1. AXI Interface
   1. General functionality
      1. The device should function only if provided either read or write address in range [C\_BASEADDR , C\_HIGHADDR].
   2. Address Decoder and Register Select(Chip select for registers)  
       --Software Reset Register Chip Select generation (Write only)
      1. Srr\_cs signal shall be set when s\_axi\_awaddr is equal to 0x[C\_BASEADDR + 40] and s\_axi\_awvalid is set.  
           
         --SPI Control Register Chip Select generation (R/W)
      2. Spicr\_cs signal shall be set when s\_axi\_awaddr is equal to 0x[C\_BASEADDR + 60] and s\_axi\_awvalid is set.
      3. Spicr\_cs signal shall be set when s\_axi\_araddr is equal to 0x[C\_BASEADDR + 60] and s\_axi\_arvalid is set.  
           
         --SPI Status Register Chip Select generation (Read only)
      4. Spisr\_cs signal shall be set when s\_axi\_araddr is equal to 0x[C\_BASEADDR + 64] and s\_axi\_arvalid is set.  
           
         --SPI Data Transmit Register Chip Select generation (Write only)
      5. Spidtr\_cs signal shall be set when s\_axi\_awaddr is equal to 0x[C\_BASEADDR + 68] and s\_axi\_awvalid is set.  
           
         --SPI Data Receive Register Chip Select generation (Read only)
      6. Spidrr\_cs signal shall be set when s\_axi\_araddr is equal to 0x[C\_BASEADDR + 6C] and s\_axi\_arvalid is set.  
           
         --SPI Slave Select Register Chip Select generation (R/W)
      7. Spissr\_cs signal shall be set when s\_axi\_awaddr is equal to 0x[C\_BASEADDR + 70] and s\_axi\_awvalid is set.
      8. Spissr\_cs signal shall be set when s\_axi\_araddr is equal to 0x[C\_BASEADDR + 70] and s\_axi\_arvalid is set.  
           
         --SPI Transmit FIFO Occupancy Register Chip Select generation (Read only)
      9. Tx\_fifo\_ocy\_cs signal shall be set when s\_axi\_araddr is equal to 0x[C\_BASEADDR + 74] and s\_axi\_arvalid is set.  
           
         --SPI Receive FIFO Occupancy Register Chip Select generation (Read only)
      10. Rx\_fifo\_ocy\_cs signal shall be set when s\_axi\_araddr is equal to 0x[C\_BASEADDR + 78] and s\_axi\_arvalid is set.  
            
          --Device Global Interrupt Enable Register Chip Select generation (R/W)
      11. Dgier\_cs signal shall be set when s\_axi\_awaddr is equal to 0x[C\_BASEADDR + 1C] and s\_axi\_awvalid is set.
      12. Dgier\_cs signal shall be set when s\_axi\_araddr is equal to 0x[C\_BASEADDR + 1C] and s\_axi\_arvalid is set.  
            
          --IP Interrupt Status Register Chip Select generation (R/TOW)
      13. Ipisr\_cs signal shall be set when s\_axi\_awaddr is equal to 0x[C\_BASEADDR + 20] and s\_axi\_awvalid is set.
      14. Ipisr\_cs signal shall be set when s\_axi\_araddr is equal to 0x[C\_BASEADDR + 20] and s\_axi\_arvalid is set.  
            
          --IP Interrupt Enable Register Chip Select generation (R/W)
      15. Ipier\_cs signal shall be set when s\_axi\_awaddr is equal to 0x[C\_BASEADDR + 28] and s\_axi\_awvalid is set.
      16. Ipier\_cs signal shall be set when s\_axi\_araddr is equal to 0x[C\_BASEADDR + 28] and s\_axi\_arvalid is set.
   3. Read Cycle Signaling  
       --State S1: Setting up address for read
      1. Axi\_state shall be set to S1 when axi\_state is S0 and s\_axi\_arvalid is set.   
           
         --State S2: Broadcast
      2. Axi\_state shall be set to S2 when axi\_state is S1 and reg\_rack is set.  
           
         --End of broadcast: return to S0(waiting)
      3. Axi\_state shall be set to S0 when axi\_state is S2 and reg\_rack is reset.  
           
         --Register Read enable signal generation
      4. Reg\_read\_enable shall be set when s\_axi\_arvalid is set.  
           
         --Address read ready signal generation
      5. S\_axi\_arready shall be set when reg\_rack is set.  
           
         --Read data output latching
      6. S\_axi\_rdata shall be set equal to reg\_rdata\_latch when s\_axi\_rready is set.
      7. Reg\_rdata\_latch shall be set equal to reg\_rdata when reg\_rack is set.  
           
         --Read response bus error signal generation
      8. S\_axi\_rresp shall be set to 01 when reg\_rerror signal is set.  
           
         --Read valid signal generation
      9. S\_axi\_rvalid shall be set after one s\_axi\_aclk clock cycle when reg\_rack is set.
   4. Write Cycle Signaling  
       --State S3: Writing to a register address
      1. Axi\_state shall be set to S3 when axi\_state is S0 and s\_axi\_awvalid is set and s\_axi\_wvalid is set and reg\_wack is reset.  
           
         --End of write sequence: return to S0(waiting)
      2. Axi\_state shall be set to S0 when axi\_state is S3 and reg\_wack is set.  
           
         --Write data passthrough
      3. Reg\_wdata shall be set to s\_axi\_wdata when s\_axi\_wvalid is set.  
           
         --Write data strobe passthrough
      4. Reg\_wstb shall be set to s\_axi\_wstb when s\_axi\_wvalid is set.  
           
         --Register Write enable signal generation
      5. Reg\_write\_enable shall be set when s\_axi\_wvalid is set.  
           
         --Address write ready signal generation
      6. S\_axi\_awready shall be set when reg\_wack is set.  
           
         --Write ready signal generation
      7. S\_axi\_wready shall be set when reg\_wack is set.  
           
         --Write response bus error signal generation
      8. S\_axi\_bresp shall be set to 01 when reg\_werror signal is set.  
           
         --Write valid signal generation
      9. S\_axi\_bvalid shall be set one cycle after reg\_wack is set and reg\_werror is not set.
2. SPI Interface
   1. Initial Conditions:
      1. SCK\_T, MOSI\_T, MISO\_T shall all be initialized as low.
      2. MOSI\_O, MISO\_O, and SCLK\_O shall all be initialized as high
   2. Input tristate signal:
      1. The MOSI port shall be equal to MOSI\_O when MOSI\_T is high.
      2. The MOSI\_I shall be equal to value at MOSI port when MOSI\_T is low.
      3. The MISO port shall be equal to MISO\_O when MISO\_T is high.
      4. The MISO\_I shall be equal to value at MISO port when MISO\_T is low.
      5. The SCK port shall be equal to SCK\_O when SCK\_T is high.
      6. The SCK\_I shall be equal to value at SCK port when SCK\_T is low.
   3. Baud Rate Generator
      1. SCK\_O shall be set to the clock signal of frequency equal to that of SCK\_I divided by C\_SCK\_RATIO when the device is in the master mode.
   4. Serializer / Deserializer
      1. The shift register shall perform a bit shift right on the rising edge of SCK\_O when lsb\_signal is high and cpha is low.
      2. The shift register shall perform a bit shift left on the rising edge of SCK\_O when lsb\_signal is low and cpha is low.
      3. The shift register shall perform a bit shift right on the falling edge of SCK\_O when lsb\_signal is high and cpha is high.
      4. The shift register shall perform a bit shift left on the falling edge of SCK\_O when lsb\_signal is low and cpha is high.
      5. Internal counter that indicates current bit position shall advance in range 0 to [C\_NUM\_TRANSFER\_BITS – 1] with value of [C\_NUM\_TRANSFER\_BITS – 1] corresponding to the complete message sent.
      6. Shift register shall perform a write into the Rx FIFO followed by read from Tx FIFO on internal counter value of [C\_NUM\_TRANSFER\_BITS – 1].
      7. The carry-in for shift register shall originate from MOSI\_I in slave mode.
      8. The carry-in for shift register shall originate from MISO\_I in master mode.
      9. The carry-out for shift register shall be connected to MISO\_O in slave mode.
      10. The carry-out for shift register shall be connected to MOSI\_O in master mode when master\_inhibit is low.
   5. Slave Selector
      1. The SS(N) bus shall output the value written into the SPISSR register when manual\_ss\_en is high.
      2. The SS(N) bus shall advance through the SS bits starting with SS(0) in zero\_hot encoding manner at internal counter value of [C\_NUM\_TRANSFER\_BITS – 1] when manual\_ss\_en is low.
   6. Control Unit
      1. The slave mode shall initiate for device when SPISEL\_I is reset.
      2. The slave mode shall initiate for device when spi\_master\_en is reset.
      3. The SCK\_O shall be inverted when cpol is set.
      4. The module shall be disabled when spi\_system\_en is reset.
      5. The MOSI and MISO lines shall be shorted when loopback\_en is set.
      6. Slave\_mode\_select shall be set when the SPISEL\_I is set.
      7. Slave\_mode\_select shall be set when the spi\_master\_en is set.
3. Registers
   1. Shared register functionality
      1. Register Size
         1. The register shall be instantiated to a size of 32 bits.
            1. Exceptions are SPIDTR and SPIDRR registers, which initialize to a size of C\_NUM\_TRANSFER\_BITS bits.
      2. Data Write
         1. The register shall only modify contents of the bits flagged by reg\_wstb by equivalent bits from reg\_wdata when reg\_write\_enable is set.
         2. The flagged bits shall set their value equal to the value of the corresponding bit from reg\_wdata.
            1. Exception to (3.-a.-ii.-2.) is IPISR register, which follows toggle-on-write procedure. (refer to IPISR requirements)
         3. The signal reg\_wack shall be set when the register write has been completed for a duration of 1 clock cycle.
      3. Data Read
         1. Reg\_rdata shall be set to the contents of the register when both reg\_read\_enable is set and corresponding chip select signal (e.g. srr\_cr for SRR) is set.
         2. The signal reg\_wack shall be set when the register write has been completed for a duration of 1 clock cycle.
      4. Chip select error correction
         1. The register shall set reg\_werror and reg\_rerror when reg\_read\_enable is set and reg\_write\_enable is set.
   2. Software Reset Register (SRR)
      1. Register shall initialize to 0x0000\_0000 upon system bootup.
      2. Register shall initialize to 0x0000\_0000 upon system reset.
      3. Soft\_reset shall be reset when the register content is equal to 0x0000\_0000.
      4. Soft\_reset shall be set when the register content is equal to 0x0000\_000A.
      5. Reg\_werror shall be set when srr\_cs is set and reg\_wdata is not equal to 0x0000\_000A.
   3. SPI Control Register (SPICR)
      1. Register shall initialize to 0x0000\_0180 upon system bootup.
      2. Register shall initialize to 0x0000\_0180 upon system reset.
      3. Lsb\_first signal shall be set when bit 9 of the register is set.
      4. Master\_inhibit signal shall be set when bit 8 of the register is set.
      5. Manual\_ss\_en signal shall be set when bit 7 of the register is set.
      6. Rx\_fifo\_reset signal shall be set when bit 6 of the register is set.
      7. Tx\_fifo\_reset signal shall be set when bit 5 of the register is set.
      8. Chpa signal shall be set when bit 4 of the register is set.
      9. Cpol signal shall be set when bit 3 of the register is set.
      10. Spi\_master\_en signal shall be set when bit 2 of the register is set.
      11. Spi\_system\_en signal shall be set when bit 1 of the register is set.
      12. Loopback\_en signal shall be set when bit 0 of the register is set.
   4. SPI Status Register (SPISR)
      1. Bit 5 of the register shall be set when slave\_mode\_select is set.
      2. Bit 4 of the register shall be set when mode\_fault\_error is set.
      3. Bit 3 of the register shall be set when tx\_full is set.
      4. Bit 2 of the register shall be set when tx\_empty is set.
      5. Bit 1 of the register shall be set when rx\_full is set.
      6. Bit 0 of the register shall be set when rx\_empty is set.
   5. Data Registers
      1. AXI to SPI Data Transmit Register (SPIDTR)
         1. Register shall initialize to empty upon system bootup.
         2. Register shall initialize to empty upon system reset.
         3. Tx\_fifo\_data shall be of size of C\_NUM\_TRANSFER bits.
         4. Tx\_fifo\_data shall be equal to the contents of the register.
      2. SPI to AXI Data Receive Register (SPIDRR)
         1. Register shall initialize to empty upon system bootup.
         2. Register shall initialize to empty upon system reset.
         3. Contents of the register shall be equal to Rx\_fifo\_data.
   6. Slave Select Register (SPISSR)
      1. Register shall initialize to 0x0000\_0001 upon system bootup.
      2. Register shall initialize to 0x0000\_0001 upon system reset.
      3. Bits [ [C\_NUM\_SS\_BITS -1] : 0] of the register shall be a one-hot encoded representation of the selected slave to transfer data to.
         1. At most one bit of the register is allowed to be set.
      4. Slave\_select shall be of size of C\_NUM\_SS\_BITS bits.
      5. Slave\_select shall be equal to the contents of the register.
   7. Transmit FIFO Occupancy Register (Tx\_FIFO\_OCY)
      1. Register shall initialize to empty upon system bootup.
      2. Register shall initialize to empty upon system reset.
      3. Bits [3:0] of the register shall be equal to the tx\_fifo\_occupancy.
   8. Receive FIFO Occupancy Register (Rx\_FIFO\_OCY)
      1. Register shall initialize to empty upon system bootup.
      2. Register shall initialize to empty upon system reset.
      3. Bits [3:0] of the register shall be equal to the rx\_fifo\_occupancy.
   9. Interrupt generation interface
      1. Global Interrupt Enable Register (DGIER)
         1. Register shall initialize to empty upon system bootup.
         2. Register shall initialize to empty upon system reset.
         3. Bit 31 of the register shall be equal to the gi\_en.
      2. IP Interrupt Status Register (IPISR)
         1. Register shall initialize to empty upon system bootup.
         2. Register shall initialize to empty upon system reset.
         3. Register bit 8 shall have its value inverted when bit 8 of reg\_wdata is set and bit 8 of reg\_wstrb is set.
         4. Register bit 8 shall be set when register bit 8 is currently reset and drr\_not\_empty is set.
         5. Register bit 7 shall have its value inverted when bit 7 of reg\_wdata is set and bit 7 of reg\_wstrb is set.
         6. Register bit 7 shall be set when register bit 7 is currently reset and slave\_select\_mode is set.
         7. Register bit 6 shall have its value inverted when bit 6 of reg\_wdata is set and bit 6 of reg\_wstrb is set.
         8. Register bit 6 shall be set when register bit 6 is currently reset and tx\_fifo\_half\_empty is set.
         9. Register bit 5 shall have its value inverted when bit 5 of reg\_wdata is set and bit 5 of reg\_wstrb is set.
         10. Register bit 5 shall be set when register bit 5 is currently reset and drr\_overrun is set.
         11. Register bit 4 shall have its value inverted when bit 4 of reg\_wdata is set and bit 4 of reg\_wstrb is set.
         12. Register bit 4 shall be set when register bit 4 is currently reset and drr\_full is set.
         13. Register bit 3 shall have its value inverted when bit 3 of reg\_wdata is set and bit 3 of reg\_wstrb is set.
         14. Register bit 3 shall be set when register bit 3 is currently reset and dtr\_underrun is set.
         15. Register bit 2 shall have its value inverted when bit 2 of reg\_wdata is set and bit 2 of reg\_wstrb is set.
         16. Register bit 2 shall be set when register bit 2 is currently reset and dtr\_empty is set.
         17. Register bit 1 shall have its value inverted when bit 1 of reg\_wdata is set and bit 1 of reg\_wstrb is set.
         18. Register bit 1 shall be set when register bit 1 is currently reset and slave\_mode\_fault\_error is set.
         19. Register bit 0 shall have its value inverted when bit 0 of reg\_wdata is set and bit 0 of reg\_wstrb is set.
         20. Register bit 0 shall be set when register bit 0 is currently reset and mode\_fault\_error is set.
      3. IP Interrupt Enable Register (IPIER)
         1. Register shall initialize to empty upon system bootup.
         2. Register shall initialize to empty upon system reset.
         3. Drr\_not\_empty\_int\_en is set when bit 8 of the register is set.
         4. Ss\_mode\_int\_en is set when bit 7 of the register is set.
         5. Tx\_fifo\_half\_int\_en is set when bit 6 of the register is set.
         6. Drr\_overrun\_int\_en is set when bit 5 of the register is set.
         7. Drr\_full\_int\_en is set when bit 4 of the register is set.
         8. Dtr\_underrun\_int\_en is set when bit 3 of the register is set.
         9. Dtr\_empty\_int\_en is set when bit 2 of the register is set.
         10. Slave\_mode\_fault\_int\_en is set when bit 1 of the register is set.
         11. Mode\_fault\_int\_en is set when bit 0 of the register is set.
4. Asynchronous FIFO(Optional)
   1. FIFO structure is generated when C\_FIFO\_EXIST is high (instance generic)
      1. Otherwise a basic double synchronizer is generated.
   2. FIFO data width is tied to C\_NUM\_TRANSFER\_BITS (instance generic)
      1. FIFO depth is always 16
   3. Double synchronizer for register signals
      1. The AXI and SPI interfaces will double synchronize read and write data from/to the selected non-data registers.
      2. Data register (SPIDRR and SPIDTR) synchronization is handled by FIFO modules.
   4. Module Signals
      1. Tx FIFO (AXI to SPI)
         1. Wdata tied to SPI Data Transmit Register (SPIDTR)
         2. Rdata is tied to SPI Module data serializer input
         3. w\_enable is set when transmit data register has been loaded
            1. S\_AXI\_BVALID
         4. r\_enable is set when SPI module requests data transfer
            1. SPI serializer requests new byte (internal count)
         5. Reset is tied to Tx FIFO Reset bit of SPI Control Register (SPICR[5])
         6. wclk is tied to AXI clock (S\_AXI\_ACLK)
         7. rclk is tied to SPI clock output (SCK\_O)
         8. full\_flag is tied to Tx\_Full bit of SPI Status Register (SPISR[3])
         9. empty\_flag is tied to Tx\_Empty bit of SPI Status Register (SPISR[2])
         10. Occupancy value is tied to Tx\_FIFO\_OCY[3:0]
      2. Rx FIFO (SPI to AXI)
         1. Wdata is tied to SPI module data output (shift register output)
         2. Rdata tied to SPI Data Receive Register (SPIDRR)
         3. w\_enable is set when SPI shift register is full (internal counter)
         4. r\_enable is set when AXI register module requests data transfer
            1. S\_AXI\_RRESP
         5. Reset tied to Rx FIFO Reset bit of SPI Control Register (SPICR[6])
         6. wclk is tied to SPI clock output (SCK\_O)
         7. rclk is tied to AXI clock (S\_AXI\_ACLK)
         8. full\_flag is tied to Rx\_Full bit of SPI Status Register (SPISR[1])
         9. empty\_flag is tied to Rx\_Empty bit of SPI Status Register (SPISR[0])
         10. Occupancy value is tied to Rx\_FIFO\_OCY [3:0]
   5. IP Interrupt Status Register Conditions
      1. Bit 8 – DRR\_Not\_Empty is set when the Rx\_FIFO recieves the first data value during SPI transaction
      2. Bit 6 – Tx FIFO Half Empty is set when the transmit FIFO occupancy decrements from “1000” to “0111”
      3. Bit 5 – DRR Overrun is set when Rx FIFO full flag is 1 and a write operation is attempted by SPI bus.
      4. Bit 4 – DRR Full is set with Rx FIFO full\_flag and one clock strobe to indicate end of SPI element transfer
      5. Bit 3 – DTR Under run is set with Tx FIFO empty and SPI request. (slave only)
      6. Bit 2 – DTR Empty is set with Tx FIFO empty to indicate end of SPI element transfer. (slave only)