S5N8947X

MCU for ADSL/Cable Modem

(Revision 0.1)

May. 23, 2000

SAMSUNG ELECTRONICS PROPRIETARY

CONTENTS

1. GENERAL DESCRIPTION

Samsung's S5N8947 16/32-bit RISC microcontroller is a cost-effective, high-performance microcontroller solution. The S5N8947 is designed as an integrated Ethernet controller for use in managed communication hubs and routers. The S5N8947 also provides ATM Layer SAR (Segmentation and Reassembly) function with UTOPIA interface and the full-rate USB (Universal Serial Bus) function.

The S5N8947 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.

Important peripheral functions including an UART channel, 2-channel GDMA, two 32-bit timers, I^2C bus controller, and programmable I/O ports are supported. Built-in logic including an interrupt controller, DRAM controller, and a controller for ROM/SRAM and flash memory are also supported. The S5N8947's System Manager includes an internal 32-bit system bus arbiter and an external memory controller.

To reduce total system cost, the S5N8947 offers a unified cache, Ethernet controller, SAR and USB. Most of the on-chip function blocks have been designed using an HDL synthesizer and the S5N8947 has been fully verified in Samsung's state-of-the-art ASIC test environment.

2. FEATURES

- ✓ 4-Kbyte unified cache
- ✓ SAR (Segmentation and Reassembly)
- ✓ UTOPIA (the Universal Test & Operations PHY Interface for ATM) Level 2 Interface
- ✓ Ethernet MAC
- ✓ Full-rate USB controller
- ✓ 2-CH GDMA (General Purpose Direct Memory Access)
- ✓ UART (Universal Asynchronous Receiver and Transmtter)
- ✓ 2 programmable 32bits Timers
- ✓ 18 Programmable I/O ports
- ✓ Interrupt controller
- ✓ I^2C controller
- ✓ Built-in PLLs for System/USB
- ✓ Cost effective JTAG-based debug solution
- ✓ Boundary scan
- ✓ Operating Voltage Range(2.5V +/- 0.2V)
- ✓ Operating Frequency Up to 50MHz
- ✓ 208 TQFP Package

3. Functional Block Descriptions

3.1. Block Diagram

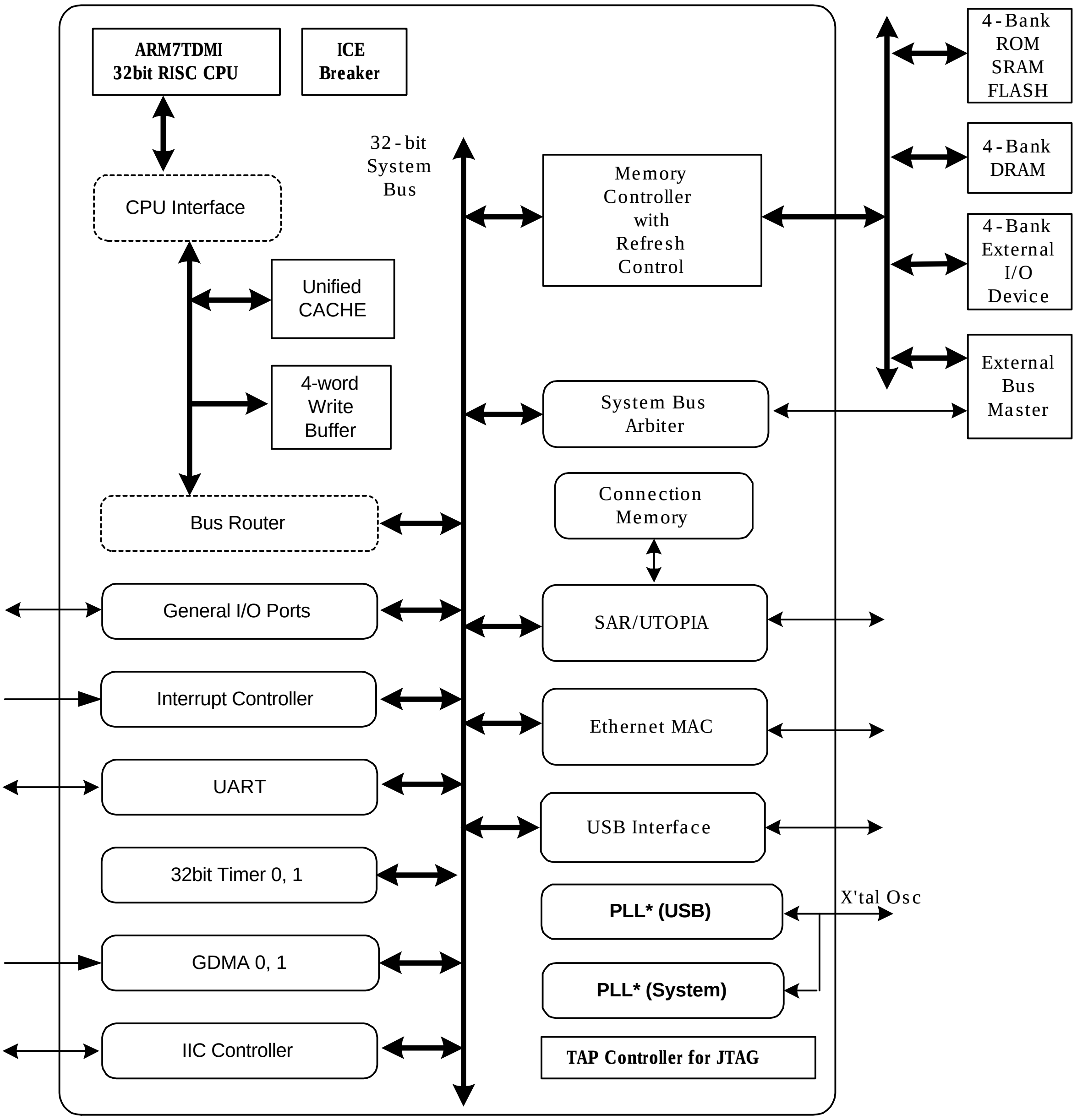


Figure 1 Top Block Diagram

3.2. Architecture

Integrated system for embedded Ethernet / USB / SAR
Fully 16/32-bit RISC architecture
Efficient and powerful ARM7TDMI core
Little/Big-Endian mode is supported basically, but the internal architecture is big-endian.
Cost-effective JTAG-based debug solution
Supports Boundary Scan

3.3. 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
Supports for EDO/normal or SDRAM
Programmable access cycle
Four-word depth write buffer
Cost-effective memory-to-peripheral DMA interface

3.4. Unified Instruction/Data Cache

Two-way set-associative unified cache (4Kbytes)
Supports for LRU (least recently used) Protocol
Cache is configurable as internal SRAM

3.5. SAR/Utopia Interface

Directly supports ATM Adaptation Layer Five (AAL5) Segmentation And Reassembly
Segments and reassembles data up to 70Mbps
A glueless UTOPIA level 2 interface is supprted (for receiving and transmitting ATM cells with SAR, it is a standard ATM interface between ATM link and physical layer).

3.6. Ethernet MAC

2 DMA engines with burst mode
Full compliance with IEEE standard 802.3
Supports MII interface (7-wire 10-Mbps interface is also supported).

3.7. USB Controller

Supports 12Mbps full rate function for universal serial bus

3.8. DMA Controller

2-channel general purpose DMA (for memory-to-memory, memory-to-USB, USB-to-memory, UART-to-memeory, memory-to-UART data transfers without CPU intervention)
Initiated by a software or a external DMA request
Increments or decrements source or destination address in 8-bit, 16-bit or 32-bit data transfers

3.9. UART (Serial I/O)

UART (Serial I/O) block with DMA-based or interrupt-based operation
Supports for 5-bit, 6-bit, 7-bit, or 8-bit serial data transmit and receive
Programmable baud rates
Infra-red (IR) TX/RX support (IrDA)

3.10. Timers

Two programmable 32-bit timers
Interval mode or toggle mode operation
Supports a watchdog timer.

3.11. Programmable I/O

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

3.12. Interrupt Controller

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

3.13. I^{2}C Serial Interface

Single Master mode operation only

3.14. PLL (for System/USB)

The external clock can be multiplied by on-chip PLLs to provide high frequency System/USB clock
The input frequency is fixed to 12 MHz
The output frequency is 4.167 times the input clock for System.
The output frequency is 4 times the input clock for USB.

4. PIN DESCRIPTIONS

4.1. Pin Configuration

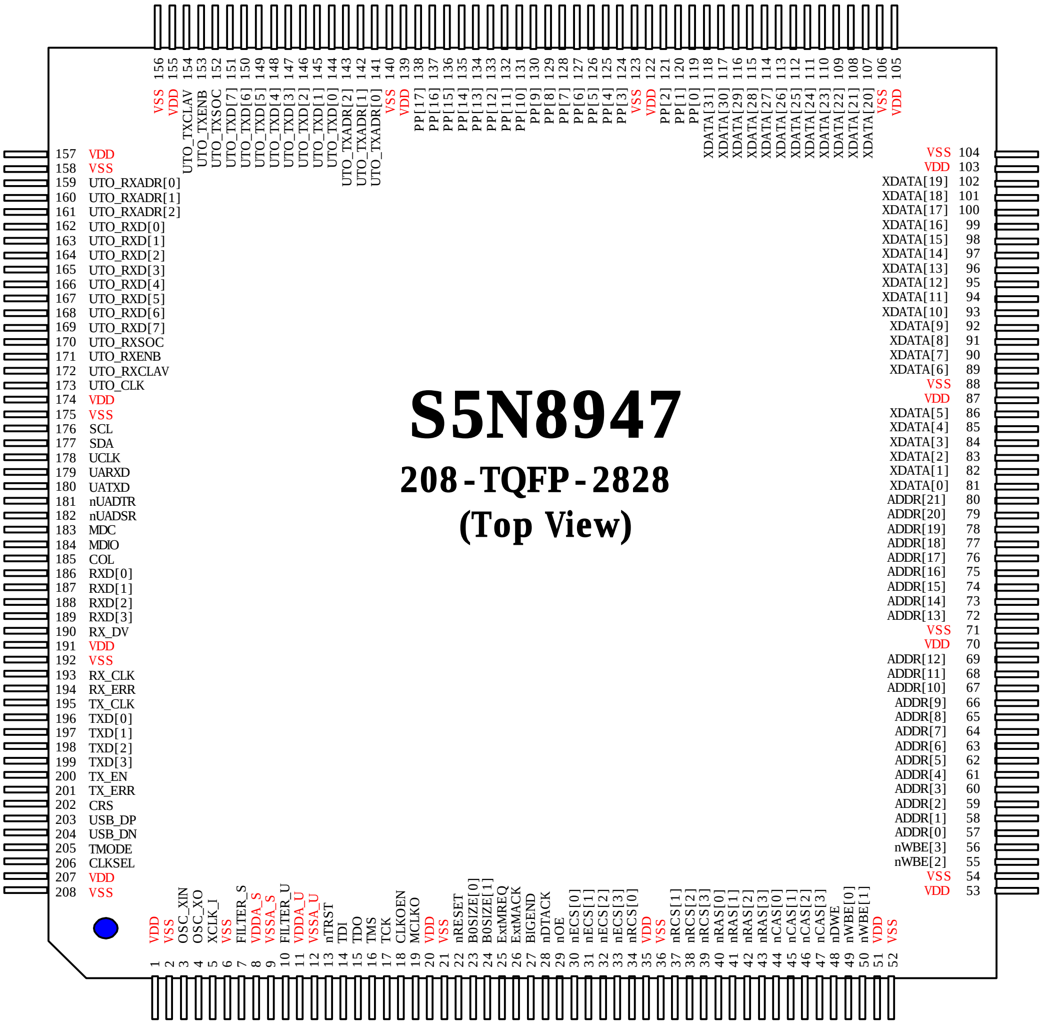


Figure 2 S5N8947 Pin Configuration

4.2. Pin Descriptions

Group	Pin Name	Pin Counts	I/O Type	Description
System Configurations (9)	XCLK_I	1	I	External System Clock Source Input.
	MCLKO	1	O	System Clock Out.
	CLKSEL	1	I	Clock Frequency Select from the internal PLL.
	nRESET	1	I	System Reset, Low Active.
	CLKOEN	1	I	System Clock Out Enable.
	BIGEND	1	I	Big endian mode select pin.
	FILTER_S	1	O	PLL filter pin for System Clock Generation.
	OSC_XIN	1	I	12MHz Reference Clock.
	OSC_XO	1	O	Crystal Clock Output.
TAP Control (5)	TMODE	1	I	Test Mode Enable.
	TCK	1	I	JTAG Test Clock Input.
	TMS	1	I	JTAG Test Mode Select.
	TDI	1	I	JTAG Test Data Input.
	TDO	1	O	JTAG Test Data Output.
	nTRST	1	I	JTAG Reset Signal, Low Active.
Memory Interface (81)	ADDR[21:0]	22	O	Address Bus.
	XDATA[31:0]	32	I/O	External Bidirectional 32bit Data Bus.
	nRAS[3:0]	4	O	Row AddressSstrobe for DRAM, Low Active.
	nCAS[3:0]	4	O	Column Address Strobe for DRAM, Low Active.
	nDWE	1	O	Write Enable, Low Active.
	nECS[3:0]	4	I/O	External I/O Select, Low Active.
	nDTACK	1	I	External Data Acknowledge Signal.
	nRCS[3:0]	4	O	ROM/SRAM/Flash Chip Select, Low Active.
	B0SIZE[1:0]	2	I	Bank 0 Data Bus Size for Boot ROM.
	nOE	1	O	Output Enable, Low Active.
	nWBE[3:0]	4	O	Write Byte Enable, Low Active.
	ExtMREQ	1	I	External Master Bus Request.
	ExtMACK	1	O	External Bus Acknowledge.
Ethernet Controller (18)	MCD	1	O	Management data clock.
	MDIO	1	I/O	Management data I/O.
	COL/COL_10M	1	I	Collision detected/Collision detected for 10M.
	TX_CLK/ TX_CLK_10M	1	I	Transmit clk/Transmit clk for 10M.
	TXD[3:0]/ TXD_10M/ LOOP_10M	4	O	Transmit data/Transmit data for 10M.
	TX_EN/ TXEN_10M	1	O	Transmit enable/Transmit enable for 10M.
	TX_ERR/ PCOMP_10M	1	O	Transmit error/Packet compression enable for 10M.
	CRS/CRS_10M	1	I	Carrier sense/Carrier sense for 10M.
	RX_CLK/ RXCLK_10M	1	I	Receive clock/Receive clock for 10M.
	RXD[3:0]/ RXD_10M	4	I	Receive data/Receive data for 10M.
	RX_DV/ LINK_10M	1	I	Receive data valid.
	RX_ERR	1	I	Receive error.
UART (5)	UCLK	1	I	External Clock Input for UART.
	UARXD	1	I	UART Receive Data.
	UATXD	1	O	UART Transmit Data.

	nUADTR	1	I	UART Data Terminal Ready, Low Active.
	nUADSR	1	O	UART Data Set Ready, Low Active.
General Purpose In/Out Ports (including xINTREQ nXDREQ nXDACK TIMER0,1) (18)	P[7:0]	8	I/O	General I/O ports for Bi-directional Data Only.
	xINTREQ[3:0] /P[11:8]	4	I/O	External Interrupt Requests/General I/O Ports.
	XDREQ[1:0]/ P[13:12]	2	I/O	External DMA Requests for GDMA/General I/O Ports.
	nXDACK[1:0]/ P[15:14]	2	I/O	External DMA Acknowledge from GDMA/General I/O Ports.
	TIMER0/ P[16]	1	I/O	TIMER0 Out/General I/O Port.
	TIMER1/ P[17]	1	I/O	TIMER1 Out /General I/O Port.
I^{2}C (2)	SCL	1	I/O	I^{2}C Serial Clock.
	SDA	1	I/O	I^{2}C Seral Data.
Utopia (Level 2) (30)	UTO_TXADR[2:0]	3	O	Transmit Address Bus.
	UTO_TXD[7:0]	8	O	Transmit Data Bus to the ATM PHY.
	UTO_TXSOC	1	O	Start Of Cell Indicator for Transmit Data.
	UTO_TXENB	1	O	Transmit Data Transfer Enable, Low Active.
	UTO_TXCLAV	1	I	Cell Buffer Available for Transmit Data.
	UTO_TXADR[2:0]	3	O	Receive Address Bus.
	UTO_RXD[7:0]	8	I	Receive Data Bus from the ATM PHY.
	UTO_RXSOC	1	I	Start Of Cell Indicator for Receive Data.
	UTO_RXENB	1	O	Receive Data Transfer Enable, Low Active.
	UTO_RXCLAV	1	I	Cell Buffer Available for Receive Data.
	UTO_CLK	1	O	Transfer/Receive interface byte clock.
USB (3)	USB_DP	1	I/O	USB data D+
	USB_DN	1	I/O	USB data D-
	FILTER_U	1	O	USB PLL filter pin.

Table 1 Signal Pin Descriptions for Each Group

4.3. Pin Descriptions with the Pin number and Pad type

Pin No	Pin Name	I/O Type	Pad type	Descriptions
1	VDD			
2	VSS			
3	OSC_XIN	I	Psoscm2	12MHz reference clock
4	OSC_XO	O		USB crystal clock out
5	XCLK_I	I	ptic	S5N8947 System Source Clock
6	VSS			
7	FILTER_S	O	Poa_bb	System PLL filter pin
8	VDDA_S	PWR	vdda	Analog power for PLL
9	VSSA/VBBA_S	GND	vbba	Analog / bulk ground for PLL
10	FILTER_U	O	poa_bb	USB PLL filter pin
11	VDDA_U	PWR	vdda	Analog power for PLL
12	VSSA/VBBA_U	GND	vssa	Analog / bulk ground for PLL
13	nTRST	I	pticu	JTAG Not Reset
14	TDI	I	pticu	JTAG Test Data In
15	TDO	O	ptot2	JTAG Test Data Out
16	TMS	I	pticu	JTAG Test Mode Select
17	TCK	I	ptic	JTAG Test Clock
18	CLKOEN	I	ptic	Clock Out Enable/Disable
19	MCLKO	O	pob8	System Clock Out
20	VDD	PWR		
21	VSS	GND		
22	nRESET	I	ptis	Not Reset
23-24	B0SIZE[0:1]	I	ptic	Bank 0 Data Bus Access Size
25	ExtMREQ	I	ptic	External Master bus request
26	ExtMACK	O	pob1	External bus Acknowledge
27	BIGEND	I	pticd	Big endian mode select pin
28	nDACK	I	ptic	Not external acknowledge signal
29	nOE	O	ptot4	Not output enable
30-33	nECS[0:3]	B	pbct4	Not external I/O select
34-39	nRCS[0:3]	O	ptot4	Not ROM/SRAM/Flash Chip select
40-43	nRAS[0:3]	O	ptot4	Not Row address strobe for DRAM
44-47	nCAS[0:3]	O	ptot4	Not Column address strobe for DRAM
48	nDWE	O	ptot4	Not Write Enable
49-50	NWBE[0:1]	O	ptot4	Not Write Byte Enable
51	VDD	PWR		
52	VSS	GND		
53	VDD	PWR		
54	VSS	GND		
55-56	NWBE[2:3]	O	ptot4	Not Write Byte Enable
57-69	ADDR[0:12]	O	ptot6	Address bus
70	VDD	PWR		
71	VSS	GND		
72-80	ADDR[13:21]	O	ptot6	Address bus
81-86	XDATA[0:5]	B	ptbsut6	External bidirectional data bus
87	VDD	PWR		
88	VSS	GND		
89-102	XDATA[6:19]	B	ptbsut6	External bidirectional data bus
103	VDD	PWR		
104	VSS	GND		

107-118	XDATA[20:31]	B	ptbsut6	External bidirectional data bus
119-121	P[0:2]	B	ptbst4sm	General I/O ports
122	VDD_P	PWR		
123	VSS_P	GND		
124-138	P[3:17]	B	ptbst4sm	General I/O ports
139	VDD_P	PWR		
140	VDD_S	GND		
141-143	UTO_TXADR[0:2]	O		Address bus for TX
144-151	UTO_TXD[0:7]	O	pob4	Data bus for TX
152	UTO_TXSOC	O	pob4	Start Of Cell for TX
153	UTO_TXENB	O	pob4	Enable data transfers (active low)
154	UTO_TXCLAV	I	ptis	Cell Buffer Available
155	VDD_P	PWR	vdd3op	I/O pad power
156	VSS_P	GND	vssop	I/O pad ground
157	VDD_P	PWR	vdd3op	I/O pad power
158	VSS_P	GND	vssop	I/O pad ground
159-161	UTOP_RXADR[0:2]	O		Address bus for RX
162-169	UTO_RXD[0:7]	I	ptis	Data bus for RX
170	UTO_RXSOC	I	ptic	Start Of Cell for RX
171	UTO_RXENB	O	pob4	Enable data transfers (active low)
172	UTO_RXCLAV	I	ptis	Cell Buffer available
173	UTO_CLK	O	pob4	Transfer/Receive interface byte clock
174	VDD_I	PWR		
175	VSS_I	GND		
176	SCL	B	ptbcd4	I^{2}C serial clock
177	SDA	B	ptbcd4	I^{2}C serial data
178	UCLK	I	ptis	UART external clock for UART
179	UARXD	I	ptic	UART receive data
180	UATXD	O	pob4	UART transmit data
181	nUADTR	I	ptic	Not UART0 data terminal ready
182	nUADSR	O	pob4	Not UART0 data set ready
183	MDC	O	pob4	Management data clock
184	MDIO	O	ptbbcut4	Management data I/O
185	COL/COL_10M	I	ptis	Collision detected/Collision detected for 10M
186-189	RXD[0:3]/RXD_10M	I	ptis	Receive data/Receive data for 10M
190	RX_DV/LINK_10M	I	ptis	Receive data valid
191	VDD	PWR		
192	VSS	GND		
193	RX_CLK/RXCLK_10M	I	ptis	Receive clock/Receive clock for 10M
194	RX_ERR	I	ptis	Receive error
195	TX_CLK/TX_CLK_10M	I	ptis	Transmit clock/Transmit clock for 10M
196-199	TXD[0:3]/TXD_10M/ LOOP_10M	O	pob4	Transmit data/Transmit data for 10M
200	TX_EN/TXEN_10M	O	pob4	Transmit enable/Transmit enable for 10M
201	TX_ERR/PCOMP_10M	O	pob4	Transmit error/Packet compression enable for 10M
202	CRS/CRS_10M	I	ptis	Carrier sense/Carrier sense for 10M

203	USB_DP	B	pbusb1	USB data D+
204	USB_DN	B		USB data D-
205	TMODE	I	ptic	Test Mode
206	CLKSEL	I	ptic	Clock Out Enable/Disable
207	VDD	PWR		
208	VSS	GND		

5. OPERATION DESCRIPTION

5.1. CPU Core Overview

The S5N8947 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 those 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.

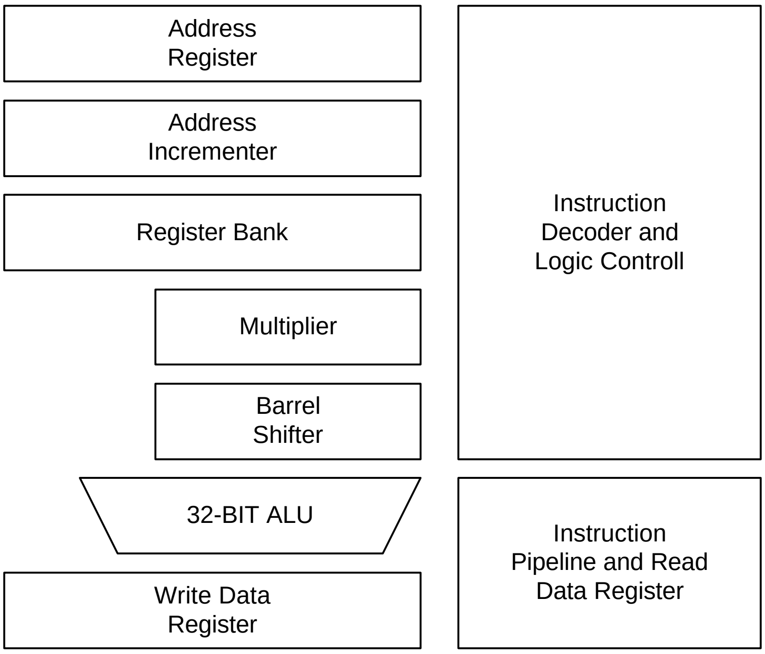


Figure 3 ARM7TDMI Core Block Diagram

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.

5.2. Instruction Set

The S5N8947 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 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.

5.3. 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).

5.4. 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.

5.5. REGISTERS

The S5N8947 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 S5N8947 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.

5.6. 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 S5N8947 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 S5N8947 core supports seven types of exceptions. Each exception has a fixed priority and a corresponding privileged processor mode, as shown in following Table

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
SWI	Supervisor mode	6 (lowest)

Table 2 S5N8947 CPU Exceptions

6. HARDWARE STRUCTURE

6.1. System Manager

6.1.3. Overview

The S5N8947 microcontroller’s System Manager has the following functions.

- To arbitrate system bus access requests from several master blocks, based on fixed priorities.
- To provide the required memory control signals for external memory accesses. For example, if a master block such as the DMA controller or the CPU generates an address which corresponds to a DRAM bank, the System Manager’s DRAM controller generates the required normal/EDO or SDRAM access signals. The interface signals for normal/EDO or SDRAM can be switched by SYSCFG[31].
- To provide the required signals for bus traffic between the S5N8947 and ROM/SRAM and the external I/O banks.
- To compensate for differences in bus width for data flowing between the external memory bus and the internal data bus.
- To support both little and big endian for external memory or I/O devices. Internal registers, however, operate under big-endian mode.

Note : By generating an external bus request (ExtMREQ), an external device can access the S5N8947’s external memory. The S5N8947 can access slow external devices using a nDTACK signal. The DTACK signal, which is generated by the external device, extends the duration of the CPU’s memory access cycle beyond its programmable value.

6.1.4. System Manager Registers

To control external memory operations, the System Manager uses a dedicated set of special registers. By programming the values in the System Manager special registers, you can specify such things as :

- Memory type
- External bus width access cycle
- Control signal timing (RAS and CAS, for example)
- Memory bank locations
- Size of each memory bank to be used for arbitrary address spacing

The System Manager uses special register setting to control the generation and processing of the control signals, addresses, and data that are required by external devices in a standard system configuration. Special registers are also used to control access to ROM/SRAM/Flash banks, up to four DRAM banks and four external I/O banks, and a special register mapping area.

The address resolution for each memory bank base pointer is 64 Kbytes (16 bits). The base address pointer is 10 bits. This gives a total addressable memory bank space of 16 M words.

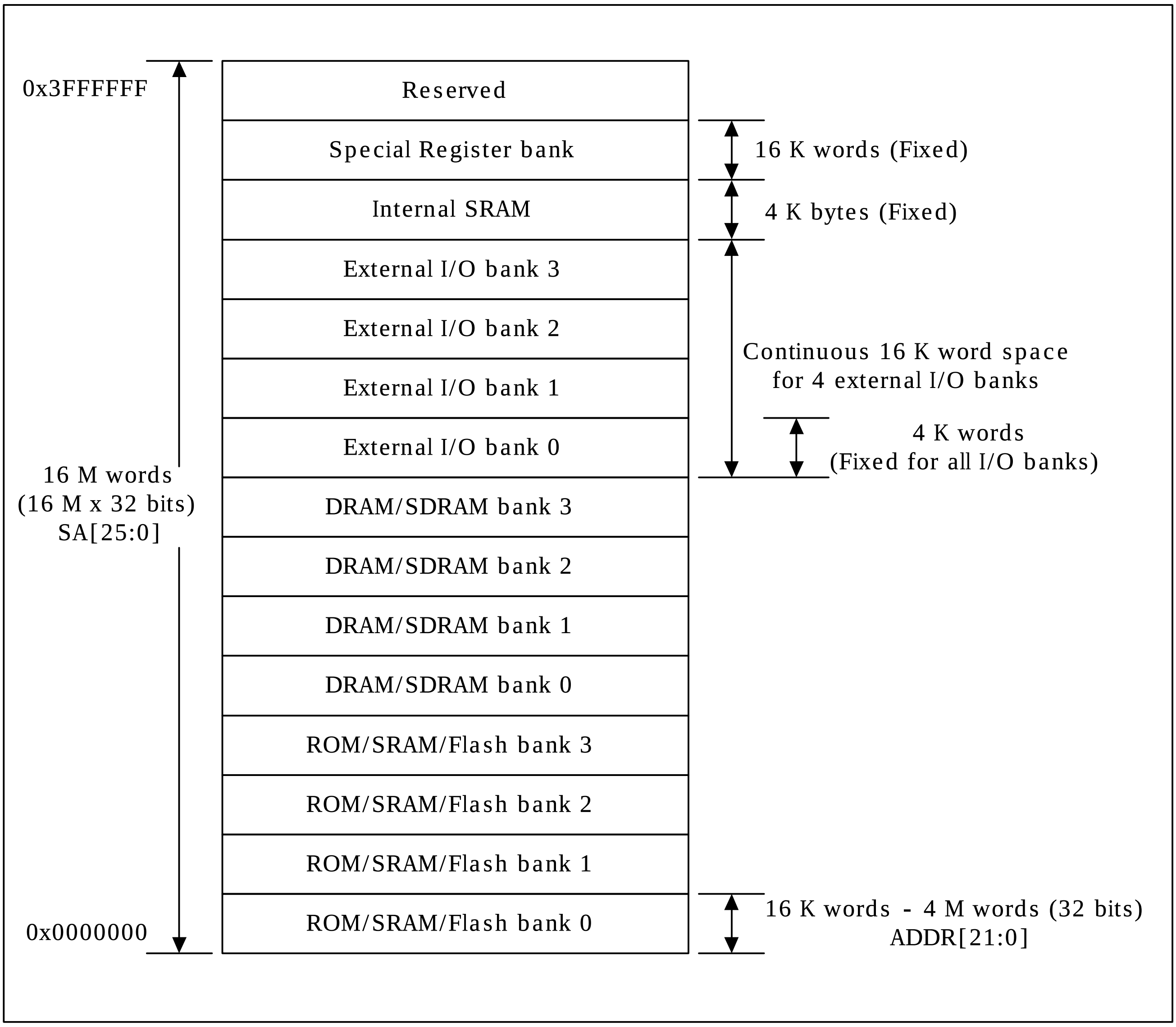


Figure 4 S5N8947 System Memory Map

6.1.5. System Memory Map

Followings are several important features to note about the S5N8947 system memory map :

- The size and location of each memory bank is determined by the register settings for “current bank base pointer” and “current bank end pointer”. You can use this base/next bank pointer concept to set up a consecutive memory map. To do this, you set the base pointer of the “next bank” to the same address as the next pointer of the “current bank”. Please note that when setting the bank control registers, the address boundaries of consecutive banks must not overlap. This can be applied even if one or more banks are disabled.

- Four external I/O banks are defined in a continuous address space. A programmer can only set the base pointer for external I/O bank 0. The start address of external I/O bank 1 is then calculated as the external I/O bank 0 start address +16 K. Similary, the start address for external I/O bank 2 is the external I/O bank 0 start address + 32 K, and the start address for external I/O bank 3 is the external I/O bank 0 start address + 48 K. Therefore, the total consecutive addressable space of the four external banks is defined as the start address of external I/O bank 0 + 64 K bytes.
- Within the addressable space, the start address of each I/O bank is not fixed. You can use bank control registers to assign a specific bank start address by setting the bank's base pointer. The address resolution is 64 K bytes. The bank's start address is defined as "base pointer << 16" and the bank's end address (except for external I/O banks) is "next pointer << 16 – 1".

After a power-on or system reset, all bank address pointer registers are initialized to their default values. In this means that a system reset automatically defines ROM bank 0 as a 32-Mbyte space with a start address of zero. This means that, except for ROM bank 0, all banks are undefined following a system startup.

The reset value for the next pointer and base pointer of ROM bank 0 are 0x200 and 0x000, respectively. This means that a system reset automatically defines ROM bank 0 as a 32-Mbyte space with a start address of zero. This initial definition of ROM bank 0 lets the system power-on or reset operation pass control to the user-supplied boot code that is stored in external ROM. (This code is located at address 0 in the system memory map.) When the boot code (i.e. ROM program) executes, it performs various system initialization tasks and reconfigures the system memory map according to the application's actual external memory and device configuration.

The initial system memory map following system startup is shown in following :

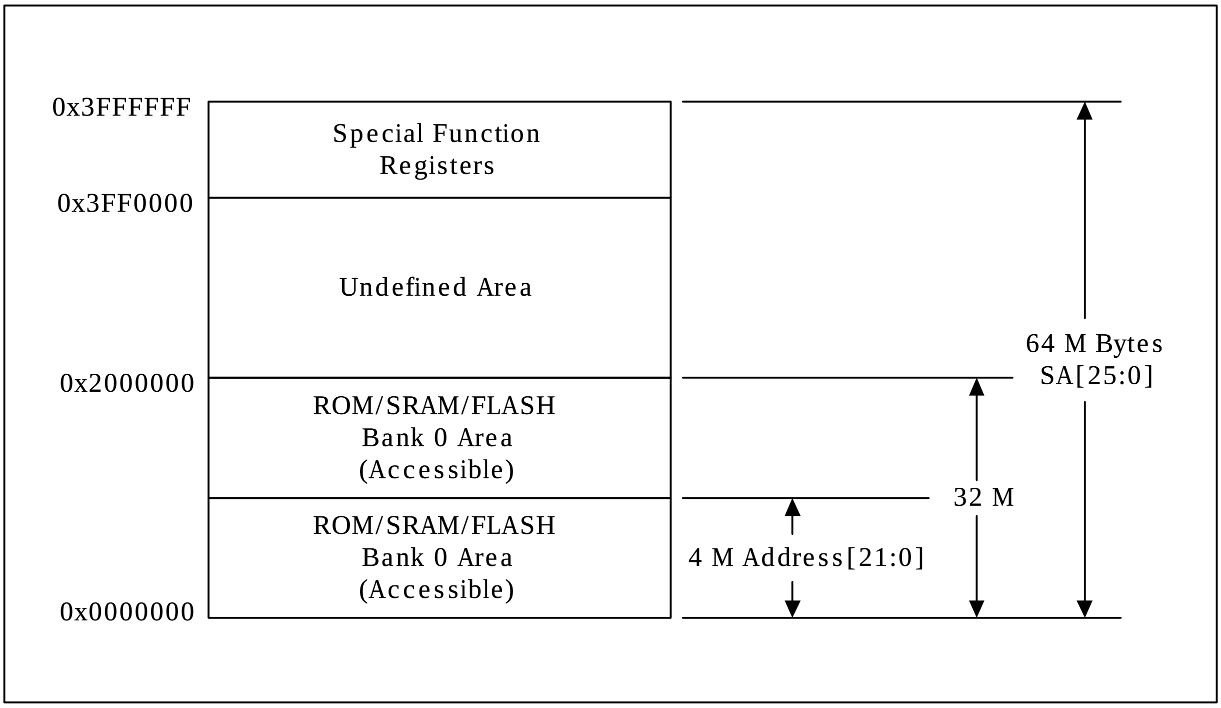
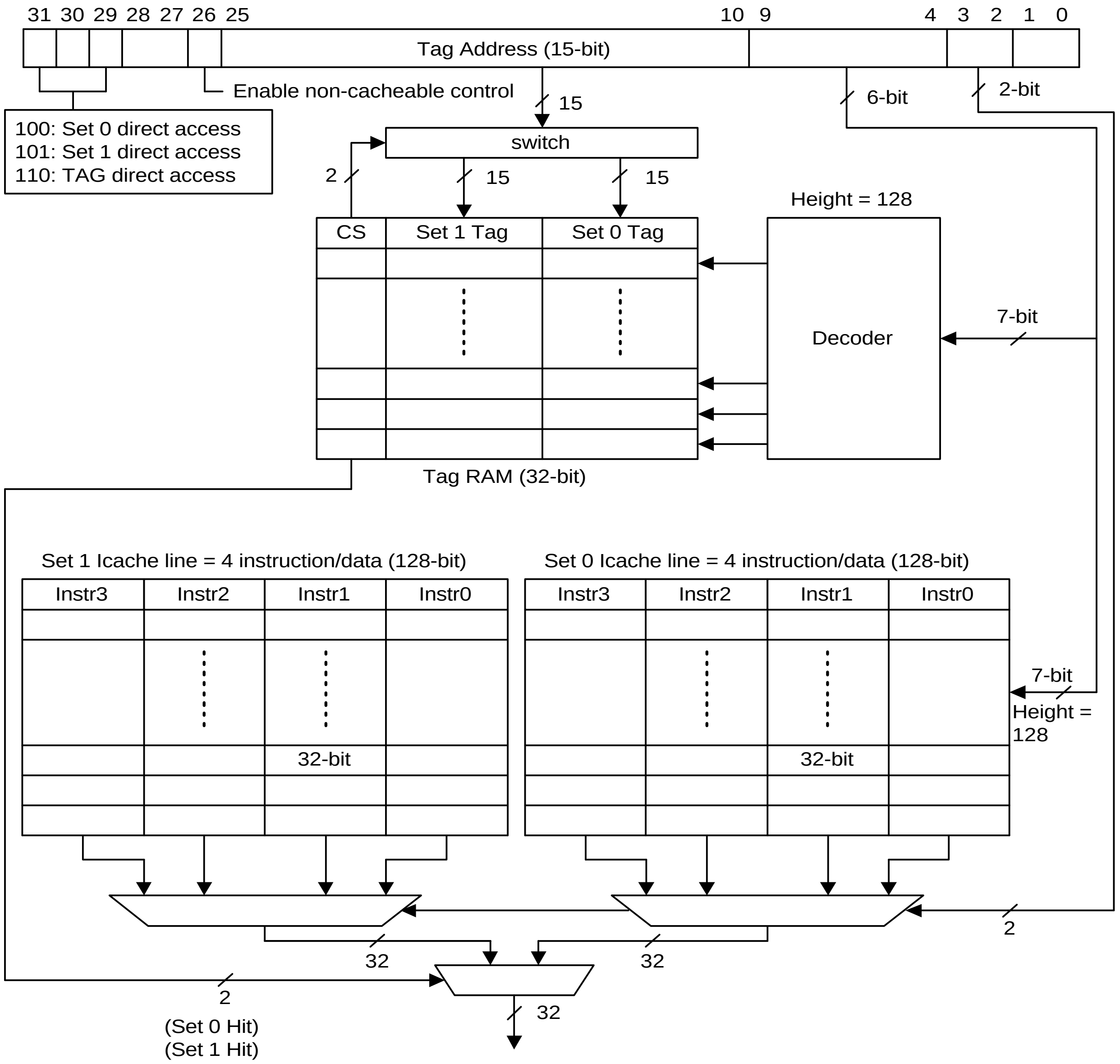

0x3FFFFFF
Special Function Registers
0x3FF0000
Undefined Area
0x2000000
ROM/SRAM/FLASH Bank 0 Area (Accessible)
ROM/SRAM/FLASH Bank 0 Area (Accessible)
0x0000000
64 M Bytes
SA[25:0]
32 M
4 M Address[21:0]

Figure 5 Initial system memory map (After reset)

6.2. Instruction / Data Cache

The S5N8947 CPU has a unified internal 4-Kbyte instruction/data cache. The cache is configured using two-way, set-associative addressing. The replacement algorithm is pseudo-LRU (Least Recently Used). The cache line size is four words (16 bytes). When a miss occurs, four word must be fetched consecutively from external memory. Typically, RISC processors take advantage of unified instruction/data caches to improve performance.



6.3. I^{2}C Bus Controller

The S5N8947's Internal IC bus (I^{2}C-bus) controller has the following important features :

- It requires only two bus lines, a serial data line (SDA) and a serial clock line (SCL). When the I^{2}C-bus is free, both lines are High level.
- Each device that is connected to the bus is software-addressable by a unique address. Slave relationships on the bus are constant. The bus master can be either a master-transmitter or a master-receiver. The I^{2}C bus controller supports only single master mode.
- It supports 8-bit, bi-directional, serial data transfers.
- The number of ICs that you can connect to the same I^{2}C-bus is limited only by the maximum bus capacitance of 400 pF.

Following figure shows a block diagram of the S5N8947's I^{2}C-bus controller.

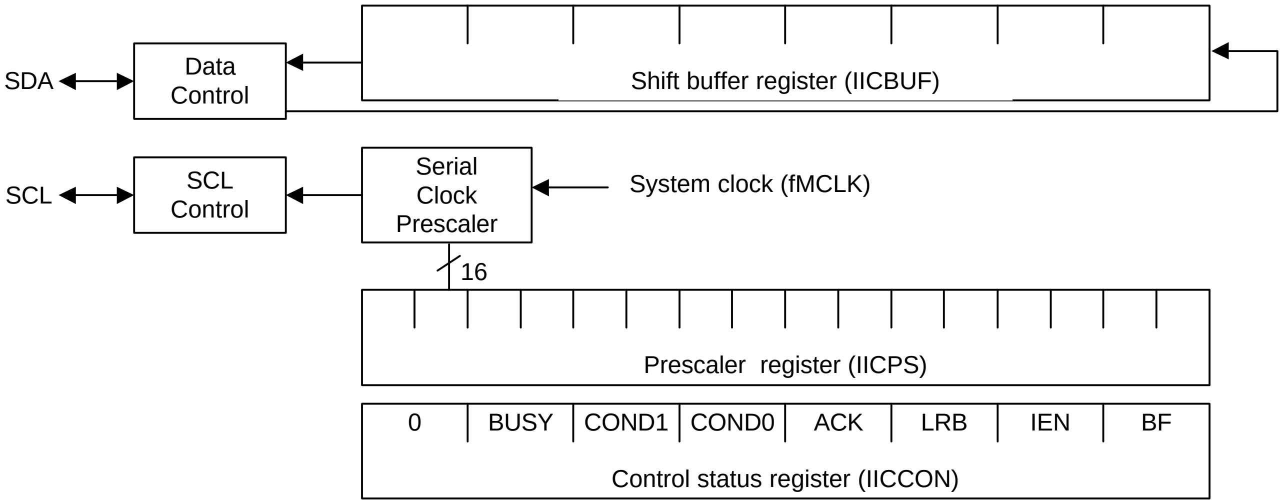


Figure 6 I^{2}C-Bus block diagram

6.4. Ethernet Controller

The S5N8947 has an Ethernet controller which operates at either 100/10-Mbits per second in half-duplex or full-duplex mode. In half-duplex mode, the controller supports the IEEE 802.3 Carrier Sense Multiple Access with Collision Detection (CSMA/CD) protocol. In full-duplex mode, it supports the IEEE 802.3 MAC Control Layer, including the Pause operation for flow control.

6.4.1. Block Diagram

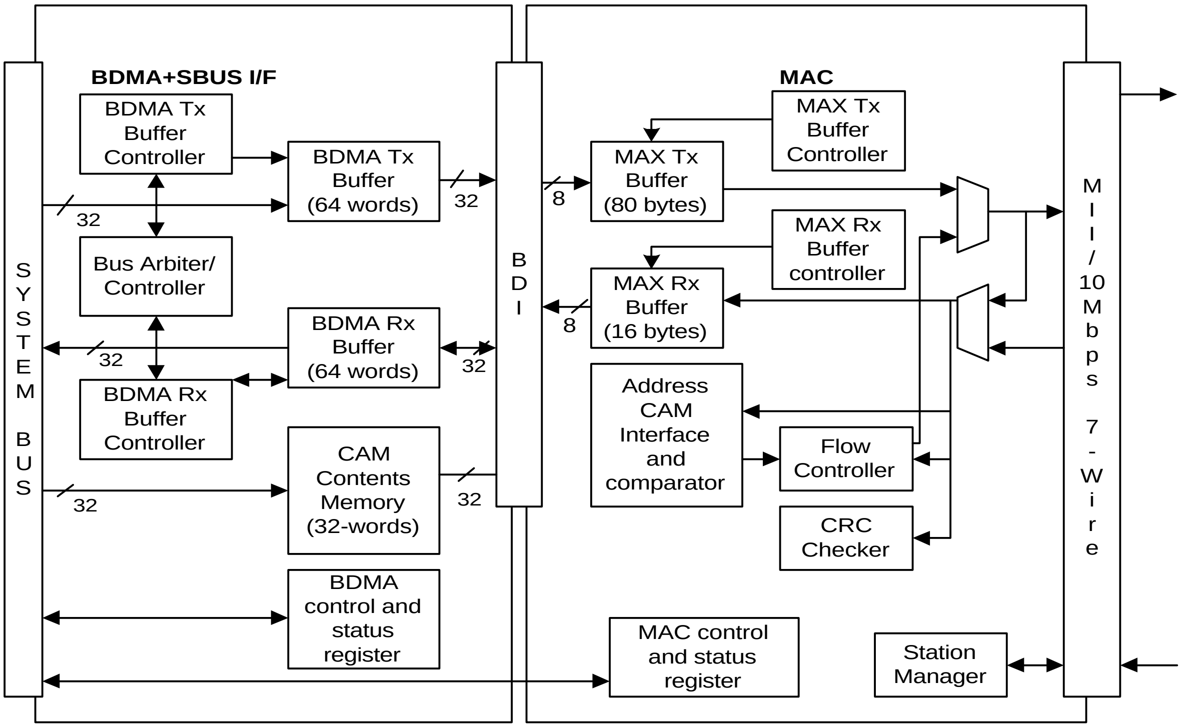

BDMA+SBUS I/F
BDMA Tx Buffer Controller
BDMA Tx Buffer (64 words)
32
32
Bus Arbiter/ Controller
BDMA Rx Buffer (64 words)
32
32
BDMA Rx Buffer Controller
CAM Contents Memory (32-words)
32
32
BDMA control and status register
SYSTEM BUS
BDI
MAC
MAX Tx Buffer Controller
MAX Tx Buffer (80 bytes)
8
MAX Rx Buffer controller
MAX Rx Buffer (16 bytes)
8
Address CAM Interface and comparator
Flow Controller
CRC Checker
MAC control and status register
Station Manager
MII/10Mbps 7-Wire

Figure 7 Ethernet controller block diagram

6.4.2. Features and Benefits

The most important features and benefits of the S5N8947 Ethernet controller are follows :

- Cost-effective connection to an external Repeater Interface Controller(RIC)/Ethernet backbone
- Buffered DMA (BDMA) engine using Burst mode
- BDMA Tx/Rx buffers (256 bytes/256 bytes)
- MAC Tx/Rx FIFOs (80 bytes/16 bytes) to support re-transmit after collision without DMA request and to handle DMA latency
- Data alignment logic
- Supports for old and new media (compatible with existing 10-Mbit/s networks)
- Full IEEE 802.3 compatibility for existing applications
- Provides a standard Media Independent Interface (MII)
- Provides an external 7-wire interface, also.

- Station Management (STA) signaling for external physical layer configuration and link negotiation
- On-chip CAM (21 addresses)
- Full-duplex mode for doubled bandwidth
- Pause operation hardware support for full-duplex flow control
- Long packet mode for specialized environments
- Short packet mode for fast testing
- PAD generation for ease of processing and reduced processing time
- Support for old and new media : Compatible with existing 100/10Mbit/s networks.
- Full IEEE 802.3 compatibility : Compatible with existing hardware and software.
- Standard CSMA/CD,Full duplex capability at 100/10 Mbit/s : Increase in data throughput performance.

6.5. SAR and Utopia Interface

The S5N8947 provides ATM layer Segmentation and Reassembly (SAR) function over a 8bit UTOPIA interface. The S5N8947 delivers an integrated solution for performing the SAR tasks required to communicate over an ATM network. The device translates packet-based data into 53-byte ATM cells that are asynchronously mapped into various physical media. The S5N8947 can be effectively applied for equipment requiring an interface between packet-based data and ATM-based networks.

6.5.1. Block Diagram

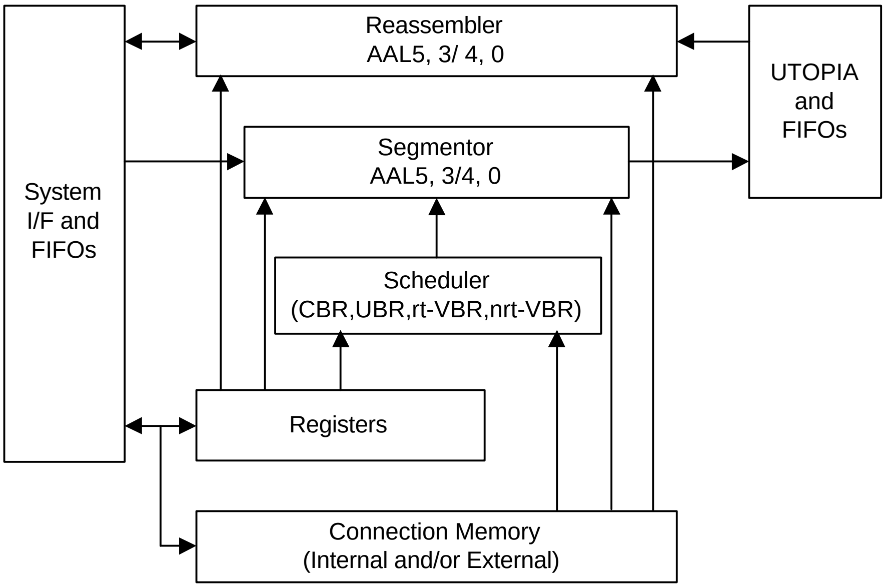


Figure 8 SAR function block diagram

6.5.2. Features and Benefits

- Supports CBR, UBR, rt-VBR and nrt-VBR traffic with rates set on a per-VC or per-VP basis.
- Supports AAL0 (raw cells) and AAL5 segmentation and reassembly.
- Segments and reassembles data up to about 70M bps via UTOPIA interface.
- Generates and verifies CRC-10 for OAM cells and AAL3/4 cells.
- Supports concurrent OAM cells and AAL5 cells on each active connection.
- Supports simultaneous segmentation and reassembly of up to 32 connections with internal memory and up to 4K connections with external memory.
- On chip 8K bytes SRAM for internal connection memory.
- Supports Contents Addressable Memory (CAM) for channel mapping (up to 32 connections).
- Supports packet sizes up to 64K bytes.
- Supports scatter and gather packet capability for large packets
- Start of Packet offset available for ease of implementing bridging and routing between different protocols.
- Provides glue-less UTOPIA level 2 interface (up to 7 PHYs).

6.6. USB Controller

The Universal Serial Bus (USB) is an industry standard bus architecture for computer peripheral attachment. The USB provides a single interface for easy, plug-and-play, hot-plug attachment of peripherals such as keyboard, mouse, speakers, printers, scanners, and communication devices. The USB allows simultaneous use of many different peripherals with a combined transfer rate of up to 12 Mbit/s.

The S5N8497 controller includes a highly flexible integrated USB peripheral controller that lets designers implement a variety of microcontroller-based USB peripheral devices for telephony, audio, or other high-end applications. The S5N8947 controller is intended for USB peripherals that use the full-speed signalling rate of 12 Mbit/s. The USB low-speed rate (1.5 Mbit/s) is not supported. An integrated USB transceiver is provided to minimize system device count and cost, but an external transceiver can be used instead, if required. The USB peripheral controller's features meet or exceed all of the USB device class resource requirements defined by the USB specification Version 1.0 and 1.1. Consult the USB specification for details about overall USB system design. The integrated USB peripheral controller provides a very efficient and easy-to-use interface, so that device software (or firmware) does not incur the overhead of managing low-level USB protocol requirements.

The USB peripheral controller hardware implements a number of USB standard commands directly; the rest can be implemented in device software. In addition, the USB peripheral controller provides a high degree of flexibility to help designers accommodate vendor- or device-class-specific commands, as well as any new features that might be added in future USB specifications.

Specialized hardware is provided to support Bulk data transfers. Using the Microcontroller's DMA features, large size of bulk transfers from an off-chip peripheral, can be automatically synchronized to the USB data rate with little or no CPU overhead.

Robust error detection and management features are provided so the device software can manage transfers in any number of ways as required by the application. The USB suspend/ resume, reset, and remote wake up features are also supported.

6.6.1. Block Diagram

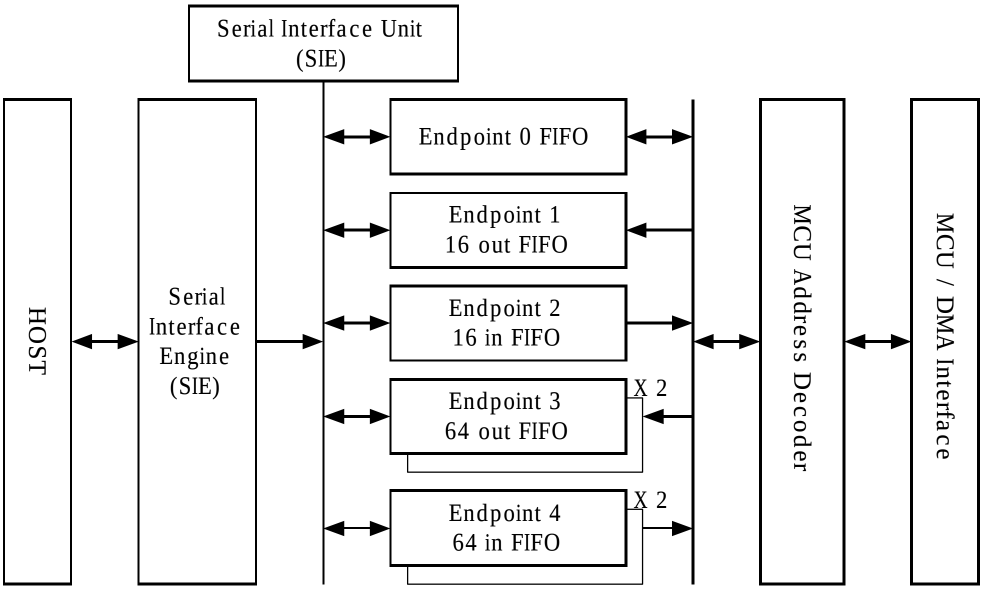


Figure 9 USB Module Block Diagram

6.7. DMA Controller

The S5N8947 has a two-channel general DMA controller, called the GDMA. The two-channel GDMA performs the following data transfers without CPU intervention:

- Memory-to-memory (memory to/from memory)
- UART-to-memory (serial port to/from memory)
- USB-to-memory (USB port to/from memory)

The on-chip GDMA can be started by software and/or by an external DMA request (nXDREQ). Software can also be used to restart a GDMA operation after it has been stopped.

The CPU can recognize when a GDMA operation has been completed by software polling and/or when it receives an appropriate internally generated GDMA interrupt. The S5N8947 GDMA controller can increment or decrement source or destination addresses and conduct 8-bit (byte), 16-bit (half-word), or 32-bit (word) data transfers.

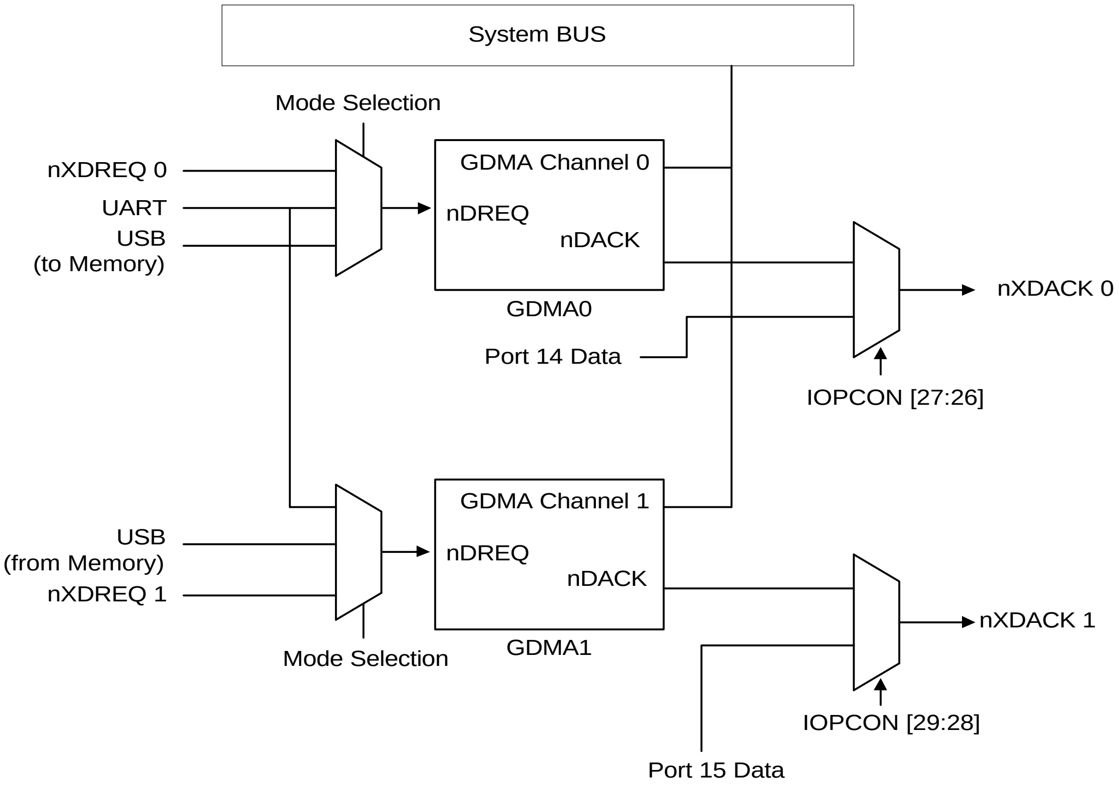


Figure 10 GDMA controller block diagram

6.8. UART(Serial I/O)

The S5N8947 UART (Universal Asynchronous Receiver/Transmitter) unit provides an asynchronous serial I/O (SIO) port. This can operate in interrupt-based or DMA-based mode. That is, the UART can generate internal interrupts or DMA requests to transfer data between the CPU and the serial I/O port.

The most important features of the S5N8947 UART include:

- Programmable baud rates
- Infra-red (IR) transmit/receive
- Insertion of one or two Stop bits per frame
- Selectable 5-bit, 6-bit, 7-bit, or 8-bit data transfers
- Parity checking

This unit has a baud rate generator, transmitter, receiver, and a control unit, as shown in next figure. The baud-rate generator can be driven by the internal system clock, MCLK. The transmitter and receiver block use this baud rate clock and have independent data buffer registers and data shifters.

Transmit data is written first to the transmit buffer register. From there, it is copied to the transmit shifter and then shifted out by the transmit data pin, UATXDn. Receive data is shifted in by the receive data pin, UARXDn. It is then copied from the shifter to the receive buffer register when one data byte has been received.

This unit provides software controls for mode selection, and for status and interrupt generation.

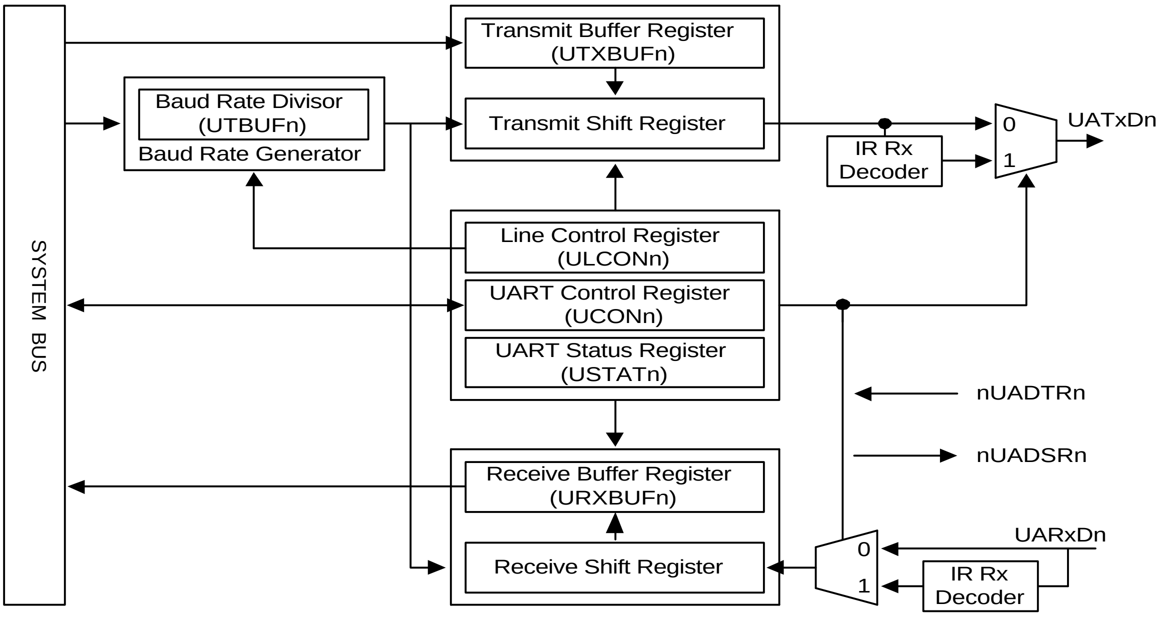


Figure 11 UART block diagram

6.9. Timers

The S5N8947 has two 32-bit timers. These timers can operate in interval mode or in toggle mode. The output signals are TOUT0 and TOUT1, respectively.

You enable or disable the timers by setting control bits in the timer mode register, TMOD. An interrupt request is generated whenever a timer count-out (down count) occurs.

Watchdog timer is also implemented in the S5N8947. The following guidelines apply to watchdog timer functions:

— When a watchdog timer is enabled, it loads a data value to its count register and begins decrementing the count register value by the system clock.

— If the reset from the watchdog timer (WDRESET) reaches to zero, the Watchdog will start its reset sequence. The reset value is then reloaded and the watchdog timer is disabled.

— The WDRESET performs the same function as the External Reset (System Reset) to each block.

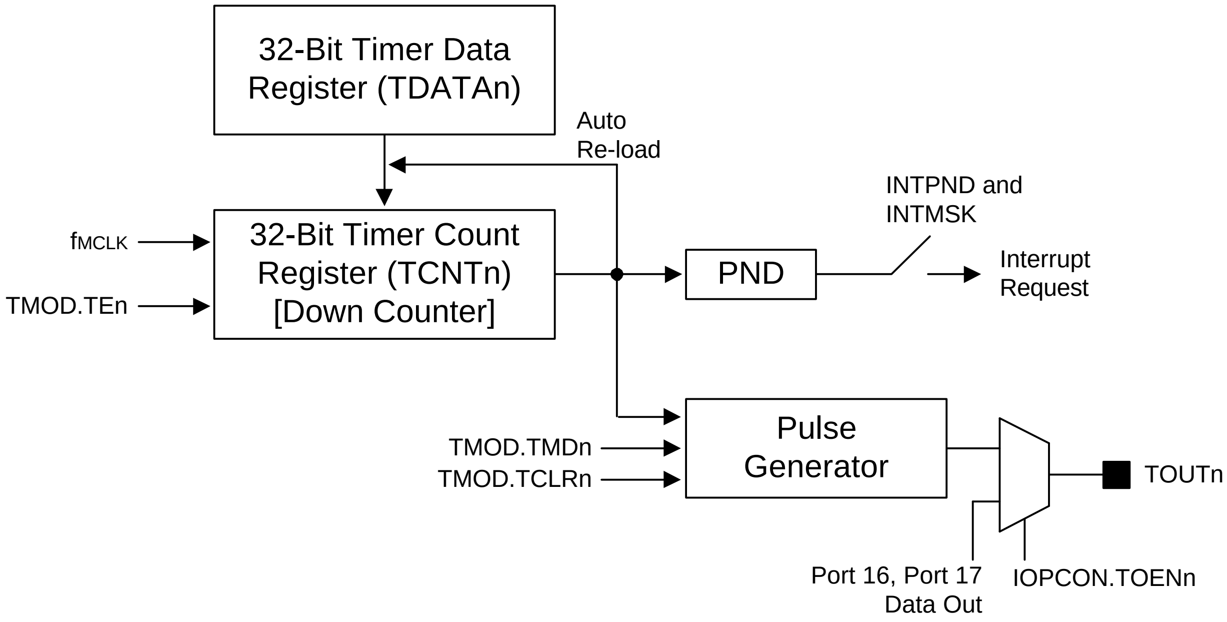


Figure 12 32-bit timer block diagram

6.10. I/O Ports

The S5N8947 has 18 programmable I/O ports. You can configure each I/O port to input mode, output mode, or special function mode. To do this, you write the appropriate settings to the IOPMOD and IOPCON registers. User can set filtering for the input ports using IOPCON register.

The modes of the ports from port0 to port7 are determined only by the IOPMOD register. But port[11:8] can be used as xINTREQ[3:0], port[13:12] as nXDREQ[1:0], port[15:14] as nXDACK[1:0], port[16] as TOUT0, or port[17] as TOUT1 depending on the settings in IOPCON register.

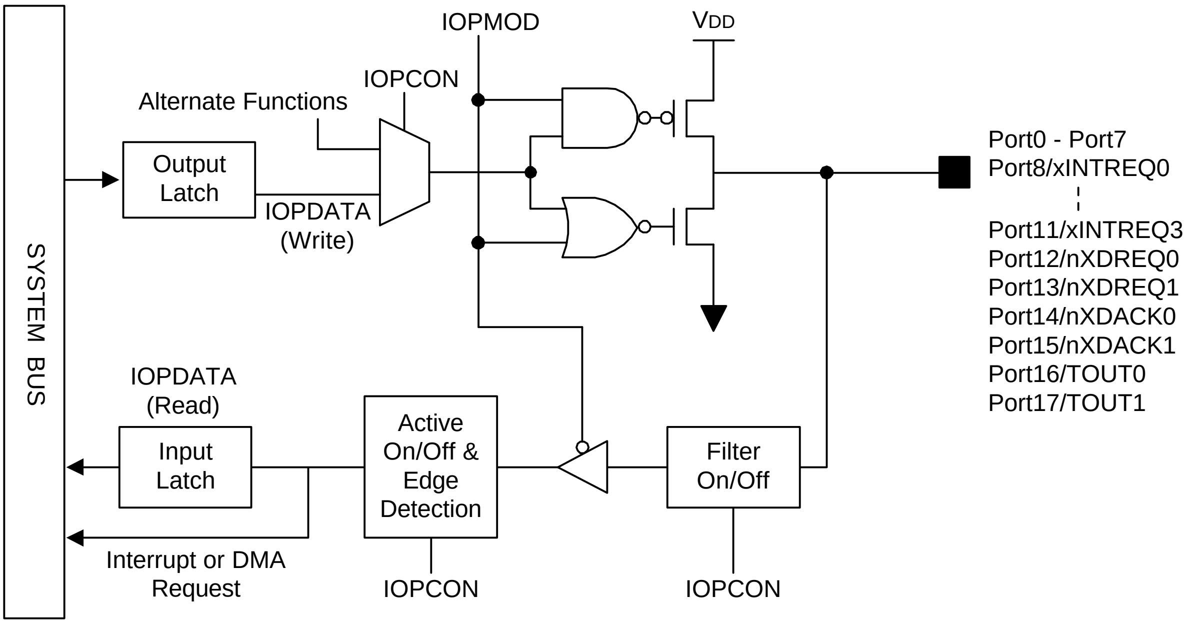


Figure 13 I/O port function diagram

6.11. Interrupt Controller

The S5N8947 interrupt controller has a total of 18 interrupt sources. Interrupt requests can be generated by internal function blocks and external pins.
The ARM7TDMI core recongnizes two kinds of interrupts: a normal interrupt request (IRQ), and a fast interrupt request (FIQ). Therefore all S5N8947 interrupts can be categorized as either IRQ or FIQ. The S5N8947 interrupt controller has an interrupt pending bit for each interrupt source.

Four special registers are used to control interrupt generation and handling:

- Interrupt priority registers. The index number of each interrupt source is written to the pre-defined interrupt priority register field to obtain that priority. The interrupt priorities are pre-defined from 0 to 17.
- Interrupt mode register. Defines the interrupt mode, IRQ or FIQ, for each interrupt source.
- Interrupt pending register. Indicates that an interrupt request is pending. If the pending bit is set, the interrupt pending status is maintained until the CPU clears it by writing a "1" to the appropriate pending register. When the pending bit is set, the interrupt service routine starts whenever the interrupt mask register is "0". The service routine must clear the pending condition by writing a "1" to the appropriate pending bit. This avoids the possibility of continuous interrupt requests from the same interrupt pending bit.
- Interrupt mask register. Indicates that the current interrupt has been disabled if the corresponding mask bit is "1". If an interrupt mask bit is "0" the interrupt will be serviced normally. If the global mask bit (bit 18) is set to "1", no interrupts are serviced. However, the source's pending bit is set if the interrupt is generated. When the global mask bit has been set to "0", the interrupt is serviced.

Index Values	Interrupt Sources
[17]	I^{2}C-bus interrupt
[16]	Ethernet controller MAC Rx interrupt
[15]	Ethernet controller MAC Tx interrupt
[14]	Ethernet controller BDMA Rx interrupt
[13]	Ethernet controller BDMA Tx interrupt
[12]	SAR Tx/Rx done interrupt
[11]	SAR Tx/Rx error interrupt
[10]	USB interrupt
[9]	GDMA channel 1 interrupt
[8]	GDMA channel 0 interrupt
[7]	Timer 1 interrupt
[6]	Timer 0 interrupt
[5]	UART receive and error interrupt
[4]	UART transmit interrupt
[3]	External interrupt 3
[2]	External interrupt 2
[1]	External interrupt 1
[0]	External interrupt 0

Table 3 S5N8947 Interrupt Sources

7. SPECIAL FUNCTION REGISTERS

Group	Registers	Offset	R/W	Description	Reset/Value
System Manager	SYSCFG	0x0000	R/W	System configuration register	0x23FF0000
	SYSCON	0x3000	R/W	System 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
	DRAMCON0	0x3024	R/W	DRAM bank 0 control register	0x00000000
	DRAMCON1	0x3028	R/W	DRAM bank 1 control register	0x00000000
	DRAMCON2	0x302C	R/W	DRAM bank 2 control register	0x00000000
	DRAMCON3	0x3030	R/W	DRAM bank 3 control register	0x00000000
	REFEXTCON	0x3034	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	R/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
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	0x00006000
	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	0x9040	R	Transmit control frame status	0x00000000
USB	FA	0x7000	R/W	Function address register	0x00000000
	PM	0x7004	R/W	Power management register	0x00000000
	EI	0x7008	R/W	Endpoint interrupt register	0x00000000
	UI	0x700C	R/W	USB interrupt register	0x00000000
	EIE	0x7010	R/W	Endpoint interrupt enable register	0x0000001F
	UIE	0x7014	R/W	USB interrupt enable register	0x00000004
	LBFN	0x7018	R	Frame number1 register	0x00000000

	HBFN	0x701C	R	Frame number2 register	0x00000000
	IE0M	0x7020	R/W	Input EP0 MAXP register	0x00000000
	*E0C	0x7024	R/W	EP0 Control register	0x00000000
	*E0BC	0x7028	R/W	EP0 Write Byte Counter	0x00000000
	*O1M	0x7030	R/W	EP1 OUT MAXP register	0x00000001
	*O1C1	0x7034	R/W	EP1 OUT Control register 1	0x00000000
	*O1C2	0x7038	R/W	EP1 OUT Control register 2	0x00000000
	*E1BC	0x703C	R/W	EP1 Write Byte Counter	0x00000000
	*I2M	0x7040	R/W	EP2 IN MAXP register	0x00000001
	*I2C1	0x7044	R/W	EP2 IN Control register 1	0x00000000
	*I2C2	0x7048	R/W	EP2 IN Control register 2	0x00000000
	*O3M	0x7050	R/W	EP3 OUT MAXP register	0x00000004
	*O3C1	0x7054	R/W	EP3 OUT Control register 1	0x00000000
	*O3C2	0x7058	R/W	EP3 OUT Control register 2	0x00000000
	*E3BC	0x705C	R/W	EP3 Write Byte Counter	0x00000000
	*I4M	0x7060	R/W	EP4 IN MAXP register	0x00000004
	*I4C1	0x7064	R/W	EP4 IN Control register 1	0x00000000
	*I4C2	0x706C	R/W	EP4 IN Control register 2	0x00000000
	*PDC	0x7070	R/W	Power-down Counter Register	0x00000000
	*EP0D	0x7100	R/W	EP0 FIFO data register	0x00000000
	*EP1D	0x7104	R/W	EP1 FIFO data register	0x00000000
	*EP2D	0x7108	R/W	EP2 FIFO data register	0x00000000
	*EP3D	0x710C	R/W	EP3 FIFO data register	0x00000000
	*EP4D	0x7110	R/W	EP4 FIFO data register	0x00000000
SAR	SW_RESET	0x00	R/W	Software reset register	0x00000000
	GLOBAL_MODE	0x08	R/W	Global mode register	0x00000000
	TIMEOUT_BASE	0x0C	R/W	Base multiple for receive packet timeout register	0x00FF7FFF
	TX_READY1	0x10	R/W	Transmit ready first packet or subpacket address	0x00000000
	TX_READY2	0x14	R/W	Transmit ready last packet or subpacket address	0x00000000
	TX_DONE_ADDR	0x18	R/W	Transmit packet done queue base address register	0x00000000
	TX_DONE_SIZE	0x1C	R/W	Transmit packet done queue size register	0x00C00000
	RX_POOL0_ADDR	0x20	R/W	Receive queue 0 base address register	0x00000000
	RX_POOL0_SIZE	0x24	R/W	Receive queue 0 size register	0x00C00000
	RX_POOL1_ADDR	0x28	R/W	Receive queue 1 base address register	0x00000000
	RX_POOL1_SIZE	0x2C	R/W	Receive queue 1 size register	0x00C00000
	RX_POOL2_ADDR	0x30	R/W	Receive queue 2 base address register	0x00000000
	RX_POOL2_SIZE	0x34	R/W	Receive queue 2 size register	0x00C00000
	RX_POOL3_ADDR	0x38	R/W	Receive queue 3 base address register	0x00000000
	RX_POOL3_SIZE	0x3C	R/W	Receive queue 3 size register	0x00C00000
	RX_DONE0_ADDR	0x40	R/W	Receive packet done queue 0 base address register	0x00000000
	RX_DONE0_SIZE	0x44	R/W	Receive packet done queue 0 size register	0x00C00000
	RX_DONE1_ADDR	0x48	R/W	Receive packet done queue 1 base address register	0x00000000
	RX_DONE1_SIZE	0x4C	R/W	Receive packet done queue 1 size register	0x00C00000
	UTOPIA_CONFIG	0x50	R/W	UTOPIA interface configuration register	0x00000000
	UTOPIA_TIMEOUT	0x54	R/W	UTOPIA interface timeout register	0xFFFFFFFF
	CLOCK_RATIO	0x64	R/W	Ratio of SAR clock freq toUNI interface speed	0x0000008E
	DONE_INT_MASK	0x70	R/W	Interrupt mask for done interrupt register	0xFFFFFFFF
	ERR_INT_MASK	0x74	R/W	Interrupt mask for error interrupt register	0xFFFFFFFF
	DONE_INT_STAT	0x78	R/W	Interrupt status for done interrupt register	0x00000000
	ERR_INT_STAT	0x7C	R/W	Interrupt status for error interrupt register	0x00000000
	1/R_LOOKUP_TBL	0x80	R/W	Base address of 1/Rate lookup table	0x00000000
	VP_LOOKUP_TBL	0x84	R/W	Base address of VP lookup table	0x00000200
	UBR_SCH_TBL	0x88	R/W	Base address and entry number of UBR schedule	0x00000300
	CBR_SCH_TBL	0x8C	R/W	Base address and entry number of CBR schedule	0x00000380

	CELL_BUFF	0x90	R/W	Base address and entry number of cell buffer	0x00000400
	SCH_CONN_TBL	0x94	R/W	Base address and entry number of scheduler connection table	0x00000500
	AAL_CONN_TBL	0x98	R/W	Base address and entry number of AAL connection table	0x00000600
	SAR_CONN_TBL	0x9C	R/W	Base address and entry number of SAR connection table	0x00000700
	CAM_VPVC/CN	0x100-1FC	R/W	CAM VPCI, VCI and connection number register	0x00000000
	CONFIGURATION	0x200	R/W	Clock control and connection memory configuration register	0x00000044
	EXT_CMBASE	0x204	R/W	External connection memory base address register	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	0x00001110
	INTPRI5	0x4020	R/W	Interrupt priority register 5	0x00000000
	INTOFFSET	0x4024	R	Interrupt offset address register	0x00000054
	INTPNDPRI	0x4028	R	Interrupt pending priority register	0x00000000
	INTPNDTST	0x402C	W	Interrupt pending test register	0x00000000
	INTOSET_FIQ	0x4030	R	FIQ interrupt offset register	0x00000054
	INTOSET_IRQ	0x4034	R	IRQ interrupt offset register	0x00000054
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
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	ULCON	0xD000	R/W	UART line control register	0xXXXXXX00
	UCON	0xD004	R/W	UART control register	0xXXXXXX00
	USTAT	0xD008	R	UART status register	0xXXXXXXC0
	UTXBUF	0xD00C	W	UART transmit holding register	Undefined
	URXBUF	0xD010	R	UART receive buffer register	Undefined
	UBRDIV	0xD014	R/W	Baud rate divisor register	0xXXXXXX00
Timers	TMOD	0x6000	R/W	Timer mode register	0x00000000
	TDATA0	0x6004	R/W	Timer 0 data register	0x00000000
	TDATA1	0x6008	R/W	Timer 1 data register	0x00000000
	TCNT0	0x600C	R/W	Timer 0 count register	0xFFFFFFFF
	TCNT1	0x6010	R/W	Timer 1 count register	0xFFFFFFFF
	WDCON	0x6014	R/W	Watchdog Timer Control register	0xFFFFFF00
	WDCNT	0x6018	R	Watchdog Timer Count register	0xFFFFFFFF

8. ELECTRIC CHARACTERISTICS

8.1. ABSOLUTE MAXIMUM RATINGS

Parameter	Symbol	Rating		Units
Supply Voltage	V_{DD}/V_{DDA}	3.6		V
DC input Voltage	V_{IN}	2.5 V I/O	3.6	V
		5 V-tolerant	6.5	
DC input current	I_{IN}	± 200		mA
Operating temperature	T_{OPR}	0 to 70		°C
Storage temperature	T_{STG}	– 65 to 150		°C

Table 4 Absolute Maximum Ratings

8.2. Recommended Operating Conditions

Parameter	Symbol	Rating	Units
Supply Voltage	V_{DD}/V_{DDA}	2.3 to 2.7	V
Oscillator frequency	f_{OSC}	12	MHz
External Loop Filter Capacitance	L_F	820	pF
Commercial temperature	T_A	0 to 70	°C

Table 5 Recommaended Operating Conditions

NOTES

- ✓ It is strongly recommended that all the supply pins (VDD/VDDA) be powered from the same source to avoid power latch-up.

8.3. DC ELECTRICAL CHARACTERISTICS

V_{DD} =2.5V+/-0.2V, VEXT = 5+/-0.25V, TA=-40 to 85 Centigrade (In case of 5V-tolerant I/O)

Parameter		Symbol	Conditions	Min	Typ	Max	Unit
High level input voltage	LVCMOS interface	V_{IH}[1]	–	1.7	–	–	V
Low level input voltage	LVCMOS interface	V_{IL}[1]	–	–	–	0.7	V
Switching threshold		VT	LVCMOS	–	0.5 V_{DD}	–	V
Schmitt trigger positive-going threshold		VT+	LVCMOS	–	–	1.9	–
Schmitt trigger negative-going threshold		VT–	LVCMOS	0.6	–	–	–
High level input current	Input buffer	I_{IH}	$V_{IN} = V_{DD}$	– 10	–	10	µA
	Input buffer with pull-up			10	25	50	
Low level input current	Input buffer	I_{LH}	$V_{IN} = V_{SS}$	– 10	–	10	µA
	Input buffer with pull-up			– 50	– 25	– 10	
High level output voltage	Type B1 to B16[2]	V_{OH}	I_{OH} = – A	V_{DD} – 0.05	–	–	V
	Type B1		I_{OH} = – 1 mA	1.9			
	Type B2		I_{OH} = – 2 mA				
	Type B4		I_{OH} = – 4 mA				
	Type B6		I_{OH} = – 6 mA				
Low level output voltage	Type B1 to B16[2]	V_{OL}	I_{OL} = – A			0.05	V
	Type B1		I_{OL} = – 1 mA			0.5	
	Type B2		I_{OL} = – 2 mA				
	Type B4		I_{OL} = – 4 mA				
	Type B6		I_{OL} = – 6 mA				
Tri-state output leakage current		I_{OZ}	V_{OUT} = V_{SS} or V_{DD}	– 10		10	µA
Maximum operating current		I_{DD}	V_{DD} = 3.6 V, f_{MCLK} = 50MHz			55	mA

Table 6 DC Electrical Characteristics

NOTES:

1. All 5V-tolerant input have less than 0.2V hysterisis.
2. Type B1 means 1mA output driver cells, and Type B6/B24 means 6mA/24mA output driver cells.

8.4. A.C Electrical Characteristics

(Ta = -40 to +85 Centigrade, V_{DD} = 2.3V to 2.7V)

Signal Name	Description	Min	Max	Unit
t_{EMz}	Memory control signal High-Z time	5.1	10.1	ns
t_{EMRs}	ExtMREQ setup time	0		
t_{EMRh}	ExtMREQ hold time	3.0		
t_{EMAr}	ExtMACK rising edge delay time	12.1	29.3	
t_{EMAf}	ExtMACK falling edge delay time	12.3	29.7	
t_{ADDRh}	Address hold time	8.5		
t_{ADDRd}	Address delay time	7.08	17.5	
t_{NRCS}	ROM/SRAM/Flash bank chip select delay time	5.2	12.4	
t_{NROE}	ROM/SRAM or external I/O bank output enable delay	5.7	13.6	
t_{NWBE}	ROM/SRAM or external I/O bank write byte enable delay	5.5	13.1	
t_{RDh}	Read data hold time	3.0		
t_{WDd}	Write data delay time (SRAM or external I/O)		17.23	
t_{WDh}	Write data hold time (SRAM or external I/O)	9.4		
t_{NRASf}	DRAM row address strobe active delay	5.6	13.4	
t_{NRASr}	DRAM row address strobe release delay	4.3	16.38	
t_{NCASf}	DRAM column address strobe active delay	5.5	13.1	
t_{NCASr}	DRAM CAS signal release delay time	4.36	13.1	
t_{NDWE}	DRAM bank write enable delay time	5.8	13.9	
t_{NDOE}	DRAM bank out enable delay time	5.7	13.6	
t_{NECS}	External I/O bank chip select delay time	5.3	12.5	
t_{WDDd}	DRAM write data delay time (DRAM)	5.9	14.2	
t_{WDDh}	DRAM write data hold time (DRAM)	7.4		
t_{Ws}	External wait setup time	0		
t_{Wh}	External wait hold time	3.0		
T_{MCLKOd}	External clock to MCLKO delay time when PLL power-down	5.0	12.45	

Table 7 AC Electrical Characteristics

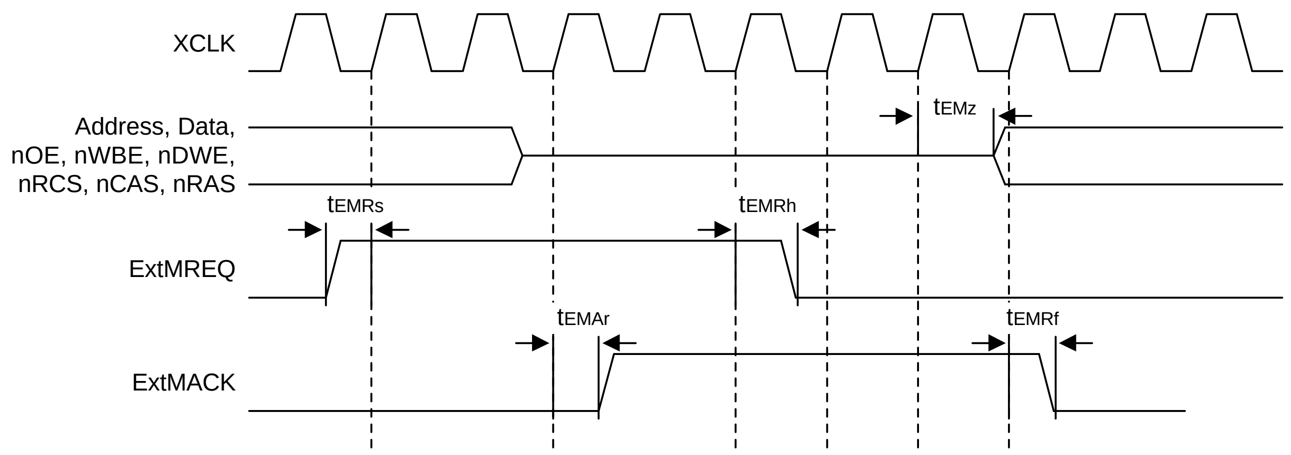


Figure 14 External Bus Request Timing

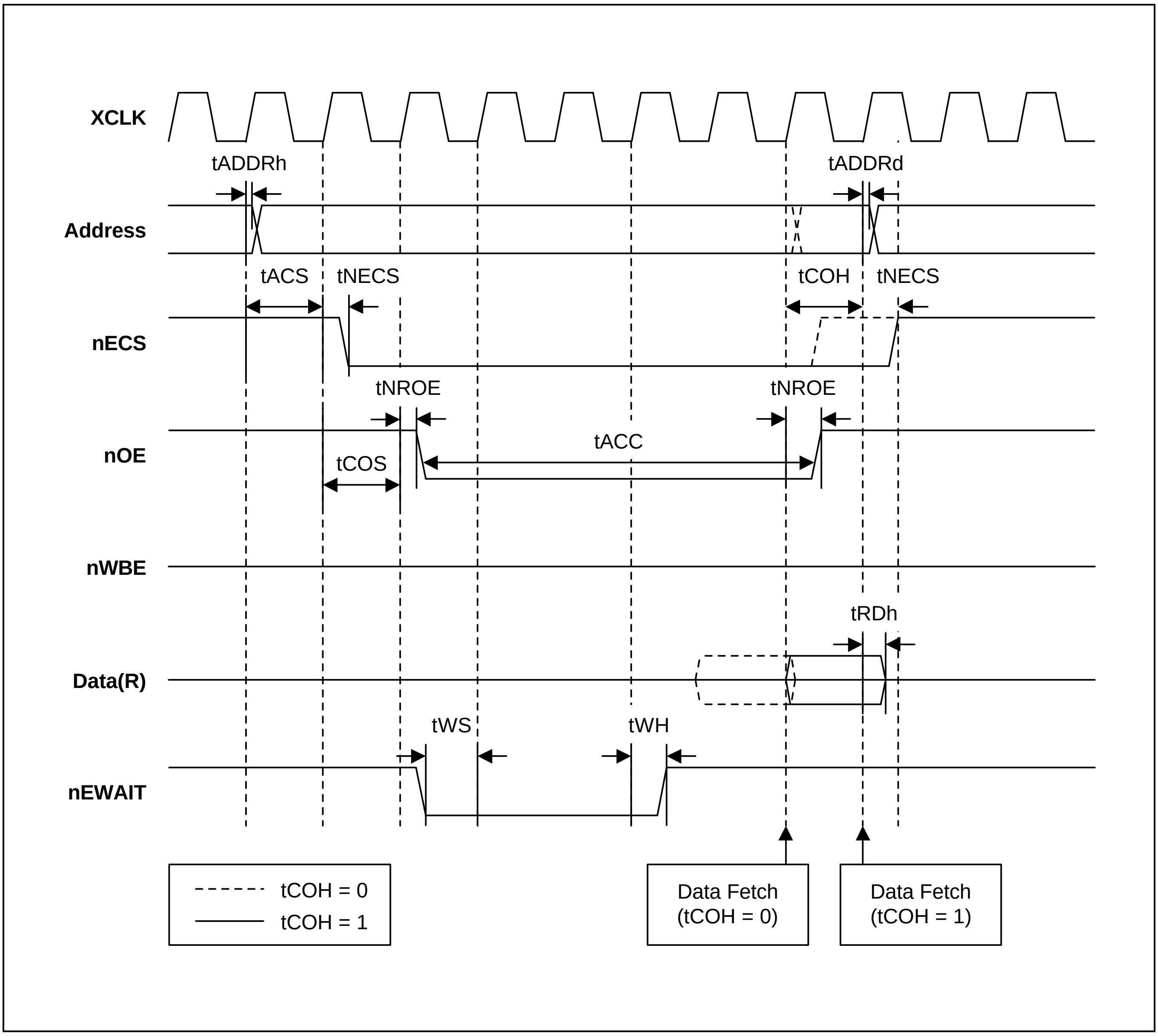


Figure 15 External I/O Read Timing with nEWAIT (t_{COH} = 1, t_{ACC} = 1, t_{COS} = 1, t_{ACS} = 1)

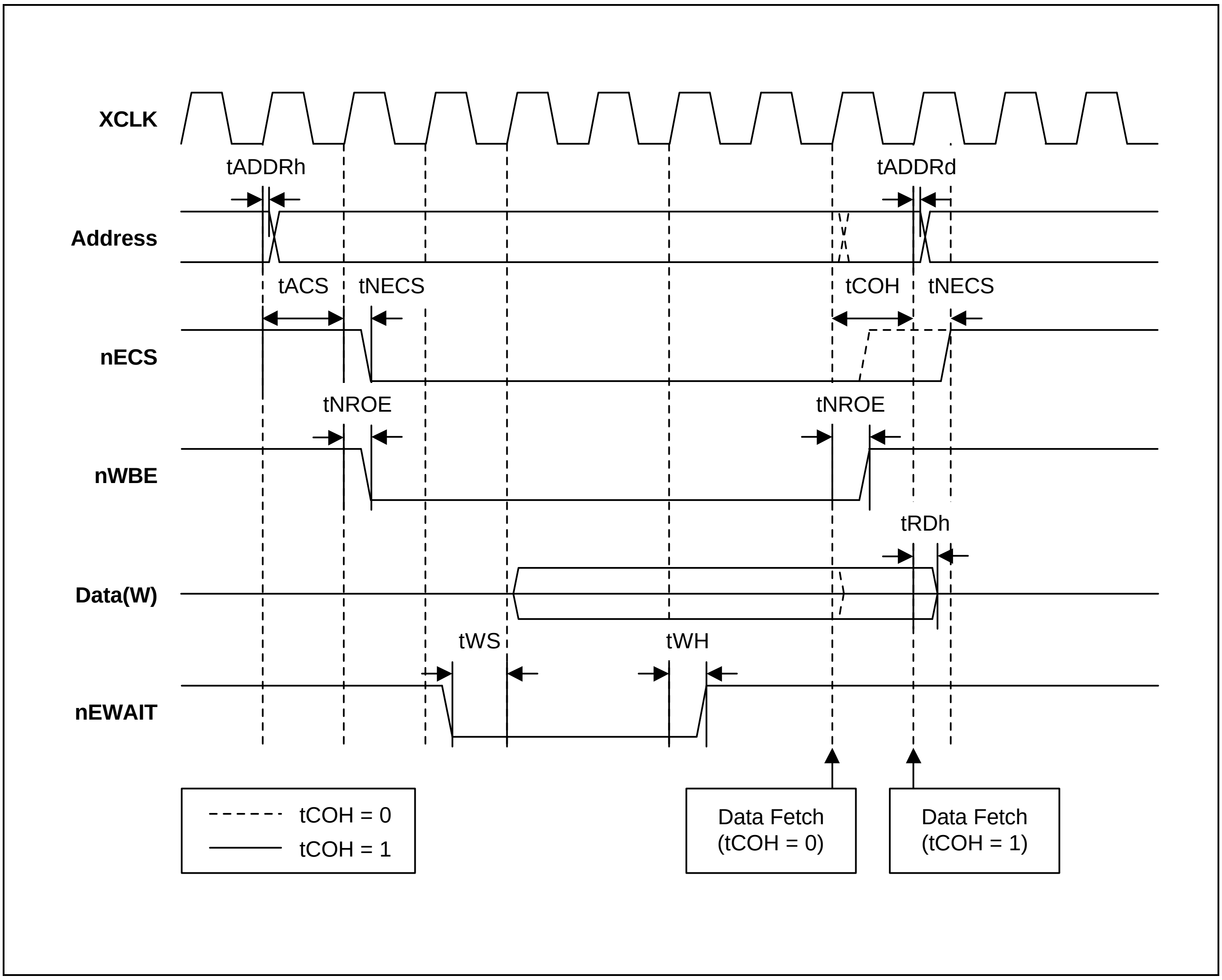


Figure 16 External I/O Write Timing with nEWAIT (t_{COH} = 1, t_{ACC} = 1, t_{COS} = 1, t_{ACS} = 1)

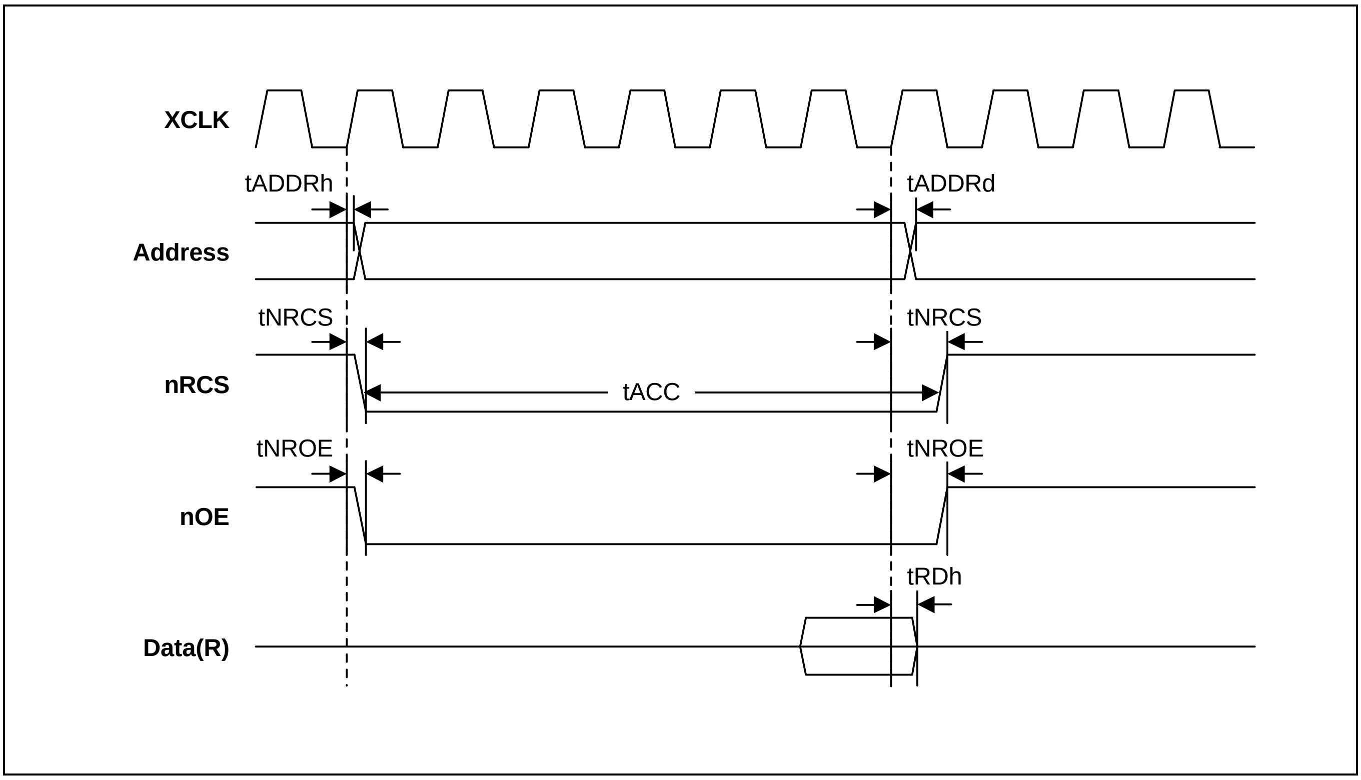


Figure 17 ROM/SRAM/Flash Read Access Timing

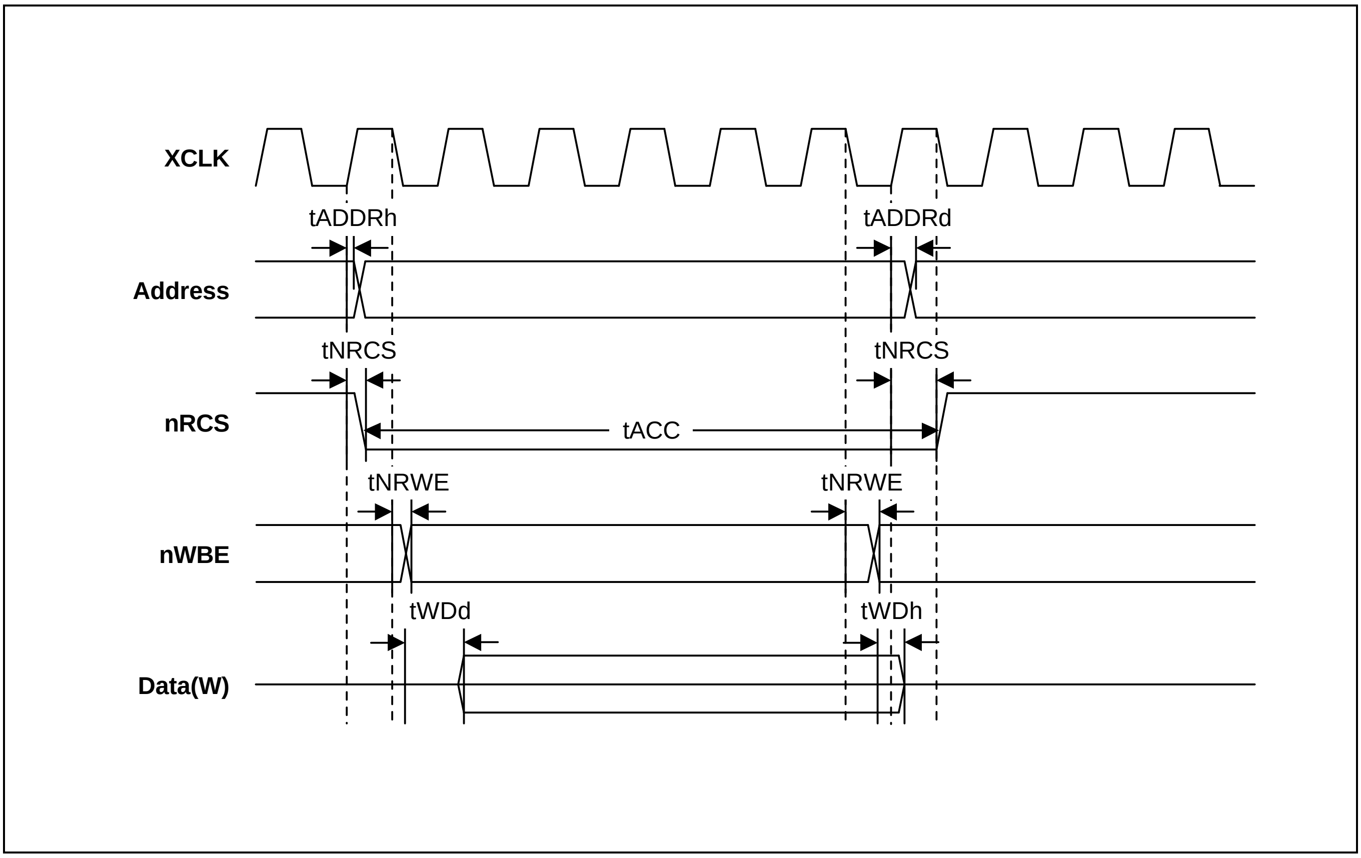


Figure 18 ROM/SRAM/Flash Write Access Timing

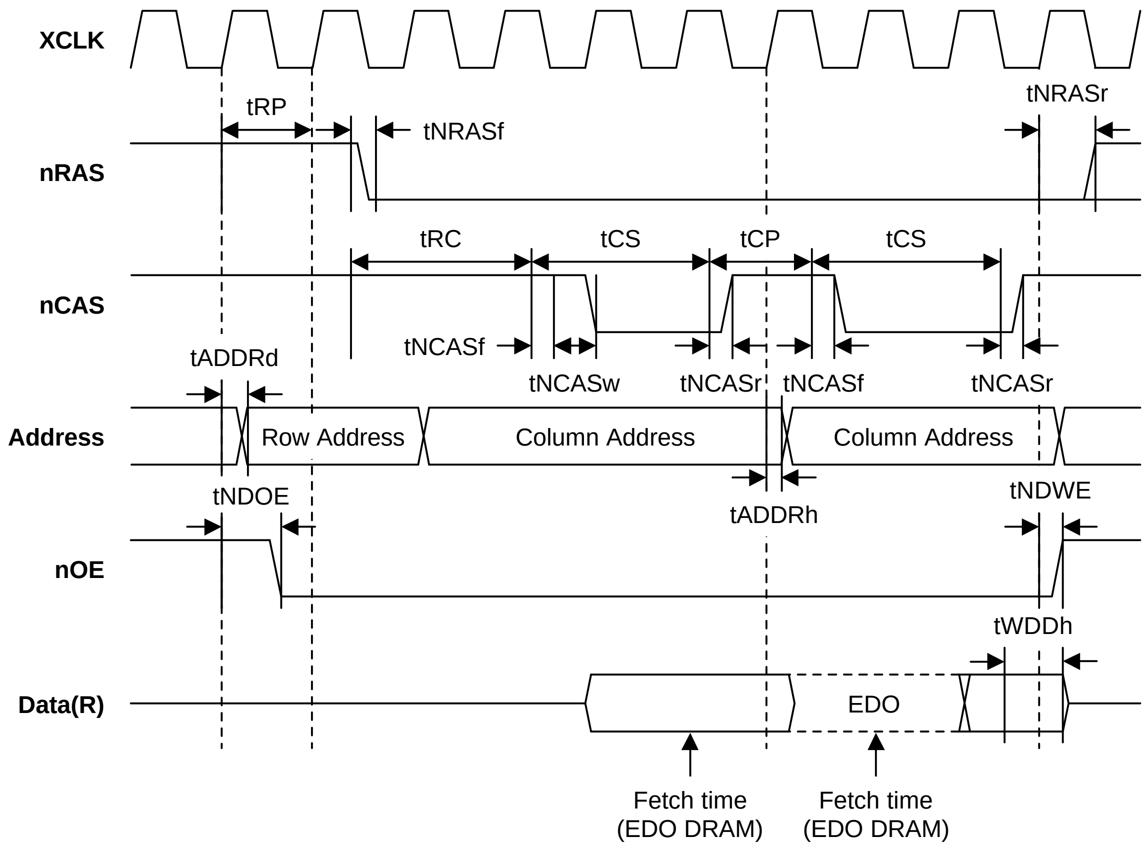


Figure 19 EDO/FP DRAM Bank Read Timing (Page Mode)

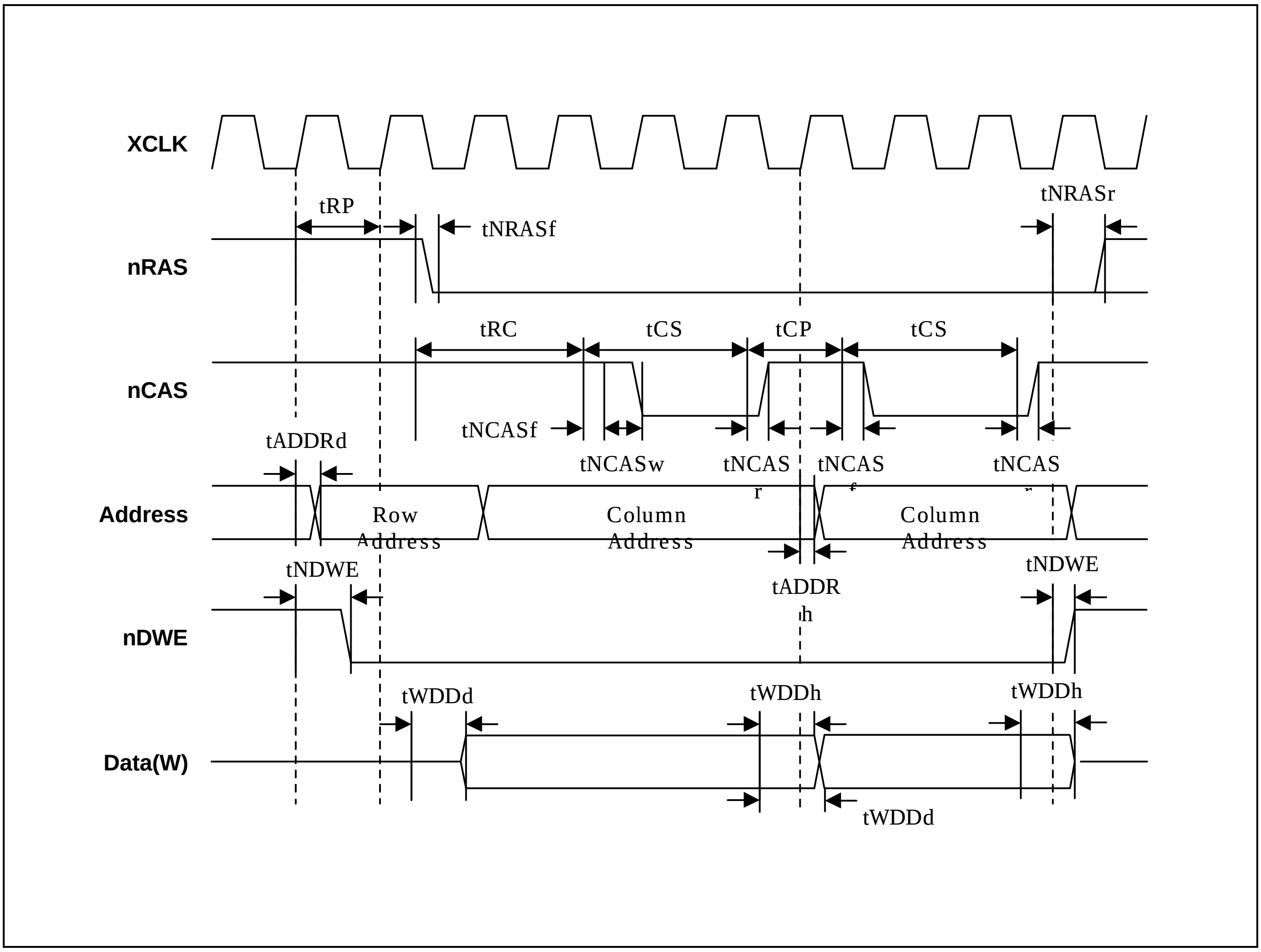


Figure 20 EDO/FP DRAM Bank Write Timing (Page Mode)

Timing Parameters for MII Transactions

The timing diagrams in this section conform to the guidelines described in the "Draft Supplement to ANSI/IEEE Std. 802.3, Section 22.3, Signal Characteristics."

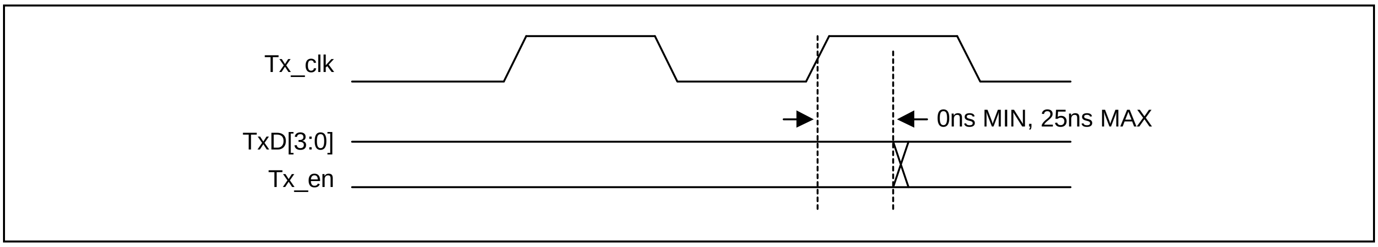


Figure 21 Transmit Signal Timing Relationship at MII

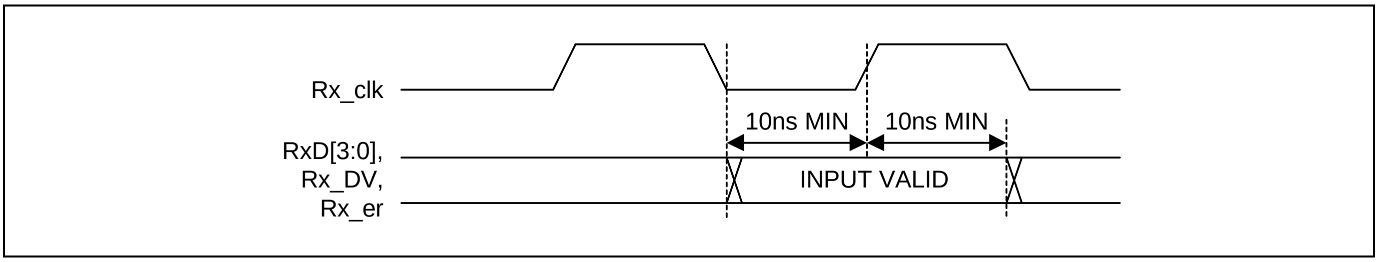


Figure 22 Receive Signal Timing Relationship at MII

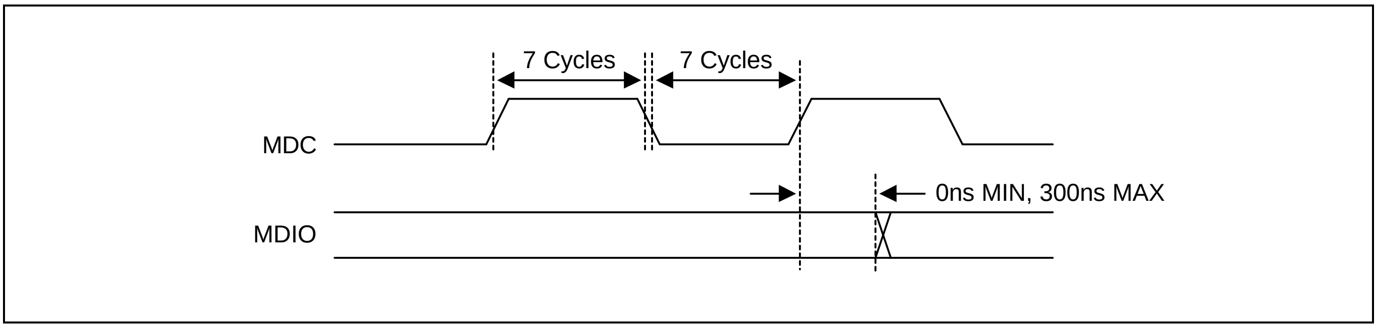


Figure 23 MDIO Sourced by PHY

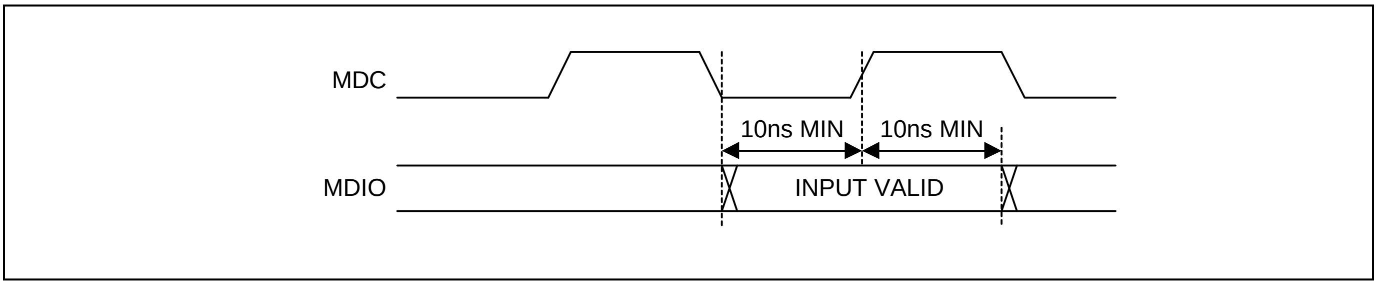


Figure 24 MDIO Sourced by STA

Timing Parameters for UTOPIA, An ATM-PHY Interface Specification

The AC characteristics are based on the timing specification for the receiver side pof a signal. The setup and hold times are defined with regard to a positive clock edge (see Figure 25). Tacking the actual used clock frequency into account (e.g. up to the max. frequency), the corresponding (min. and max.) transmit side "clock to output" propagation delay specifications can be derived. The timing references (tT5 to tT12) are according to Table 8 and 9.

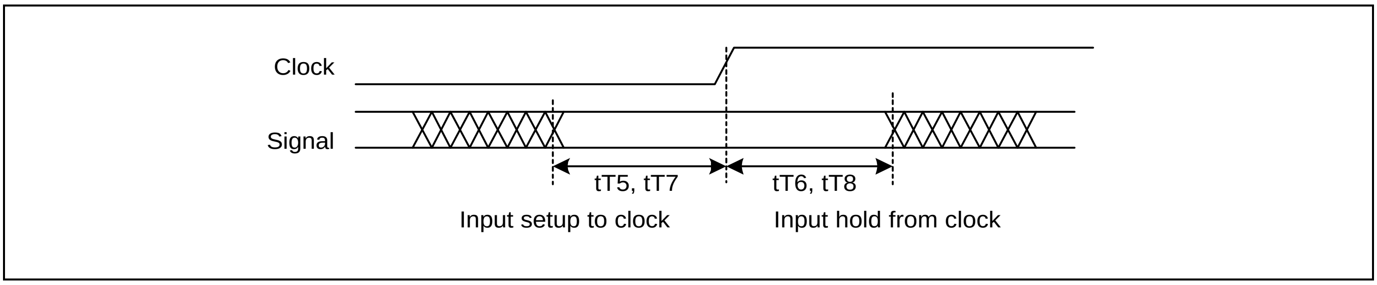


Figure 25 Aetup and hold time definition (single- and multi-PHY)

Figure 26 shows the tri-state timing for the multi-PHY application (multiple PHY devices, multiple output signals are multiplexed together).

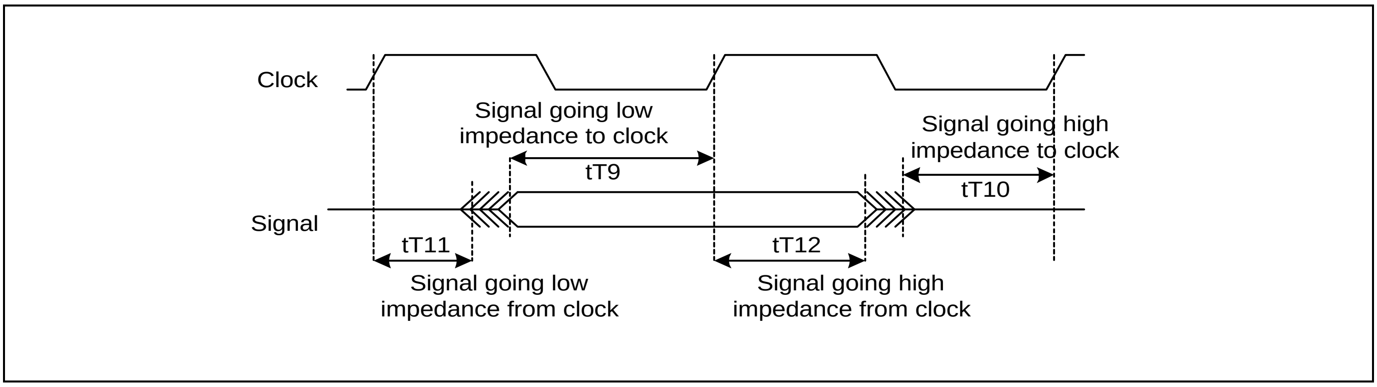


Figure 26 Tri-state timing (multi-PHY, multiple devices only)

In the following Tables, A⇒P (column DIR, direction) defines a signal from the ATM layer (transmitter, driver) to the PHY layer (receiver), A⇐P defines a signal from the PHY layer (transmitter, driver) to the ATM layer (receiver).

Signal Name	DIR	Item	Description	Min	Max
TxClk	A⇒P	f1	TxClk frequency (nominal)	0	33MHz
		tT2	TxClk duty cycle	40%	60%
		tT3	TxClk peak-to-peak jitter	-	5%
		tT4	TxClk rise/fall time	-	3ns
TxData[7:0], TxPrty, TxSOC, TxEnb*, TxAddr[4:0]	A⇒P	tT5	Input setup to TxClk	8ns	-
		tT6	Input hold from TxClk	1ns	-
TxFull*/TxClav[3:0]	A⇐P	tT7	Input setup to TxClk	8ns	-
		tT8	Input hold from TxClk	1ns	-
		tT9	Signal going low impedance to TxClk	8ns	-
		tT10	Signal going high impedance to TxClk	0ns	-
		tT11	Signal going low impedance from TxClk	1ns	-
		tT12	Signal going high impedance from TxClk	1ns	-

Table 8 Transmit timing (8-bit data bus, ☠33MHz at cell interface, multi-PHY)

Signal Name	DIR	Item	Description	Min	Max
RxClk	A⇒P	f1	RxClk frequency (nominal)	0	33MHz
		tT2	RxClk duty cycle	40%	60%
		tT3	RxClk peak-to-peak jitter	-	5%
		tT4	RxClk rise/fall time	-	3ns
RxEnb*, RxAddr[4:0]	A⇒P	tT5	Input setup to RxClk	8ns	-
		tT6	Input hold from RxClk	1ns	-
RxData[7:0], RxPrty, RxSOC, RxEmpty*/RxClav[3:0]	A⇐P	tT7	Input setup to RxClk	8ns	-
		tT8	Input hold from RxClk	1ns	-
		tT9	Signal going low impedance to RxClk	8ns	-
		tT10	Signal going high impedance to RxClk	0ns	-
		tT11	Signal going low impedance from RxClk	1ns	-
		tT12	Signal going high impedance from RxClk	1ns	-

Table 9 Receive timing (8-bit data bus, ☠33MHz at cell interface, multi-PHY)

9. PACKAGE DIMENSION

This section describes the mechanical data for the S5N8947 208-pin TQFP package.

208-TQFP-2828 PACKAGE DIMENSIONS

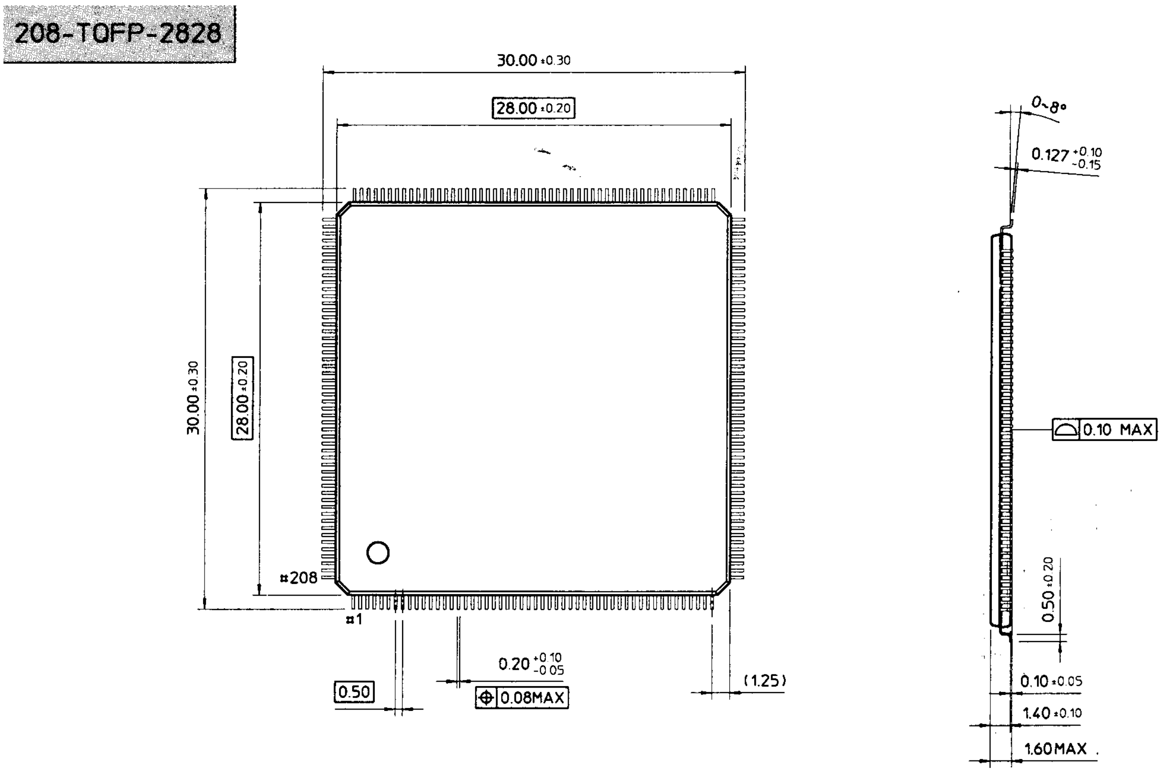


Figure 27 208-TQFP-2828 Package Dimensions

Revision History

Revision No.	Date	Description
0.1	2000-05-23	**S5N8947X** (Rev.0.1) Released.

IMPORTANT NOTICE

The information furnished by Samsung Electronics in this document is belived to be accurate and reliable. However, no resposibility is assumed by Samsung Electronics for its use, nor for any infringements of patents or other rights of third parties resulting from its use. No license is granted under any patents or patent rights of Samsung Electronics. Samsung Electronics reserves the right to make changes to its products or to discontinue any semiconductor product or service without notice, and advises its customers to obtain the latest version of relevant information to verify, before placing orders, that the information being relied on is current and complete.

For More Information

Tel: (82)-(31)-209-8301, Fax: (82)-(31)-209-8309

E-mail: kimil@sec.samsung.com

http://www.intl.samsungsemi.com