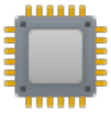
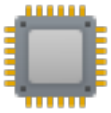


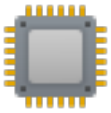
SSP332i User Manual

JUNE 2021

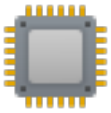
Doc. Name: SSP332i User Manual

Contents

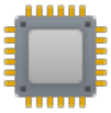




List of Figures



List of Tables



1

Block Diagram

1.1 Core

The SSP332i SoC has a 32bit 3-stage RISCV32I core at its heart. It supports all the RISCV32I instructions and the M mode privilege level. There is provison for the core to raise an exception on illegal instruction, load-store address misalignment, instruction address misallignment, software interrupt, timer interrupt and external interrupt. All memory access are single cycled. The pipeline executes by fowarding the data in case of a dependent instruction. All load and store causes a single cycle stall in the pipeline. There is no branch prediction nor any cache or any other fancy stuff.

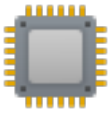
The reset values of various important registers is given in the table 1.1.

1.1.1 Clock and Reset

The system clock is set as 24Mhz on the Tang primer board and reset is active low. For further configuration use the PLL IP.

Register	Reset Value
PC	0x000010000
Stack Pointer	0x0001FFFF

Table 1.1: Reset value of core registers



1.2 Bus Interface

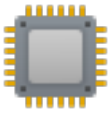
The core has 2 separate bus for interfacing the Instruction memory and Data memory. The instruction memory is a simple address, data interface. The data memory conforms to a standard APB interface standard.

1.2.1 Advanced Peripheral bus protocol

The Advanced Peripheral Bus (APB) is part of the Advanced Microcontroller Bus Architecture (AMBA) protocol family. It defines a low-cost interface that is optimized for minimal power consumption and reduced interface complexity. The APB protocol is not pipelined. It is used connect to low-bandwidth peripherals that do not require the high performance of the AXI protocol. The APB protocol relates a signal transition to the rising edge of the clock, to simplify the integration of APB peripherals into any design flow. Every transfer takes at least two cycles.

1.2.2 AMBA APB signals

- PCLK - The rising edge of PCLK times all transfers on the APB.
- PRESETn -The APB reset signal is active LOW. This signal is normally connected directly to the system bus reset signal.
- PADDR - This is the APB address bus. It can be up to 32 bits wide and is driven by the peripheral bus bridge unit.
- PPROT - This signal indicates the normal, privileged, or secure protection level of the transaction and whether the transaction is a data access or an instruction access.
- PSELx - The APB bridge unit generates this signal to each peripheral bus slave. It indicates that the slave device is selected and that a data transfer is required. There is a PSELx signal for each slave.
- PENABLE (APB bridge Enable) This signal indicates the second and subsequent cycles of an APB transfer.
- PWRITE (APB bridge Direction) - This signal indicates an APB write access when HIGH and an APB read access when LOW.
- PWDATA (APB bridge Write data) - This bus is driven by the peripheral bus bridge unit during write cycles when PWRITE is HIGH. This bus can be up to 32 bits wide.



- PSTRB (APB bridge Write strobes) - This signal indicates which byte lanes to update during a write transfer. There is one write strobe for each eight bits of the write data bus.
- PREADY - Slave interface Ready. The slave uses this signal to extend an APB transfer.
- PRDATA - Slave interface Read Data. The selected slave drives this bus during read cycles.
- PSLVERR - Slave interface error, This signal indicates a transfer failure.

1.2.3 Write transfers

There are two types of write transfers:

- Write transfer without wait state
- Write transfer with wait states

A write transfer starts with address PADDR, write data PWDATA, write signal PWRITE, and select signal PSEL, being registered at the rising edge of PCLK. This is called the Setup phase of the write transfer. At the next clock cycle enable signal PENABLE, and ready signal PREADY, are registered at the rising edge of PCLK. When asserted, PENABLE indicates the start of the Access phase of the transfer. When asserted, PREADY indicates that the slave can complete the transfer at the next rising edge of PCLK. The address PADDR, write data PWDATA, and control signals all remain valid until the transfer completes. The enable signal PENABLE, is de-asserted at the end of the transfer. The select signal PSEL, is also de-asserted unless the transfer is to be followed immediately by another transfer to the same peripheral

The PREADY signal can be extended low if the slave device is not able to accept the write transfer, in such a case the PENABLE is extended. Such and transfer is called write transfer with wait states. The core however **does not support wait states**.

1.2.4 Read transfers

Two types of read transfer are described in this section:

- With no wait states

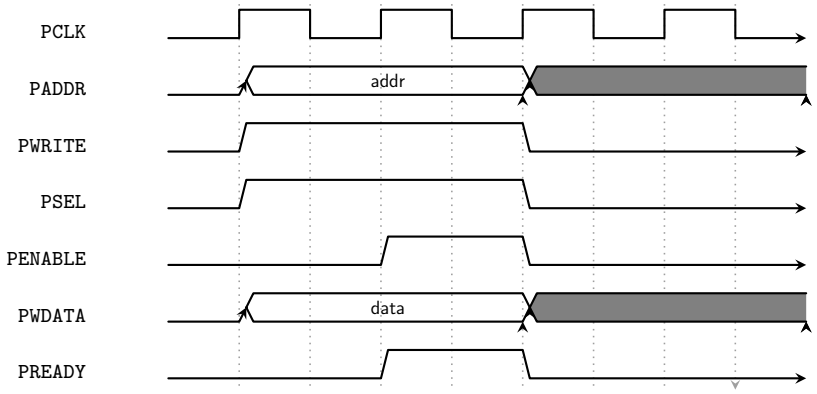


Figure 1.1: Write transfer with no wait states

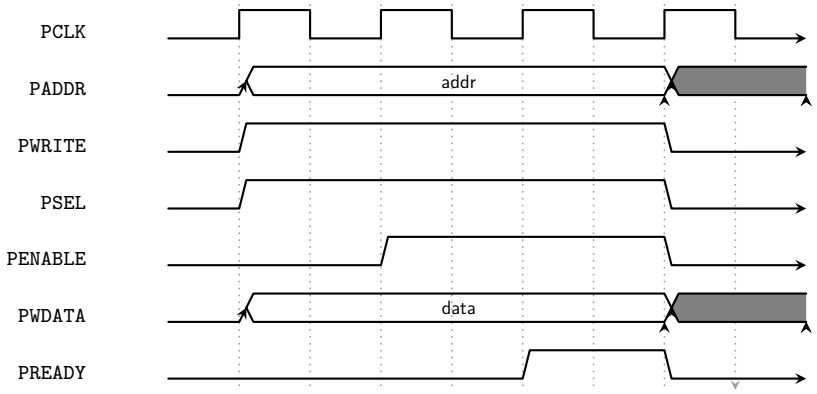


Figure 1.2: Write transfer with wait states **not supported by the core**

- With wait states.

The signals timings for read transfer with and without waits states is shown in the figure.

1.3 Addesss Mapping

The address map for the 12 peripherals is shown below :

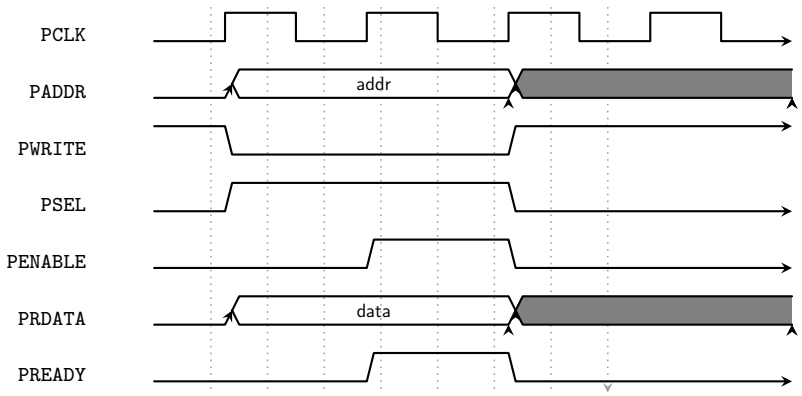


Figure 1.3: Read transfer with no wait states

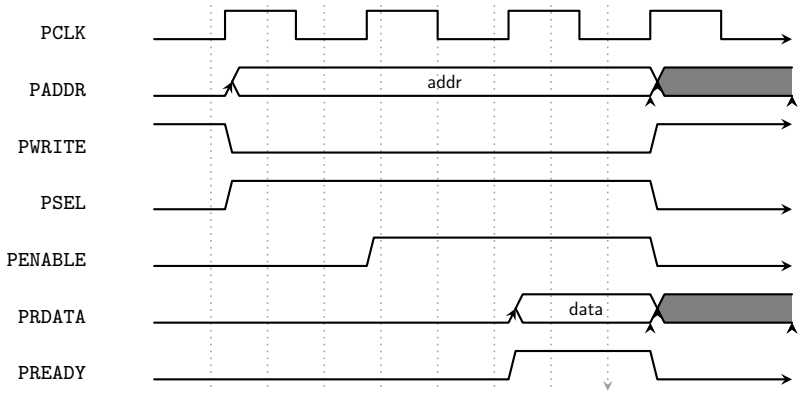
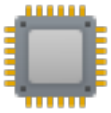
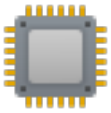


Figure 1.4: Read transfer with wait states **not supported by the core**



Peripheral Number	Peripheral	Low Addesss	High Address
RAM 8KB	0	0x1000_0000	0x1000_1FFF
UART	1	0x4000_0000	0x4000_FFFF
PLIC	2	0x5000_0000	0x5000_FFFF
GPIO	3	0x6000_0000	0x6000_FFFF
SPI	4	0x7000_0000	0x7000_FFFF
ROM	5	0x0000_0000	0x0000_FFFF
SD	6	0x8000_0000	0x8000_FFFF
I2C	7	0x9000_0000	0x9000_FFFF
I2S	8	0xA000_0000	0xA000_FFFF
DFF_RAM 1KB	9	0xB000_0000	0xB000_03FF
TIMER	10	0xC000_0000	0xC000_FFFF
PWM	11	0xD000_0000	0xD000_FFFF

Table 1.2: Address Mapping



2

Peripherals

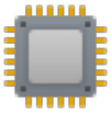
The SoC consists of the following peripherals :-

- UART
- PLIC
- GPIO
- SPI
- SD Card interface
- I2C
- I2S
- TIMER
- PWM

2.1 UART

The UART supports the following features :

- APB interface for register access and data transfers
- Supports default core configuration for 9600 baud, 8 bits data length, 1 stop bit and no parity
- 5, 6, 7 or 8 bits per character
- Odd, Even or no parity.



- 1, 2 stop bit.
- Internal baud rate generator.
- Modem control functions.
- Prioritized transmit, receive, line status and modem control interrupts.
- False start bit detection and recover.
- Line break detection and generation.
- Internal loopback diagnostic functionality.
- 16 character transmit and receive FIFOs.
- Designed with the same address space and register naming as Xilinx AXI UART.

2.1.1 Register Space

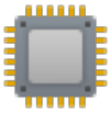
LCR[7]	Addesss Offset	Name	R/W	Description
0	0x1000	RBR	RO	Receiver Buffer Register
0	0x1000	THR	WO	Transmitter Holding Register
0	0x1004	IER	R/W	Interrupt Enable Register
x	0x1008	IIR	RO	Interrupt Identification Register
x	0x1008	FCR	WO	FIFO Control Register
1	0x1008	FCR	RO	FIFO Control Register
x	0x100C	LCR	R/W	Line Control Register
x	0x1010	MCR	R/W	Modem Control Register
x	0x1014	LSR	R/W	Line Status Register
x	0x1018	MSR	R/W	R/W Modem Status Register
x	0x101C	SCR	R/W	Scratch Register
1	0x1000	DLL	R/W	Divisor Latch (LSB) Register
1	0x1004	DLM	R/W	Divisor Latch (MSB) Register

Table 2.1: Address Mapping

2.1.2 Receiver Buffer Register - (RBR) [Read Only]

Address - 0x1000

Received character is available while reading this register.



2.1.3 Transmitter Holding Register - THR [Write Only]

Address - 0x1000

Write transmit character to this address.

2.1.4 Interrupt Enable Register (IER) [Read/Write]

Address - 0x1004

The Interrupt Enable register contains the bits which enable interrupts. The bit definitions for the register are shown in table **??**

Reserved	EMSI	ELSI	ETSI	ERSI
31:4	3	2	1	0

- EMSI - Enable Modem status Interrupt
- ELSI - Enable line status Interrupt
- ETSI - Enable Transmitter fifo status interrupt.
- ERSI - Enable Receiver fifo status interrupt.

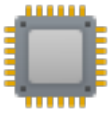
2.1.5 Interrupt Identification Register (IIR) [Read Only]

Address - 0x1008

The Interrupt Identification register contains the priority interrupt identification. The bit definitions for the register are shown ine **??**

Reserved	IID	IP
31:4	3:1	0

- IP - Interrupt pending
- IID - Interrupt ID
  - 011 = Receiver line status
  - 010 = Received Data Available
  - 110 = Character Timeout
  - 001 = Transmit holding register empty
  - 000 = Modem status Interruot



2.1.6 Line Control Register (LCR) [Read Only]

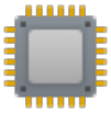
Address - 0x100C

The Line Control register contains the serial communication configuration bits. The bit definitions for the register are shown below :

RSV	DLAB	RSV	Sticky parity	Even parity	Odd parity	Stop Bits	Word lendth
31:8	7	6	5	4	3	2	1: 0

- Word length
  - 00 = 5 bits/char
  - 01 = 6 bits/char
  - 10 = 7 bits/char
  - 11 = 8 bits/char
- Stop Bits
  - 0 = 1 stop bits
  - 1 = 2 stop bits
- Odd parity
  - 0 = deselect odd parity
  - 1 = select odd parity
- Even parity
  - 0 = deselect even parity
  - 1 = select even parity
- Sticky parity
  - 0 = deselect Sticky parity
  - 1 = select Sticky parity
- DLAB
  - 1 = Allows access to the Divisor Latch Registers and reading of the FIFO Control Register.
  - 0 = Allows access to RBR, THR, IER and IIR registers.

RSV - Reserved - Write Zero to all such fields in this document



2.1.7 Modem Control Register - (MCR) [Read Write]

Address - 0x1010 The Modem Control register contains the modem signaling configuration bits

RSV	Loop	Out2	Out1	RTS	DTR
31:5	4	3	2	1	0

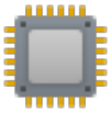
- Loop Back.
  - 1 = Enables loopback.
  - 0 = Disables loopback.
- User Output 2.
  - 1 = Drives OUT2N Low.
  - 0 = Drives OUT2N High.
- User Output 1.
  - 1 = Drives OUT1N Low.
  - 0 = Drives OUT1N High.
- RTS Request To Send.
  - 1 = Drives RTSN Low.
  - 0 = Drives RTSN High.
- DTR Data Terminal Ready.
  - 1 = Drives DTRN Low.
  - 0 = Drives DTRN High

2.1.8 Line Status Register - (LSR) [Read Write]

Address - 0x1014 The Line Status register contains the current status of receiver and transmitter.

RSV	ERR	TEMT	THRE	BI	FE	PE	OE	DR
31:8	7	6	5	4	3	2	1	0

- ERR - error in receiver fifo - parity , framing or break
- TEMT - Transmitter Empty



  - 1 = Transmitter Empty
  - 0 = Transmitter not Empty

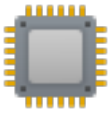
- THRE - Receiver Empty
  - 1 = Receiver Empty
  - 0 = Receiver not Empty
- BI - Break Interrupt - Sin low for entire period
- FE - Framing Error - Stop bit missing.
- PE - Parity error
- OE - Overrun error - Receiver fifo full and character came in.
- DR - Data ready - data in receiver fifo.

2.1.9 Modem Status Register - (MSR) [Read Write]

Address - 0x1018 The Line Status register contains the current status of receiver and transmitter.

RSV	DCD	RI	DSR	CTS	DDCD	TERI	DDS	DCTS
31:8	7	6	5	4	3	2	1	0

- DCD Data Carrier Detect - Complement of DCDN input.
- RI - Ring Indicator - Complement of RIN input.
- DSR - Data Set Ready - Complement of DSRN input.
- CTS - Clear To Send - Complement of CTSN input.
- DDCD - Delta Data Carrier Detect - Change in DCDN after last MSR read.
- TERI - Trailing Edge Ring Indicator - RIN has changed from a Low to a High.
- DDS - Delta Data Set Ready - Change in DSRN after last MSR read.
- DCTS - Delta Clear To Send - Change in CTSN after last MSR read.



2.1.10 Divisor Latch (Least Significant Byte) Register - DLL

Address - 0x1000

The Divisor Latch (Least Significant Byte) register holds the least significant byte of the Baud rate generator counter. This can bre written only if DLAB in LCR is '1'.

RSV	DLL
31:8	7:0

2.1.11 Divisor Latch (Most Significant Byte) Register - DLH

Address - 0x1004

The Divisor Latch (Most Significant Byte) register holds the most significant byte of the Baud rate generator counter.

RSV	DLH
31:8	7:0

Both DLL and DLH can be accessed when DLAB is set to '1'

2.2 GPIO

Base Address - 0x6000_0000 The GPIO is fairly simple. There are only 3 register to configure.

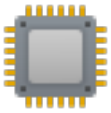
2.2.1 GPO - General Purpose Output - Read/Write

Address - Base Address + 0x0

Default - 0x00000000 The GPO is a 32 bit register with each bit value to control the 32 output.

2.2.2 GPO - General Purpose Input - Read Only

Address - Base Address + 0x4 The GPI is a 32 bit register with each bit value representing the value on the 32 GPIO.



2.2.3 GPO - General Purpose Direction - GPD - Read/Write

Address - Base Address + 0xC

Default - 0xFF000000 The GPD register is used to configure the GPIO pin's as output or input. A value of '1' set the corresponding pin as input and '0' set the corresponding pin as output. On reset the upper 8 bits are configured as Inputs and rest as outputs.

2.3 PWM

Base Address - 0x9000_0000 There are 2 PWM out availabe in the system, each of which can be configured independently. The corresponding Register space are :

2.3.1 PWMENA - PWM Enable Register - Read/Write

Address - Base Address + 0x0

Default - 0x00000000

RSV	PWMEN1	PWMEN0
31:2	1	0

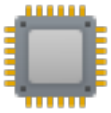
- PWMEN0
  - 1 = Enable PWM 0
  - 0 = Disable PWM 0
- PWMEN1
  - 1 = Enable PWM 1
  - 0 = Disable PWM 1

2.3.2 PWM0CMPL - PWM0 Compare Low - Read/Write

Address - Base Address + 0x4

Default - 0x00000000

The PWM0CMP is 32 bit register holding the value for PWM0 low period. The PWM0 is low for PWM0CMPL * SYS_CLK_PERIOD.



2.3.3 PWM0CMPH - PWM0 Compare high - Read/Write

Address - Base Address + 0xc

Default - 0x00000000

The PWM0CMP is 32 bit register holding the value for PWM0 high period. The PWM0 is high for PWM0CMPH * SYS_CLK_PERIOD.

2.3.4 PWM1CMPL - PWM1 Compare Low - Read/Write

Address - Base Address + 0x10

Default - 0x00000000

The PWM1CMP is 32 bit register holding the value for PWM1 low period. The PWM1 is low for PWM1CMPL * SYS_CLK_PERIOD.

2.3.5 PWM1CMPH - PWM1 Compare high - Read/Write

Address - Base Address + 0x14

Default - 0x00000000

The PWM1CMP is 32 bit register holding the value for PWM1 high period. The PWM1 is high for PWM1CMPH * SYS_CLK_PERIOD.

2.4 PLIC

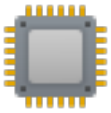
TODO

2.5 I2C

TODO

2.6 I2S

TODO



3

Software Environment

3.1 Transmission/Reception

For transmitting a UART character 8 bit/character 1 stop bit and no parity bit , perform the following operations :-

- Write 0x0000_0003 to line control register (0x000c) .
- Write data in THR (0x0000).
- Check TEMT (6th bit in LSR (0x0014). If TEMT is '1' , repeat this step else write character to THR.

For receiving a UART character 8 bit/character 1 stop bit and no parity bit , perform the following operations :-

- Write 0x0000_0003 to line control register (0x000c) .
- Check THRE (5th bit in LSR (0x0014). If TEMT is '0' , read from RBR (address 0x0000).