20.3-32-5000A

Data Sheet

KS32C5000A RISC Microcontroller

Revision 0.1

SAMSUNG

ELECTRONICS

INTRODUCTION

Samsung's KS32C5000AA 16/32-bit RISC microcontroller is a cost-effective, high-performance microcontroller solution for Ethernet-based systems. The KS32C5000AA is designed as an integrated Ethernet controller for use in managed communication hubs and routers.

The KS32C5000A is built around an outstanding CPU core: the 16/32-bit ARM7TDMI RISC processor designed by Advanced RISC Machines, Ltd. The ARM7TDMI core is a low-power, general purpose, microprocessor macro-cell that was developed for use in application-specific and custom-specific integrated circuits. Its simple, elegant, and fully static design is particularly suitable for cost-sensitive and power-sensitive applications.

To reduce total system cost, the KS32C5000A offers a configurable 8-Kbyte unified cache/SRAM and Ethernet controller. Most of the on-chip function blocks have been designed using an HDL synthesizer and the KS32C5000A has been fully verified in Samsung's state-of-the-art ASIC test environment.

Important peripheral functions include two HDLCs, two UART channels, 2-channel GDMA, two 32-bit timers, and 18 programmable I/O ports. On-board logic includes an interrupt controller, DRAM controller, and a controller for ROM/SRAM and flash memory. The System Manager includes an internal 32-bit system bus arbiter and an external memory controller.

The following integrated on-chip functions are described in detail in this user's manual:

- 8-Kbyte unified cache/SRAM
- I^2C interface
- Ethernet controller
- HDLC
- GDMA
- UART
- Timers
- Programmable I/O ports
- Interrupt controller

FEATURES

Architecture

- Integrated system for embedded Ethernet applications
- Fully 16/32-bit RISC architecture
- Little/Big-Endian memory system
- Efficient and powerful ARM7TDMI core
- Cost-effective JTAG-based debug solution
- Boundary scan

System Manager

- 8/16/32-bit external bus support for ROM/SRAM, flash memory, DRAM, and external I/O
- One external bus master with bus request/ acknowledge pins
- Support for EDO DRAM
- Programmable access cycle (0–7 wait cycles)
- Four-word depth write buffer
- Cost-effective memory-to-peripheral DMA interface

Unified Instruction/Data Cache

- Two-way, set-associative, unified 8-Kbyte cache
- Support for LRU (least recently used) protocol
- Cache is configurable as internal SRAM

I^{2}C Serial Interface

- Master mode operation only
- Baud rate generator for serial clock generation

Ethernet Controller

- DMA engine with burst mode
- DMA Tx/Rx buffers (256 bytes Tx, 256 bytes Rx)
- MAC Tx/Rx FIFO buffers (80 bytes Tx, 16 bytes Rx)
- Data alignment logic
- Endian translation
- 100/10-Mbit per second operation
- Full compliance with IEEE standard 802.3
- MII and 7-wire 10-Mbps interface
- Station management signaling
- On-chip CAM (up to 21 destination addresses)
- Full-duplex mode with PAUSE feature
- Long/short packet modes
- PAD generation

HDLCs

- HDLC protocol features:
  - Flag detection and synchronization
  - Zero insertion and deletion
  - Idle detection and transmission
  - FCS encoding and detection (16-bit)
  - Abort detection and transmission
- Address search mode (expandable to 4 bytes)
- Selectable CRC or No CRC mode
- Automatic CRC generator preset
- Digital PLL block for clock recovery
- Baud rate generator
- NRZ/NRZI/FM/Manchester data formats for Tx/Rx
- Loop-back and auto-echo modes
- Tx/Rx FIFOs have 8-word (8 × 32-bit) depth
- Selectable 1-word or 4-word data transfer mode
- Data alignment logic
- Endian translation
- Programmable interrupts
- Modem interface
- Up to 4 Mbps operation using an external receive clock
- Up to 2 Mbps (32-Mhz system clock) for FM
- DPLL encoding support
- Up to 1 Mbps (32-Mhz system clock) for NRZI encoding using DPLL
- HDLC frame length based on octets
- 2-channel DMA engine for Tx/Rx on each HDLC

DMA Controller

- 2-channel General DMA for memory-to-memory, memory-to-UART, UART-to-memory data transfers without CPU intervention
- Initiated by a software or external DMA request
- Increments or decrements source or destination address in 8-bit, 16-bit or 32-bit data transfers
- 4-data burst mode

UARTs

- Two UART (serial I/O) blocks with DMA-based or interrupt-based operation
- Support for 5-bit, 6-bit, 7-bit, or 8-bit serial data transmit and receive
- Programmable baud rates
- 1 or 2 stop bits
- Odd or even parity
- Break generation and detection
- Parity, overrun, and framing error detection
- External UART clock input
- 16bytes Rx FIFO
- ×16 clock mode
- Infra-red (IR) Tx/Rx support (IrDA)

Timers

- Two programmable 32-bit timers
- Interval mode or toggle mode operation

Programmable I/O

- 18 programmable I/O ports
- Pins individually configurable to input, output, or I/O mode for dedicated signals

Interrupt Controller

- 21 interrupt sources, including 4 external interrupt sources
- Normal or fast interrupt mode (IRQ, FIQ)
- Prioritized interrupt handling

Operating Voltage/Frequency

- V_{DD} = 4.75 V to 5.25 V @33MHz(T_A = 0 °C to + 70 °C)

Package Types

- 208-pin QFP

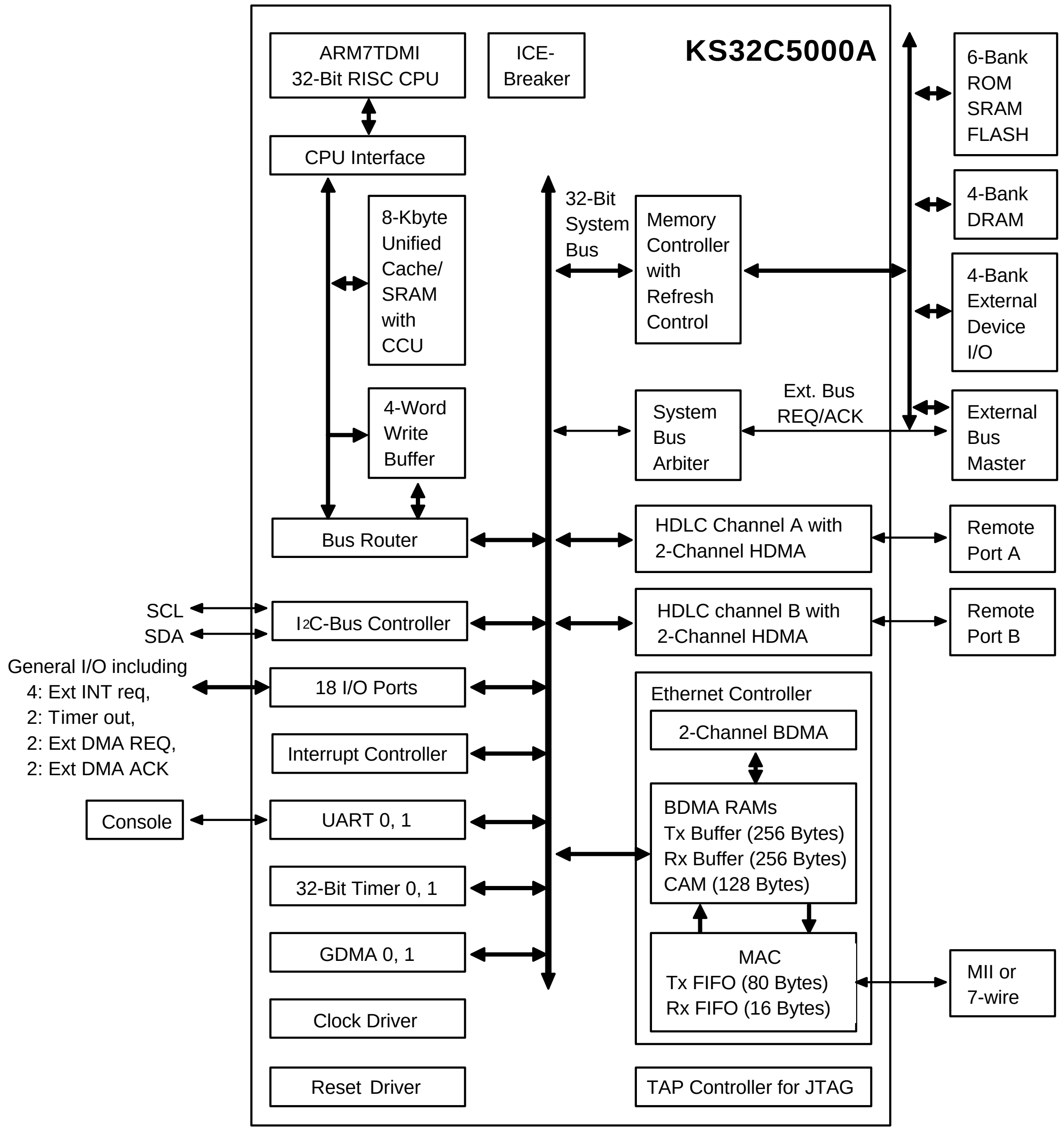


Figure 1. KS32C5000A Block Diagram

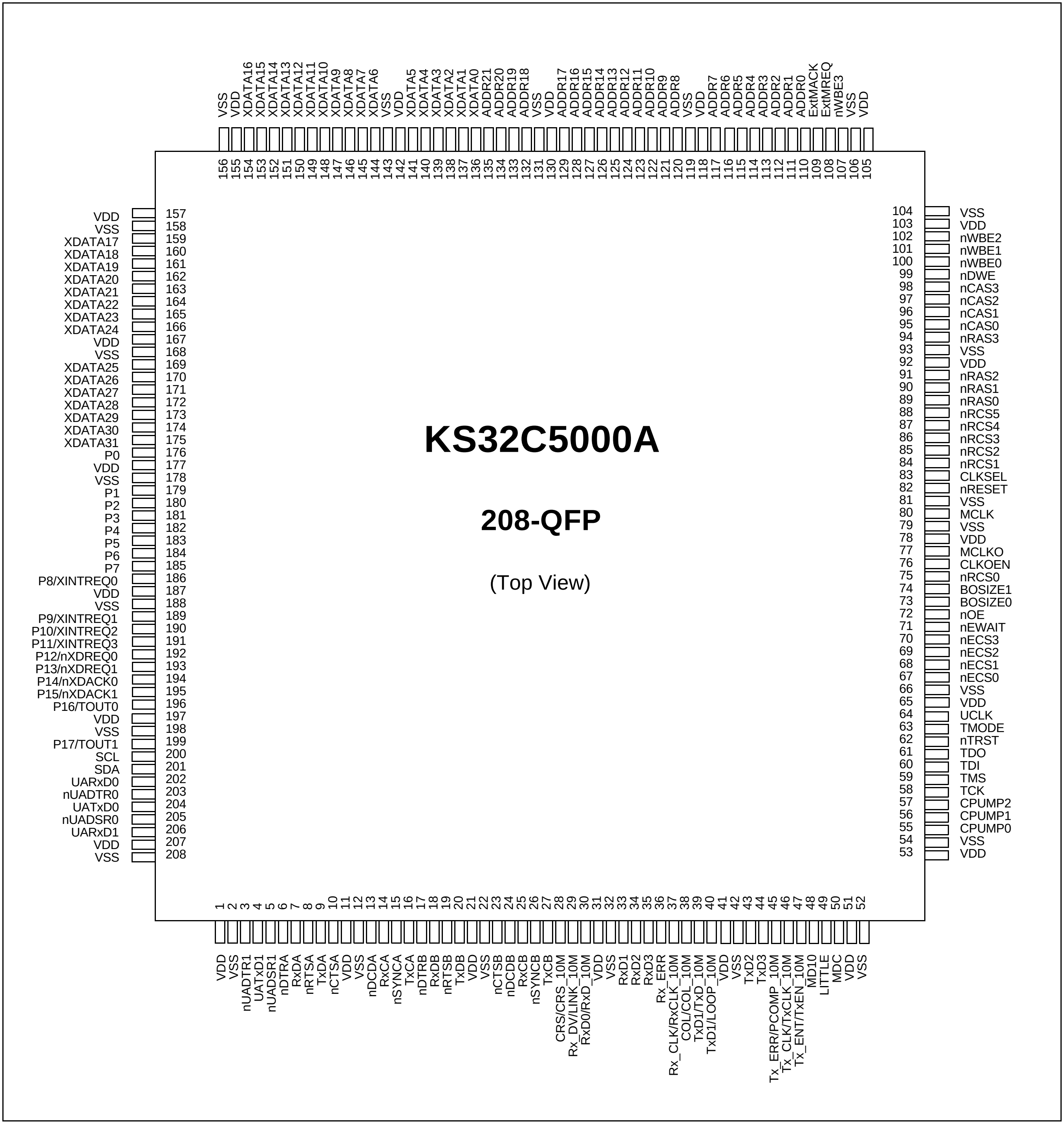


Figure 2. KS32C5000A Pin Assignment Diagram

SIGNAL DESCRIPTIONS

Table 1. KS32C5000A Signal Descriptions

Signal	Pin No.	Type	Description
MCLK	80	I	KS32C5000A System Source Clock. If CLKSEL is Low, MCLK is directly used as the KS32C5000A internal system clock. In this case, MCLK should have a 50% duty cycle. If CLKSEL is High, MCLK is divided by two and MCLK/2 is used as the KS32C5000A internal system clock. In this case, MCLK may not have a 50% duty cycle.
MCLKO	77	O	System Clock Out. MCLKO is monitored as the reversed phase of internal system clock, MCLK. (See Figure 1-3.)
CLKSEL	83	I	Clock Select. If CLKSEL is Low, MCLK is used as the master clock. If CLKSEL is High, MCLK/2 is used as the master clock.
nRESET	195	I	Not Reset. nRESET is the global reset input for the KS32C5000A. For a system reset, and to allow for internal digital filtering, nRESET must be held to Low level for at least 540 master clock cycles .
CLKOEN	76	I	Clock Out Enable/Disable. (See the pin description for MCLKO.)
TMODE	63	I	Test Mode. The TMODE bit settings are interpreted as follows: ’0’ = normal operating mode, ’1’ = core test mode.
UCLK	64	I	UCLK is External UART clock input pin. Cautious : UCLK pin have to be assigned Vdd or Vss level in case of not using external UART clock.
CPUMP	55–57	O	CPU Mode Mapping. These bits determine the CPU's operating mode. Illegal mode (’111’) indicates that the CPU has entered an unrecoverable state. If this occurs, a reset should be applied. The three CPU Mode Mapping pins are interpreted as follows: ’000’ = User mode, ’001’ = FIQ, ’010’ = IRQ, ’011’ = Supervisor, ’100’ = Abort, ’101’ = Undefined, ’110’ = System, and ’111’ = Illegal mode.
TCK	58	I	JTAG Test Clock. The JTAG test clock shifts state information and test data into, and out of, the KS32C5000A during JTAG test operations. This pin internally is pull-downed by internal resistor.
TMS	59	I	JTAG Test Mode Select. This pin controls JTAG test operations in the KS32C5000A. This pin is internally connected to pull-up resistor.
TDI	60	I	JTAG Test Data In. The TDI level is used to serially shift test data and instructions into the KS32C5000A during JTAG test operations. This pin is internally connected to pull-up resistor.
TDO	61	O	JTAG Test Data Out. The TDO level is used to serially shift test data and instructions out of the KS32C5000A during JTAG test operations.
nTRST	62	I	JTAG Not Reset. Asyncronous reset of the JTAG logic. This pin is internally connected to pull-up resistor.
ADDR[21:0]	117–110, 129–120, 135–132	O	Address Bus. The 22-bit address bus, ADDR[21:0], covers the full 4 M word address range of each ROM/SRAM, flash memory, and DRAM, and of the external I/O banks. Internal 23bit system address used to generate DRAM address. The number of column address in DRAM bank can be programmed 8bit to 11bits use by DRAMCON register.

Table 1. KS32C5000A Signal Descriptions

Signal	Pin No.	Type	Description
XDATA[31:0]	136–141, 144–154, 159–166, 169–175	I/O	External (bi-directional, 32-bit) Data Bus. The KS32C5000A data bus supports external 8-bit, 16-bit, and 32-bit bus sizes.
nRAS[3:0]	94, 91, 90, 89	O	Not Row Address Strobe for DRAM. The KS32C5000A supports up to four DRAM banks. One nRAS output is provided for each bank.
nCAS[3:0]	98, 97, 96, 95	O	Not column address strobe for DRAM. The four nCAS outputs indicate the byte selections whenenver a DRAM bank is accessed.
nDWE	99	O	DRAM Not Write Enable. This pin is provided for DRAM bank write operations. (nWBE[3:0] is used for write operations to the ROM/ SRAM/flash memory banks.)
nECS[3:0]	70, 69, 68, 67	O	Not External I/O Chip Select. Four external I/O banks are provided for external memory-mapped I/O operations. Each I/O bank stores up to 64 Kbytes. nECS signals indicate which of the four external I/O banks is selected.
nEWAIT	71	I	Not External Wait. This signal is activated when an external I/O device or ROM/SRAM/flash memory needs more access cycles than those defined in the corresponding control register.
nRCS[5:0]	88–84, 75	O	Not ROM/SRAM/Flash Chip Select. The KS32C5000A can access up to six external ROM/SRAM/Flash banks. By controlling the nRCS signals, you can map CPU addresses into the physical memory banks.
B0SIZE	74, 73	I	Bank 0 Data Bus Access Size. Bank 0 is used for the boot program. You use these pins to set the size of the bank 0 data bus, as follows: ’01’ = one byte, ’10’ = half-word, ’11’ = one word, and ’00’ = reserved.
nOE	72	O	Not Output Enable. Whenever a memory access occurs, the nOE output controls the output enable port of the specific memory device.
nWBE[3:0]	100–102, 107	O	Not Write Byte Enable. Whenever a memory write access occurs, the nWBE output controls the write enable port of the specific memory device (except for DRAM). For DRAM banks, CAS[3:0] and nDWE are used for the write operation.
ExtMREQ	108	I	External Bus Master Request. An external bus master uses this pin to request the external bus. When it activates the ExtMREQ signal, the KS32C5000A drives the state of external bus pins to high impedance. This lets the external bus master take control of the external bus. Which it has control, the external bus master assumes responsibility for DRAM refresh operations. The ExtMREQ signal is deactivated when the external bus master releases the external bus. When this occurs, ExtMACK goes Low level and the KS32C5000A assumes control of the bus.
ExtMACK	109	O	External Bus Acknowledge. (See the ExtMREQ pin description.)
MDC	50	O	Management Data Clock. The signal level at the MDC pin is used as a timing reference for data transfers that are controlled by the MDIO signal.

Table 1. KS32C5000A Signal Descriptions

Signal	Pin No.	Type	Description
MDIO	48	I/O	Management Data I/O. When a read command is being executed, data that is clocked out of the PHY is presented on this pin. When a write command is being executed, data that is clocked out of the controller is presented on this pin for the Physical Layer Entity, PHY.
LITTLE	49	I	Little endian mode select pin. If LITTLE is High, KS32C5000A operate in little endian mode. If Low, then in Big endian mode. Default value is low because this pin is pull-downed internally.
COL/COL_10M	38	I	Collision Detected/Collision Detected for 10M. COL is asserted asynchronously with minimum delay from the start of a collision on the medium in MII mode. COL_10M is asserted when a 10-Mbit/s PHY detects a collision.
TX_CLK/ TXCLK_10M	46	I	Transmit Clock/Transmit Clock for 10M. The controller drives TXD[3:0] and TX_EN from the rising edge of TX_CLK. In MII mode, the PHY samples TXD[3:0] and TX_EN on the rising edge of TX_CLK. For data transfers, TXCLK_10M is provided by the 10-Mbit/s PHY.
TXD[3:0] TXD_10M LOOP_10M	39, 40, 43, 44	O	Transmit Data/Transmit Data for 10 M/Loop-back for 10M. Transmit data is aligned on nibble boundaries. TXD[0] corresponds to the first bit to be transmitted on the physical medium, which is the LSB of the first byte and the fifth bit of that byte during the next clock. TXD_10M is shared with TXD[0] and is a data line for transmitting to the 10-Mbit/s PHY. LOOP_10M is shared with TXD[1] and is driven by the loop-back bit in the control register.
TX_EN/ TXEN_10M	47	O	Transmit Enable/Transmit Enable for 10M. TX_EN provides precise framing for the data carried on TXD[3:0]. This pin is active during the clock periods in which TXD[3:0] contains valid data to be transmitted, from the preamble stage through CRC. When the controller is ready to transfer data, it asserts TXEN_10M.
TX_ERR/ PCOMP_10M	45	O	Transmit Error/Packet Compression Enable for 10M. TX_ERR is driven synchronously to TX_CLK and is sampled continuously by the Physical Layer Entity, PHY. If asserted for one or more TX_CLK periods, TX_ERR causes the PHY to emit one or more symbols which are not part of the valid data or delimiter set somewhere in the frame that is being transmitted. PCOMP_10M is asserted immediately after the packet's DA field is received. PCOMP_10M is used with the Management Bus of the DP83950 Repeater Interface Controller (from National Semiconductor). The MAC can be programmed to assert PCOMP if there is a CAM match, or if there is not a match. The RIC (Repeater Interface Controller) uses this signal to compress (shorten) a received packet for management purposes and to reduce memory usage. (See the DP83950 Data Sheet, published by National Semiconductor, for details on the RIC Management Bus.) This pin should be controlled by a special register. Using this register, you can define the polarity and assertion method (CAM match active or not match active) of the PCOMP signal.

Table 1. KS32C5000A Signal Descriptions

Signal	Pin No.	Type	Description
CRS/CRS_10M	28	I	Carrier Sense/Carrier Sense for 10M. CRS is asserted asynchronously with minimum delay from the detection of a non-idle medium in MII mode. CRS_10M is asserted when a 10-Mbit/s PHY has data to transfer. A 10-Mbit/s transmission also uses this signal.
RX_CLK/ RXCLK_10M	37	I	Receive Clock/Receive Clock for 10M. RX_CLK is a continuous clock signal. Its frequency is 25 MHz for 100-Mbit/s operation, and 2.5 MHz for 10-Mbit/s. RXD[3:0], RX_DV, and RX_ERR are driven by the PHY off the falling edge of RX_CLK, and are sampled on the rising edge of RX_CLK. To receive data, the TXCLK_10 M clock comes from the 10-Mbit/s PHY.
RXD[3:0]/ RXD_10M	30, 33, 34, 35	I	Receive Data/Receive Data for 10M. RXD is aligned on nibble boundaries. RXD[0] corresponds to the first bit received on the physical medium, which is the LSB of the byte in one clock period and the fifth bit of that byte in the next clock. RXD_10M is shared with RXD[0] and is a line for receiving data from the 10-Mbit/s PHY.
RX_DV/ LINK_10M	29	I	Receive Data Valid/Link Status for 10M. PHY asserts RX_DV synchronously and holds it active during the clock periods in which RXD[3:0] contains valid received data. PHY asserts RX_DV no later than the clock period when it places the first nibble of the start frame delimiter (SFD) on RXD[3:0]. If PHY asserts RX_DV prior to the first nibble of the SFD, then RXD[3:0] carries valid preamble symbols. LINK_10M is shared with RX_DV and is used to convey the link status of the 10-Mbit/s endec. The value is stored in a status register.
RX_ERR	36	I	Receive Error. PHY asserts RX_ERR synchronously whenever it detects a physical medium error (e.g., a coding violation). PHY asserts RX_ERR only when it asserts RX_DV.
TXDA	9	O	HDLC Ch-A Transmit Data. The serial output data from the transmitter is coded in NRZ/NRZI/FM/Manchester data format.
RXDA	7	I	HDLC Ch-A Receive Data. The serial input data received by the device should be coded in NRZ/NRZI/FM/Manchester data format. The data rate should not exceed the rate of the KS32C5000A internal master clock.
nDTRA	6	O	HDLC Ch-A Data Terminal Ready. nDTRA output indicates that the data terminal device is ready for transmission and reception.
nRSTA	8	O	HDLC Ch-A Request To Send. The nRSTA output is controlled by the Request to Send control bit. When a RTS bit goes High, the nRTS output is driven Low. When the RTS bit returns to Low level, the nRTS output remains Low until 1) the end of frame is reached, and 2) there is no more data in the TxFIFO for a new frame.
nCTSA	10	I	HDLC Ch-A Clear To Send. The KS32C5000A stores each positive transition of nCTS to ensure that its occurrence will be acknowledged by the system.

Table 1. KS32C5000A Signal Descriptions

Signal	Pin No.	Type	Description
nDCDA	13	I	HDLC Ch-A Data Carrier Detected. A High level on this pin resets and inhibits the receiver register. Data from a previous frame that may remain in the RxFIFO is retained. The KS32C5000A stores each positive transition of nDCD.
nSYNCA	15	I	HDLC Ch-A Sync is detected. This indicates the reception of a flag. The nSYNC output goes low for one bit time beginning at the last bit of the flag.
RXCA	14	I	HDLC Ch-A Receiver Clock. When this clock input is used as the receiver clock, the receiver samples the data on the positive edge of RXCA clock. This clock can be the source clock of the receiver, the baud rate generator, or the DPLL.
TXCA	16	I/O	HDLC Ch-A Transmitter Clock. When this clock input is used as the transmitter clock, the transmitter shifts data on each negative transition of the TXCA clock input. If you do not use TXCA as the transmitter clock, you can use it as an output pin for monitoring internal clocks such as the transmitter clock, receiver clock, and baud rate generator output clocks.
TXDB	20	O	HDLC Ch-B Transmit Data. See the TXDA pin description.
RXDB	18	I	HDLC Ch-B Receive Data. See the RXDA pin description.
nDTRB	17	O	HDLC Ch-B Data Terminal Ready. See the nDTRA pin description.
nRTSB	19	O	HDLC Ch-B Request To Send. See the nRTSA pin description.
nCTSB	23	I	HDLC Ch-B Clear To Send. See the nCTSA pin description.
nDCDB	24	I	HDLC Ch-B Data Carrier Detected. See the nDCDA pin description.
nSYNCB	26	O	HDLC Ch-B Sync is detected. See the nSYNCA pin description.
RXCB	25	I	HDLC Ch-B Receiver Clock. See the RXCA pin description.
TXCB	27	I/O	HDLC Ch-B Transmitter Clock. See the TXCA pin description.
UARXD0	202	I	UART0 Receive Data. RXD0 is the UART 0 input signal for receiving serial data.
UATXD0	204	O	UART0 Transmit Data. TXD0 is the UART 0 output signal for transmitting serial data.
nUADTR0	203	I	Not UART0 Data Terminal Ready. This input signals the KS32C5000A that the peripheral (or host) is ready to transmit or receive serial data.
nUADSR0	205	O	Not UART0 Data Set Ready. This output signals the host (peripheral) that UART 0 is ready to transmit or receive serial data.
UARXD1	206	I	UART1 Receive Data. See the RXD0 pin description.
UATXD1	4	O	UART1 Transmit Data. See the TXD0 pin description.
nUADTR1	3	I	Not UART1 Data Terminal Ready. See the DTR0 pin description.
nUADSR1	5	O	Not UART1 Data Set Ready. See the DSR0 pin description.

Table 1. KS32C5000A Signal Descriptions

Signal	Pin No.	Type	Description
P[7:0]	185–179, 176	I/O	General I/O ports.
xINTREQ[3:0] P[11:8]	191–189, 186	I/O	External interrupt request lines or general I/O ports.
xDREQ[1:0]/ P[13:12]	193, 192	I/O	External DMA requests for GDMA or general I/O ports.
nXDACK[1:0] P[15:14]	195, 194	I/O	External DMA acknowledge from GDMA or general I/O ports.
TOUT0/P[16]	196	I/O	Timer 0 out or general I/O port.
TOUT1/P[17]	199	I/O	Timer 1 out or general I/O port.
SCL	200	I/O	I^2C serial clock.
SDA	201	I/O	I^2C serial data.
VDDP	1,21,41,53, 78,103,118, 142,157,177 ,197	Power	I/O pad power
VDDI	11,31,51,65, 92,105,130, 155,167,187 ,207	Power	Internal core power
VSSP	2,22,42,54, 79,93,106, 131,156,168 ,188,208	GND	I/O pad ground
VSSI	12,32,52,66, 81,104,119, 143,158,178 ,198	GND	Internal core ground

Table 2. KS32C5000A Pin List and PAD Type

Group	Pin Name	Pin Counts	I/O Type	Pad Type	Description
System Configuration (11)	MCLK	1	I	pit	KS32C5000A system source clock.
	MCLKO	1	O	pob4	System clock out.
	CLKSEL	1	I	pit	Clock select.
	nRESET	1	I	pil	Not reset.
	CLKOEN	1	I	pit	Clock out enable/disable.
	TMODE	1	I	pit	Test mode.
	CPUMP[2:0]	3	O	pob1	CPU mode mapping.
	LITTLE	1	I	pitd	Little endian mode select pin.
TAP Control (5)	TCK	1	I	pitd	JTAG test clock.
	TMS	1	I	pitu	JTAG test mode select.
	TDI	1	I	pitu	JTAG test data in.
	TDO	1	I	pot2	JTAG test data out.
	nTRST	1	I	pitu	JTAG not reset.
Memory Interface (83)	ADDR[21:0]	22	O	pot8sm	Address bus.
	XDATA[31:0]	32	I/O	pblut8sm	External, bi-directional, 32-bit data bus.
	nRAS[3:0]	4	O	pot4	Not row address strobe for DRAM.
	nCAS[3:0]	4	O	pot4	Not column address strobe for DRAM.
	nDWE	1	O	pot4	Not write enable
	nECS[3:0]	4	O	pot4	Not external I/O chip select.
	nEWAIT	1	I	pit	Not external wait signal.
	nRCS[5:0]	6	O	pot4	Not ROM/SRAM/flash chip select.
	B0SIZE[1:0]	2	I	pit	Bank 0 data bus access size.
	nOE	1	O	pot4	Not output enable.
	nWBE[3:0]	4	O	pot4	Not write byte enable.
	ExtMREQ	1	I	pit	External master bus request.
	ExtMACK	1	O	pob1	External bus acknowledge.

Table 2. KS32C5000A Pin List and PAD Type

Group	Pin Name	Pin Counts	I/O Type	Pad Type	Description
Ethernet Controller (18)	MDC	1	O	pob4	Management data clock.
	MDIO	1	I/O	pbtut4	Management data I/O.
	COL/ COL_10M	1	I	pil	Collision detected/collision detected for 10M.
	TX_CLK/ TXCLK_10M	1	I	pil	Transmit data/transmit data for 10M.
	TXD[3:0]/ TXD_10M/ LOOP_10M	4	O	pob4	Transmit data/transmit data for 10M.
	TX_EN/ TXEN_10M	1	O	pob4	Transmit enable or transmit enable for 10M.
	TX_ERR/ PCOMP_10M	1	O	pob4	Transmit error/packet compression enable for 10M.
	CRS/ CRS_10M	1	I	pil	Carrier sense/carrier sense for 10M.
	RX_CLK/ RXCLK_10M	1	I	pil	Receive clock/receive clock for 10M.
	RXD[3:0]/ RXD_10M	4	I	pil	Receive data/receive data for 10M.
	RX_DV/ LINK_10M	1	I	pil	Receive data valid.
	RX_ERR	1	I	pil	Receive error.
HDLC Channel A (9)	TXDA	1	O	pob4	HDLC channel A transmit data.
	RXDA	1	I	pil	HDLC channel A receive data.
	nDTRA	1	O	pob4	HDLC channel A data terminal ready.
	nRTSA	1	O	pob4	HDLC channel A request to send.
	nCTSA	1	I	pil	HDLC channel A clear to send.
	nDCDA	1	I	pil	HDLC channel A data carrier detected.
	nSYNCA	1	O	pob4	HDLC channel A sync is detected.
	RXCA	1	I	pil	HDLC channel A receiver clock.
	TXCA	1	I/O	pblut1	HDLC channel A transmitter clock.

Table 2. KS32C5000A Pin List and PAD Type

Group	Pin Name	Pin Counts	I/O Type	Pad Type	Description
HDLC Channel B (9)	TXDB	1	O	pob4	HDLC channel B transmit data.
	RXDB	1	I	pil	HDLC channel B receive data.
	nDTRB	1	O	pob4	HDLC channel B data terminal ready.
	nRTSB	1	O	pob4	HDLC channel B request to send.
	nCTSB	1	I	pil	HDLC channel B clear to send.
	nDCDB	1	I	pil	HDLC channel B data carrier detected.
	nSYNCB	1	O	pob4	HDLC channel B sync is detected.
	RXCB	1	I	pil	HDLC channel B receiver clock.
	TXCB	1	I/O	pblut1	HDLC channel B transmitter clock.
UART 0 (4)	UCLK	1	I	pit	UART external clock input.
	UARXD0	1	I	pit	UART 0 receive data.
	UATXD0	1	O	pob4	UART 0 transmit data.
	nUADTR0	1	I	pit	Not UART 0 data terminal ready.
	nUADSR0	1	O	pob4	Not UART0 data set ready.
UART 1 (4)	UARXD1	1	I	pit	UART 1 receive data.
	UATXD1	1	O	pob4	UART 1 transmit data.
	nUADTR1	1	I	pit	Not UART 1 data terminal ready.
	nUADSR1	1	O	pob4	Not UART 1 data set ready.
General-Purpose I/O Ports, xINTREQ, nXDACK, Timer 0, 1 (18)	P[7:0]	8	I/O	pblt4sm	General I/O ports.
	xINTREQ[3:0] /P[11:8]	4	I/O	pblt4sm	External interrupt requests or general I/O ports.
	xXDREQ[1:0]/ P[13:12]	2	I/O	pblt4sm	External DMA requests for GDMA or general I/O ports.
	nXDACK[1:0]/ P[15:14]	2	I/O	pblt4sm	External DMA acknowledge from GDMA or general I/O ports.
	TIMER0/P[16]	1	I/O	pblt4sm	Timer 0 out or general I/O port.
	TIMER1/P[17]	1	I/O	pblt4sm	Timer 1 out or general I/O port.
I^{2}C (2)	SCL	1	I/O	pbtd4	I^{2}C serial clock.
	SDA	1	I/O	pbtd4	I^{2}C serial data.

Table 3. KS32C5000A PAD Type

Pad Type	I/O Type	Current Drive	Cell Type	Feature	Slew-Rate Control
pit	I	–	TTL level	–	–
pitu	I	–	TTL level	Pull-up	–
pitd	I	–	TTL level	Pull-down	–
pil	I	–	TTL schmitt trigger level	–	–
pob1	O	1 mA	–	–	–
pob4	O	4 mA	–	–	–
pot4	O	4 mA	–	–	–
pot8sm	O	8 mA	–	Tri-state	–
pbtd4	I/O	4 mA	TTL level	Open-drain	Medium
pbtut4	I/O	4 mA	TTL level	Tri-state	–
pblut1	I/O	1 mA	TTL schmitt trigger level	Tri-state pull-up	–
pblt4sm	I/O	4 mA	TTL schmitt trigger level	Tri-state	Medium
pblut8sm	I/O	8 mA	TTL Schmitt trigger level	Tri-state pull-up	Medium

NOTE For detailed information about pad types, see input/output cell on Chapter 4 in the "STD85/STDM85 0.5-Micron 5V/3V Standard Cell Library Data Book," published by the ASIC Team of Samsung Electronics Co., Ltd., Kiheung, South Korea.

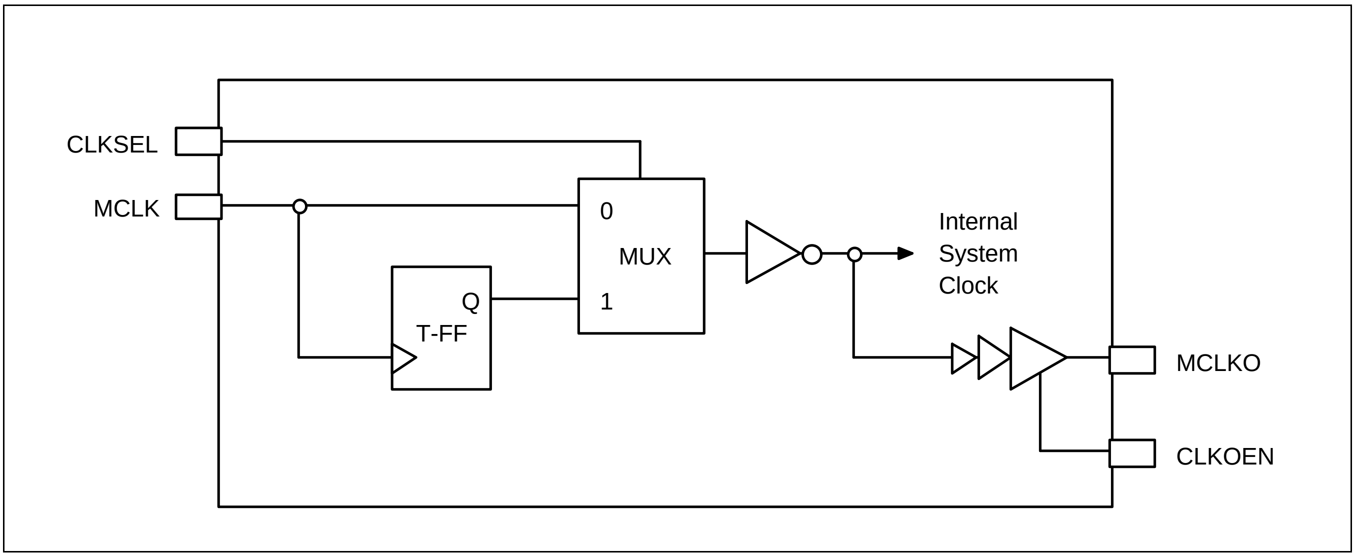


Figure 3. System Clock Circuit

NOTE

In the system clock circuit shown in Figure 3, MCLK is delayed and inverted out to the MCLKO pin. The elapsed time of the rising and falling edge ranges from 17.1 ns (best case) to 16.95 ns (worst case), given V_{DD} = 4.5 V, ambient temperature = 70° F, and load capacitance = 50 pF.

CPU CORE OVERVIEW

The KS32C5000A CPU core is the ARM7TDMI processor, a general purpose, 32-bit microprocessor developed by Advanced RISC Machines, Ltd. (ARM). The core's architecture is based on Reduced Instruction Set Computer (RISC) principles. The RISC architecture makes the instruction set and its related decoding mechanisms simpler and more efficient than with microprogrammed Complex Instruction Set Computer (CISC) systems. The resulting benefit is high instruction throughput and impressive real-time interrupt response. Pipelining is also employed so that all components of the processing and memory systems can operate continuously. The ARM7TDMI has a 32-bit address bus.

An important feature of the ARM7TDMI processor, and one which differentiates it from the ARM7 processor, is a unique architectural strategy called *THUMB*. The THUMB strategy is an extension of the basic ARM architecture and consists of 36 instruction formats. These formats are based on the standard 32-bit ARM instruction set, but have been re-coded using 16-bit wide opcodes.

Because THUMB instructions are one-half the bit width of normal ARM instructions, they produce very high-density code. When a THUMB instruction is executed, its 16-bit opcode is decoded by the processor into its equivalent instruction in the standard ARM instruction set. The ARM core then processes the 16-bit instruction as it would a normal 32-bit instruction. In other words, the Thumb architecture gives 16-bit systems a way to access the 32-bit performance of the ARM core without incurring the full overhead of 32-bit processing.

Because the ARM7TDMI core can execute both standard 32-bit ARM instructions and 16-bit Thumb instructions, it lets you mix routines of Thumb instructions and ARM code in the same address space. In this way, you can adjust code size and performance, routine by routine, to find the best programming solution for a specific application.

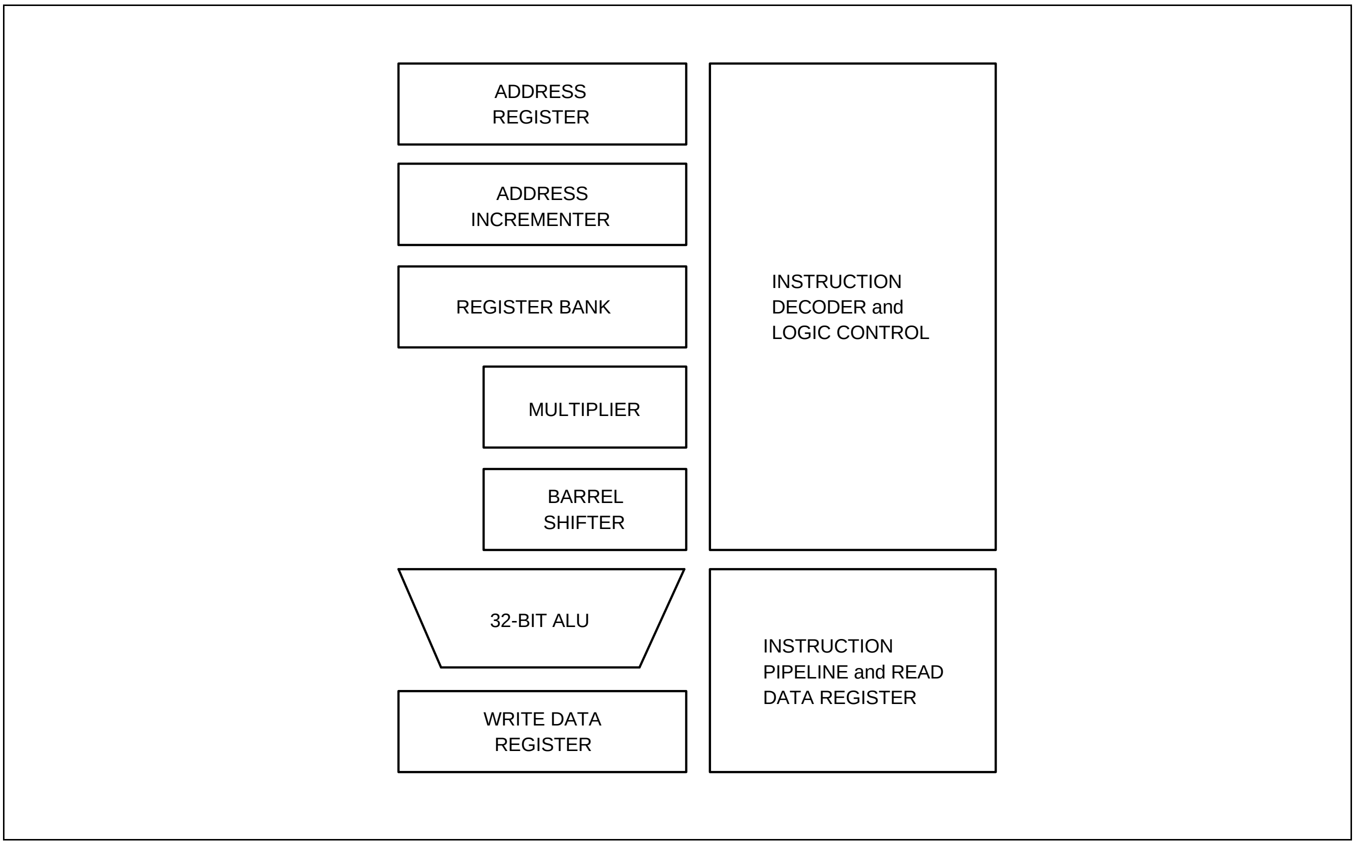


Figure 4. ARM7TDMI Core Block Diagram

INSTRUCTION SET

The KS32C5000A instruction set is divided into two subsets: a standard *32-bit ARM instruction set* and a *16-bit THUMB instruction set*.

The 32-bit ARM instruction set is comprised of thirteen basic instruction types which can, in turn, be divided into four broad classes:

- Four types of branch instructions which control program execution flow, instruction privilege levels, and switching between ARM code and THUMB code.
- Three types of data processing instructions which use the on-chip ALU, barrel shifter, and multiplier to perform high-speed data operations in a bank of 31 registers (all with 32-bit register widths).
- Three types of load and store instructions which control data transfer between memory locations and the registers. One type is optimized for flexible addressing, another for rapid context switching, and the third for swapping data.
- Three types of co-processor instructions which are dedicated to controlling external co-processors. These instructions extend the off-chip functionality of the instruction set in an open and uniform way.

NOTE

All 32-bit ARM instructions can be executed conditionally.

The 16-bit THUMB instruction set contains 36 instruction formats drawn from the standard 32-bit ARM instruction set. The THUMB instructions can be divided into four functional groups:

- Four branch instructions.
- Twelve data processing instructions, which are a subset of the standard ARM data processing instructions.
- Eight load and store register instructions.
- Four load and store multiple instructions.

NOTE

Each 16-bit THUMB instruction has a corresponding 32-bit ARM instruction with the identical processing model.

The 32-bit ARM instruction set and the 16-bit THUMB instruction sets are good targets for compilers of many different high-level languages. When assembly code is required for critical code segments, the ARM programming technique is straightforward, unlike that of some RISC processors which depend on sophisticated compiler technology to manage complicated instruction interdependencies.

Pipelining is employed so that all parts of the processor and memory systems can operate continuously. Typically, while one instruction is being executed, its successor is being decoded, and a third instruction is being fetched from memory.

MEMORY INTERFACE

The CPU memory interface has been designed to allow the highest performance potential to be realized without incurring high costs in the memory system. Speed-critical control signals are pipelined so that system control functions can be implemented in standard low-power logic. These pipelined control signals let you fully exploit the fast local access modes offered by industry standard dynamic RAMs.

OPERATING STATES

From a programmer’s point of view, the ARM7TDMI core is always in one of two operating states. These states, which can be switched by software or by exception processing, are:

- *ARM state* (when executing 32-bit, word-aligned, ARM instructions), and
- *THUMB state* (when executing 16-bit, half-word aligned THUMB instructions).

OPERATING MODES

The ARM7TDMI core supports seven operating modes:

- *User mode*: the normal program execution state
- *FIQ (Fast Interrupt Request) mode*: for supporting a specific data transfer or channel process
- *IRQ (Interrupt ReQuest) mode*: for general purpose interrupt handling
- *Supervisor mode*: a protected mode for the operating system
- *Abort mode*: entered when a data or instruction pre-fetch is aborted
- *System mode*: a privileged user mode for the operating system
- *Undefined mode*: entered when an undefined instruction is executed

Operating mode changes can be controlled by software, or they can be caused by external interrupts or exception processing. Most application programs execute in User mode. Privileged modes (that is, all modes other than User mode) are entered to service interrupts or exceptions, or to access protected resources.

REGISTERS

The KS32C5000A CPU core has a total of 37 registers: 31 general-purpose, 32-bit registers, and 6 status registers. Not all of these registers are always available. Which registers are available to the programmer at any given time depends on the current processor operating state and mode.

NOTE

> When the KS32C5000A is operating in ARM state, 16 general registers and one or two status registers can be accessed at any time. In privileged mode, mode-specific banked registers are switched in.

Two register sets, or banks, can also be accessed, depending on the core's current state: the *ARM state register set* and the *THUMB state register set*:

- The ARM state register set contains 16 directly accessible registers: R0–R15. All of these registers, except for R15, are for general-purpose use, and can hold either data or address values. An additional (seventeenth) register, the CPSR (Current Program Status Register), is used to store status information.
- The THUMB state register set is a subset of the ARM state set. You can access eight general registers, R0–R7, as well as the program counter (PC), a stack pointer register (SP), a link register (LR), and the CPSR. Each privileged mode has a corresponding banked stack pointer, link register, and saved process status register (SPSR).

The THUMB state registers are related to the ARM state registers as follows:

- THUMB state R0–R7 registers and ARM state R0–R7 registers are identical
- THUMB state CPSR and SPSRs and ARM state CPSR and SPSRs are identical
- THUMB state SP, LR, and PC map directly to ARM state registers R13, R14, and R15, respectively

In THUMB state, registers R8–R15 are not part of the standard register set. However, you can access them for assembly language programming and use them for fast temporary storage, if necessary.

EXCEPTIONS

An *exception* arises whenever the normal flow of program execution is interrupted. For example, when processing must be diverted to handle an interrupt from a peripheral. The processor's state just prior to handling the exception must be preserved so that the program flow can be resumed when the exception routine is completed. Multiple exceptions may arise simultaneously.

To process exceptions, the KS32C5000A uses the banked core registers to save the current state. The old PC value and the CPSR contents are copied into the appropriate R14 (LR) and SPSR register. The PC and mode bits in the CPSR are forced to a value which corresponds to the type of exception being processed.

The KS32C5000A core supports seven types of exceptions. Each exception has a fixed priority and a corresponding privileged processor mode, as shown in Table 4.

Table 4. KS32C5000A CPU Exceptions

Exception	Mode on Entry	Priority
Reset	Supervisor mode	1 (Highest)
Data abort	Abort mode	2
FIQ	FIQ mode	3
IRQ	IRQ mode	4
Prefetch abort	Abort mode	5
Undefined instruction	Undefined mode	6 (Lowest)
SWI	Supervisor mode	6

SPECIAL REGISTERS

Table 5. KS32C5000A Special Registers

Group	Registers	Offset	R/W	Description	Reset Value
System Manager	SYSCFG	0x0000	R/W	System configuration register	0x3FFFF91
	CLKCON	0x3000	R/W	Clock control register	0x00000000
	EXTACON0	0x3008	R/W	External I/O timing register 1	0x00000000
	EXTACON1	0x300C	R/W	External I/O timing register 2	0x00000000
	EXTDBWTH	0x3010	R/W	Data bus width for each memory bank	0x00000000
	ROMCON0	0x3014	R/W	ROM/SRAM/Flash bank 0 control register	0x20000060
	ROMCON1	0x3018	R/W	ROM/SRAM/Flash bank 1 control register	0x00000060
	ROMCON2	0x301C	R/W	ROM/SRAM/Flash bank 2 control register	0x00000060
	ROMCON3	0x3020	R/W	ROM/SRAM/Flash bank 3 control register	0x00000060
	ROMCON4	0x3024	R/W	ROM/SRAM/Flash bank 4 control register	0x00000060
	ROMCON5	0x3028	R/W	ROM/SRAM/Flash bank 5 control register	0x00000060
	DRAMCON0	0x302C	R/W	DRAM bank 0 control register	0x00000000
	DRAMCON1	0x3030	R/W	DRAM bank 1 control register	0x00000000
	DRAMCON2	0x3034	R/W	DRAM bank 2 control register	0x00000000
	DRAMCON3	0x3038	R/W	DRAM bank 3 control register	0x00000000
	REFEXTCON	0x303C	R/W	Refresh and external I/O control register	0x83FD0000
Ethernet (BDMA)	BDMATXCON	0x9000	R/W	Buffered DMA receive control register	0x00000000
	BDMARXCON	0x9004	R/W	Buffered DMA transmit control register	0x00000000
	BDMATXPTR	0x9008	R/W	Transmit trame descriptor start address	0x00000000
	BDMARXPTR	0x900C	R/W	Receive frame descriptor start address	0x00000000
	BDMARXLSZ	0x9010	R/W	Receive frame maximum size	Undefined
	BDMASTAT	0x9014	R/W	Buffered DMA status	0x00000000
	CAM	0x9100–0x917C	W	CAM content (32 words)	Undefined
	BDMATXBUF	0x9200–0x92FC	R/W	BDMA Tx buffer (64 words) for test mode addressing	Undefined
	BDMARXBUF	0x9800–0x99FC	R/W	BDMA Rx buffer (64 words) for test mode addressing	Undefined

Table 5. KS32C5000A Special Registers

Group	Registers	Offset	R/W	Description	Reset Value
Ethernet (MAC)	MACON	0xA000	R/W	Ethernet MAC control register	0x00000000
	CAMCON	0xA004	R/W	CAM control register	0x00000000
	MACTXCON	0xA008	R/W	MAC transmit control register	0x00000000
	MACTXSTAT	0xA00C	R/W	MAC transmit status register	0x00000000
	MACRXCON	0xA010	R/W	MAC receive control register	0x00000000
	MACRXSTAT	0xA014	R/W	MAC receive status register	0x00000000
	STADATA	0xA018	R/W	Station management data	0x00000000
	STACON	0xA01C	R/W	Station management control and address	0x00008000
	CAMEN	0xA028	R/W	CAM enable register	0x00000000
	EMISSCNT	0xA03C	R/W	Missed error count register	0x00000000
	EPZCNT	0xA040	R	Pause count register	0x00000000
	ERMPZCNT	0xA044	R	Remote pause count register	0x00000000
	ETXSTAT	0xA048	R	Transmit control frame status	0x00000000
HDLC Channel A	HCON0	0x7000	W	HDLC control register 0	0x00000000
	HCON1	0x7004	W	HDLC control register 1	0x00000000
	HSTAT	0x7008	R	HDLC status register	0x00010400
	HINTEN	0x700C	R/W	HDLC interrupt enable register	0x00000000
	HTXFIFOC	0x7010	W	TxFIFO frame base pointer register	–
	HTXFIFOT	0x7014	W	TxFIFO frame limit pointer register	–
	HRXFIFO	0x7018	R	HDLC receive FIFO entry register	0x00000000
	HSADR	0x701C	R/W	HDLC station address register	0x00000000
	HBRG	0x7020	R/W	HDLC baud rate generator time constant	0x00000000
	HPRMB	0x7024	R/W	HDLC preamble constant	0x00000000
HDLC (HDMA Channel A)	HDMATXMA	0x7028	R/W	HDMA memory address for Tx	0x00000000
	HDMARXMA	0x702C	R/W	HDMA memory address for Rx	0x00000000
	HDMATXCNT	0x7030	R/W	HDMA transfer count for Tx	0x00000000
	HDMARXCNT	0x7034	R/W	HDMA transfer count for Rx	0x00000000
	HDMABCNT	0x7038	R/W	HDMA transferred bytes count for Rx	0x00000000

Table 5. KS32C5000A Special Registers

Group	Registers	Offset	R/W	Description	Reset Value
HDLC Channel B	HCON0	0x8000	W	HDLC control register 0	0x00000000
	HCON1	0x8004	W	HDLC control register 1	0x00000000
	HSTAT	0x8008	R	HDLC status register	0x00010400
	HINTEN	0x800C	R/W	HDLC interrupt enable register	0x00000000
	HTXFIFOC	0x8010	W	TxFIFO frame base pointer register	–
	HTXFIFOT	0x8014	W	TxFIFO frame limit pointer register	–
	HRXFIFO	0x8018	R	HDLC receive FIFO entry register	0x00000000
	HSADR	0x801C	R/W	HDLC station address register	0x00000000
	HBRG	0x8020	R/W	HDLC baud rate generator time constant	0x00000000
	HPRMB	0x8024	R/W	HDLC preamble constant	0x00000000
HDLC (HDMA Channel B)	HDMATXMA	0x8028	R/W	HDMA memory address for Tx	0x00000000
	HDMARXMA	0x802C	R/W	HDMA memory address for Rx	0x00000000
	HDMATXCNT	0x8030	R/W	HDMA transfer count for Tx	0x00000000
	HDMARXCNT	0x8034	R/W	HDMA transfer count for Rx	0x00000000
	HDMABCNT	0x8038	R/W	HDMA transferred bytes count for Rx	0x00000000
I/O Ports	IOPMOD	0x5000	R/W	I/O port mode register	0x00000000
	IOPCON	0x5004	R/W	I/O port control register	0x00000000
	IOPDATA	0x5008	R/W	Input port data register	Undefined
Interrupt Controller	INTMOD	0x4000	R/W	Interrupt mode register	0x00000000
	INTPND	0x4004	R/W	Interrupt pending register	0x00000000
	INTMSK	0x4008	R/W	Interrupt mask register	0x003FFFFF
	INTPRI0	0x400C	R/W	Interrupt priority register 0	0x03020100
	INTPRI1	0x4010	R/W	Interrupt priority register 1	0x07060504
	INTPRI2	0x4014	R/W	Interrupt priority register 2	0x0B0A0908
	INTPRI3	0x4018	R/W	Interrupt priority register 3	0x0F0E0D0C
	INTPRI4	0x401C	R/W	Interrupt priority register 4	0x13121110
	INTPRI5	0x4020	R/W	Interrupt priority register 5	0x00000014
	INTOFFSET	0x4024	R	Interrupt offset address register	Undefined
I^{2}C Bus	IICCON	0xF000	R/W	I^{2}C bus control status register	0x00000000
	IICBUF	0xF004	R/W	I^{2}C bus shift buffer register	Undefined
	IICPS	0xF008	R/W	I^{2}C bus prescaler register	0x00000000
	IICCOUNT	0xF00C	R	I^{2}C bus prescaler counter register	0x00000000

Table 5. KS32C5000A Special Registers

Group	Registers	Offset	R/W	Description	Reset Value
GDMA	GDMACON0	0xB000	R/W	GDMA channel 0 control register	0x00000000
	GDMACON1	0xC000	R/W	GDMA channel 1 control register	0x00000000
	GDMASRC0	0xB004	R/W	GDMA source address register 0	Undefined
	GDMADST0	0xB008	R/W	GDMA destination address register 0	Undefined
	GDMASRC1	0xC004	R/W	GDMA source address register 1	Undefined
	GDMADST1	0xC008	R/W	GDMA destination address register 1	Undefined
	GDMACNT0	0xB00C	R/W	GDMA channel 0 transfer count register	Undefined
	GDMACNT1	0xC00C	R/W	GDMA channel 1 transfer count register	Undefined
UART	ULCON0	0xD000	R/W	UART channel 0 line control register	0x00000000
	ULCON1	0xE000	R/W	UART channel 1 line control register	0x00000000
	UCON0	0xD004	R/W	UART channel 0 control register	0x00000000
	UCON1	0xE004	R/W	UART channel 1 control register	0x00000000
	USTAT0	0xD008	R	UART channel 0 status register	0x000000C0
	USTAT1	0xE008	R	UART channel 1 status register	0x000000C0
	UTXBUF0	0xD00C	W	UART channel 0 transmit holding register	Undefined
	UTXBUF1	0xE00C	W	UART channel 1 transmit holding register	Undefined
	URXBUF0	0xD010	W	UART channel 0 receive buffer register	Undefined
	URXBUF1	0xE010	W	UART channel 1 receive buffer register	Undefined
	UBRDIV0	0xD014	R/W	Baud rate divisor register 0	0x00000000
	UBRDIV1	0xE014	R/W	Baud rate divisor register 1	0x00000000
Timers	TMOD	0x6000		Timer mode register	0x00000000
	TDATA0	0x6004		Timer 0 data register	0x00000000
	TDATA1	0x6008		Timer 1 data register	0x00000000
	TCNT0	0x600C		Timer 0 count register	0xffffffff
	TCNT1	0x6010		Timer 1 count register	0xffffffff

ELECTRICAL DATA

This chapter describes the KS32C5000A electrical data.

ABSOLUTE MAXIMUM RATINGS

Table 6. Absolute Maximum Ratings

Symbol	Parameter	Rating	Unit
V_{DD}	DC supply voltage	– 0.3 to 7	V
V_{IN}	DC input voltage	– 0.3 to V_{DD} + 0.3	V
I_{IN}	DC input current	± 10	mA
T_{STG}	Storage temperature	– 40 to 125	° C

RECOMMENDED OPERATING CONDITIONS

Table 7. Recommended Operating Conditions

Symbol	Parameter	Rating	Unit
V_{DD}	DC supply voltage	4.75 to 5.25	V
T_A	Commercial temperature	0 to 70	° C

D.C. ELECTRICAL CHARACTERISTICS

Table 8. D.C. Electrical Characteristics

Symbol	Parameter		Condition	Min	Typ	Max	Unit
V_{IH}	High level input voltage	CMOS		3.5			V
		TTL		2.0			
V_{IL}	Low level input voltage	CMOS				1.5	V
		TTL				0.8	
VT	Switching threshhold	CMOS			2.5		V
		TTL			1.4		
VT+	Schmitt trigger positive-going threshold	CMOS				4.0	V
		TTL				2.0	
VT–	Schmitt trigger negative-going threshold	CMOS		1.0			V
		TTL		0.8			
I_{IH}	High level input current	Input buffer	$V_{IN} = V_{DD}$	– 10		10	µA
		Input buffer with pull-down		10	50	100	
I_{IL}	Low level input current	Input buffer	$V_{IN} = V_{SS}$	– 10		10	µA
		Input buffer with pull-down		– 100	– 50	– 10	
V_{OH}	High level output voltage	pob1, pbtut1	I_{OH} = – 1 mA	2.4			V
		pob4, pot4, pbtd4, pblut4sm, pob4	I_{OH} = – 4 mA				
		pot8sm, pblut8sm	I_{OH} = – 8 mA				
V_{OL}	Low level output voltage	pob1, pblut1	I_{OL} = 1 mA			0.4	V
		pob4, pot4, pbtd4, pbtut4	I_{OL} = 4 mA				
		pot8sm, pblut8sm	I_{OL} = 8 mA				
I_{OZ}	Tri-state output leakage current		$V_{OUT} = V_{SS}$ or V_{DD}	– 10		10	µA
I_{DD}	Maximum operating current.		V_{DD} = 5.25V, f_{MCLK} = 33MHz.			320	mA

T_A = 0 °C to + 70 °C, V_{DD} = 4.75 V to 5.25 V.

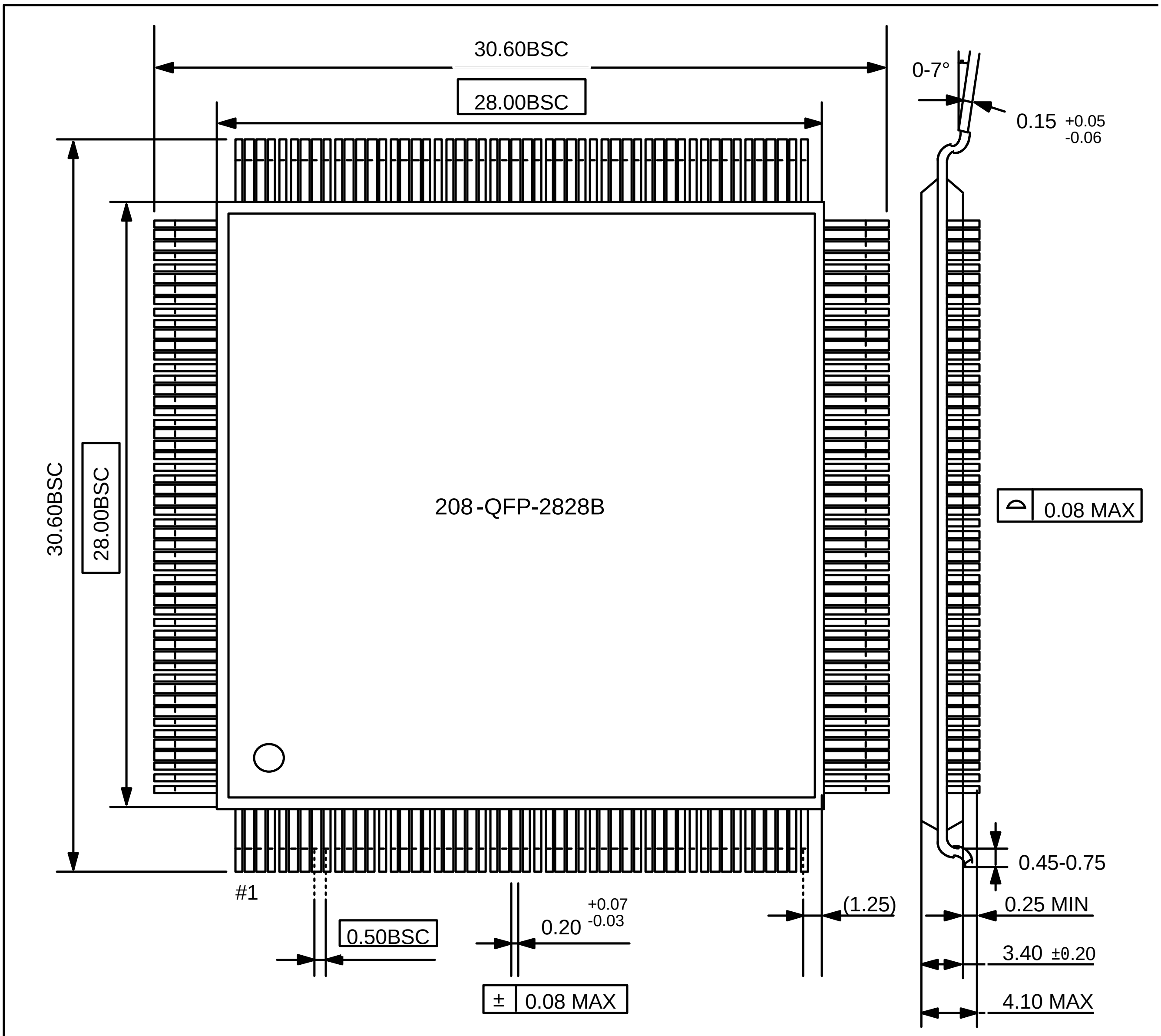
A.C. ELECTRICAL CHARACTERISTICS

Table 9. A.C. Electrical Characteristics (Preliminary)

Signal Name	Description	Min	Max	Unit
t_{EMz}	Memory control signal High-Z time	5.46	14.32	ns
t_{EMRs}	ExtMREQ setup time			
t_{EMRh}	ExtMREQ hold time			
t_{EMAr}	ExtMACK rising edge delay time	11.63	36.4	
t_{EMAf}	ExtMACK falling edge delay time	10.15	31.9	
t_{ADDRh}	Address hold time	8.5		
t_{ADDRd}	Address delay time		27.0	
t_{NRCS}	ROM/SRAM/Flash bank chip select delay time	6.1	18.0	
t_{NROE}	ROM/SRAM or external I/O bank output enable delay	5.9	18.5	
t_{NWBE}	ROM/SRAM or external I/O bank write byte enable delay	5.9	18.9	
t_{RDh}	Read data hold time	3.0		
t_{WDd}	Write data delay time (SRAM or external I/O)		9.8	
t_{WDh}	Write data hold time (SRAM or external I/O)	26.3		
t_{NRASf}	DRAM row address strobe active delay		18.5	
t_{NRASr}	DRAM row address strobe release delay		24.3	
t_{NCASf}	DRAM column address strobe active delay		16.1	
t_{NCASr}	DRAM CAS signal release delay time		17.1	
t_{NCASw}	DRAM CAS write active delay		19.8	
t_{NDWE}	DRAM bank write enable delay time		24.4	
t_{NDOE}	DRAM bank out enable delay time		23.5	
t_{NECS}	External I/O bank chip select delay time		20.6	ms
t_{WDDd}	DRAM write data delay time (DRAM)		14.2	ms
t_{WDDh}	DRAM writre data hold time (DRAM	7.4		ms
t_{Ws}	External wait setup time	0	0	ms
t_{Wh}	External wait hold time		1	ms

NOTE: T_A = 0 °C to + 70 °C, V_{DD} = 4.75 V to 5.25 V. Load capacitance = 50pF.

PACKAGE DIMENSIONS



NOTE: Dimensions are in millimeters.