* + Inputs/Outputs  
     --Chip Select Signals
    - srr\_cs : Chip select signal for Software Reset Register (SRR)
    - spicr\_cs : Chip select signal for SPI Control Register (SPICR)
    - spisr\_cs : Chip select signal for SPI Status Register (SPISR)
    - spidtr\_cs : Chip select signal for AXI to SPI Data Transmit Register (SPIDTR)
    - spidrr\_cs : Chip select signal for SPI to AXI Data Receive Register (SPIDRR)
    - spissr\_cs : Chip select signal for SPI Slave Select Register (SPISSR)
    - tx\_fifo\_ocy\_cs : Chip select signal for Transmit FIFO Occupancy Register (Tx\_FIFO\_OCY)
    - rx\_fifo\_ocy\_cs : Chip select signal for Receive FIFO Occupancy Register (Rx\_FIFO\_OCY)
    - dgier\_cs : Chip select signal for Global Interrupt Enable Register (DGIER)
    - ipisr\_cs : Chip select signal for IP Interrupt Status Register (IPISR)
    - ipier\_cs : Chip select signal for IP Interrupt Enable Register (IPIER)  
        
      --Register read signals
    - reg\_rack : Read acknowledgement signal for currently accessed register.
      * Derived after a successful read is performed and data is in the reg\_rdata.
    - reg\_read\_enable : Sets up currently accessed register into the read mode.
      * Being set alongside reg\_write\_enable shall generate an error.
    - reg\_rdata : Data bus for register read.
    - reg\_rerror : Read error flag  
        
      --Register write signals
    - reg\_wack : Write acknowledgement signal for currently accessed register.
      * Derived after a successful read is performed and data is in the register.
    - reg\_wdata : Data bus for the register write.
    - reg\_wstr : Strobe mask for partial writes.
    - reg\_write\_enable : Sets up currently accessed register into the write mode.
      * Being set alongside reg\_read\_enable shall generate an error
    - reg\_werror : Write error flag  
        
      --SRR derived signals
    - Soft\_reset : software reset signal.  
        
      --SPICR derived signals
    - Lsb\_first : dictates whether the least significant bit will be transmitted first for a given byte
    - Master\_inhibit : disables master transactions
    - Manual\_ss\_en : turns off automatic slave select instead selecting as dictated in the SPISSR
    - Rx\_fifo\_reset : resets Rx FIFO
    - Tx\_fifo\_reset : resets Tx FIFO
    - Chpa : selects the clock phase
    - Cpol : selects clock polarity
    - Spi\_master\_en : selects betweeen master and slave configurations
    - Spi\_system\_en : turns entire module on or off
    - Loopback\_en : connects the MOSI and MISO ports for local loopback.  
        
      --SPISR derived signals
    - Slave\_mode\_select : Asserted when the core is in the slave mode
    - Mode\_fault\_error : Turns on when device is in master mode, but there’s another master steering the bus
    - Tx\_full : Indicates that Tx FIFO is full
    - Tx\_empty : Indicates that Tx FIFO is empty
    - Rx\_full : Indicates that Rx FIFO is full
    - Rx\_empty : Indicates that Rx FIFO is empty  
        
      --SPIDTR derived signals
    - Tx\_fifo\_data : Holds the value to be written into the Tx FIFO   
        
      --SPIDRR derived signals
    - Rx\_fifo\_data : Holds the value read from the Rx FIFO  
        
      --SPISSR derived signals:
    - slave\_select : outputs the slave to be selected in the manual slave select mode.(See SPISR signals)  
        
      -- Tx\_FIFO\_OCY derived signals:
    - Tx\_fifo\_occupancy : reports current number of elements in a Tx FIFO  
        
      -- Rx\_FIFO\_OCY derived signals:
    - Rx\_fifo\_occupancy : reports current number of elements in a Rx FIFO  
        
      --DGIER derived signals
    - Gi\_en : enables global interrupts for te module  
        
      --IPISR derived signals
    - Drr\_not\_empty : Indicates data to be received from SPIDRR.
      * Used when the C\_FIFO\_EXIST = 1 and device is in slave mode.
    - Slave\_select\_mode : Is asserted when this device functioning as a slave is selected by SPI master.
    - Tx\_fifo\_half\_empty : Asserted when transmit FIFO goes from having 9 elements to 8 elements(reaching half capacity)
    - Drr\_overrun : Indicates that SPIDRR and by extension Rx FIFO is losing data due to a write from SPI into a full FIFO.
    - Drr\_full : Indicates that the Rx FIFO is full.
    - Drr\_underrun : Indicates an attempted read from an empty Rx FIFO
    - Dtr\_empty : Indicates an empty Rx FIFO
    - Slave\_mode\_fault : asserts when device is in slave mode and is not active when the activity on SS bus is detected.
    - Mode\_fault\_error : asserts when activity from outside the module is detected when device is in master mode  
        
      --IPIER derived signals
    - Drr\_not\_empty\_int\_en : enables the drr\_not\_empty signal to assert interrupt from module
    - Ss\_mode\_int\_en : enables the Slave\_select\_mode signal to assert interrupt from module
    - Tx\_fifo\_half\_int\_en : enables the Tx\_fifo\_half\_empty signal to assert interrupt from module
    - Drr\_overrun\_int\_en : enables the Drr\_overrun signal to assert interrupt from module
    - Drr\_full\_int\_en : enables the Drr\_full signal to assert interrupt from module
    - Dtr\_underrun\_int\_en : enables the Drr\_underrun signal to assert interrupt from module
    - Dtr\_empty\_int\_en : enables the Dtr\_empty signal to assert interrupt from module
    - Slave\_mode\_fault\_int\_en : enables the Slave\_mode\_fault signal to assert interrupt from module
    - Mode\_fault\_int\_en : enables the Mode\_fault\_error signal to assert interrupt from module
  + Software Reset Register (SRR)
    - Represents one signal: soft\_reset. Soft\_reset is the built-in software reset for the system that is actuated when a value of 0x0000000A is written into SRR. On any other write, error is generated via reg\_werror signal.
  + SPI Control Register (SPICR)
    - Bit 9: Lsb\_first signal
    - Bit 8: Master\_inhibit signal
    - Bit 7: Manual\_ss\_en signal
    - Bit 6: Rx\_fifo\_reset signal
    - Bit 5: Tx\_fifo\_reset
    - Bit 4: Chpa signal
    - Bit 3: Cpol signal
    - Bit 2: Spi\_master\_en signal
    - Bit 1: Spi\_system\_en signal
    - Bit 0: Loopback\_en signal
  + SPI Status Register (SPISR)
    - Bit 5 : slave\_mode\_select signal
    - Bit 4 : mode\_fault\_error signal
    - Bit 3 : tx\_full signal
    - Bit 2 : tx\_empty signal
    - Bit 1 : rx\_full signal
    - Bit 0 : rx\_empty signal
  + Data Registers
    - AXI to SPI Data Transmit Register (SPIDTR)
      * Holds tx\_fifo\_data of size of C\_NUM\_TRANSFER bits.
    - SPI to AXI Data Receive Register (SPIDRR)
      * Holds rx\_fifo\_data of size of C\_NUM\_TRANSFER bits.
  + SPI Slave Select Register (SPISSR)
    - Holds slave\_select of size of C\_NUM\_SS\_BITS bits.
  + Transmit FIFO Occupancy Register (Tx\_FIFO\_OCY)
    - Bits [3:0] : tx\_fifo\_occupancy.
  + Receive FIFO Occupancy Register (Rx\_FIFO\_OCY)
    - Bits [3:0] : rx\_fifo\_occupancy.
  + Interrupt generation interface
    - Global Interrupt Enable Register (DGIER)
      * Bit 31 : gi\_en.
    - IP Interrupt Status Register (IPISR)
      * Bit 8: drr\_not\_empty signal
      * Bit 7: slave\_select\_mode signal
      * Bit 6: tx\_fifo\_half\_empty signal
      * Bit 5: drr\_overrun signal
      * Bit 4 : drr\_full signal
      * Bit 3 : drr\_underrun signal
      * Bit 2 : dtr\_empty signal
      * Bit 1 : slave\_mode\_fault signal
      * Bit 0 : mode\_fault\_error signal
    - IP Interrupt Enable Register (IPIER)
      * Bit 8 : Drr\_not\_empty\_int\_en signal
      * Bit 7 : Ss\_mode\_int\_en signal
      * Bit 6 : Tx\_fifo\_half\_int\_en signal
      * Bit 5 : Drr\_overrun\_int\_en signal
      * Bit 4 : Drr\_full\_int\_en signal
      * Bit 3 : Dtr\_underrun\_int\_en signal
      * Bit 2 : Dtr\_empty\_int\_en signal
      * Bit 1 : Slave\_mode\_fault\_int\_en signal
      * Bit 0 : Mode\_fault\_int\_en signal