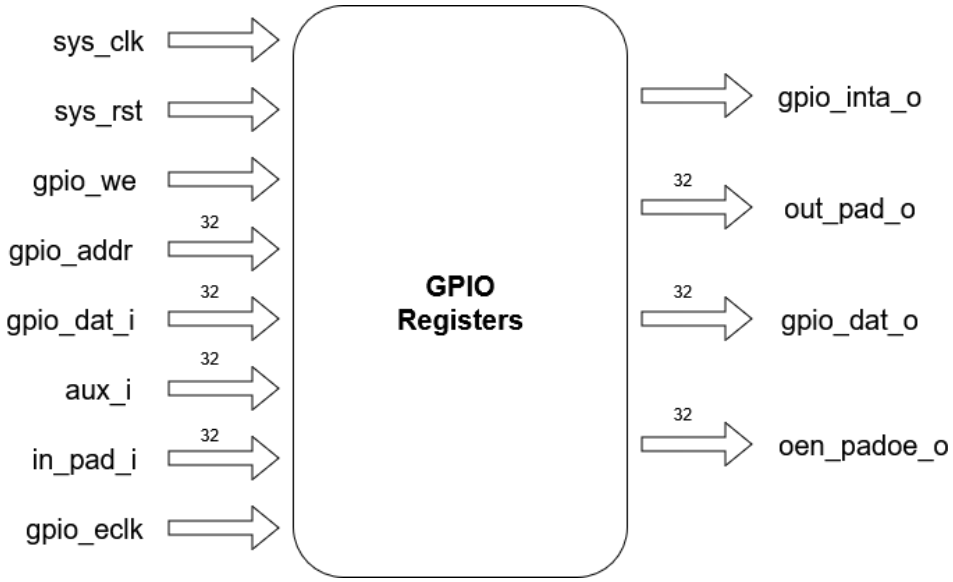


Working Principle:

- The APB slave interface is a Finite State Model (FSM) with 3 states: Idle, Setup and Enable.
- When psel = 1 and penable = 0, the system enters into setup state phase and indicates a data transfer is going to occur.
- During the setup phase unless the penable = 1, the system remains in the setup phase waiting for the enable signal to commence the transfer.
- When psel = 1 and penable = 1, the system enters into enable state, initiating the continuous transfer of data until psel or penable revert to 0.
- During enable state, if penable = 0, the system goes back to setup state for next transfer. If both psel = 0 and penable = 0, the system resets to idle state waiting for new transfer request.

GPIO registers:

The GPIO registers are memory mapped registers that stores necessary information regarding the operations performed and the data during the transfer between the GPIO peripherals and the master system.



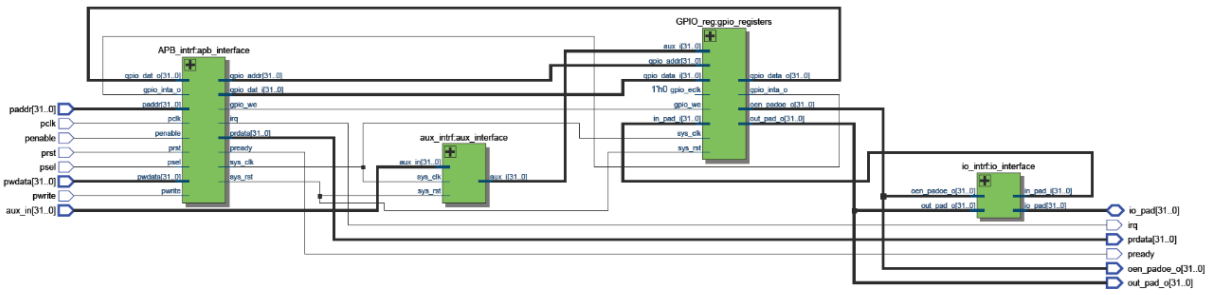
Signal	Type	Function
sys_clk	Input	System clock for GPIO core
sys_rst	Input	System reset for GPIO core
gpio_we	Input	Signal for write enable
gpio_addr	Input	32-bit address from APB slave interface
gpio_dat_i	Input	32-bit data from the APB slave interface
aux_i	Input	32-bit data from the auxiliary interface
in_pad_i	Input	32-bit data from the GPIO peripherals
gpio_eclk	Input	Signal for eternal clock usage
gpio_inta_o	Output	32-bit interrupt signal for 32 GPIO peripherals
gpio_dat_o	Output	32-bit data to the APB slave
out_pad_o	Output	32-bit data sent to the GPIO peripherals
oen_padoe_o	Output	Output enable signal for each GPIO peripheral

There are 10 unique GPIO registers in the GPIO core:

Register	Address	Purpose
GPIO_IN	0x0	Stores the 32-bit data from the GPIO peripheral (input)
GPIO_OUT	0x4	Sends the 32-bit data to the GPIO peripheral (output)
GPIO_OE	0x8	Determines if the data is sent to or received from GPIO peripheral
GPIO_INTE	0xC	Enables interrupt generation for each GPIO peripheral
GPIO_PTRIG	0x10	Indicates the interrupt generation at positive clock edge
GPIO_AUX	0x14	Stores 32-bit data from the auxiliary interface
GPIO_CNTRL	0x18	Enables global interrupt
GPIO_INTS	0x1C	Status for interrupts for each GPIO peripheral
GPIO_ECLK	0x20	Enables the use of external clock
GPIO_NEC	0x24	Indicates the interrupt generation at negative clock edge

RTL view of GPIO core:

The RTL view is generated from the Quartus Prime. The APB slave receives the inputs and the IO interface sends the outputs to the external GPIO pins. The signals (irq, prdata, pready) goes back to the processor or the master system. The signals oen_padoe_o and out_pad_o are used to direct the data and direction of transfer.



Output Waveforms:

The testbench for the GPIO core involves reset for initialization of the system. During the next cycle reset is high (active low reset), and data (0Xabcd1234) is written to GPIO_OUT following the protocols (psel = 1 and penable = 1). GPIO_OE is set to enable the 4 pins for LSB to drive the outputs to the GPIO peripherals.