

APPLICATION NOTE

Product Name	AI7933CLD
Version	C
Doc No	902-13901
Date	2023/02/20



Document History

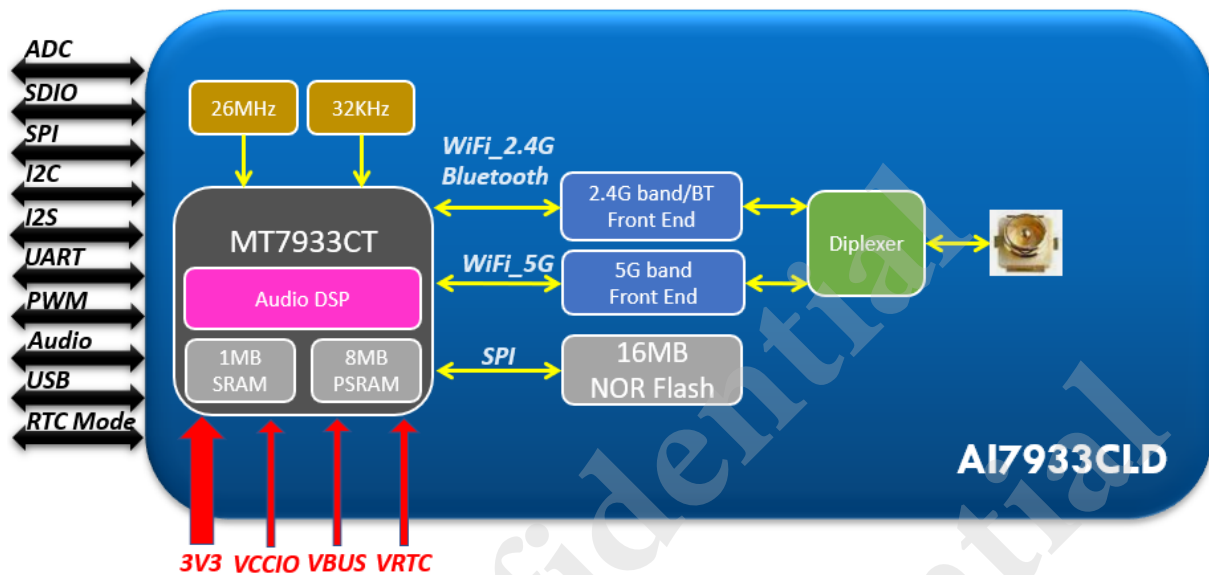
Date	Revised Contents	Revised By	Version
2022/07/15	Initial Version	Ivan	A
2022/11/10	Update PWM Patterns	Ivan	B
2023/02/20	Modify Recommended Operating Range Modify Bootstrap Function	Ivan	C

INDEX

1	Block Diagram	4
1.1	Specification.....	4
1.2	Recommended Operating Range	4
2	Functional Description	5
2.1	System Initialization	5
2.1.1	Chip Power-On.....	5
2.1.2	Bootstrap Function	6
2.1.3	System Boot Sequence	7
2.2	Peripherals	7
2.2.1	Auxiliary ADC Function	7
2.2.1.1	Functional Description	7
2.2.1.2	Features and IO	8
2.2.2	SPI Master Controller	9
2.2.2.1	Functional Description	9
2.2.2.2	Pin Description	9
2.2.2.3	Transmission Formats.....	10
2.2.2.4	Features.....	10
2.2.3	SPI Slave Controller.....	12
2.2.3.1	Features.....	12
2.2.3.2	Pin Description	13
2.2.4	SDIO Slave.....	14
2.2.4.1	Functional Description.....	14
2.2.4.2	Features.....	14
2.2.4.3	Block Diagram.....	15
2.2.4.4	Functions Description.....	15
2.2.4.5	Pin Description	16
2.2.4.6	SDIO Timing Waveform (3.3V).....	17
2.2.5	I2C.....	20
2.2.5.1	Introduction.....	20
2.2.5.2	Features.....	20
2.2.5.3	Block Diagram.....	21
2.2.6	UART	22
2.2.6.1	Introduction.....	22
2.2.6.2	Features.....	22
2.2.7	PWM.....	23

2.2.7.1	Functions Description.....	23
2.2.8	USB	24
2.2.8.1	USB Host Controller.....	24
2.2.8.1.1	Introduction.....	24
2.2.8.1.2	Features.....	24
2.2.8.1.3	Block Diagram.....	24
2.2.9	DMA.....	25
2.2.9.1	Command Queue (CQ) DMA	25
2.2.9.2	Application Processor (AP) DMA	25
2.2.9.2.1	Peripheral DMA	26
2.2.9.2.2	Virtual FIFO.....	27
2.2.9.2.3	APDMA Channels and Priority Control.....	28
2.2.10	GPT.....	29
2.2.10.1	Introduction.....	29
2.2.10.2	Features.....	29
2.2.10.3	Block Diagram.....	30
2.2.10.4	Theory of Operations	30
2.2.11	WDT.....	31
2.2.11.1	Introduction.....	31
2.2.11.2	Functional Description	31
2.2.11.3	Event Notification for CM33.....	31
2.2.12	GPIO	32
2.2.12.1	Overview	32
2.2.12.2	I/O Pull up or Down Control Truth Table.....	33
2.3	Audio System	34
2.3.1	Audio Digital Signal Processor (DSP)	34
2.3.1.1	Introduction.....	34
2.3.1.2	Features.....	34
2.3.2	Audio Channels.....	35
2.3.2.1	Functional Description.....	35
2.4	Interrupt	36
2.4.1	Introduction.....	36
2.4.2	Block Diagram.....	36
2.4.3	Interrupt Table.....	37
3	Recommend Footprint (Typ.).....	42
3.1	Recommend Stencil.....	43
4	Other Information	43

1 Block Diagram



1.1 Specification

AI7933CLD	
Chipset	MT7933CT (Wi-Fi 6 + BT5.2 + Hi-Fi 4 DSP)
Core	ARM Cortex-M33 with FPU
FPU Clock Speed	300MHz
SRAM	1MB
PSRAM for Applications	8MB
NOR Flash	16MB
Antenna connector	I-PEX

1.2 Recommended Operating Range

Temperature & Humidity Range

Temperature	Operating : -40°C ~ +85°C Storage : -40°C ~ +105°C
Humidity	Operating : 10 ~ 95% (Non-Condensing) Storage : 5 ~ 95% (Non-Condensing)

Voltage Range

Symbol	Parameter	Min.	Typ.	Max.	Unit	Input / Output
BASE_3V3	3.3V Supply Voltage	2.97	3.3	3.63	V	I
RTC_3V3	RTC Supply Voltage	2.5		4.2	V	I
VCCIO_L	Internal Flash Supply Voltage & SDIO Domain	2.97	3.3	3.63	V	I
IC_VCCIO	GPIO Domain	1.62 2.97	1.8 3.3	1.98 3.63	V	I
PHYLDO_OUT	1.8V Output	1.62	1.8	1.98	V	O
MIC_MICBIAS0	MICBIAS Output	1.8	1.85	2.2	V	O

2 Functional Description

2.1 System Initialization

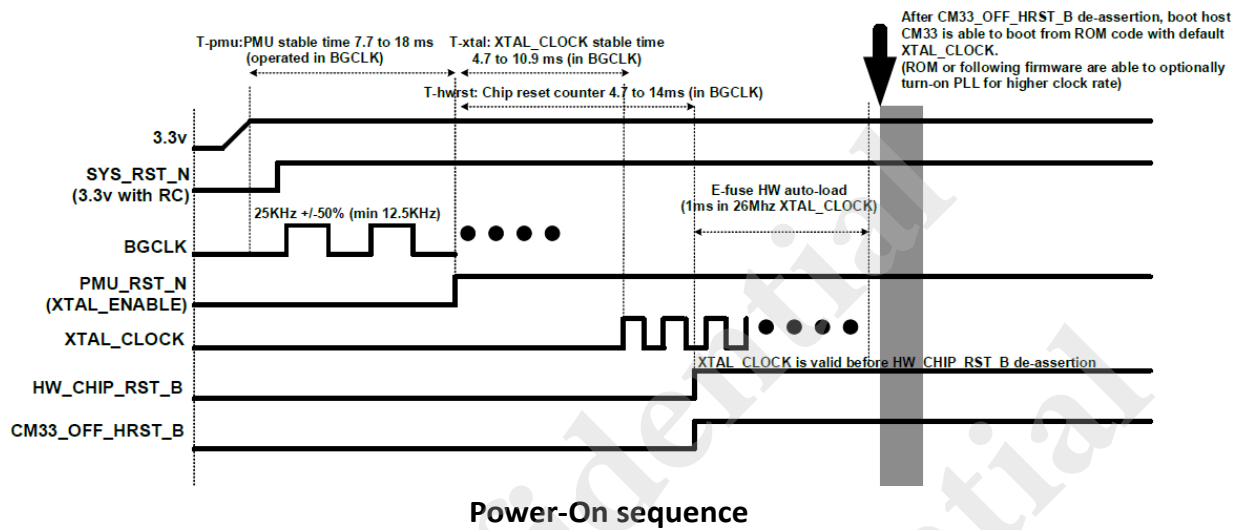
The initialization of the MT7933CT chip system inside the AI7933CLD is described in two sections here. The chip hardware power-on sequence is described in Section 2.1.1. The system boot up sequence after the secure boot master CM33 takes over the control is described in Section 2.1.3

2.1.1 Chip Power-On

This section explains the chip power-on sequence from 3.3V power supply getting stable to the fundamental chip hardware reset de-assertion (HW_CHIP_RST_B). After this fundamental chip hardware reset de-assertion, secure boot master CM33 takes over the system and boots from its ROM code. After 3.3V power is stable, the BGCLK (band gap clock) in PMU starts to generate the 25-kHz fundamental clock, the first MT7933CT clock after power on. However, there is +/- 50% uncertainty out of this BGCLK such that this clock is only used for crystal clock control circuit. The power-on sequence is shown in the diagram below and is described in the following steps.

- Step 1: 3.3V becomes stable and PMU BGCLK starts to work for PMU initialization
- Step 2: After T-pmu (7.7 to 18ms), PMU will be valid and de-assert PMU_RST_N
- Step 3: After PMU_RST_N de-assertion, XTAL control circuit will start to work
- Step 4: After T-xtal (4.7 to 10.9ms), the crystal clock (XTAL_CLOCK) will be valid
- Step 5: T-hwrst (4.7 to 14ms) guarantees that the XTAL_CLOCK is valid before HW_CHIP_RST_B
- Step 6: After HW_CHIP_RST_B, IO setting CR will be auto-loaded from eFuse (8ms in 26 MHz)
- Step 7: Secure boot master CM33 starts to boot from its ROM

Note that there is a dedicated hardware input pin SYS_RST_N, which is also able to hold the chip reset state. However, this reset will always be de-asserted before PMU_RST_N if the application circuit on the PCB connects this pin to some RC circuit from 3.3V power supply.



2.1.2 Bootstrap Function

The section describes the bootstrap function.

The chip modes are sensed from the device pin during power up. After chip reset, the pull configurations are stored in a register and the settings determine the device operation mode.

AI7933CLD Strapping pins and modes

Pin name	Pin description	Pin description	Description
GPIO_B_0	GPIO12	Chip Mode Sel	1: Normal mode: Connect 10kΩ to VCCIO_B 0: RSVD
GPIO_B_2	GPIO14	Download and Normal boot Sel	00: CM33 UART download mode 01: CM33 Flash normal boot mode 10: CM33 SDIO download mode 11: CM33 USB download mode e.g. 01=GPIO_B_2 connects 10kΩ to GND, GPIO_B_12 connects 10kΩ to VCCIO_B
GPIO_B_12	GPIO24		
SF_QPI_CS	GPIO1	XTAL mode Sel	0: XTAL buffer mode 1: XTAL normal mode

Pins GPIO1, GPIO12, GPIO14, and GPIO24 are used for bootstrap. The system design should follow the following guidelines:

- Those pins shall not be used as input functions because the signals from other devices might affect the values sensed.
- Those pins shall not be used as an open-drain function because the pull-up resistor would affect the values sensed.

- AI7933CLD Module Pin-Out does not include GPIO1, which has been set to XTAL Normal Mode in the module.
- Update FW default mode: CM33 UART download mode.
 - CM33 UART TX = Pin_A8 / GPIO_T_9 (KPCOL_0)
 - CM33 UART RX = Pin_B8 / GPIO_T_7 (KPROW_1)

2.1.3 System Boot Sequence

The system boot sequence after secure boot master CM33 reset de-assertion is described as below. CM33 will boot up first from BROM, and BROM code will verify security of bootloader and execute code directly on flash by XiP. Besides the boot loader, the flash code also contains the RTOS image for CM33, and the driver and firmware necessary for Wi-Fi, Bluetooth and Audio subsystems. By executing the boot loader, CM33 will verify the security of other flash codes, then jump to RTOS entry point on flash, and fetch the corresponding driver and firmware to enable Wi-Fi, Bluetooth, and Audio subsystems.

The chip initial power state is by default set to ON for Cortex-M33 platform, crypto engine, infra bus and peripherals. But the chip initial state is by default set to OFF for Wi-Fi, Bluetooth and Audio subsystems. Thus, the boot sequence will take care of the power on procedure for those subsystems before enabling them.

2.2 Peripherals

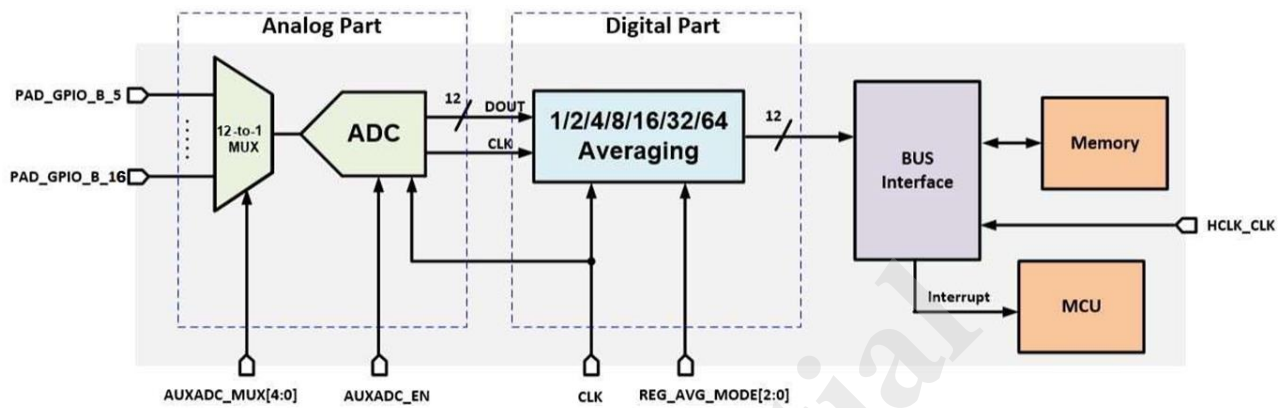
2.2.1 Auxiliary ADC Function

2.2.1.1 Functional Description

The AI7933CLD features one auxiliary ADC function. The ADC function contains a 12-channel analog switch, a single-end input asynchronous 12-bit SAR (Successive Approximation Register) ADC, dithering function and a digital averaging function.

The digital averaging function can perform on-the-fly averaging function of 1/2/4/8/16/32/64 points.

The ADC features the dithering function to enhance the DNL performance



Auxiliary ADC block diagram.

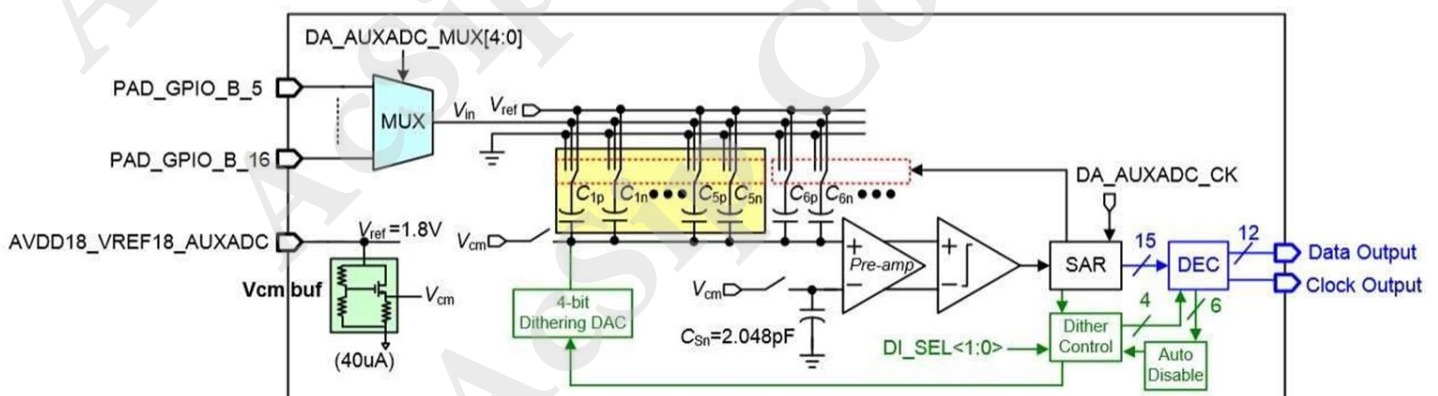
2.2.1.2 Features and IO

The features of auxiliary ADC are listed below:

- Input channel number: 12 channels
- Sampling and output data rate: 2MS/s (default)
- DNL without dithering and averaging: $< \pm 2\text{LSB}$
- DNL with dithering and averaging: $< \pm 1\text{LSB}$
- Dithering function: 16 levels with step size of 4LSB

The IOs of auxiliary ADC can be set as either analog IO for ADC function or digital IO for GPIO function:

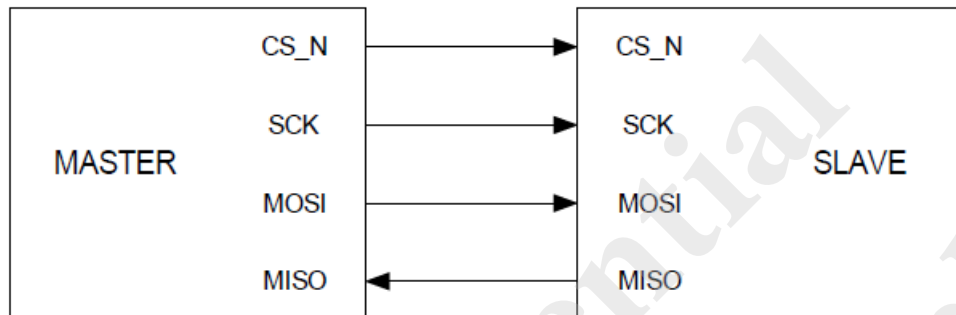
- Analog Mode: Used for ADC application. Input voltage is 1V.



Auxiliary ADC analog IP block diagram

2.2.2 SPI Master Controller

2.2.2.1 Functional Description



Pin connection between SPI master and SPI slave

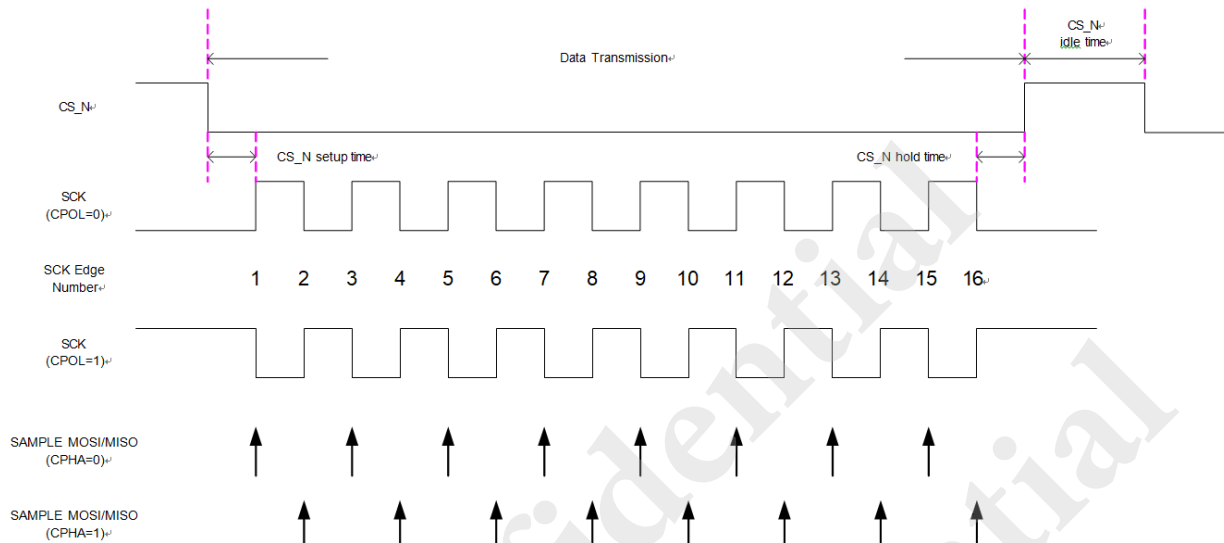
The SPI interface is a bit-serial, four-pin transmission protocol. The above figure is an example of the connection between the SPI master and SPI slave. The SPI interface controller is a master responsible for the data transmission with the slave.

2.2.2.2 Pin Description

SPI controller interface

Signal name	Type	Description
CS_N	O	Low active chip selection signal
SCK	O	The (bit) serial clock
MOSI	O	Data signal from master output to slave input
MISO	I	Data signal from slave output to master input

2.2.2.3 Transmission Formats



SPI transmission formats

The above figure shows the waveform during the SPI transmission. The low active CS_N determines the start point and end point of one transaction. The CS_N setup time, hold time and idle time are also depicted.

CPOL defines the clock polarity in the transmission. Two types of polarity can be adopted, i.e. polarity 0 and polarity 1. The above figure is an example of both clock polarities (CPOL).

CPHA defines the legal timing to sample MOSI and MISO. Two different methods can be adopted.

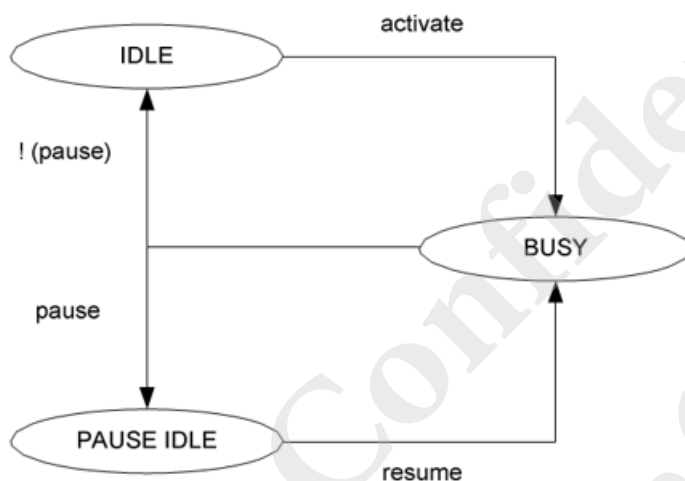
2.2.2.4 Features

The features of the SPI controller (master) are:

- Configurable CS_N setup time, hold time and idle time
- Programmable SCK high time and low time
- Configurable transmitting and receiving bit order
- Two configurable modes for the source of the data to be transmitted. 1) In Tx DMA mode, the SPI controller automatically fetches the transmitted data (to be put on the MOSI line) from memory. 2) In Tx FIFO mode, the data to be transmitted on the MOSI line are written to FIFO before the start of the transaction.
- Two configurable modes for destination of the data to be received. 1) In Rx DMA mode, the SPI controller automatically stores the received data (from MISO line) to memory. 2) In Rx FIFO mode, the received data keep being in Rx FIFO of the SPI controller. The processor must read

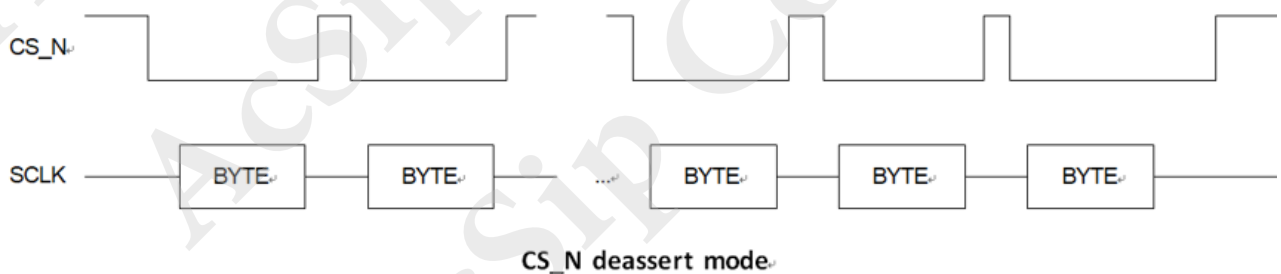
back the data by itself.

- Adjustable endian order from/to memory system
- Programmable byte length for transmission
- Unlimited length for transmission. This is achieved by the operation of PAUSE mode. In PAUSE mode, the CS_N signal will keep being active (low) after the transmission. At this time, the SPI controller is in PAUSE_IDLE state, ready to receive the resume command. The state transition is shown in the figure below.



Operation flow with or without PAUSE mode.

- Configurable option to control CS_N deassert between byte transfers. The controller supports a special transmission format called CS_N deassert mode. The figure below illustrates the waveform in this transmission format.

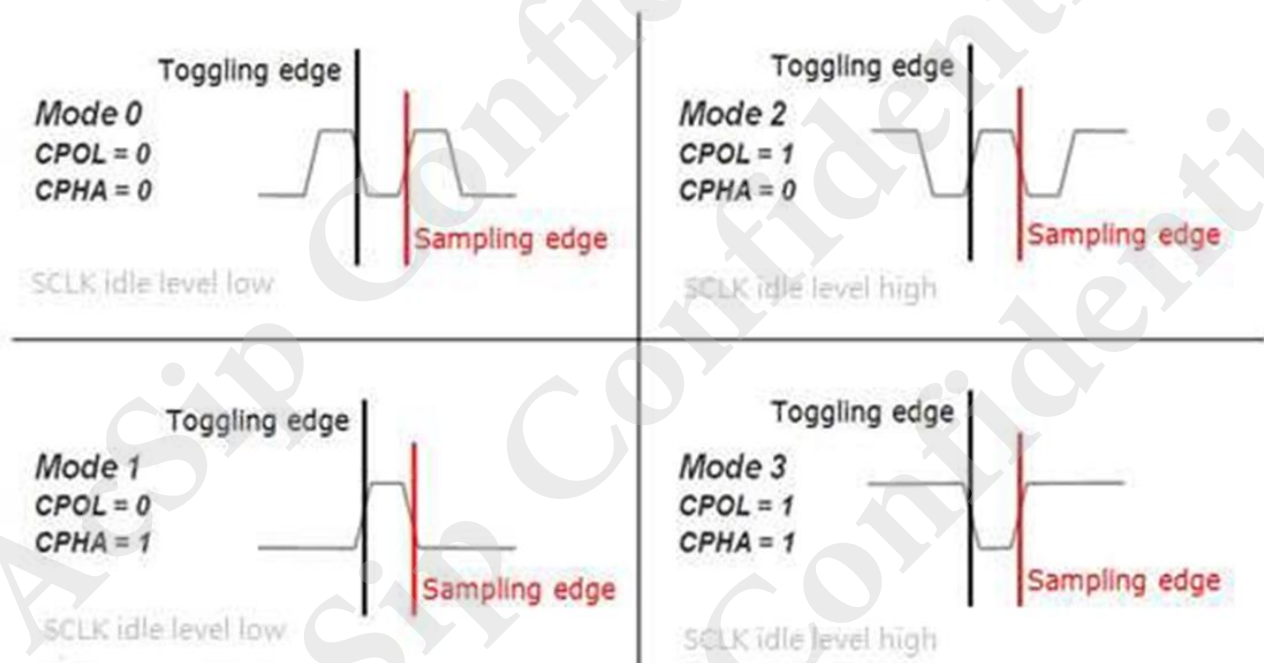


2.2.3 SPI Slave Controller

2.2.3.1 Features

The features of the SPI controller (slave) are listed below:

- The supported SPI_CLK is up to 25MHz.
- Configurable bit transmitting and receiving order: Two options of bit order – MSB or LSB first
- Four communication modes are available (MODE 0, 1, 2, 3) – that basically define the SCLK edge on which the MISO line toggles and the slave samples the MOSI line. They also define the SCLK signal steady level (namely the clock level, high or low, when the clock is not active). Each mode is formally defined with a pair of parameters called “clock polarity” (CPOL) and “clock phase” (CPHA).



Four communication modes waveform

- Enable/Disable Transmit and Receive mode
- Default Tx FIFO data (default is 0x00): If Tx FIFO is empty and the SPI Master wants to get the data from the SPI slave, then the SPI Slave will output configurable constant byte value on MISO, and the default value is 0x00.
- Rx/Tx FIFO data status: There will be Rx/Tx FIFO pointer and the number of bytes transmitted/received in status register. These can be read for status checking.
- Interrupt support: There will be Rx full interrupt and transfer done interrupt for indication.

- Support PIO mode and DMA mode transfer: Both DMA and PIO mode are supported on SPI slave Tx/Rx channel. (Note: Under DMA mode, the value of SPIS_TX_SRC (SPISn Base address+0x0020)[31:0] and SPIS_RX_DST(SPISn Base address+0x001C)[31:0] must be 4-byte aligned)
- Programmable byte length for transmission: The length of Tx DMA can be programmable from 1 to 1 M bytes.
- Supported Rx DMA byte length is from 1 to (1 M-4) bytes.
- Each FIFO depth of Tx/Rx is 32 x 4 byte

2.2.3.2 Pin Description

SPI slave controller interface

Signal Name	Type	Description
CS_N	INPUT	Low active chip selection signal
SCK	INPUT	The (bit) serial clock
MOSI	INPUT	Data signal from master output to slave input
MISO	OUTPUT	Data signal from slave output to master input

2.2.4 SDIO Slave

2.2.4.1 Functional Description

The SD Input/Output (SDIO) card is based on and compatible with the SD memory card. The controller fully supports the SD memory card bus protocol as defined in SD Memory Card Specification Part 1 Physical Layer Specification version 2.0 and SDIO Specification version 2.0.

SDIO provides high-speed data IO with low power consumption. SDIO module provides an SDIO2.0 card interface connected to the host and can support multiple speed modes including default speed mode and high speed mode.

2.2.4.2 Features

SDIO 2.0 basic features

- 1-bit and 4-bit SD data transfer modes
- Default mode: Variable clock rate 0-25 MHz, up to 12.5 MB/sec interface speed (using 4 parallel data lines)
- High-Speed mode: Variable clock rate 0-50 MHz, up to 25 MB/sec interface speed (using 4 data lines)

CR and data port access

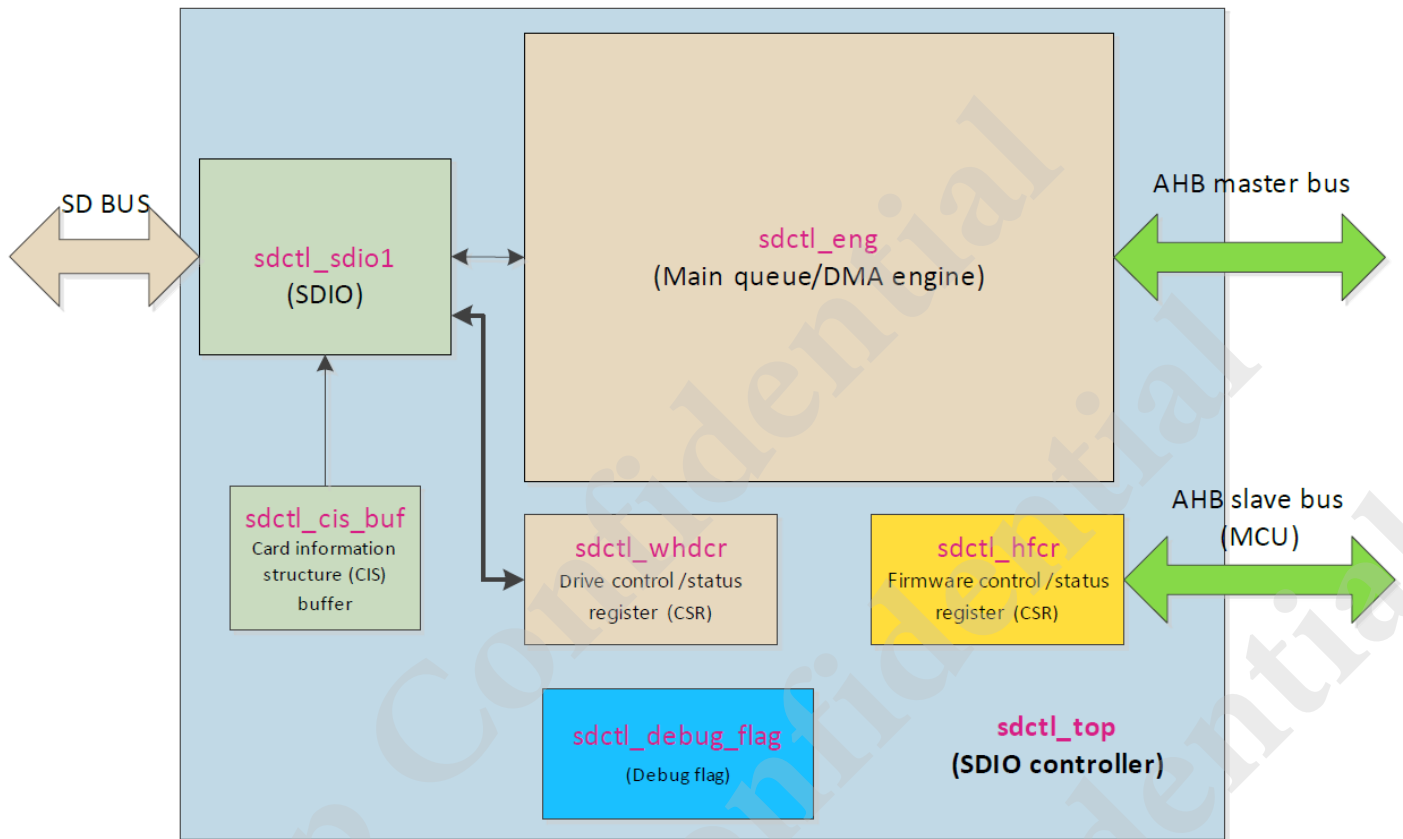
- Supports control register (CR) port single read/write access (AHB slave)
- Supports data port single and burst read/write access (AHB master)

DMA function

- One TX channel and two Rx channels
- Moves TX data from HIF buffer to SYSRAM, TCM
- Moves RX data or firmware prepared data from SYSRAM, TCM to HIF buffer

2.2.4.3 Block Diagram

The block diagram of the SDIO controller

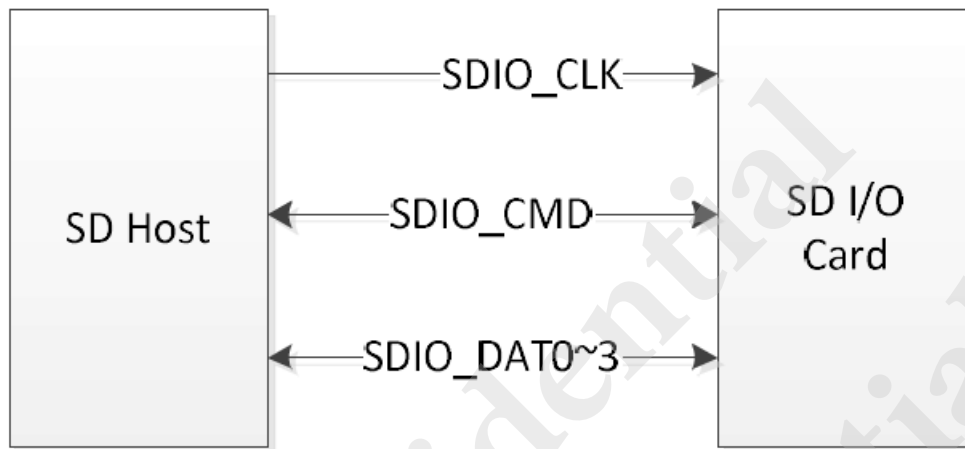


SDIO controller block diagram

2.2.4.4 Functions Description

From the external view, the SDIO interface mainly includes the SD bus and AHB master and slave. The AHB master is used for DMA operations and the AHB slave is used for register access from the MCU. The SD bus provides an interface for SD specification.

2.2.4.5 Pin Description

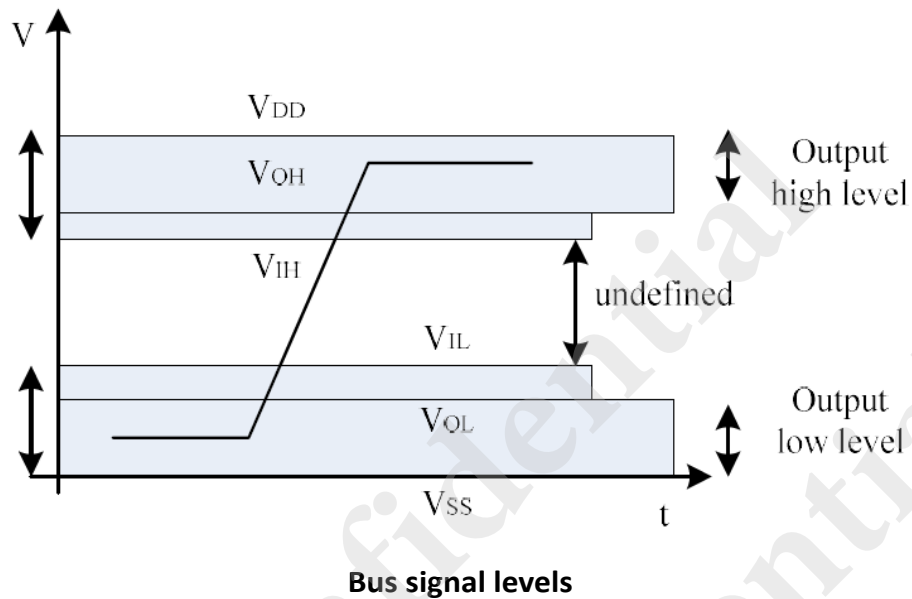


Signal connections to 4-bit SDIO cards

SDIO pin definitions

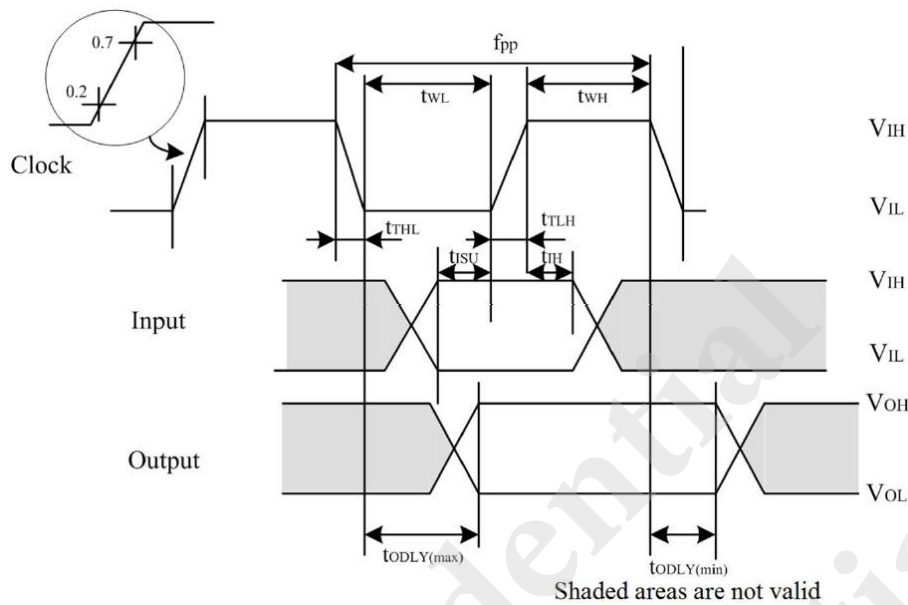
Pin	Name	SD 4-bit mode		SD 1-bit mode	
1	SDIO_DAT3	DAT[3]	Data line3	N/C	Not used
2	SDIO_CMD	CMD	Command line	CMD	Command line
3	VSS1	VSS1	Ground	VSS1	Ground
4	VDD	VDD	Supply voltage	VDD	Supply voltage
5	SDIO_CLK	CLK	Clock	CLK	Clock
6	VSS2	VSS2	Ground	VSS2	Ground
7	SDIO_DAT0	DAT[0]	Data line0	DATA	Data line
8	SDIO_DAT1	DAT[1]	Data line1 or interrupt	IRQ	Interrupt
9	SDIO_DAT2	DAT[2]	Data line2	RW	Not used

2.2.4.6 SDIO Timing Waveform (3.3V)



Bus signal voltage

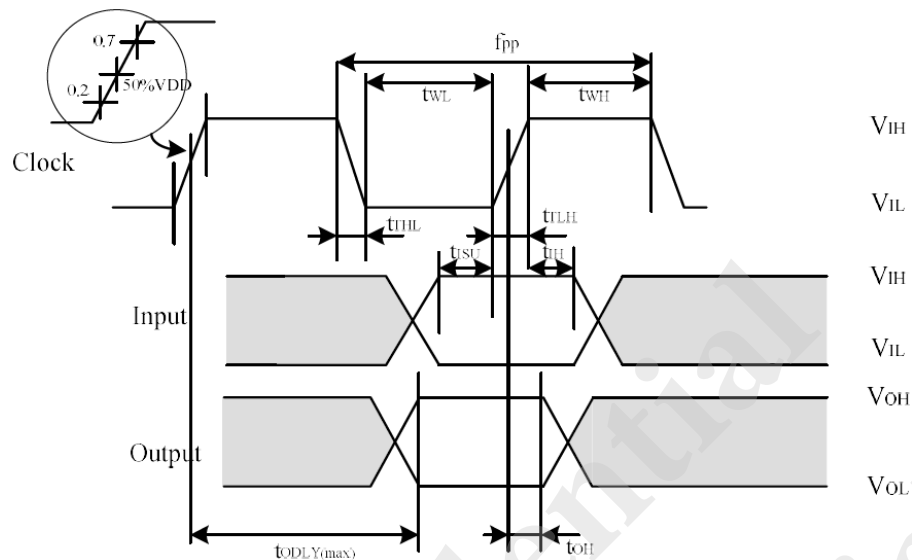
Parameter	Symbol	Min.	Max.	Unit	Conditions
Output High Voltage	VOH	0.75*VDD		V	IOH=-2mA VDD min
Output Low Voltage	VOL		0.125*VDD	V	IOL = 2mA VDD min
Input High Voltage	VIH	0.625*VDD	VDD+0.3	V	
Input Low Voltage	VIL	Vss-0.3	0.25*VDD	V	



Bus timing diagram (default)

Bus timing parameter values (default)

Parameter	Symbol	Minimum	Maximum	Unit	Remark
Clock CLK (All values are referred to min (VIH) and max (VIL))					
Clock frequency data transfer mode	fPP	0	25	MHz	CCARD ≤ 10 pF (1 card)
Clock frequency identification mode	fOD	0/100	400	kHz	CCARD ≤ 10 pF (1 card)
Clock low time	tWL	10		ns	CCARD ≤ 10 pF (1 card)
Clock high time	tWH	10		ns	CCARD ≤ 10 pF (1 card)
Clock rise time	tTLH		10	ns	CCARD ≤ 10 pF (1 card)
Clock fall time	tTHL		10	ns	CCARD ≤ 10 pF (1 card)
Inputs CMD, DAT (referred to CLK)					
Input set-up time	tISU	5		ns	CCARD ≤ 10 pF (1 card)
Input hold time	tIH	5		ns	CCARD ≤ 10 pF (1 card)
Outputs CMD, DAT (referred to CLK)					
Output delay time during data transfer mode	tOLDY	0	14	ns	CL ≤ 40 pF (1 card)
Output delay time during identification mode	tOLDY	0	50	ns	CL ≤ 40 pF (1 card)



High-speed timing diagram

High-speed timing parameter values

Parameter	Symbol	Minimum	Maximum	Unit	Remark
Clock CLK (All values are referred to min (V_{IH}) and max (V_{IL}))					
Clock frequency data transfer mode	f_{PP}	0	50	MHz	CCARD ≤ 10 pF (1 card)
Clock low time	t_{WL}	7		ns	CCARD ≤ 10 pF (1 card)
Clock high time	t_{WH}	7		ns	CCARD ≤ 10 pF (1 card)
Clock rise time	t_{TLH}		3	ns	CCARD ≤ 10 pF (1 card)
Clock fall time	t_{THL}		3	ns	CCARD ≤ 10 pF (1 card)
Inputs CMD, DAT (referred to CLK)					
Input set-up time	t_{ISU}	6		ns	CCARD ≤ 10 pF (1 card)
Input hold time	t_{IH}	2		ns	CCARD ≤ 10 pF (1 card)
Outputs CMD, DAT (referred to CLK)					
Output delay time during data transfer mode	t_{OLDY}		14	ns	CL ≤ 40 pF (1 card)
Output hold time	t_{OH}	2.5		ns	CL ≥ 40 pF (1 card)
Total system capacitance for each line (1)	CL		40	pF	1 card

(1) In order to satisfy the serving time, the host shall drive only one card.

2.2.5 I2C

2.2.5.1 Introduction

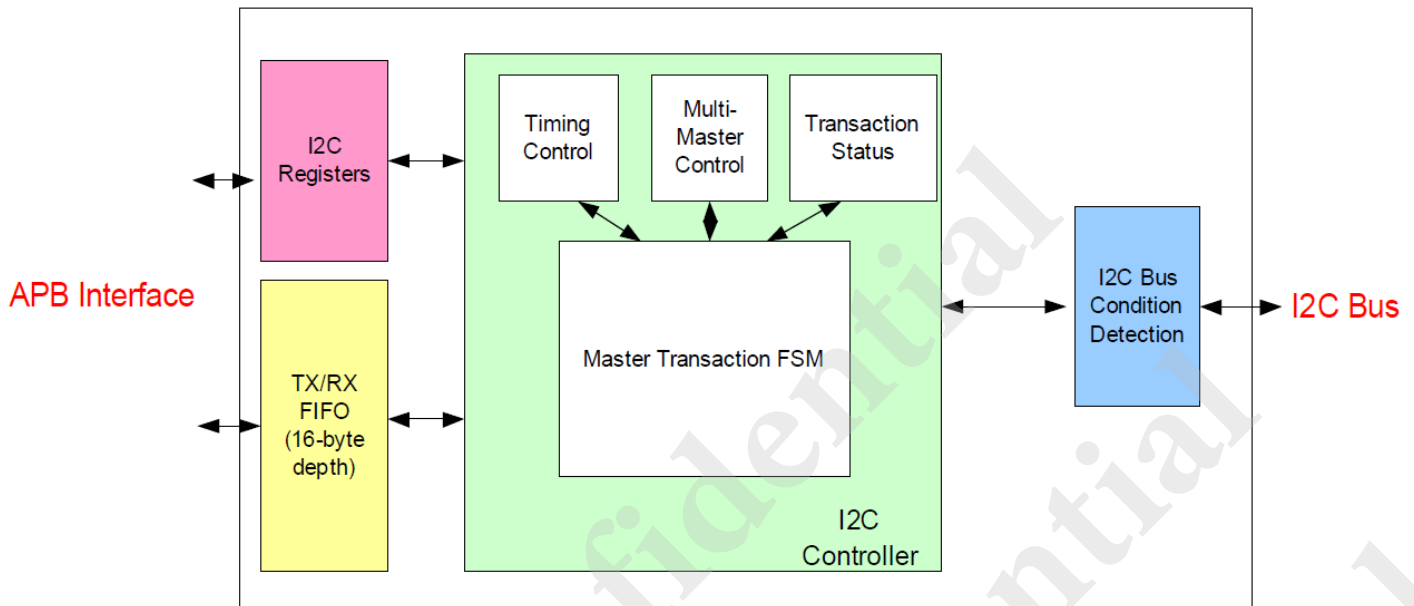
There are two I2C master channels in the AI7933CLD with the same HW architecture. I2C is a two-wire serial interface with two signals, SCL and SDA. SCL is a clock signal driven by the master. SDA is a bi-directional data signal that can be driven either by the master or by the slave. This generic controller supports the master role and conforms to the I2C specification.

2.2.5.2 Features

The main supported features of I2C Master are as follows:

- I2C compliant master mode operation
- Adjustable clock speed for Fast-mode Plus
- 7-bit address
- Clock stretching feature
- START/STOP/repeated START conditions
- I2C_FIFO mode
- DMA transfer mode
- Multi-write per transfer
- Multi-read per transfer
- Multi-transfer per transaction
- Combined format transfer with length change capability
- Multi-transfer with repeated START condition

2.2.5.3 Block Diagram



Block diagram of I2C

- I2C Registers: I2C configuration and status registers.
- Tx/Rx FIFO: Store the data to be sent to I2C slave or the data received from I2C slave.
- Master Transaction FSM: I2C master finite-state machine used to indicate the current transfer stage.
- Timing Control: Control the frequencies of SCL according to the configuration of CLOCK_DIV, SAMPLE_CNT_DIV and STEP_CNT_DIV.
- Multi-Master Control: Execute arbitration when multiple masters exist on the I2C bus.
- Transaction Status: Record the number of bytes that has been transferred and the number of transfers that has been done. It can be used to judge if all transactions have been completed.
- I2C Bus Condition Detection: Detect START/STOP/repeated START conditions and clock stretching on the I2C bus.

2.2.6 UART

2.2.6.1 Introduction

The UART controller provides full duplex serial communication channels between the MT7933CT chip and external devices. The UART controller has M16C450 and M16550A operation modes, which are compatible with a range of standard software drivers. The extensions are designed to be broadly software compatible with 16550A variants, but certain areas offer no consensus.

In common with the M16550A, the UART controller supports word lengths from 5 to 8 bits, an optional parity bit and one or two stop bits, and this word length is fully programmable by a CPU interface. A 16-bit programmable baud rate generator and an 8-bit scratch register are included, along with separate transmission and received FIFOs. Eight modem control lines and a diagnostic loop-back mode are provided. The UART_TOP also includes two DMA handshake lines which are used to indicate when the FIFOs are ready to transfer data to the CPU. Interrupts can be generated from any of the several sources.

After hardware reset, the UART controller is in M16C450 mode. Its FIFOs can be enabled and the UART controller can enter M16550A mode. The UART controller adds further functionality beyond M16550A mode. Each of the extended functions can be selected individually under software control.

2.2.6.2 Features

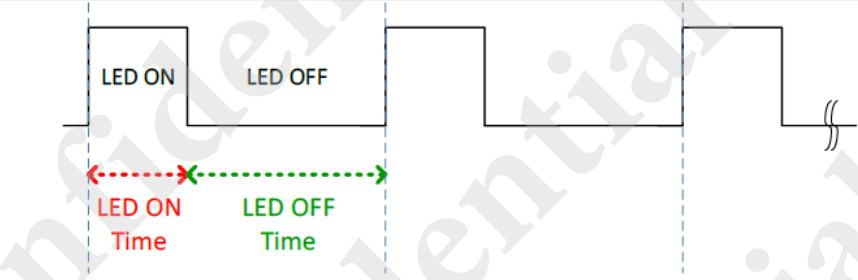
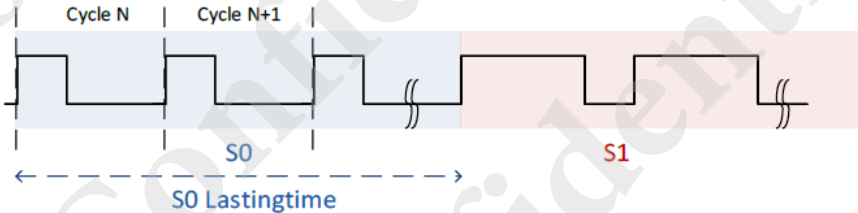
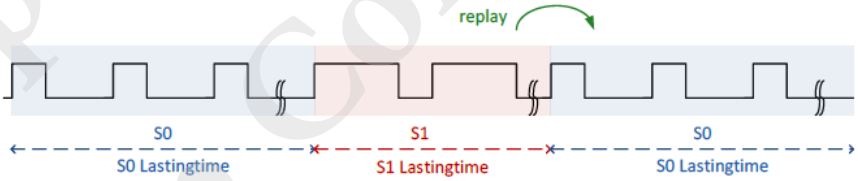
- Provide 4 channels of UART controller
- UART_TOP0, UART_TOP1 and CM33_UART are 4-pin (TX, RX, CTS, RTS) UART_TOP channel
- DSP_UART is a 2-pin (TX, RX) UART_TOP channel
- Support both M16C450 and M16550A operation modes
- Compatible with standard software drivers
- Transfer system: Asynchronous
- Data length: 5 to 8 bits
- Hardware flow control: CTS/RTS-based automatic transmission and reception of control
- Software flow control: Use special character XON/XOFF to do software flow control
- Baud rate is programmable from 300 bps to 3 Mbps
- Baud rate error: Less than 0.25 %
- Interrupt request: Receive interrupts/transmit interrupts
- Data transfer: DMA (Transmit/Receive) transfer is supported

2.2.7 PWM

2.2.7.1 Functions Description

The AI7933CLD features 12 generic PWMs to generate pulse sequences with programmable frequency and duration for LCD, vibrators, and other devices.

The PMU features three configurable pattern options.

Mode	Description	Waveform
1	Basic PWM: LED ON time (duration) and LED OFF time (duration) are configurable.	
2	Two-State PWM: There are two configurable states (S0 and S1) for PWM LED.	
3	Two-State replay mode: Users can set replay mode with specified S1_Lasting_Time. PWM LED would act as [S0 -> S1 -> S0 -> S1 -> S0] with period time of (S0_Lasting_Time + S1_Lasting_Time)	

PWM Patterns

2.2.8 USB

2.2.8.1 USB Host Controller

2.2.8.1.1 Introduction

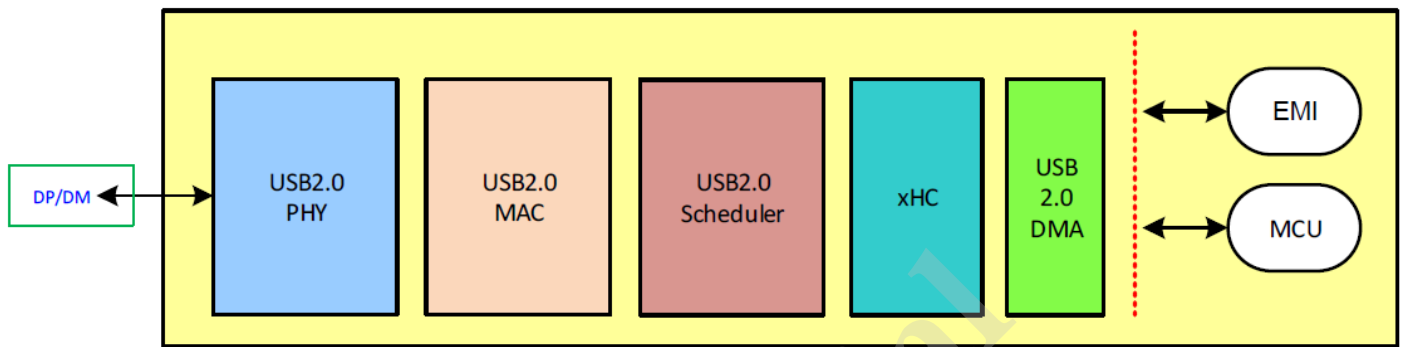
The AI7933CLD features a USB host controller which provides one USB 2.0 host port. The operation model of the host functional blocks conforms to eXtensible Host Controller Interface (xHCI) for Universal Serial Bus (USB) specification.

2.2.8.1.2 Features

- The hardware supports USB 2.0 with LS (Low-speed) 1.5 Mbps/FS (Full-speed) 12 Mbps/HS (High-speed) 480 Mbps
- Embedded USB 2.0 PHY with 16-bit/30-MHz UTMI interface
- AHB interface for register access
- AXI interface for DMA access
- xHCI-based host controller
- LPM (Lower Power Management) on the USB 2.0 port
- Dedicated DMA channel for USB 2.0 data transfer
- Supports all USB compliant data transfer types with control/bulk/interrupt/isochronous transfer and split transactions
- Compatible with USB 2.0 hub
- Supports up to 15 devices
- Supports up to 64 endpoints

2.2.8.1.3 Block Diagram

The architecture of the USB host. It has one port configured as USB 2.0 host mode. All the resources of endpoint and device are handled by the xHCI controller. The software could dynamically allocate resources for different ports and turn on or off each port separately. The xHCI controller controls the exchange of data and PHY, and controls DMA to write data to or read data from EMI.



USB host architecture

2.2.9 DMA

Direct memory access (DMA) is used to transfer data between memory and memory as well as memory and peripherals without MCU interventions.

2.2.9.1 Command Queue (CQ) DMA

The purpose of CQ_DMA is to perform data transfer between memory and memory without CPU interventions.

The supported features of CQDMA are as below:

- Up to three channels of simultaneous data transfer are supported.
- Comply with the system bus (AXI).
- Two out-standings for three channels
- Source or destination address fix mode
- Source or destination address-wrapping mode
- Source and destination address-increment mode
- Fix-pattern mode
- Bandwidth limiter
- TrustZone
- Round-Robin (RR) for scheduling scheme

2.2.9.2 Application Processor (AP) DMA

The purpose of APDMA is to perform data transfer between memory and peripherals.

The supported features of APDMA are as below:

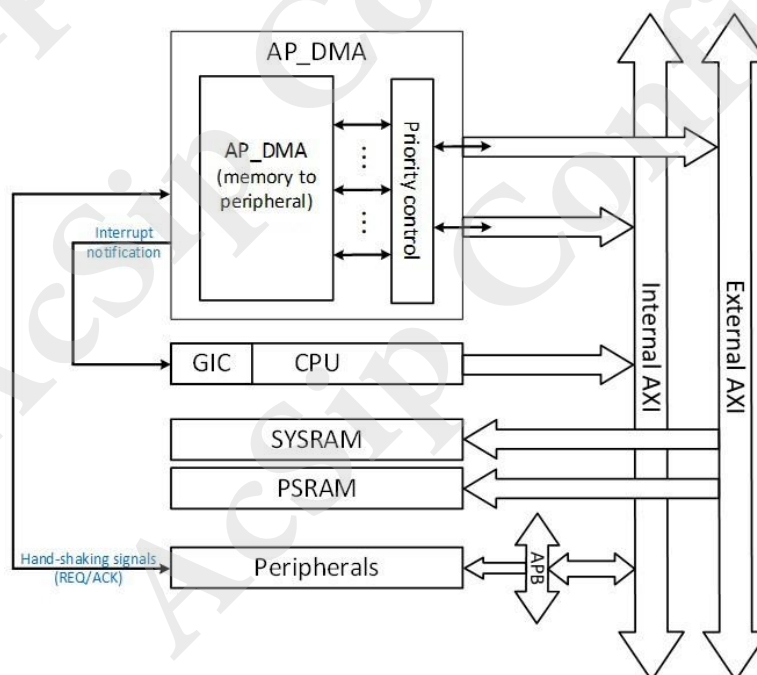
- Up to 9 channels of simultaneous data transfer are supported.
- Comply with the system Bus (AXI)
- TrustZone
- Interrupt notification
- Round-Robin (RR) for scheduling scheme

There are 2 types of DMA channels supported in AI7933CLD

- Peripheral DMA: The behavior of Peripheral DMA is like that of CQ-DMA. The major difference is that the source or destination is peripherals FIFO, not memory.
- Virtual FIFO DMA (VFF DMA): It's a peripheral DMA with an additional FIFO control engine. It is used to provide the buffering capacity for peripherals.

2.2.9.2.1 Peripheral DMA

The behavior of Peripheral DMA is like that of CQ-DMA. The major difference is that the source or destination is peripheral FIFO, not memory. And there are handshaking signals (REQ / ACK) between DMA and peripheral. Because of handshaking signals, the DMA channels with corresponding peripheral channels are fixed.



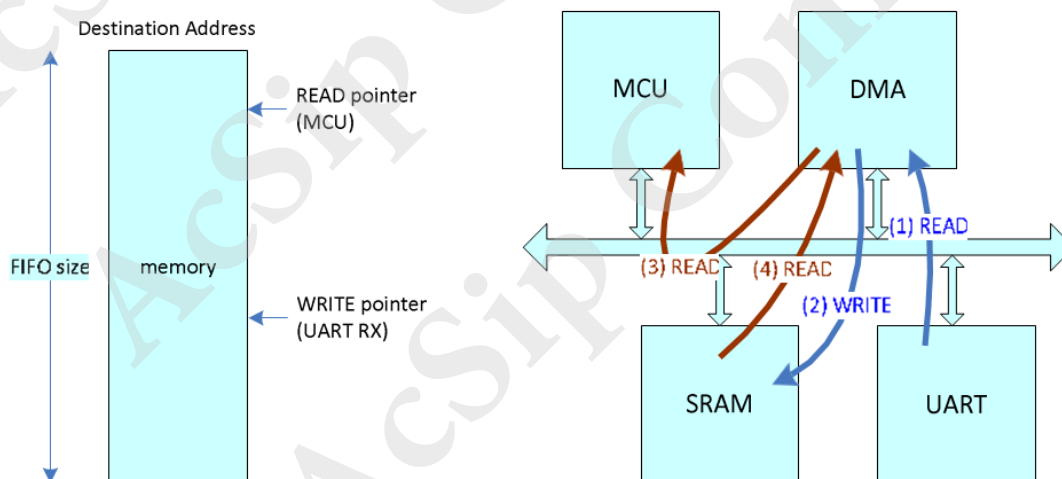
APDMA block diagram

2.2.9.2.2 Virtual FIFO

Virtual FIFO DMA is designed to offload the control of the serial interface. The difference between the virtual FIFO DMA and the peripheral DMA is that the virtual FIFO DMA contains an additional FIFO controller. VFF is like the ring buffer, and uses two address pointers (VFF_WPT/VFF_RPT) to control VFF condition. According to these two address pointers, two symbols are defined (VFF_VALID_SIZE/VFF_LEFT_SIZE) to represent the valid data and available space in VFF.

The figure below illustrates the operations of virtual FIFO DMA used for UART RX.

- (1) READ: DMA controller reads data from UART and increments the WRITE pointer of the FIFO controller.
- (2) WRITE; DMA controller writes data that was area from UART to SRAM in the area defined before enabling the virtual FIFO.
- (3) READ: MCU reads data when FIFO is not empty and the amount of data is over a pre-defined threshold. The read transaction will be finished only when DMA controller reads back the data from SRAM.
- (4) READ: DMA controller reads data from SRAM and increments the READ pointer of the FIFO controller.



Virtual FIFO concept

2.2.9.2.3 APDMA Channels and Priority Control

There are totally 7 virtual FIFO DMA channels and 2 peripherals DMA channels in the AI7933CLD.

Hardware IP	DMA Type
Radio (Bluetooth)	Virtual FIFO DMA x 2
UART (x2)	Virtual FIFO DMA x 4
AUXADC	Virtual FIFO DMA x 1
I2C	Peripheral DMA x 2

DMA type for hardware IP

2.2.10 GPT

2.2.10.1 Introduction

The Application Processor X General Purpose Timer (APXGPT) module includes five 32-bit GPTs and one 64-bit GPT. Each GPT supports four operation modes, which are ONE-SHOT, REPEAT, KEEP-GO and FREERUN. Each GPT can be operated on one of the two clock sources, RTC clock (32.768 kHz) or system clock (13 MHz).

2.2.10.2 Features

The table below provides details of the four operating modes GPT, ONE-SHOT, REPEAT, KEEP-GO and FREERUN.

Mode	Auto Stop	Interrupt Supported	Counting Behavior	When GPTn_COUNT equals GPTn_COMPARE	Example: Compare value is set to 2 (The bold value means interrupt asserted)
ONE-SHOT	Yes	Yes	The GPT stops counting when GPTn_COUNT equals GPTn_COMPARE	EN is reset to 0	0,1,2,2,2,2,2,2,2,2,...
REPEAT	No	Yes	The GPT count is reset to 0 when GPTn_COUNT equals GPTn_COMPARE	Count is reset to 0	0,1,2,0,1,2,0,1,2,0,1,2...
KEEP-GO	No	Yes	The GPT count is reset to 0 when overflow occurs		0,1,2,3,4,5,6,7,8,9,10,...
FREERUN	No	No	The GPT count is reset to 0 when overflow occurs		0,1,2,3,4,5,6,7,8,9,10,...

Operation mode of GPT

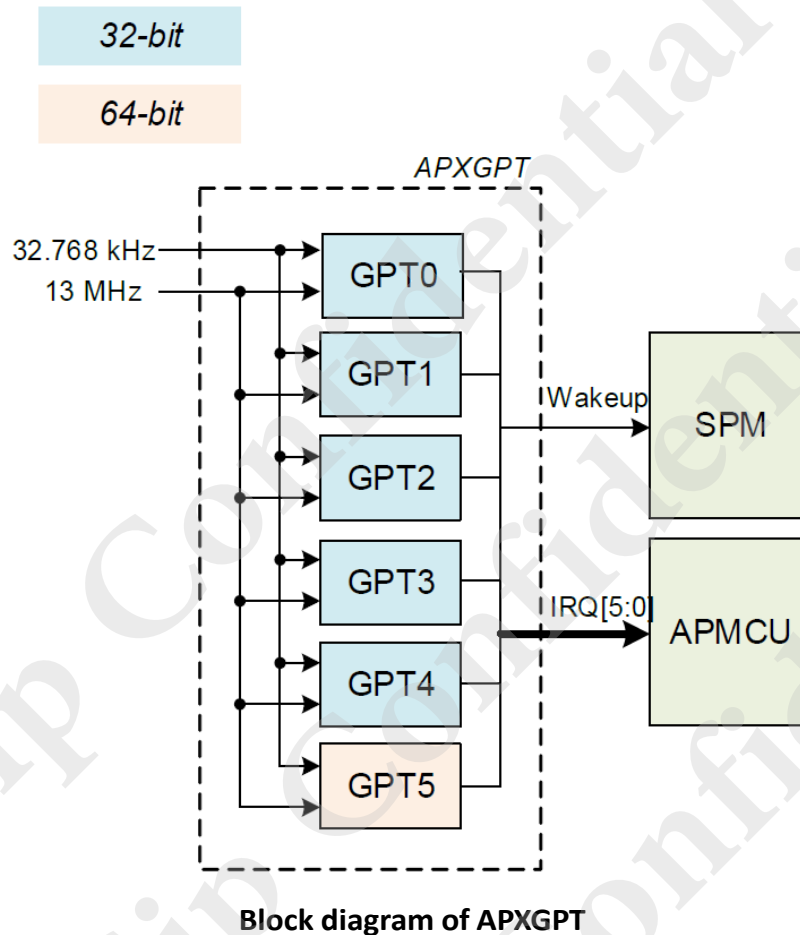
Note:

Each timer's operation is independent and can be programmed to select the clock source, RTC clock (32.768 kHz) or system clock (13 MHz). After the clock source is determined, the division ratio of the selected clock can be programmed. The division ratio can be fine-granulated as 1 to 13 and coarse-granulated as 16, 32 and 64.

2.2.10.3 Block Diagram

APXGPT consists of five sets of 32-bit GPTs and one set of 64-bit GPTs.

When the GPT triggers an IRQ, it will also issue a wakeup signal to "Sleep Control", and then "Sleep Control" can wake the MCU if the MCU is in sleep mode.



2.2.10.4 Theory of Operations

For the GPT5 64-bit timer, the read operation of the 64-bit timer value will be separated into 2 APB reads since an APB read is of 32-bit width. To perform the read operation of 64-bit timer value, the lower word should be read first and then the higher word. The read operation of the lower word will freeze the "read value" of the higher word but will not freeze the timer counting. This ensures that the separated read operation acquires the correct timer value.

To program and use the GPT, please note:

The counter value can be read at any time when the clock source is system clock. The counter value can be read at any time even when the clock source is RTC clock.

The comparative value can be programmed at any time. When the comparative value is rewritten during count operation, the counter will be reset to 0 and restart counting.

2.2.11 WDT

2.2.11.1 Introduction

The MT7933CT features a watchdog timer (WDT) for the Arm Cortex-M33 processor (referred to as CM33 in this document). CM33_WDT is the watchdog timer for the Cortex-M33 processor. When the Cortex-M33 processor hangs due to some malfunctions, the watchdog timer is used to generate system alarm and trigger whole chip reset.

2.2.11.2 Functional Description

The WDT provides the counter with the clock source of 32 kHz and asserts interrupt when needed.

There is an interrupt counter, and it can operate as follows:

- One-shot mode: the timer stops when the timer counts down to zero.
- The timer counts from a programmable initial value and asserts interrupt when counting to zero.

The interrupt will be level active low. The unit of the counter can be 1 x 32 kHz cycle.

The WDT provides two ways to generate the WDT event.

- Triggered by the time-out event to configure WDT HW reset mode.
- The WDT has an 11-bit counter and uses the 32-kHz clock. The software regularly restarts the timer to prevent it from expiring. If the timer fails to restart the WDT, it will expire and generate a WDT event.
- The programmable period length is $32\text{ms} \times (1 \sim 2048)$ which is ranged from 32ms to 65.5s.
- Triggered by software programming to write SW reset KEY.

2.2.11.3 Event Notification for CM33

The WDT provides the following options when a WDT event is generated.

- Trigger whole chip reset
- Interrupt itself

Note: The reset whole chip option is included in top chip reset register space.

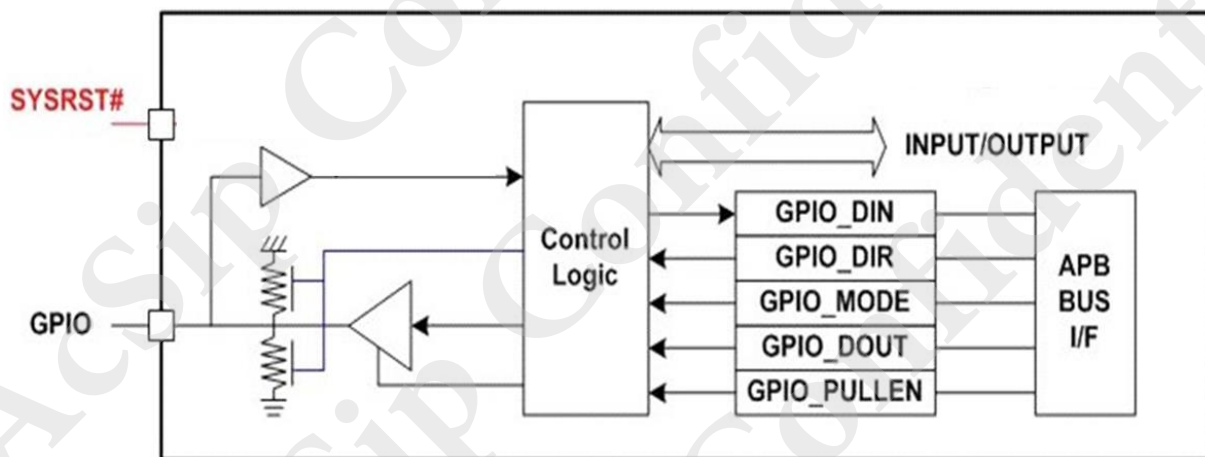
The WDT module can only be reset by the external reset (SYS_RST_N) and the PMU reset. Some WDT control registers feature a key protection mechanism to prevent unintentional access.

2.2.12 GPIO

2.2.12.1 Overview

The AI7933CLD platform offers 46 General-Purpose Input/Output (GPIO) pins. By setting the control registers, the MCU software can control the direction and the output value of the GPIO pins and read the input value of these pins. The GPIO pins and GPO (General-Purpose Output) pins are multiplexed with other functions to reduce the pin count.

The clock to send data outside the chip is software configurable. There are six clock-out ports and each clock-out port can be programmed to output the appropriate clock source. In addition, when two GPIO pins function for the same peripheral IP, the smaller GPIO serial number has higher priority over the bigger one.



AI7933CLD GPIO Controller

2.2.12.2 I/O Pull up or Down Control Truth Table

All AI7933CLD GPIO pins support both 1.8V and 3.3V IO power and different Pull up/Pull down resistors can be selected.

PU/PD resistance Spec

E	PUPD	R1	R0	R Value of 1.8V/3.3V IO Power
0	0	0	0	High-Z
0	0	0	1	PU10Kohm
0	0	1	0	PU50Kohm
0	0	1	1	PU10Kohm//50Kohm
0	1	0	0	High-Z
0	1	0	1	PD10Kohm
0	1	1	0	PD50Kohm
0	1	1	1	PD10Kohm//50Kohm
1	X	X	X	High-Z

2.3 Audio System

2.3.1 Audio Digital Signal Processor (DSP)

2.3.1.1 Introduction

DSP is responsible for running operating system and application programs. It comprises:

- One-core Cadence HiFi4 Audio Engine DSP
- AXI3 Bus Interface Unit (in-house MTK Bus)
- SPM control

Cadence's HiFi4 Audio Engine is a highly optimized audio processor geared for efficient execution of audio and voice codecs and pre/post-processing modules.

The processor goes beyond the HiFi3 with support for four 32x32-bit MACs, some support for 72-bit accumulators, limited ability to support eight 32x16-bit MACs, a fourth VLIW slot and the ability to issue two 64-bit loads per cycle. There is an optional floating point unit available, providing up to four single-precision IEEE floating point MACs per cycle. The extra resources provide significant performance improvements as compared to the HiFi3, particularly on pre/post-processing algorithms.

The HiFi4, a Single-Instruction/Multiple-Data (SIMD) processor, has the ability to work in parallel on two 32-bit data items or four 16-bit data items. HiFi4 is a VLIW architecture, supporting the execution of four operations in parallel.

2.3.1.2 Features

One-core Cadence's HiFi4 DSP operates at 300 MHz (0P7V)/600 MHz (0P8V), including:

- 32KB L1 I-cache
- 64KB L1 D-cache
- Pre-fetch buffer, I-cache, D-cache, ITag, and Dtag don't support data retention
- 32x32-bit MACs
- JTAG
- 25 interrupts
- Support SPM to control power sequence
- UART

2.3.2 Audio Channels

2.3.2.1 Functional Description

The audio system provides the ability to exchange audio data. The interfaces are listed as follows:

- Support 1* master 2-ch I2S output sample rates: 8 kHz to 192 kHz; bit resolution up to 32 bits
- Support 1* master/slave 2-ch I2S input sample rates: 8 kHz to 192 kHz; bit resolution up to 32 bits
- Support one built-in master TDM input interface: supported sampling rates include 8, 12, 16, 24, 32, 48, 96, 192 kHz; channel number up to 8
- Support 2-ch DMIC Pulse Density Modulation (PDM) interface x 2: one wire mode; 8, 16, 32, 48 kHz; 24 bits
- 2-ch built-in internal audio DAC: 8 to 48 kHz
- 3-ch built-in internal audio ADC: 8, 16, 32, 48 kHz; support dedicated HW gain: -75 dB to 24 dB
- Support 1-pair stereo general-purpose ASRC for sampling rate converter and slave mode clock tracking
- Support 2-ch HW gain for fade in and fade out
- Support 16-ch channel merge
- APLL x 1, can support two clock rates at the same time.
- Built-in 64KB audio internal SRAM

2.4.3 Interrupt Table

CM33 interrupt

MCU IRQ NUM	Interrupt Name	Interrupt Description	Polarity	Edge / Level	EINT
0	wic_int_wake_up	WIC wakeup interrupt to CM33	H	edge(HW) level(SW)	X
1	Reserved				
2	Reserved				
3	~wdt_irq_b0	CM33 WDT interrupt (wdt_irq_b[0])	H	edge	X
4	~uart_irq_b	CM33 UART interrupt	H	level	X
5	infra_bus_int	Infra bus error interrupt	H	level	X
6	dbgsys_CDBGPWRUPREQ	CoreSight debug power up req	H	level	X
7	dbgsys_CDBGPWRUPACK	CoreSight debug power up ack	H	level	X
8	~wdt_irq_b1	CM33 WDT interrupt (wdt_irq_b[1])	H	level	X
9	dsp_to_cm33_irq	DSP to CM33 IRQ	H	level	X
10	~apxgpt_irq_b0[0]	GPT_32_0	H	level	X
11	~apxgpt_irq_b0[1]	GPT_32_1	H	level	X
12	~apxgpt_irq_b0[2]	GPT_32_2	H	level	X
13	~apxgpt_irq_b0[3]	GPT_32_3	H	level	X
14	~apxgpt_irq_b0[4]	GPT_32_4	H	level	X
15	~apxgpt_irq_b0[5]	GPT_64	H	level	X
16	~devapc_INFRA_AON_secure_vio_irq_b	devapc secure violate	H	level	X
17	~devapc_AUD_BUS_PDN_secure_vio_irq_b	AUDIO_BUS devapc secure violate	H	level	X
18	conn_ap_bus_req_rise_irq	conn_ap_bus_req rising edge IRQ	H	level	X
19	conn_ap_bus_req_fall_irq	conn_ap_bus_req falling edge IRQ	H	level	X
20	conn_apsrc_req_rise_irq	conn_apsrc_req rising edge IRQ	H	level	X
21	conn_apsrc_req_fall_irq	conn_apsrc_req falling edge IRQ	H	level	X
22	conn_ap_bus_req_high_irq	conn_ap_bus_req level high IRQ	H	level	X
23	conn_ap_bus_req_low_irq	conn_ap_bus_req level low IRQ	H	level	X
24	conn_apsrc_req_high_irq	conn_apsrc_req level high IRQ	H	level	X
25	conn_apsrc_req_low_irq	conn_apsrc_req level low IRQ	H	level	X
26	infra_bus_timeout_irq	infra bus timeout IRQ	H	level	X
27	cm33_local_bus_int	CM33 local bus interrupt	H	level	X
28	adsp_infra_bus_timeout_irq	dsp_infra bus timeout IRQ	H	level	X
29	Reserved				
30	Reserved				
31	Reserved				
32	~dsp_uart_irq_b	DSP UART	H	level	X

MCU IRQ NUM	Interrupt Name	Interrupt Description	Polarity	Edge / Level	EINT
33	~top_uart0_irq_b	TOP UART	H	level	X
34	~top_uart1_irq_b	TOP UART	H	level	X
35	~i2c0_irqb	I2C transfer done or error	H	level	X
36	~i2c1_irqb	I2C transfer done or error	H	level	X
37	~sdctl_top_fw_irq_b	SDIO slave interrupt	H	level	X
38	~sdctl_top_fw_irq_b_qout	SDIO slave interrupt	H	level	X
39	~spi_irq_b	SPI master transfer done or pause	H	level	X
40	~spi_irq_b	SPI master transfer done or pause	H	level	X
41	~spis_irq_b	SPI slave transfer done	H	level	X
42	~kp_irq_b	keypad key pressed scanning done	H	edge	X
43	~irrx_irq_b	IRRX decoding done	H	level	X
44	Reserved				
45	Reserved				
46	Reserved				
47	Reserved				
48	ssusb_xhci_int	USB xHCI event interrupt	H	level	X
49	ssusb_otg_int	USB OTG interrupt	H	level	X
50	ssusb_dev_int	USB device interrupt	H	level	X
51	~audio_irq_mcu_b	AFE interrupt	H	level	X
52	~rtc_irq_b	RTC interrupt	H	level	X
53	sysram_top_int	SYSRAM out-of-bound access error	H	level	X
54	~mpu_irq_b_l2_pwr	illegal accesses to asic_mpu instances in L2	H	level	X
55	~mpu_irq_b_psrpm_pwr	illegal accesses to asic_mpu for PSRAM	H	level	X
56	~cq_dma_irq_b[0]	CQ_DMA Channel 0 finishes operation	H	level	X
57	~cq_dma_irq_b[1]	CQ_DMA Channel 1 finishes operation	H	level	X
58	~cq_dma_irq_b[2]	CQ_DMA Channel 2 finishes operation	H	level	X
59	~msdc_irq_b_0p	normal IRQ	H	level	X
60	~msdc_wakeup_ps_0p	wakeup IRQ	H	level	X
61	~dspwdt_irq_b	DSP WDT IRQ	H	level	X
62	dsp2cpu_irq	DSP to CPU IRQ	H	level	X
63	~ap_dma_irq_b[0]	AP_DMA Channel 0 (I2C 0) finishes operation	H	level	X
64	~ap_dma_irq_b[1]	AP_DMA Channel 1 (I2C 1) finishes operation	H	level	X
65	~ap_dma_irq_b[2]	AP_DMA Channel 2 (Reserved) finishes operation	H	level	X
66	~ap_dma_irq_b[3]	AP_DMA Channel 3 (Reserved) finishes operation	H	level	X

MCU IRQ NUM	Interrupt Name	Interrupt Description	Polarity	Edge / Level	EINT
67	~ap_dma_irq_b[4]	AP_DMA Channel 4 (UART 0 TX) finishes operation	H	level	X
68	~ap_dma_irq_b[5]	AP_DMA Channel 5 (UART 0 RX) finishes operation	H	level	X
69	~ap_dma_irq_b[6]	AP_DMA Channel 6 (UART 1 TX) finishes operation	H	level	X
70	~ap_dma_irq_b[7]	AP_DMA Channel 7 (UART 1 RX) finishes operation	H	level	X
71	~ap_dma_irq_b[8]	AP_DMA Channel 8 (BTIF TX) finishes operation	H	level	X
72	~ap_dma_irq_b[9]	AP_DMA Channel 9 (BTIF RX) finishes operation	H	level	X
73	~ap_dma_irq_b[10]	AP_DMA Channel 10 (Reserved) finishes operation	H	level	X
74	~ap_dma_irq_b[11]	AP_DMA Channel 11 (AUXADC RX) finishes operation	H	level	X
75	~btif_host_irq_b	BTIF host interrupt	H	level	X
76	~flash_int_b	Flash Controller Interrupt	H	level	X
77	~conn2ap_wfdma_irq_b	Wi-Fi host interrupt	H	level	X
78	~bgf2ap_wdt_irq_b	bgf WDT interrupt	H	edge	X
79	~bgf2ap_btif0_wakeup_out_b	bgf BTIF wakeup interrupt	H	level	X
80	~conn2ap_sw_irq_b	bgf software interrupt for debug	H	level	X
81	~bt2ap_isoch_irq_b	bt iso channel interrupt	H	level	X
82	~bt_cvds_int_b	bt CVSD interrupt	H	level	X
83	~ccif_wf2ap_sw_irq_b	wf software interrupt for debug from ccif trigger	H	level	X
84	~ccif_bgf2ap_sw_irq_b	bgf software interrupt for debug from ccif trigger	H	level	X
85	~wm_conn2ap_wdt_irq_b		H	edge	X
86	~sema_release_inform_m2_int_b	semaphore release IRQ for m2	H	level	X
87	~sema_release_inform_m3_int_b	semaphore release IRQ for m3	H	level	X
88	~sema_m2_timeout_int_b	semaphore timeout IRQ for m2	H	level	X
89	~sema_m3_timeout_int_b	semaphore timeout IRQ for m3	H	level	X
90	~conn_bgf_hif_on_host_int_b				
91	Reserved				
92	ssusb_spm_int	USB SPM wakeup IRQ	H	level	X
93	~wf2ap_sw_irq_b	wf2ap software IRQ	H	level	X
94	cq_dma_sec_abort	CQ_DMA APB secure violation happens	H	level	X
95	ap_dma_sec_abort	AP_DMA APB secure violation happens	H	level	X
96	~sdio_sdio_cmd_i	SDIO slave wakeup interrupt	H	edge	X
97	Reserved				

MCU IRQ NUM	Interrupt Name	Interrupt Description	Polarity	Edge / Level	EINT
98	~adc_comp_irq_b	ADC comparator IRQ	H	level	X
99	~adc_fifo_int_b	ADC FIFO mode interrupt	H	level	X
100	~gcpu_irq_b	GCPU interrupt	H	level	X
101	~ecc_irq_b	ECC interrupt	H	level	X
102	~trng_irq_b	TRNG interrupt	H	level	X
103	~sej_apxgpt_irq_b	SEJ GPT interrupt	H	level	X
104	~sej_wdt_irq_b	SEJ WDT interrupt	H	level	X
105	Reserved				
106	Reserved				
107	Reserved				
108	gpio_irq[0]	GPIO IRQ	H	configurable	V
109	gpio_irq[1] / sdio_sdio_cmd_i	GPIO IRQ / sdio_sdio_cmd_i	H	configurable	V
110	gpio_irq[2]	GPIO IRQ	H	configurable	V
111	gpio_irq[3]	GPIO IRQ	H	configurable	V
112	gpio_irq[4]	GPIO IRQ	H	configurable	V
113	gpio_irq[5]	GPIO IRQ	H	configurable	V
114	gpio_irq[6]	GPIO IRQ	H	configurable	V
115	gpio_irq[7]	GPIO IRQ	H	configurable	V
116	gpio_irq[8]	GPIO IRQ	H	configurable	V
117	gpio_irq[9]	GPIO IRQ	H	configurable	V
118	gpio_irq[10]	GPIO IRQ	H	configurable	V
119	gpio_irq[11]	GPIO IRQ	H	configurable	V
120	gpio_irq[12]	GPIO IRQ	H	configurable	V
121	gpio_irq[13]	GPIO IRQ	H	configurable	V
122	gpio_irq[14]	GPIO IRQ	H	configurable	V
123	gpio_irq[15]	GPIO IRQ	H	configurable	V
124	gpio_irq[16]	GPIO IRQ	H	configurable	V
125	gpio_irq[17]	GPIO IRQ	H	configurable	V
126	gpio_irq[18]	GPIO IRQ	H	configurable	V
127	gpio_irq[19]	GPIO IRQ	H	configurable	V
128	gpio_irq[20]	GPIO IRQ	H	configurable	V
129	gpio_irq[21]	GPIO IRQ	H	configurable	V
130	gpio_irq[22]	GPIO IRQ	H	configurable	V
131	gpio_irq[23]	GPIO IRQ	H	configurable	V
132	gpio_irq[24]	GPIO IRQ	H	configurable	V
133	gpio_irq[25]	GPIO IRQ	H	configurable	V
134	gpio_irq[26]	GPIO IRQ	H	configurable	V
135	gpio_irq[27]	GPIO IRQ	H	configurable	V

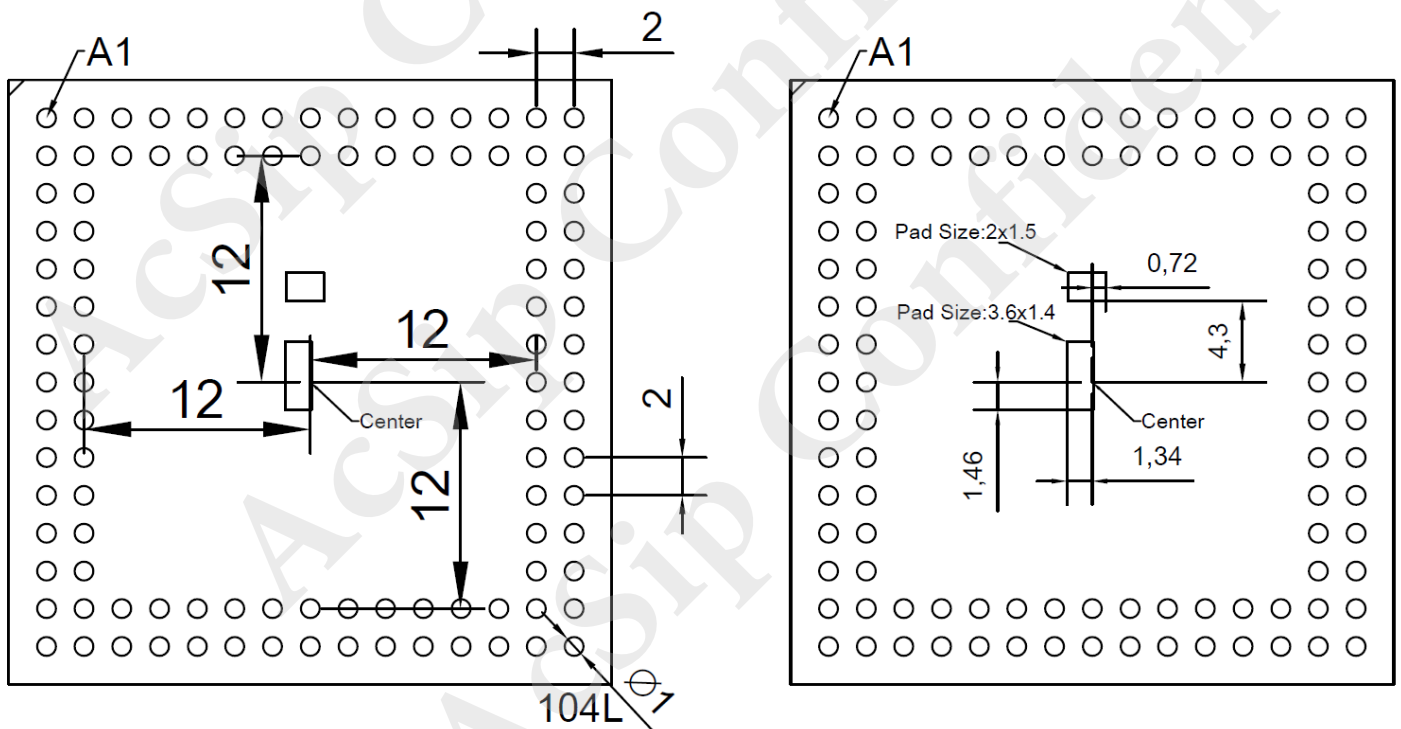
MCU IRQ NUM	Interrupt Name	Interrupt Description	Polarity	Edge / Level	EINT
136	gpio_irq[28]	GPIO IRQ	H	configurable	V
137	gpio_irq[29]	GPIO IRQ	H	configurable	V
138	gpio_irq[30] / ~sdio_sdio_cmd_i	GPIO IRQ / ~sdio_sdio_cmd_i	H	configurable	V
139	~cm33_pad_uart_rx / ~sdio_sdio_cmd_i	~cm33_pad_uart_rx / ~sdio_sdio_cmd_i	H	configurable	V

3 Recommend Footprint (Typ.)

Unit: mm

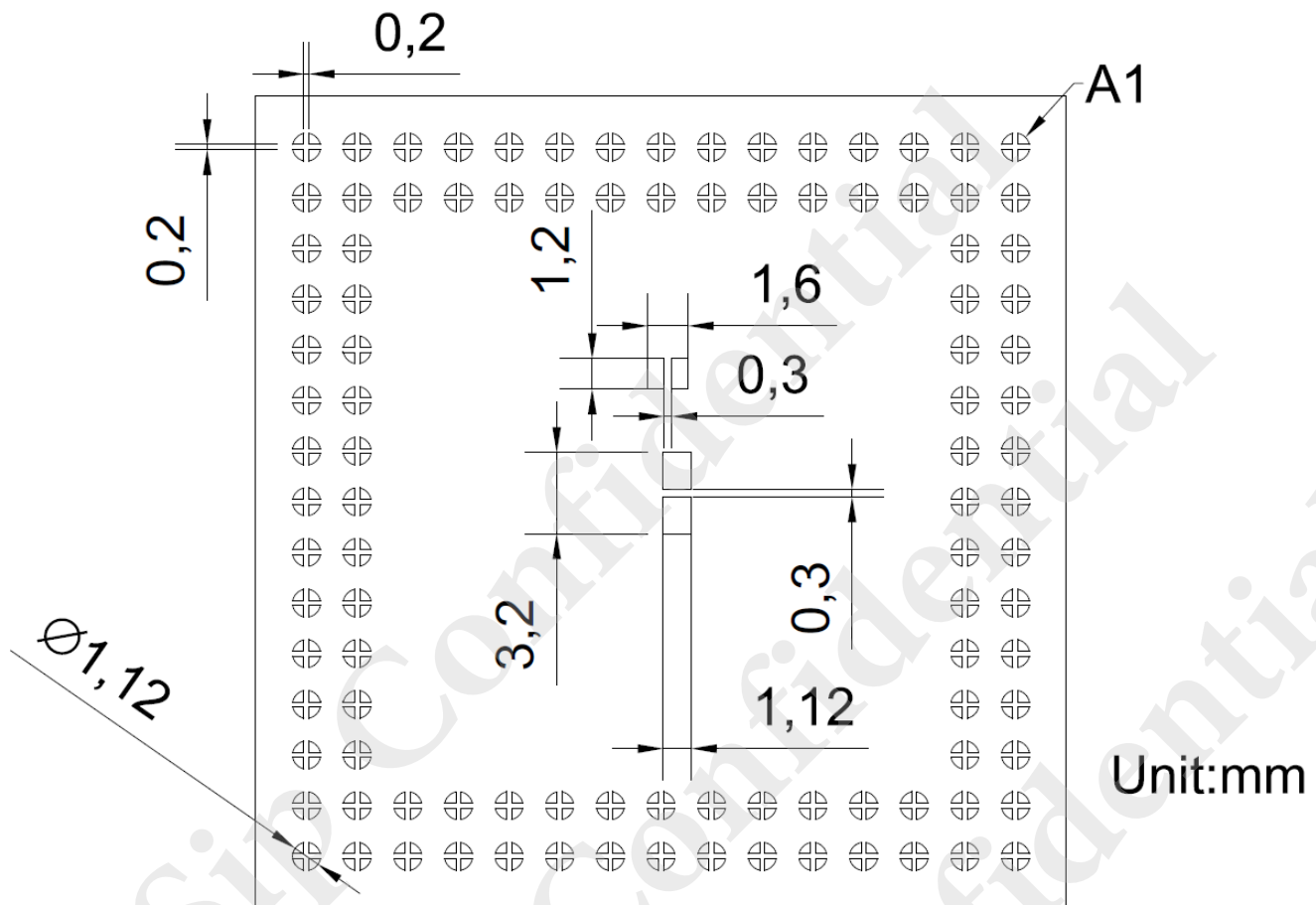


TOP View



Bottom View

3.1 Recommend Stencil



4 Other Information

- Discuss with AcSiP engineer after schematic and layout finished.