Related Documentation From ARCA

Arca Instruction Set Architecture Manual – version 2

Arca210 Hardware Manual

Draco Development Kit for Arca210 Hardware User’s Guide

Draco Development Kit for Arca210 Software User’s Guide

ARCABoot User’s Guide

Table of Contents

List of Figure

List of Table

1 Introduction

The Arca^{TM} is a Reduced Instruction Set Computer (RISC) architecture which is designed specifically to address the applications requirements and challenges for IA (Information Application) and embedded market. As the second generation of Arca series RISC microprocessor, Arca210 is a high performance system-on-chip (SOC) microprocessor that integrates Arca RISC core, memory controller, PCI controller and other peripheral controllers such as DMA, MAC, USB, AC97, UART, IrDA, SCC, I2C, GPIO and so on.

This document is addressed to present recommendation on schematics and PCB of application design based on Arca210 microprocessor.

1.1 Block Diagram

The system architecture of Arca210 is shown in the following diagram:

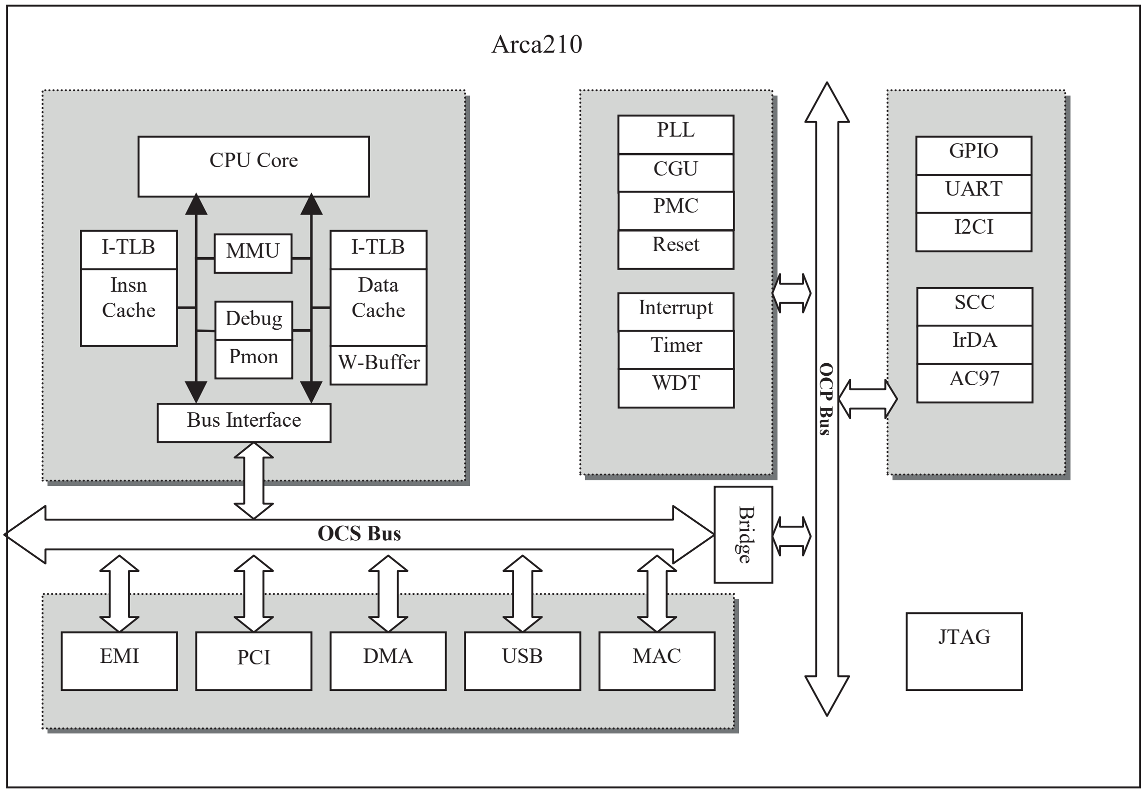

Figure 1-1 Block Diagram of Arca210

Arca210 consists of the following three functional groups:

- **Processor Core**

  The processor core contains CPU, memory management unit, caches and a bus interface unit. The processor core adopts a Harvard memory architecture that contains 16k byte instruction cache and 16k byte data cache. For data cache, 16 word write back buffer is provided to reduce the performance lost due to bus conflict. Instruction address and data address are translated through separate 128 entry Instruction TLB and 128 entry data TLB.

A debug module with JTAG interface is embedded in the core to assist software, especially kernal code debugging. PMON module is used to monitor CPU-core's performance such as, clock number, instruction executed number, cache miss number and etc.

- **OCS Bus Devices**

Devices requiring high performance memory bandwidth or to be in close proximity to the CPU are connected to OCS bus. Arca210 OCS bus devices include memory controller, PCI bus interface controller, DMA controller, USB controller and Ethernet controller. The high speed of the OCS bus guarantees the high bandwidth requirements between CPU and external high speed devices.

- **OCP Bus Devices**

Devices that don't need a high bandwidth connection are connected in OCP (On Chip Peripheral) bus. These include clock generation unit, power management unit, interrupt controller, timer, real time clock, watchdog timer, general purpose IO, UART, smart card controller, IrDA, I^{2}C interface and AC97 controller.

1.2 Features

The Arca210 is a highly integrated, low-cost implementation of the Arca RISC architecture, that includes many function units, can be used in a single board design. The features of Arca210 are listed in Table 1-1.

Table 1-1 Arca210 Features

Item	Features
Integer Unit	• Arca version 2 architecture, 32-bit Arca instruction set. • 32-bit internal data bus • Thirty-two 32-bit general registers • 5-stage pipeline • Virtual address space: 4 G-Bytes • Space identifier ASID: 8 bits, 256 virtual address space
Memory Manager Unit (MMU)	• 4 G-Bytes of address space, 256 address space (8 bits ASID) • Full associative 32-entry instruction TLB (ITLB) and 32-entry data TLB (DTLB), with round robin replacement algorithm • Four different page size: 4KB, 16KB, 1MB and 16MB in any entry • Support entry lock • Translate 32-bit virtual address to 32-bit physical address • Support five address spaces
Data Cache	• 16K-Byte, physically-indexed, virtually-tagged • Hardware solve alias issue • 32-way set associative: 8 sets with each set containing 32 ways • Each way contains 32 bytes (one cache line) • Write-back, write-through, round robin algorithm • 4-deep write buffer • Support lock, allocate operations
Instruction Cache	• 16K-Byte, physically-indexed, virtually-tagged • 32-way set associative: 8 sets with each set containing 32 ways • Each way contains 32 bytes (one cache line) • Support lock operation
DEBUG	• JTAG interface to host machine • ASID match • Two instruction break point or one maskable instruction break

	point • Two data break point or one maskable data break point • Software break • Asynchronous break from host machine • Asynchronous boot from host machine
Performance Monitor (PMON)	• One 32-bits internal clock counter • Two 32-bits signal counter, each of which can be set to count 1 from 14 signals • Count overflow interrupt support
External Memory Interface (EMI)	• Static memory controller – Direct interface to ROM, Burst ROM, SRAM, Flash and memory like devices – 8, 16 or 32-bit bus width – 4 banks up to 256M (64M X 4) – Programmable wait and external wait signal • Synchronous DRAM controller – Both DIMM and SODIMM are supported – 32-bit bus width – Four banks up to 1G (256M x 4) is supported – Supports burst operation – Page mode supported – Has both auto-refresh and self-refresh functions
PCI Controller	• PCI revision 2.1 • 32-bit, 33MHz • Supports (in host mode) up to 4 external bus masters or slaves • Supports (in host mode) bus arbitration between host and 4 external masters • Support host and satellite mode • Support transaction between Arca210 CPU and PCI slaves • Support direct transaction between external master and system device • Four 8-deep FIFO for each transaction
Direct Memory Access controller (DMA)	• Four independent DMA channels with fix or round robin priority • Up to 16M transfer count • Transfer data unit: byte, half-word, word, 4-word or 8-word • Support external DMA request
Ethernet Controller (ETHC)	• Compliant with IEEE802.3, 802.3u Specification • 10/100 Mbps data transfer rates • IEEE802.3 compliant MII interface to talk to an external PHY • VLAN support • Full and half duplex modes • Support CSMA/CD protocol in half duplex mode • Supports flow-control for full-duplex operation • Provides External and internal loop back capability on the MII Interface
USB Host Controller (UHC)	• Open Host Controller Interface (OHCI)-compatible and USB Revision 1.1-compatible • Support low-speed (1.5 Mbps) and full-speed (12 Mbps) USB devices • Two downstream ports are provided
Clock Generation Unit (CGU)	• On-chip phase-locked loop (PLL) with programmable multiple-ratio • On-chip oscillator circuit • 5 clock modes can be selected by MD0 ~ MD2 • The PLL on/off is programmable by software • Input clock source can be selected from oscillator or an external

	clock input • Generates three clocks: internal clock (iclk) for CPU core, peripheral clock (pclk) for peripheral devices, system clock (sclk) for system bus devices • Iclk, pclk and sclk frequency can be changed separately for software by setting division ratio
Interrupt controller (INT)	• Eight external interrupt pins (IRQ0 – IRQ7) • On-chip peripheral interrupts
Universal Asynchronous Receiver/Transmitter (UART)	• Full-duplex communication • 16-byte FIFO for transmission/reception • DMA can be transferred • Modem control functions (RTS and CTS) are provided
Watchdog Timer (WDT)	• Generates power-on reset or manual reset • 16-bit counter
Timer Unit (TMU)	• 3-channel auto-reload-type 32-bit timer • 5 types of counter input clocks can be selected • Auto-reload function is provided for each channel
Power/reset Management Controller (PMC)	• Supports three low-power modes and function: sleep mode, standby mode, module stop function • Reset sequence control
I^{2}C-Bus Interface (I2CI)	• Supports only single master mode • Support of I²C standard-mode and F/S-mode up to 400 kHz • Independent, programmable serial clock generator • The number of devices that you can connect to the same I^{2}C-bus is limited only by the maximum bus capacitance of 400pF
Smart Card Controller (SCC)	• Conforms to the ISO/IEC standard 7816-3 • Support asynchronous character (T=0) and block (T=1) communication modes • Receiver and Transmit mode error signal detection and automatic re-transmission of data (T = 0) • Supports both direct convention and inverse convention • A straightforward extension of UART: When SCC is disabled, UART can work as a normal UART
Infrared Serial Interface (IrDA)	• Based on the IrDA 1.0 specification • Polarity of transmitted and received signals selectable • When transmitting, support normal 3/16 and IrDA low-power mode bit duration • When receiving, normal 3/16 and low-power mode bit duration signal will both be received as low-power mode bit duration • A straightforward extension of UART2: When IrDA is disabled, UART2 can work as a normal UART
AC97 Controller (AC97)	• Compliant with AC'97 Component Specification 2.2 • Support 16, 18, 20bit sample size • Programmable Output channels and Input channels Support • Power Down Mode • Two Wake-Up mode Support • Support DMA transfer mode
General Purpose Input/Output Port (GPIO)	• 8 bits dedicated ports • 39 shared ports

1.3 Pin Description

The pin description of Arca210 is shown as Table 1-1.

Table 1-1 Arca210 Pin Description (total: 304pin)

Pin Name	I/O	Pin Description
1. EMI pins (total 85)		
D[31:0]	IO	Memory data bus
A[25:0]	O	SDRAM memory address For SDRAM, A[16:2] is used as SDRAM address signals /A[20:18]/MD_CLK[2:0]: used as the clock mode configuration when power-up, should be pull-up or pull-down with 4.7-10KΩ resistor for clock mode configuration, it will be latched at the rising-edge of the RESETP_ signal Add[20:18] Core-CLK : OCS_CLK(SDAM Clock CKO[3:0]) 000 1 : 1 001 1 : 2 010 1 : 3 (default, pull-down/pull-up internal) 011 1 : 4 100 Test Mode 101 1 : 6 110 1 : 8 111 No used /A[1]/MD_Endian: used as the Endian configuration when power-up, should be pull-up or pull-down with 4.7-10KΩ resistor for Arca2 Endian configuration, it will be latched at the rising-edge of the RESETP_ signal 0: Big-Endian (default, pull-down internal) 1: Little-Endian
CS0_	O	Static memory bank0 chip select
CS1_	O	Static memory bank1 chip select
CS2_	O	Static memory bank2 chip select
CS3_	O	Static memory bank3 chip select
CS4L_	O	SDRAM bank4 chip select When SDRAM is used, chip select signal to indicate that bank 4 is being accessed
CS4H_	O	SDRAM bank4 upper chip select When SDRAM SODIMM is used, selection signal for D32-D39. When SDRAM DIMM is used, chip select signal to indicate that upper half of bank 4 is being accessed
CS5L_	O	SDRAM bank5 chip select When SDRAM is used, chip select signal to indicate that bank 5 is being accessed
CS5H_	O	SDRAM bank5 upper chip select When SDRAM SODIMM is used, selection signal for D40-D47. When SDRAM DIMM is used, chip select signal to indicate that upper half of bank 5 is being accessed
CS6L_	O	SDRAM bank6 chip select When SDRAM is used, chip select signal to indicate that bank 6 is being accessed

CS6H_	O	SDRAM bank6 upper chip select When SDRAM SODIMM is used, selection signal for D48-D55. When SDRAM DIMM is used, chip select signal to indicate that upper half of bank 6 is being accessed
CS7L_	O	SDRAM bank7 chip select When SDRAM is used, chip select signal to indicate that bank 7 is being accessed
CS7H_	O	SDRAM bank7 upper chip select When SDRAM SODIMM is used, selection signal for D56-D63. When SDRAM DIMM is used, chip select signal to indicate that upper half of bank 7 is being accessed
RAS_	O	SDRAM Row address strobe
CAS_	O	SDRAM Column address strobe
WE0_	O	Byte 0 write enable When static memory is used, selects D7-0 write strobe signal. When SDRAM is used, selection signal for D7–D0
WE1_	O	Byte 1 write enable When static memory is used, selects D15-8 write strobe signal When SDRAM is used, selection signal for D15–D8
WE2_	O	Byte 2 write enable When static memory is used, selects D23-16 write strobe signal When SDRAM is used, selection signal for D23–D16
WE3_	O	Byte 3 write enable When static memory is used, selects D31-24 write strobe signal When SDRAM is used, selection signal for D31–D24
RDWR_	O	Data bus direction designation signal Also used as SDRAM write designation signal 1 – read, 0 – write
RD_	O	When using static memory , RD# signal, indicates a read cycle
WE_	O	When using static memory: WE# signal, indicates a write cycle
CKE	O	Clock enable for SDRAM
CKO0	O	SDRAM Clock0
CKO1	O	SDRAM Clock1
CKO2	O	SDRAM Clock2
CKO3	O	SDRAM Clock3
WAIT_	I	External Wait state request signal for static memory, pull-up internally
2. PCI pins (total 52)		
AD[31..0]	IO	PCI: Address/Data bus
C/BE[3..0]	IO	PCI: Command/Byte enable
PAR	IO	PCI: Even parity across AD[31:0] and C_[3:0]
CLK	I	PCI clock
RST_	IO	PCI reset
FRAME_	IO	PCI frame
TRDY_	IO	Target ready
IRDY_	IO	initiator ready
STOP_	IO	current target request master to stop current transaction
DEVSEL_	IO	device select
LOCK_	I	Lock access to memory
IDSEL_	I	Initialization device select
PERR_	IO	parity error
SERR_	O	system error
INTA_	IO	Interrupt A
INTB_	I	Interrupt B
INTC_	I	Interrupt C
INTD_	I	Interrupt D

REQ_0_	IO	PCIART: bus request 0, used by Arca210
REQ_1_	I	PCIART: bus request 1
REQ_2_	I	PCIART: bus request 2
REQ_3_	I	PCIART: bus request 3
REQ_4_	I	PCIART: bus request 4
GNT_0_	IO	PCIART: bus grant 0, used by Arca210
GNT_1_	O	PCIART: bus grant 1
GNT_2_	O	PCIART: bus grant 2
GNT_3_	O	PCIART: bus grant 3
GNT_4_	O	PCIART: bus grant 4
3. DMA/GPIO D 6~7/GPIO E 0~2 pins (total 8)		
DREQ0_ /GPE0	IO	DMA external request 0 /GPIO E Inout0
DREQ1_ /GPE1	IO	DMA external request 1 /GPIO E Inout1
DREQ2_ /GPE2	IO	DMA external request 2 /GPIO E Inout2
DACK0 /MD_TAP_SEL	IO	DMA external data acknowledge 0 /MD_TAP_SEL : used as the TAP (Test Access Port, JTAG) mode configuration when power-up, should be pull-up or pull-down with 4.7-10KΩ resistor for TAP mode configuration, it will be latched at the rising-edge of the RESETP_ signal 1: debug 0: boundary (default, pull-down internal)
DACK1 /MD_MEM[0]	IO	DMA external data acknowledge 1 /MD_MEM[0]: used as the boot mode configuration when power-up, should be pull-up or pull-down with 4.7-10KΩ resistor for boot mode configuration, it will be latched at the rising-edge of the RESETP_ signal
DACK2 /MD_MEM[1]	IO	DMA external data acknowledge 2 /MD_MEM[1]: used as the boot mode configuration when power-up, should be pull-up or pull-down with 4.7-10KΩ resistor for boot mode configuration, it will be latched at the rising-edge of the RESETP_ signal MD_MEM[1:0] Boot-Mode 00 8-bit (default, pull-down internal) 01 16-bit 10 32-bit 11 Reserved
AEN /GPD6	IO	DMA transaction enable /GPIO D Inout6
EOP /GPD7	IO	DMA transaction complete /GPIO D Inout7
4. UHC pins (total 9)		
UHC_CLK	I	USB 48MHz Clock Input
DPLS0	IO	USB Port 0 Data Plus
DMNS0	IO	USB Port 0 Data Minus
OVC0	I	USB Port 0 Overcurrent to indicate there is a power supply problem with the port
PPWR0	O	USB Port 0 Power Switch used to enable or disable the external voltage supplying power to the port and is de-asserted when a power supply problem is detected at OVC0 pin
DPLS1	IO	USB Port 1 Data Plus
DMNS1	IO	USB Port 1 Data Minus
OVC1	I	USB Port 1 Overcurrent to indicate there is a power supply problem with the port

PPWR1	O	USB Port 1 Power Switch used to enable or disable the external voltage supplying power to the port and is de-asserted when a power supply problem is detected at OVC1 pin
5. ETHC/GPIO E 3~7/GPIO F pins (total 17)		
MII_COL /GPF7	IO	Ethernet Collision MII_COL shall be asserted by the Ethernet PHY Controller chip upon detection of a collision on the medium, and shall remain asserted while the collision condition persists. The transitions on the MII_COL signal are not synchronous to either the MII_TX_CLK or the MII_RX_CLK. The MAC ignores the signal MII_COL during Full Duplex Operation. /GPIO F Inout0
MII_CRS /GPF6	IO	Ethernet Carrier Sense MII_CRS shall be asserted by the Ethernet PHY Controller Chip when either transmit or receive medium is non idle. MII_CRS shall be de-asserted by the PHY when both the TX medium and the RX medium are idle. The PHY shall ensure that MII_CRS remains asserted throughout the duration of a collision condition. The transitions on the MII_CRS signal are not synchronous to either the MII_TX_CLK or the MII_RX_CLK. /GPIO F Inout1
MII_TX_CLK	I	Ethernet Transmit Clock MII_TX_CLK is a continuous clock that provides for the timing reference for the transfer of the MII_TX_EN, MII_TX_ER, and MII_TXD signals from the MAC to the Ethernet PHY Controller. MII_TX_CLK is sourced by the Ethernet PHY Controller chip. The operating frequency of the MII_TX_CLK is 25 MHz when operating at 100-Mb/s and 2.5 MHz when operating at 10-Mb/s.
MII_TXD[3:0] /GPF[5:2]	IO	Ethernet Transmit Data MII_TXD is a bundle of 4 data signals MII_TXD[3:0] that are driven by the MAC. MII_TXD[3:0] will transition synchronously with respect to the MII_TX_CLK. For each MII_TX_CLK period in which MII_TX_EN is asserted, MII_TXD[3:0] will have the data to be accepted by the Ethernet PHY Controller chip. MII_TXD[0] is the least significant bit. While MII_TX_EN is de-asserted the data presented on MII_TXD[3:0] should be ignored. /MII_TXD[3]: GPIO F Inout5 /MII_TXD[2]: GPIO F Inout4 /MII_TXD[1]: GPIO F Inout3 /MII_TXD[0]: GPIO F Inout2
MII_TX_EN /GPF1	IO	Ethernet Transmit Enable MII_TX_EN indicates that the MAC is presenting nibbles on the MII for transmission. It will be asserted by the Mac with the first nibble of the preamble and will remain asserted while all nibbles to be transmitted are presented to the MII. MII_TX_EN will be negated prior to the first MII_TX_CLK following the final nibble of the frame. MII_TX_EN is driven by the MAC and will transition synchronously with respect to the MII_TX_CLK. When asserted the MII_TX_EN will be at logic '1' and it will be at logic '0' while de-asserted. /GPIO F Inout1
MII_RX_CLK	I	MII_RX_CLK is a continuous clock that provides the timing reference for the transfer of the MII_RX_DV, MII_RX_ER, and MII_RXD signals from the Ethernet PHY Controller to the MAC. MII_RX_CLK is sourced by the Ethernet PHY Controller chip. The MII_RX_CLK shall have a frequency equal to 25% of the data rate of the received signal on the Ethernet Cable.
MII_RX_DV /GPF0	IO	MII_RX_DV is driven by the external Ethernet PHY Controller to indicate the MAC that it is presenting the recovered and decoded

		nibbles on the MII_RXD[3:0] bundle and that the data on MII_RXD[3:0] is synchronous to MII_RX_CLK. MII_RX_DV shall transition synchronously with respect to the MII_RX_CLK. MII_RX_DV shall remain be asserted continuously from the first recovered nibble of the frame through the final recovered nibble, and shall be negated prior to the first MII_RX_CLK that follows the final nibble. When asserted the MII_RX_DV will be at logic '1' and it will be at logic '0' while de-asserted. /GPIO F Inout0
MII_RXD[3:0] /GPE[7:4]	IO	MII_RXD is a bundle of four data signals MII_RXD [3:0] that transition synchronously with respect to the MII_RX_CLK. RXD[3:0] are driven by the Ethernet PHY Controller chip. For each MII_RX_CLK period in which MII_RX_DV is asserted, MII_RXD [3:0] transfer four bits of recovered data from the PHY to the ETHC. MII_RXD [0] is the least significant bit. While MII_RX_DV is de-asserted, MII_RXD [3:0] will have no effect on the ETHC. MII_RXD[3]: GPIO E Inout7 MII_RXD[2]: GPIO E Inout6 MII_RXD[1]: GPIO E Inout5 MII_RXD[0]: GPIO E Inout4
MII_RX_ER /GPE3	IO	MII_RX_ER is driven by the Ethernet PHY Controller chip. MII_RX_ER shall be asserted for one or more MII_RX_CLK periods to indicate to the MAC that an error (e.g., a coding error, or any error that the PHY is capable of detecting, and that otherwise be undetectable by the MAC) was detected some where in the frame presently being transferred from the PHY to the MAC. MII_RX_ER shall transition synchronously with respect to MII_RX_CLK. While MII_RX_DV is de-asserted, MII_RX_ER will have no effect on the MAC. When asserted the MII_RX_ER will be at logic '1' and it will be at logic '0' while de-asserted. /GPIO E Inout3
MII_MDC	O	Ethernet Management Clock The MAC sources the MII_MDC signal to the Ethernet PHY Controller as the timing reference for transfer of information on the MII_MDIO signal. MII_MDC is an aperiodic signal that has no maximum high or low times. The minimum high and low times for MII_MDC will be 160 ns each, and the minimum period for MII_MDC will be 400 ns, regardless of the nominal period of TXClk and RXClk.
MII_MDIO	IO	Ethernet Management Data Inout When input, MII_MDIO is the data input signal from the Ethernet PHY. The Read Data is driven by the PHY synchronously with respect to the MII_MDC clock during the read cycles. When output, MII_MDIO is the data output signal from the MAC that is used to drive the control information during the Read/Write cycles to the Ethernet PHY. The MAC drives the MII_MDIO signal synchronously with respect to the MII_MDC.
6. AC97/GPIO C 0 ~ 4 pins (total 5)		
SDATA_OUT /GPC0	IO	SDATA_OUT: Serial audio data out, to Codec /GPIO C Inout0
SDATA_IN /GPC1	IO	SDATA_IN : Serial audio data in, from Codec /GPIO C Inout1
SYNC /GPC2	IO	SYNC: Frame Synchronization, 48 kHz fixed rate sample sync /GPIO C Inout2
RESET_ /GPC3	IO	RESET_ : AC97 Master hardware reset, active low /GPIO C Inout3

BIT_CLK	I	12.288MHz Serial Data Clock Input
7. UART/GPIO C 6~7 (total 2)		
TxD /GPC6	IO	TxD: Transmit data output /GPIO C Inout6
RxD/GPC7	IO	RxD: Receive data input /GPIO C Inout7
8. IrDA/UART2/GPIO D 0~3 (total 4)		
RTS_/GPD0	IO	RTS_: Transmission request (Request To Send) /GPIO D Inout0
CTS_/GPD1	IO	CTS_: Transmission enabled (Clear To Send) /GPIO D Inout1
TxD2/GPD2	IO	TxD2: Transmit data output /GPIO D Inout2
RxD2 /GPD3	IO	RxD2: Receive data input /GPIO D Inout3
9. SCC/GPIO D 4~5 (total 2)		
SCC_DATA /GPD4	IO	Transmit/Receive data connects SCC and smart card /GPIO D Inout4
SCC_CLK /GPD5	IO	Serial clock connects SCC and smart card /GPIO D Inout5
10. I2CI pins (total 2)		
SDA	IO	Serial Data Inout
SCL	IO	Serial Clock
11. GPIO A pins (total 8)		
GPA[7:0]	IO	GPIO A Inout[7:0]
12. IRQ/GPIO B pins (total 8)		
IRQ0 /GPB0	IO	Interrupt Request Input 0 /GPIO B Inout0
IRQ1 /GPB1	IO	Interrupt Request Input1 /GPIO B Inout1
IRQ2 /GPB2	IO	Interrupt Request Input2 /GPIO B Inout2
IRQ3 /GPB3	IO	Interrupt Request Input3 /GPIO B Inout3
IRQ4 /GPB4	IO	Interrupt Request Input4 /GPIO B Inout4
IRQ5 /GPB5	IO	Interrupt Request Input5 /GPIO B Inout5
IRQ6 /GPB6	IO	Interrupt Request Input6 /GPIO B Inout6
IRQ7 /GPB7	IO	Interrupt Request Input7 /GPIO B Inout7
13. JTAG pins (total 5)		
TRST_	O	JTAG Reset. Pull-down Internally
TMS	IO	JTAG Mode Select
TDI	I	JTAG Serial Data Input
TCK	O	JTAG Clock, Pull-down Internally
TDO	O	JTAG Serial Data Output
14. SYSTEM pins (total 7)		
EXTAL	I	System Crystal Input
XTAL	O	Crystal Output
RESETOUT_	O	Core Reset output
RESETP_	I	System Power on reset input
RESETM_	I	System Manual reset input, Pull-up- Internally
TEST_SEN	I	scan enable for scan-reg

TEST_MODE	I	Chip test mode
15. Test port / MD pins (total 5)		
TEST_PORT0	IO	Test port 0
TEST_PORT1	IO	Test port 1
TEST_PORT2	IO	Test port 2
TEST_PORT3 /MD_PCI	IO	Test port 3 /MD_PCI: used as the PCI mode configuration when power-up, should be pull-up or pull-down with 4.7-10KΩ resistor for PCI mode configuration, it will be latched at the rising-edge of the RESETP_ signal 0: Satellite 1: Host (default, Pull-up internal)
TEST_PORT4 /MD_PCIARB	IO	Test port 3 /MD_PCIARB: used as the PCI arbiter mode configuration when power-up, should be pull-up or pull-down with 4.7-10KΩ resistor for PCI arbiter mode configuration, it will be latched at the rising-edge of the RESETP_ signal 0: Built-in Arbiter Enable (default pull-down internal) 1: Build-in Arbiter Disable
16. Power pins (total 64)		
VDDIO	POW	Power supply for IO pad(3.3 V) (10pin)
VDDCORE	POW	Power supply for core(1.8 V) (10pin)
VDD(PLL)	POW	PLL Power supply for digital(1.8 V) (2pin)
VSS(PLL)	POW	PLL ground supply for digital (2pin)
VDD (USB)	POW	USB power supply for IO and USB cell (analog 3.3V) (2pin)
VSS (USB)	POW	USB ground supply for IO and USB cell (analog 0 V) (2pin)
VSSIO	POW	Ground supply for IO pad(0 V) (10pin)
VSSCORE	POW	Ground supply for core(0 V) (10pin)
G/T	POW	Ground for thermal radiation (0V) (16pin)
NC	**NC**	**Not Connect Pin**

1.4 Package

The Arca210 package is shown below: