CS5954AM
USB Controller for NAND Flash

Cypress Semiconductor Corporation • 3901 North First Street • San Jose • CA 95134 • 408-943-2600
Document #: 38-08025 Rev. ** Revised May 28, 2002

Table of Contents

List of Figures

List of Tables

1.0 Definitions

USB	**U**niversal **S**erial **B**us
CS5954	The CS5954AM is a Cypress **USB** Controller, which provides multiple functions on a single chip.
QT	**Q**uick stream data **T**ransfer engine, which contains a small set of RISC instructions designed for the **CS5954AM USB** controller.
QTU	"**QT**" is a naming convention that represents QT Engine utility tools. For example: "**QTU**" indicates all tools that interface with the USB port.
R/W	**R**ead/**W**rite
PLL	**P**hase **L**ock **L**oop
WDT	**W**atchdog **T**imer
RAM	**R**andom **A**ccess **M**emory
2-wire serial interface	2-wire serial EEPROM interface
R0-R15	CS5954AM Registers R0-R7 data registers or general-purpose registers R8-R14 address/data registers, or general-purpose registers R15 stack pointer register
CS5954AM BIOS	A simulation model similar to 80×86 BIOS

2.0 References

[Ref. 1] SL11R_BIOS

[Ref. 2] SL11R Family Tools

[Ref. 3] Universal Serial Bus Specification 2.0

3.0 Introduction

3.1 Overview

The CS5954AM is a low-cost, full-speed Universal Serial Bus (USB) RISC-based controller specifically designed for mass storage applications using NAND Flash technology. It contains a 16-bit RISC processor with built-in BIOS ROM to greatly reduce firmware development work. Its 2-wire serial EEPROM interface offers low cost storage for USB device configuration and customer's product-specific functions. New functions can be programmed into the 2-wire serial interface by downloading them from a USB Host PC. This unique architecture provides the ability to upgrade products in the field without changing the peripheral hardware.

The CS5954AM Processor can execute code from either internal ROM/RAM or external ROM and SRAM. The CS5954AM Programmable bidirectional data port supports I/O mode. A built-in USB port supports data transfers up to 12 MBits/sec which is the maximum full speed USB transfer rate. All USB protocol modes are supported: Isochronous (up to 1024 bytes/packet), Bulk, Interrupt, and Control. The CS5954AM requires a 3.3V power supply, which can be powered via a USB host PC or a Hub. Suspend/Resume, and Low power modes are available.

The CS5954AM offers a cost effective solution for NAND Flash products.

3.2 Features

- **Cypress offers a development kit for each of its product lines. These development kits include multiple peripheral mini-port class drivers for Microsoft® Windows® 98/ME/2000, firmware source code and demo USB source code for a variety of applications. Also available is a debugger and assembler with a reference demo board**.
- **48-MHz 16-bit RISC processor**.
- **Up to 16 bits of programmable bidirectional data I/O**.
- **Up to 32 bits of general-purpose I/O (GPIO)**.
- **6K × 8 internal mask ROM with built-in BIOS supporting a comprehensive list of interrupt calls (see [Ref. 1] SL11R_BIOS for detailed information). These include USB functions, 2-wire serial interface boot-up option (boot-up from 2-wire serial interface or external ROM). Executable code can also run from 8-bit or 16-bit external memory**.
- **3K × 8 internal RAM that can be configured as the USB Ping-Pong buffer for USB DATA0 and DATA1 packets. It can also be used for data and/or code**.
- **Two-wire serial EEPROM interface port with CS5954AM BIOS support to allow on-board EEPROM programming**.
- **Flexible programmable external memory wait-states and a 8/16 data path**.
- **Up to 16-bit address for extended memory interface port for external SRAM and ROM**.

- **Supports 12-MHz/48-MHz external crystal or clock**.
- **Executable code or data can be loaded from the USB port. The code/data is moved to RAM for debugging purposes (using a break point register), or to be programmed via a two-wire serial EEPROM**.
- **USB port (12 Mbits/sec), including a built-in USB transceiver. All USB standard protocol modes are supported: Isochronous mode (up to 1024 packet size), Bulk, Interrupt, and Control modes**.
- **There are four available endpoints. Data can be sent/received to/from the data port independently**.
- **Two general-purpose timers and a Watchdog timer (WDT)**.
- **Suspend/resume and low-power modes are supported**.
- **USB generic mini-port driver for WIN98/2000 is available**.
- **Debugger and QT-Assembler are available**.
- **Package: 100 PQFP**.
- **Power requirements: 3.3V**.

3.3 CS5954AM 16-bit RISC Processor

The CS5954AM can be used as a general-purpose 16-bit embedded processor. It includes a USB interface and up to 32 bits of GPIO supporting a variety of functions and modes. Also, the CS5954AM contains a 2-wire serial EEPROM interface, an additional SRAM interface for extended memory, two timers, a Watchdog timer, an internal mask BIOS ROM (3k×16) and an SRAM (3K×8). The CS5954AM is optimized to offer maximum flexibility in the implementation of a variety of USB-to-GPIO devices such as a NAND flash controller.

The CS5954AM contains a specialized instruction set (RISC) that is highly optimized to provide efficient coding for a variety of USB based applications. The CS5954AM includes a simple software interface for all USB transaction processing, which supports Bulk mode (up to 64 Bytes/packet), Isochronous mode (up to 1024 Bytes/packet), and all Interrupt and Control modes.

3.4 3K×16 Mask ROM and BIOS

The CS5954AM has a built-in 3K×16 Mask ROM that contains the CS5954AM BIOS. This BIOS ROM provides the software interface for the USB and a boot-up option for a 2-wire serial interface or an external 8/16 EEPROM.

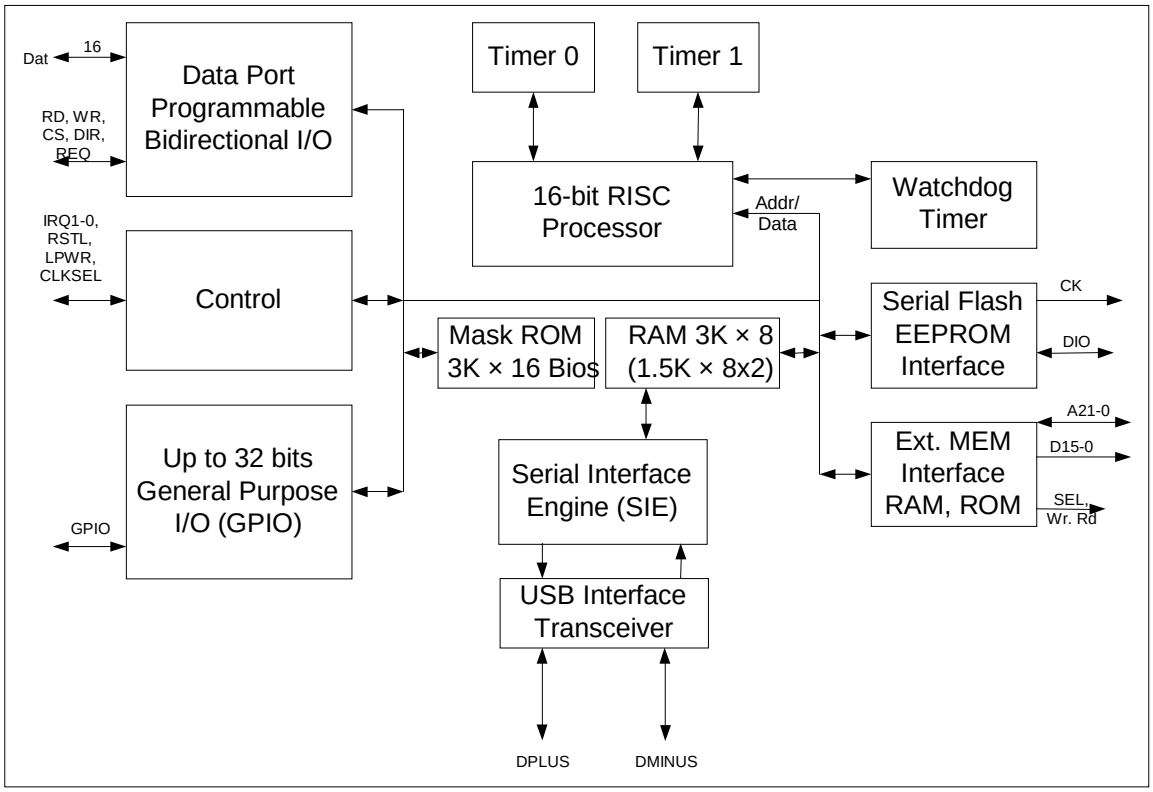


Figure 3-1. CS5954AM Block Diagram

3.5 Internal RAM

The CS5954AM contains 3K×8 internal RAM. The RAM can be used for code/program, variables, buffer I/O, and USB packets. This memory can be accessed by the 16-Bit processor for data manipulation or by the SIE (Serial Interface Engine), which receives or sends USB host data.

3.6 Clock Generator

A 12- or 48-MHz external crystal, or logic-level clock can be used with the CS5954AM. Two pins, XIN and XOUT, are provided to connect a low-cost crystal circuit to the device. If a logic-level clock is available, it may be connected directly to the XIN pin instead of using a crystal.

Register C006 must be configured appropriately depending on the frequency used.

3.7 USB Interface

The CS5954AM has a built-in SIE and USB transceiver that meet the USB specification v2.0. The transceiver is capable of transmitting or receiving serial data at the USB maximum data rate of 12 Mbits/sec. The CS5954AM controller supports four endpoints. Endpoint 0 is the default pipe and is used to initialize and manipulate the peripheral device. It also provides access to the peripheral device's configuration information, and supports control transfers. Endpoint 1, 2, and 3 support interrupt transfers, Bulk transfers (up to 64 Bytes/packet), or Isochronous transfers (up to 1024 Bytes/packet size).

3.8 Processor Control Registers

The CS5954AM provides software control registers that can be used to configure chip mode, clock generator, and software breakpoint, and to read the BIOS version.

3.9 Interrupts

The CS5954AM provides 128 interrupt vectors for its BIOS software interface (see [Ref. 1] SL11R_BIOS).

3.10 2-Wire Serial EEPROM Interface

The CS5954AM provides an interface to an external serial EEPROM. The interface is implemented using general-purpose I/O signals. A variety of serial EEPROM formats can be supported; currently the BIOS ROM supports a 2-wire serial EEPROM. A serial EEPROM can be used to store specific peripheral USB configuration and value-added functions. In addition, serial EEPROM can be used for field product upgrades.

3.11 External SRAM Interface

The CS5954AM provides a multiplexed address port and an 8-/16-bit data port. This port can be configured to interface to an external SRAM.

3.12 General Timers and Watchdog Timer

The CS5954AM has two built-in programmable timers that can provide an interrupt to the CS5954AM engine. On every clock tick (which is one microsecond), the timers decrement. An interrupt occurs when the timer reaches zero. A separate Watchdog timer is also provided to provide a fail-safe mechanism. The Watchdog timer can also interrupt the CS5954AM processor.

3.13 Special GPIO Functionality for Suspend, Resume, and Low-power Modes

The CS5954AM CPU supports suspend, resume, and CPU low-power modes. The CS5954AM BIOS assigns GPIO29 for the USB DATA+ line pull-up (this pin can simulate USB cable removal or insertion while the USB power is still applied to the board) and the GPIO20 for controlling the power-off function.

3.14 CS5954AM Interface Modes

The CS5954AM has a general-purpose I/O interface mode.[1]

3.14.1 General-purpose I/O Mode (GPIO)

In the GPIO mode, the CS5954AM has up to 32 general-purpose I/O signals available. However, four pins that are used by the 2-wire serial interface cannot be used as GPIO pins. On any other available general-purpose programmable I/O, the pins can be programmed for peripheral control and/or status.

Note:

1. The 2-wire serial interface I/O pins are fixed in all cases.

4.0 Interface

4.1 Internal Masked ROM: 0xE800–0xFFFF

The CS5954AM has a built-in 3K×16 internal masked ROM that contains software bootstrap code to allow programs in an external 8-/16-bit ROM to be executed. The ROM code can also load data from the 2-wire serial interface into internal RAM for execution. In addition, the internal BIOS ROM contains the interrupt service routines (see [Ref. 1] SL11R_BIOS for information) that support the USB, 2-wire serial interface boot-up option (boot-up from 2-wire serial interface or external ROM). This CS5954AM BIOS ROM eases software development of all CS5954AM interfaces. The CS5954AM chip is ready for all the USB enumeration and download of program code.

The CS5954AM internal masked ROM (i.e. SL11R BIOS) is mapped from address 0xE800 to 0xFFFF. On power-up or hardware reset, the CS5954AM processor jumps to the address of 0xFFF0, which contains a long jump to the beginning of the internal ROM of address 0xE800. See *Table 4-1*.

Table 4-1. Internal Masked ROM (CS5954AM BIOS)

Address	Memory Description
0xE800–0xFFEF	CS5954AM BIOS Code/Data Space
0xFFF0–0xFFF3	Jump to 0xE800
0xFFF4–0xFFF9	Reserved for Future Use
0xFFFA–0xFFFB	ROM BIOS Checksum
0xFFFC–0xFFFD	CS5954AM BIOS Revision
0xFFFE–0xFFFE	Peripheral Revision
0xFFFF–0xFFFF	QT Engine Instruction Revision

4.2 External ROM: 0xC100–0xE800

The CS5954AM BIOS ROM reserves addresses from 0xC100 to 0xE800 for external ROM. During BIOS initialization, the CS5954AM will scan for the signature ID (0xCB36) at location 0xC100. After a valid signature is detected, execution will begin at address 0xC102 (see [Ref. 1] SL11R_BIOS for more information). The signal nXROMSEL is used to enable the external ROM. It is mapped from 0xC100 to 0xE800 by default. However, the Extended Memory Control can be used to configure multiple windows for external ROM set-up.[2]

4.3 Internal RAM: 0x0000–0x0BFF

The CS5954AM contains a 1.5K×16 internal RAM. This memory is used to buffer USB packets and is accessed by the 16-bit processor and the SIE (Serial Interface Engine). USB transactions are automatically routed to the memory buffer. The CS5954AM BIOS uses this internal RAM for USB buffers, BIOS variables and user data/code. Executable code or data can reside in multiple locations: internal masked ROM (3K×16), internal RAM (3K×8), external ROM, and external SRAM. Program code or data can also be loaded to either the internal or the external RAM from the USB port, the RS232 port, or the 2-wire serial interface.

The CS5954AM Internal RAM is mapped from 0x0000 to 0x0BFF. See internal RAM memory usage in *Table 4-2* below.

Table 4-2. Internal RAM Memory Usage

Address	Memory Description
0x0000–0x00FF	Hardware/Software Interrupts
0x0100–0x01FF	Register Banks/USB Control/Software Stack
0x0200–0x021F	Hardware Interrupts Stack
0x0220–0x0343[3]	CS5954AM BIOS Internal Buffers and Variables
0x0344–0x0BFF	User's Programming Space

- **The addresses from 0x0000 to 0x00FF are reserved for hardware and software interrupt vectors (see [Ref. 1] SL11R_BIOS for more information).**
- **Addresses from 0x0100 to 0x01FF are reserved for Internal Register Banks (CS5954AM register R0-R15 bank 0 and R0-R15 bank 1) and the software stack. Others are reserved for USB Control registers and other read/write control registers.**
- **Addresses from 0x0200 to 0x021F are reserved for the hardware interrupt stack.**
- **Addresses from 0x0220 to 0x0343 are available internal RAM for application software. Software can be downloaded via the USB port (see [Ref. 1] SL11R_BIOS for more information).**

Notes:

2. The Address space from 0x8000-0xC100 can also be used as the external ROM (see the External Memory Control Set-up for more detail).
3. This address may be changed due to CS5954AM BIOS revision updates. The new CS5954AM BIOS may require more internal memory for its variable usage in any new CS5954AM BIOS.

4.4 Clock Generator

The CS5954AM has an option to use either a 48-MHz or 12-MHz external crystal or oscillator as its clock source. CS5954AM includes an internal PLL that can be configured by software. At power-up, the CS5954AM BIOS default configuration sets the processor clock to run at 2/3 of XIN (of the external provided clock).

Example 1 Changing CS5954AM CPU Speed

The default of the CS5954AM BIOS assumes a 48-MHz input clock, so the CS5954AM processor clock is (2/3)*48 MHz = 32 MHz. See example below.

```
mov     [0xC006],0x10     ;clock = 2/3*XIN
mov     [0xC008],0        ;CPU clock at 32 MHz
```

If the XIN input clock is 48 MHz, then the maximum speed of the CS5954AM processor can be set as follows:

```
mov     [0xC006],0        ;clock = set up at XIN clock input
mov     [0xC008],0        ;CPU clock at 48 MHz
```

If the XIN input clock is 12 MHz, then the maximum speed of the CS5954AM processor can be set to:

```
mov     [0xC006],0x40     ;clock = 4*XIN
mov     [0xC008],0        ;CPU clock at 48 MHz
```

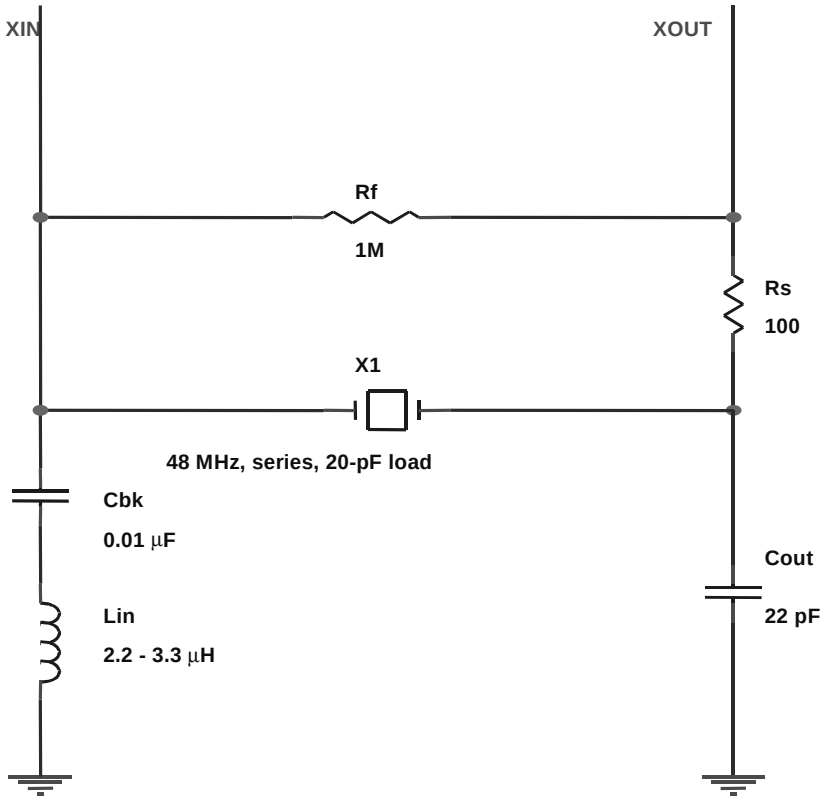


Figure 4-1. 48-MHz Crystal Circuit

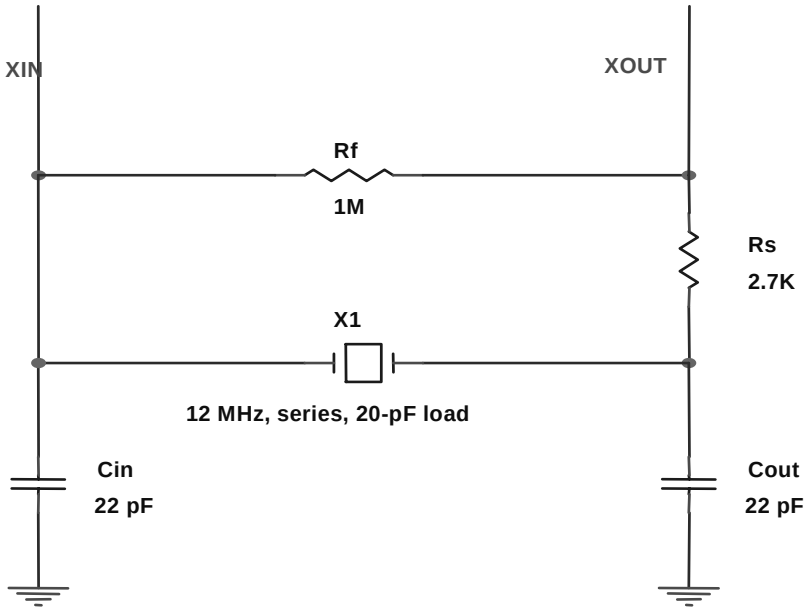


Figure 4-2. 12-MHz Crystal Circuit[4]

4.5 USB Interface

The CS5954AM has a built-in transceiver that meets USB specification v2.0. The transceiver connects directly to the physical layer of the USB engine. It is capable of transmitting or receiving serial data at the full speed USB maximum data rate of 12 Mbits/sec. The 16-bit processor has the ability to set up pointers and block sizes in buffer memory for USB transactions.

The CS5954AM controller contains a number of registers that provide overall control and status functions for USB transactions. The first set of registers is for control and status functions, while the second group is dedicated to specific endpoint functions. Communication and data flow on the USB is implemented using endpoints. These uniquely identifiable entities are the terminals of communication flow between a USB host and USB devices. Each USB device is composed of a collection of independently operating endpoints. Each endpoint has a unique identifier: the endpoint number. (See USB specification v2.0. sec 5.3.1.)

The CS5954AM also includes the CS5954AM BIOS that provides a set of subroutines via interrupt calls for all USB interface functions required to communicate with a USB host (refer to [Ref. 1] SL11R_BIOS for more information). The CS5954AM BIOS greatly simplifies the firmware/software development cycle.

4.5.1 USB Global Control and Status Register (0xC080: R/W)

The USB Global Control and Status Register allows high-level control. The Global Control and Status Register bits are defined as follows.

D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
0	0	0	0	0	0	0	0	0	0	0	0	UA	US	UR	UE

D15-D4	Reserved	
D0	UE	USB Enable = “1,” Overall USB enable/disable bit
D1	UR	USB Reset = “1,” USB received Reset command
D2	US	USB SOF = “1,” USB received SOF command
D3	UA	USB Activity = “1,” Activity seen

- **Suspend state should be entered if there is no activity after 3 mS (UA).**
- **The US and UA bits are automatically cleared after they are read by the CS5954AM processor.**
- **D15–D4 are the reserved bits, should be written with zeros.**
- **The CS5954AM BIOS will set the UE = 1 upon reset.**

Note:
4. Bit C2 =1 must be set from configuration address (0xC006). See section 4.10.1 for CPU control speed.

4.5.2 USB Frame Number Register (0xC082: Read-only)

The Frame Number Register contains the 11-bit ID number of the last SOF received by the device from the USB host.[5]

D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
0	0	0	0	0	S10	S9	S8	S7	S6	S5	S4	S3	S2	S1	S0

D15-D11	Reserved	set to all zeros.
D10-D0	S10-S0	SOF ID number of last SOF received.

4.5.3 USB Address Register (0xC084: R/W)

Address Register holds the USB address of the device assigned by the Host–initialized to address 0x0000 upon power-up.[6]

D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
0	0	0	0	0	0	0	0	0	A6	A5	A4	A3	A2	A1	A0

D15-D7	Reserved	set to all zeros.
D6-D0	A6-A0	USB address of device after assignment by host.

4.5.4 USB Command Done Register (0xC086: Write-only)

This is the USB Command Done Register. It is only used by the control point (endpoint 0).[7]

D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	E

D15–D1	Reserved	set to all zeros
D0	E	set E = 0 for successful command completion
		set E = 1 for error command completion.

4.6 USB Endpoint 0 Control and Status Register (0xC090: R/W)

See the USB Endpoint 3 control and status register for more information.

4.7 USB Endpoint 1 Control and Status Register (0xC092: R/W)

See the USB Endpoint 3 control and status register for more information.

4.8 USB Endpoint 2 Control and Status Register (0xC094: R/W)

See the USB Endpoint 3 control and status register for more information.

4.9 USB Endpoint 3 Control and Status Register (0xC096: R/W)

4.9.1 General Description for All Endpoints from Endpoint 0 to Endpoint 3

The CS5954AM controller supports four endpoints. Endpoint 0 is the default pipe and is used to initialize and control the peripheral device. It also provides access to the peripheral device's configuration information, and supports control transfers. Endpoints 1, 2, and 3 support interrupt transfers, bulk transfers up to 64 Bytes/packet, or Isochronous transfers up to 1024 Bytes/packet size.

4.9.2 USB Endpoint Control (for Writing)

Each of the endpoint control registers, when written, have the following functions assigned.

Bit Position	Bit Name	Function
D0	ARM	Allows enabled transfers when set to "1." Cleared to "0" when transfer is complete.
D1	Enable	When set to "1" it allows transfers to this endpoint. When set to "0" USB transactions are ignored. If enable = "1" and Arm = "0" the endpoint will return NAK to USB transmissions.
D2	DIR	When set to "1" it transmits to Host (IN). When "0" receive from Host (OUT).
D3	ISO	When set to "1" it allows Isochronous mode for this endpoint.
D4	Stall	When set to "1" it sends Stall in response to next request on this endpoint.
D5	Zero Length	When set to "1" it sends a zero length packet.
D6–D15	Not Defined	Set to logic "0"s.

Notes:

5. The CS5954AM BIOS uses this register to detect USB activity for the internal idle task.
6. The CS5954AM BIOS modifies this register upon receiving the SET_ADDRESS from the host. (See [Ref. 3] Universal Serial Bus Specification v2.0 sec. 9 for more information.)

4.9.3 USB Endpoints Status (for Reading)

Reading the Endpoint Status Register provides Status information relative to the packet that has been received or transmitted. The register is defined as follows.

Bit Position	Bit Name	Function
D0	Arm	If “1,” the endpoint is armed.
D1	Enable	If “1,” the endpoint is enabled.
D2	DIR	Direction bit. If “1,” set to transmit to Host (IN). If “0,” set to receive from Host (OUT).
D3	ISO	If “1,” isochronous mode selected for this endpoint.
D4	Stall	If “1,” endpoint will send stall on USB when requested.
D5–D12	Not used	Read returns logic “0”s.
D13	Setup	If “1,” a Setup packet has been received.
D14	Error	If “1,” an error condition occurred on last transaction for this endpoint.
D15	Done	If “1,” transaction completed. Arm Bit is cleared to “0” when Done Set.

- **Endpoint 0 is set up as a control endpoint. The DIR bit is read-only, and indicates the direction of the last completed transaction. If the direction is incorrect, it is the firmware’s responsibility to handle the error. On other endpoints, DIR bit is written, and if the direction of the transfer does not match the DIR bit, then the transaction is ignored.**
- **At the end of any transfer to an armed and enabled endpoint (with the correct DIR bit), an interrupt occurs, and vectors to a different location depending upon whether an error occurred or not. At the end of this transfer, the corresponding endpoint is disarmed (the Arm bit is cleared), and the DATA0/DATA1 toggle bit is advanced if no error occurred. If a packet is received with an incorrect toggle state, the packet is ignored so that the host will resend the data.**
- **The DATA0/DATA1 bit is automatically toggled by the hardware. To reset this DATA0/DATA1 toggle bit to DATA0, the Enable on the D1 bit should be cleared to “0” and then set to “1.”**
- **When the zero length bit (D5) is set, the host will receive the zero length USB packet, regardless of the number of bytes in the USB count register.**
- **The CS5954AM BIOS has full control of USB endpoint 0. The CS5954AM BIOS responds to all numeration from the host. On other endpoints, the CS5954AM BIOS can be used to control under BIOS interrupt calls (see [Ref. 1] SL11R_BIOS).**
- **The CS5954AM BIOS will set all USB control and status registers for endpoint 1 through 3 to zero upon receiving the SET_CONFIG command from host. (See [Ref.3] Universal Serial Bus Specification v2.01, sec. 9 for more information.)**

4.9.4 USB Endpoint 0 Address Register (0x0120: R/W)

This is the pointer to memory buffer location for USB reads and writes to this endpoint. At the end of any transfer, this register will contain its original value plus the value in the USB endpoint count register.

4.9.5 USB Endpoint 1 Address Register (0x0124: R/W)

See USB Endpoint 0 Address Register (0x0120: R/W).

4.9.6 USB Endpoint 2 Address Register (0x0128: R/W)

See USB Endpoint 0 Address Register (0x0120: R/W).

4.9.7 USB Endpoint 3 Address Register (0x012C: R/W)

See USB Endpoint 0 Address Register (0x0120: R/W).

4.9.8 USB Endpoint 0 Count Register (0x0122: R/W)

This register is used to set the maximum packet size for the USB transfer. At the end of a successful transfer, the USB endpoint Count Register is set to zero.

4.9.9 USB Endpoint 1 Count Register (0x0126: R/W)

See USB Endpoint 0 Count Register (0x0122: R/W).

4.9.10 USB Endpoint 2 Count Register (0x012A: R/W)

See USB Endpoint 0 Count Register (0x0122: R/W).

4.9.11 USB Endpoint 3 Count Register (0x012E: R/W)

See USB Endpoint 0 Count Register (0x0122: R/W).

Note:

7. The CS5954AM BIOS modifies this register upon command completion on endpoint 0.

4.10 Processor Control Registers

The CS5954AM provides software control registers that can be used to configure the chip mode, clock control, read software version, and software breakpoint control.

4.10.1 Configuration Register (0xC006: R/W)

The Configuration Register is used to configure the CS5954AM into the appropriate mode, and to select a clock multiplier.[8]

D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
0	0	0	0	0	0	0	0	0	C2	C1	C0	CD	M1	M0	MD

C2	C1	C0	PCLK	RCLK	OE
0	0	0	XIN	XIN	0
0	0	1	2/3*XIN	XIN	0
0	1	0	X_PCLK	XIN	0
0	1	1	2/3*XIN	XIN	1
1	0	0	4*XIN	4*XIN	0
1	0	1	8/3*XIN	4*XIN	0
1	1	0	4*XIN	4*XIN	1
1	1	1	8/3*XIN	4*XIN	1

D3 CD

If Clock Disable bit = “1,” this Clock Configuration register can no longer be modified through software writes. It is a “sticky bit” used to lock the configuration through a write to this bit in the boot prom code.

- **On the CS5954AM chip set, this bit will be set to zero.**
- **There is one mode defined in this document: general-purpose IO (GPIO) mode.**

D2, D1 M1,M0: CS5954AM modes are selected as shown here.

M1	M0	Mode
0	0	GPIO
0	1	Reserved
1	0	Reserved
1	1	Reserved

D0 MD

If Mode Disable bit = “1,” this Configuration register can no longer be modified through software writes. It is a “sticky bit” used to lock the configuration through a Write to this bit in the boot prom code.[9]

D15-D7 Reserved should be set to all zeros

where

PCLK is connected to the CS5954AM processor clock.

RCLK is the resulting clock that connects to other modules (i.e., USB engine).

OE when **OE** = 1, the **X_PCLK** (pin 59) will become an output pin of the **PCLK** value.

- **When the XIN input pin is fed with a 12-MHz signal, the software should set C2 to “1” to enable the PLL.**
- **X_PCLK is a bidirectional pin allowing an additional clock input for PCLK when selected or an observation pin for PCLK when OE = “1.”**
- **The X_PCLK can be used as the input clock like XIN, but only when mode C2=0, C1=1, C0=0.**
- **Upon reset, the CS5954AM BIOS will set this register equal to 0x0010 (i.e., C2=0, C1=0, C0=1, PCLK=XIN, RCLK=XIN, OE=0, M1–M0=0=GPIO Mode).**

Notes:

8. D6–4 and C2–0 are Clock Configuration bits. These bits select the clock source. The clock may come from an outside pin (XIN or X_PCLK) or it may come from the PLL multiplier, as indicated in the table.
9. By default, this bit will be set to zero by the CS5954AM BIOS.

4.10.2 Speed Control Register (0xC008: R/W)

The Speed Control Register allows the CS5954AM processor to operate at a number of speed selections. A four-bit divider (SPD3–0 + 1) selects the speed as shown below. Speed will also depend on the clock multiplier. See Configuration Register (0xC006: R/W) for more information.

D15–D4	D3	D2	D1	D0
0	SPD3	SPD2	SPD1	SPD0

D3–D0 SPD3–SPD0 Speed selection bits

SPD3–0	CS5954AM Speed[10]
0000	48.00 MHz
0001	24.00 MHz
0010	16.00 MHz
0011	12.00 MHz
0100	09.60 MHz
0101	08.00 MHz
0110	06.86 MHz
0111	06.00 MHz
1000	05.33 MHz
1001	04.80 MHz
1010	04.36 MHz
1011	04.00 MHz
1100	03.69 MHz
1101	03.42 MHz
1110	03.20 MHz
1111	03.00 MHz

D15–D4 Reserved should be set to all zeros.

4.10.3 Power-down Control Register (0xC00A: R/W)

During power down mode, the peripherals are put in a **"pause"** state. All counters and timers stop incrementing.

D15–D6	D5	D4	D3	D2	D1	D0
0	USB	GPIO	PUD1	PUD0	SUSPEND	HALT

There are two ways to enter power-down mode: **Suspend** or **Halt**.

D5 USB Enable restarts on USB transition resulting in device power-up.

D4 GPIO Enable restarts on GPIO transition resulting in device power-up (see GPIO Interrupt Control Register (0xC01C:R/W)).

D3–D2 PUD1–PUD0 Power-up Delay Selection. Four delays are provided and selected using these select bits. This is time from power-up until processor starts executing allowing clock to settle.

PUD1	PUD0	Power-up Delay
0	0	0 milliseconds
0	1	1 milliseconds
1	0	8 milliseconds
1	1	64 milliseconds

D1 SUSPEND To save power, Suspend mode stops all clocks in the CS5954AM.

This mode ends with a transition on either USB or any Interrupt. It is followed by a delay set in the power-up delay bit fields.

D0 HALT ends with an interrupt.

D15–D6 Reserved should be set to all zeros.

Note:

10. Upon reset, the lowest speed is selected for low-power operation. The CS5954AM BIOS will configure the clock to 24 MHz as part of its initialization.

4.10.4 Breakpoint Register (0xC014: R/W)

The Breakpoint Register holds the breakpoint address. Access to this address causes an INT127.

D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
A15	A14	A13	A12	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0

D15–0 A15–0 Breakpoint address.

4.11 Interrupts

The CS5954AM provides 128 interrupt vectors. The first 64 vectors are hardware interrupts and the next 64 are software interrupts (see the [Ref. 1] SL11R_BIOS for more information).

4.11.1 Hardware Interrupts

The CS5954AM allocates addresses from 0x0000 to 0x003E for hardware interrupts. The hardware interrupt vectors are shown below.

Table 4-3. Hardware Interrupt Table

Interrupt Number	Vector Address	Interrupt Type
0	0x0000	Timer0[11]
1	0x0002	Timer1[12]
2	0x0004	GP IRQ0[12]
3	0x0006	GP IRQ1[12]
4–6	0x0008–0x000C	Reserved
7	0x000E	USB Reset
8	0x0010	USB SOF[13]
9	0x0012	USB Endpoint0 No Error[11]
10	0x0014	USB Endpoint0 Error[11]
11	0x0016	USB Endpoint1 No Error
12	0x0018	USB Endpoint1 Error
13	0x001A	USB Endpoint2 No Error
14	0x001C	USB Endpoint2 Error
15	0x001E	USB Endpoint3 No Error
16	0x0020	USB Endpoint3 Error
17–63	0x0022–0x007E	Reserved

All these vector interrupts are read/write accessible. You can overwrite these default software interrupt vectors by replacing your interrupt service subroutine.

The addresses from 0x0000 to 0x003E are read/write accessible and can be used for variables.

4.11.2 Interrupt Enable Register (0xC00E: R/W)

This is a global hardware interrupt enable register that allows control of the hardware interrupt vectors. The CS5954AM BIOS default set-up of this register is 0x28 (i.e., USB bits are set).

D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
0	0	0	0	0	0	0	0	0	0	USB	0	0	GP	T1	T0

D5 USB USB Interrupt Enable

D2 GP General-purpose I/O pins Interrupt enables (see GPIO Interrupt Control Register (0xC01C: R/W))

D1 T1 Timer1 Interrupt Enable

D0 T0 Timer0 Interrupt Enable

Notes:

11. These hardware interrupt vectors are reserved for internal CS5954AM-BIOS usage. You should not attempt to overwrite these functions.
12. These hardware interrupt vectors are initialized to return on the interrupt.
13. The SOF interrupt is generated when there is an incoming SOF on the USB.

4.11.3 GPIO Interrupt Control Register (0xC01C: R/W)

This register defines the polarity of the GPIO interrupt on IRQ1 (GPIO25) and IRQ0 (GPIO24). The **GPIO** bit on the Interrupt Enable Register must be set in order for this register to operate.[14]

D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
0	0	0	0	0	0	0	0	0	0	0	0	P1	E1	P0	E0

D3	P1	IRQ1 polarity is rising edge if “1,” falling edge if “0”
D2	E1	Enable IRQ1 if set to “1”
D1	P0	IRQ0 polarity is rising edge if “1,” falling edge if “0”
D0	E0	Enable IRQ0 if set to “1.”

4.11.4 Software Interrupts

The CS5954AM allocates addresses from 0x0040 to 0x00FE for software interrupts. The software interrupt vectors are shown in *Table 4-4*.

Table 4-4. Software Interrupt Table

Interrupt Number	Vector Address	Interrupt Type
64 (0x40)	0x0080	2-wire serial interface_INT[15]
65 (0x41)	0x0082	Reserved for future extension of other serial EEPROM
66 (0x42)	0x0084	Reserved
67 (0x43)	0x0086	SCAN_INT[15]
68 (0x44)	0x0088	ALLOC_INT[15]
69 (0x45)	0x008A	Data: start of free memory. Default = 0x200[16]
70 (0x46)	0x008C	IDLE_INT
71 (0x47)	0x008E	IDLER_INT
72 (0x48)	0x0090	INSERT_IDLE_INT
73 (0x49)	0x0092	PUSHALL_INT[15]
74 (0x4a)	0x0094	POPALL_INT[15]
75 (0x4b)	0x0096	FREE_INT[15]
76 (0x4c)	0x0098	REDO_ARENA[15]
77 (0x4d)	0x009A	HW_SWAP_REG[15]
78 (0x4e)	0x009C	HW_REST_REG[15]
79 (0x4f)	0x009E	SCAN_DECODE_INT
80 (0x50)	0x00A0	USB_SEND_INT[15]
81 (0x51)	0x00A2	USB_RECEIVE_INT[15]
82 (0x52)	0x00A4	Reserved
83 (0x53)	0x00A6	USB_STANDARD_INT
84 (0x54)	0x00A8	Data: Standard loader vector. Default = 0[16]
85 (0x55)	0x00AA	USB_VENDOR_INT
86 (0x56)	0x00AC	Data: USB_Vendor loader. Default = 0xff[16]
87 (0x57)	0x00AE	USB_CLASS_INT
88 (0x58)	0x00B0	Data: USB_Class_Loader. Default = 0[16]
89 (0x59)	0x00B2	USB_FINISH_INT
90 (0x5a)	0x00B4	Data: Device Descriptor. Default = Cypress device desc[16]

Notes:

14. The interrupts can be enabled for “Suspend mode” by the Power-down Register or enabled for interrupts by the Interrupt Enable Register.
15. These software vectors are reserved for the internal CS5954AM-BIOS. The user should not overwrite these functions.
16. These vectors are used as the data pointers. The user should not execute code (i.e. **JMP** or **INT**) to these vectors. See [Ref. 1] SL11R_BIOS for more information.

Table 4-4. Software Interrupt Table (continued)

Interrupt Number	Vector Address	Interrupt Type
91 (0x5b)	0x00B6	Data: configuration desc. Default = Cypress configuration[6]
92 (0x5c)	0x00B8	Data: string descriptor. Default = Cypress string desc.[6]
93 (0x5d)	0x00BA	USB_PARSE_CONFIG_INT
94 (0x5e)	0x00BC	USB_LOADER_INT
95 (0x5f)	0x00BE	USB_DELTA_CONFIG_INT
96 (0x60)	0x00C0	USB_PULLUP_INT
97–104	0xC2–0xD0	Reserved for future addition secondary USB port
105 (0x69)	0x00D2	POWER_DOWN_SUBROUTINE
106–109	0xD4–0xDA	Reserved for future secondary USB Port
110–124	0xDE–0xF8	User's ISR or internal peripheral interrupt
125–127	0xFA–0xFE	Reserved for the debugger

All these vector interrupts are Read/Write accessible. User can overwrite these default software interrupt vectors by replacing the user's interrupt service subroutine.

4.12 Serial EEPROM Interface (2-wire serial interface)

The CS5954AM provides an interface to an external serial EEPROM. The interface is implemented using General-purpose I/O signals. A variety of serial EEPROM formats can be supported: currently the BIOS ROM supports the two-wire serial EEPROM type. The serial EEPROM can be used to store specific peripheral USB configuration and add on value functions. It can also be used for field product upgrades

The CS5954 BIOS uses an interrupt to read and write to/from an external serial EEPROM. The recommended serial EEPROM device is a 2-wire serial CMOS EEPROM (AT24CXX device family). Currently, the CS5954AM BIOS Revision 1.1 allows reading/writing to/from EEPROM, up to 2K Bytes (16K bits), 2-wire serial interface device (i.e., AT24C16).

The user's program and USB vendor/device configuration can be programmed and stored into the external EEPROM device. On power-up the content of the EEPROM will be downloaded into RAM and may be executed as code or used as data, or both. The advantage of the 2-wire serial interface/EEPROM interface is the space and cost saving when compared to using an external eight-bit PROM/EPROM.

The CS5954AM BIOS uses two GPIO pins, GPIO31 and GPIO30 to interface to an external serial EEPROM (see *Figure 4-3*):

- **GPIO31 is connected to the Serial Clock Input (SCL).**
- **GPIO30 is connected to the Serial Data (SDA).**
- **We recommend you add a 5K to 15K pull-up resistor on the Data line (e.g., GPIO30).**
- **Pin 1 (A0), Pin 2 (A1), Pin 3 (A2), Pin 4 (GND) and Pin 7 Write Protect) are connected to Ground.**

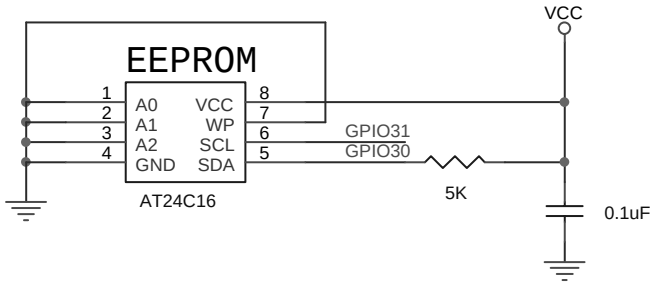


Figure 4-3. 2-Wire Serial Interface 2K-byte Connection

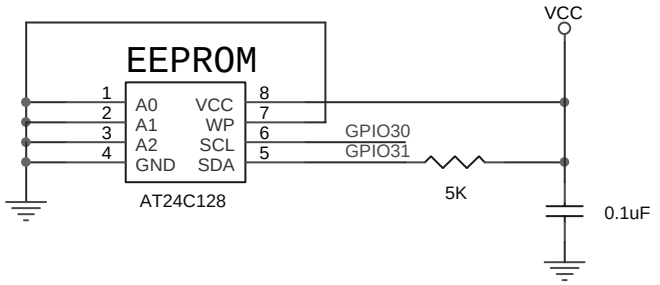


Figure 4-4. 2-Wire Serial Interface 16K Connection

The current CS5954AM BIOS only support up to a 2Kbyte serial EEPROM. To read and write to a device that is larger than 2Kbytes, the CS5954AM-BIOS requires additional serial EEPROM to be connected as shown in *Figure 4-4*.

In this example, the CS5954AM BIOS will first access the (small) program residing on **IC1** serial EEPROM, and then it will access the second **IC2 EEPROM** (see [Ref. 1] SL11R_BIOS for more information).

4.13 External SRAM

The CS5954AM has a multiplexed address port and 16-bit data port. These interface signals are provided to interface to an external SRAM. At boot up stage, the CS5954AM BIOS configures the CS5954AM for external SRAM and serial EEPROM. In addition, the external memory interface is set up as 16-bit and seven wait states for both external SRAM and EEPROM.

Example 2 CS5954AM extended memory set-up:[17]

```
internal_rom_start:
        mov     [0xC03A],0x0077     ;set 16-bit ROM and 7 wait
        cmp     [0xC100],0xCB36     ;check for special pattern in external ROM
        je      0xC102              ;if it's there, jump to it
        mov     [0xC006],0x10       ;2/3 clock
        mov     [0xC008],1          ;at 24 MHz
        mov     [0xC03E],3          ;extra wait state for ROM and Debug
        cmp     [0xC100],0xC3B6     ;external ROM has 0xC3B6 as first 16 bits
        je      xrom_ok
        cmp     b[0xC100],0xB6      ;check 0xc3b6 for 8-bit ROM
        jne     xrom_ok
        or      [0xC03A],0x80       ;set for 8-bit ROM
xrom_ok:
        mov     [0xC00],0xC3B6      ;check 0xC3B6 for 16-bit RAM
        cmp     [0xC00],0xC3B6
        je      xram_ok
        or      [0xC03A],8          ;set for 8-bit external RAM
xram_ok:
```

4.13.1 Memory Control Register (0xC03E: R/W)

This register provides control of Wait states for the internal RAM and ROM.

D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
0	0	0	0	0	0	0	0	0	0	0	0	0	RA	RO	DB

D2	RA	If “1,” one-wait state for internal RAM is added
D1	RO	If “1,” one-wait state for internal ROM is added
D0	DB	If “1,” DEBUG mode is enabled. Internal address bus is echoed to external address pins.

4.13.2 Extended Memory Control Register (0xC03A: R/W)

This register provides control of Wait states for the external SRAM/EPROM.[18]

D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
0	0	0	RM	EM3	EM2	EM1	EM0	RO3	RO2	RO1	RO0	RA3	RA2	RA1	RA0

D12	RM	ROM Merge. If “1,” nXROMSEL is active if nXMEMSEL is active.
D11	EM3	Extended Memory Width (“0” = 16, “1” = 8)
D10–8	EM2–0	Extended Memory Wait states (0–7)
D7	RO3	External ROM Width (“0” = 16, “1” = 8)
D6–4	RO2–0	External ROM wait states (0–7)
D3	RA3	External RAM Width (“0” = 16, “1” = 8)
D2–0	RA2–0	External RAM Wait States (0–7)

Notes:

17. The external memory devices can be 8 or 16 bits wide, and can be programmed to have up to seven wait-states. External SRAM/PROM requires one wait state.
18. The default Wait State setting on power-up or reset is seven wait states.

4.13.3 Extended Page 1 Map Register (0xC018: R/W)

This register contains the Page 1 high order address bits. These bits are always appended to accesses to the Page 1 Memory mapped space. The default is 0x0000.

D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
0	0	0	0	0	0	0	0	A20	A19	A18	A17	A16	A15	A14	A13

D7–o	A20–13	Page 1 high order address bits. The address pins on A21–A13 will reflect the content of this register when CS5954AM accesses the address 0x8000–0x9FFF.

4.13.4 Extended Page 2 Map Register (0xC01A: R/W)

This register contains the Page 2 high order address bits. These bits are always appended to accesses to the Page 2 Memory mapped space. The default is 0x0000.

D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
0	0	0	0	0	0	0	0	A20	A19	A18	A17	A16	A15	A14	A13

D7–o	A20–13	Page 2 high order address bits. The address pins on A21–A13 will reflect the content of this register when CS5954AM access the address 0xA000–0xBFFF.

4.13.5 Memory Map

The total memory space allocated by the CS5954AM is 64K-bytes. Program, data, and I/O space are contained within a 64K-byte address space. The program code or data can be stored in either external RAM or external ROM.

The CS5954AM Controller provides a 16-bit Memory interface that can support a wide variety of external, RAM and ROM devices. The CS5954AM Controller memory space is byte addressable and is divided as shown in *Table 4-5*.

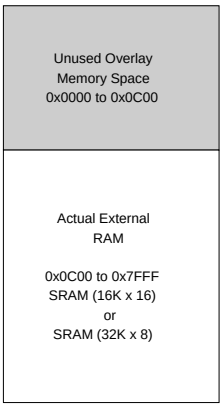
Table 4-5. Memory Map

Function	Address
Internal RAM	0x0000–0x0BFF
External RAM	0x0C00–0x7FFF[19]
Memory Mapped Registers	0xC000–0xC0FF
External ROM	0xC100–0xE7FF[20]
Internal ROM	0xE800–0xFFFF

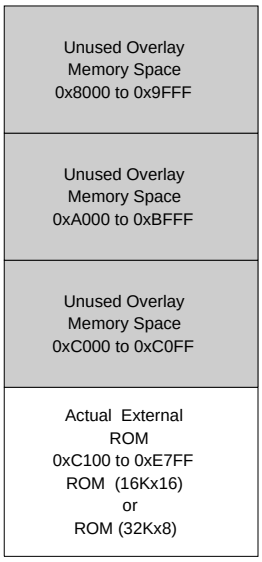
Each External memory space can be 8 or 16 bits wide, and can be programmed to have up to seven wait-states.

Notes:

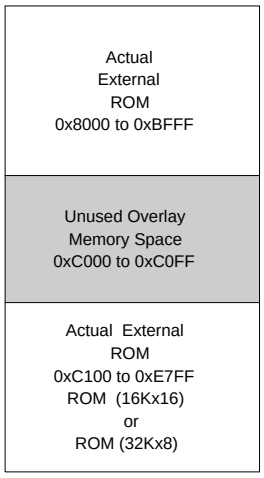
19. The External RAM address from 0x0000 to 0x0C00 will not be accessible from the CS5954AM processor. This is an overlay memory space between internal RAM and external RAM. The addressable external RAM will occupy from 0x0C00–0x7FFF, which is 29 Kbyte. The signal name nXRAMSEL on CS5954AM–pin56 will be active when the CPU access address from 0x0C00–0x7FFF.
20. When bit 12 (ROM Merge Bit) of the Extended Memory Controller Register at address 0xC03A is “0,” then the External ROM address space will be mapped from 0xC100–0xE7FF. The address from 0x8000–0xC100 and the address from 0xE800 to 0xFFFF are the overlay memory spaces. The actual total size of the external ROM will be (0xE800–0xC100), which is 9.75 Kbyte. The signal nXROMSEL on the CS5954AM (pin 57) will be active when the CPU accesses the address from 0xC100–0xE7FF. The signal nXMEMSEL on the CS5954AM (pin 58) will be active when the CPU accesses the address from 0x8000–0xBFFF. When bit 12 (ROM Merge Bit) of the Extended Memory Controller Register at address 0xC03A is “1,” then the External ROM address space will be mapped into these windows: 0x8000–0xBFFF and 0xC100–0xE7FF. The address from 0xC000 to 0xC100 and the address from 0xE800–0xFFFF are the overlay memory spaces. The actual total size of the external ROM will be (0xC000–0x8000) and (0xE800–0xC100), which is 16 Kbyte + 9.75 Kbytes, or 25.75K.



Bit 12 (ROM Merge) of the Extended Memory Controller Register = 0



Bit 12 (ROM Merge) of the Extended Memory Controller Register = 1



4.14 General Timers and Watchdog Timer

The CS5954AM Controller has two built-in programmable timers that can provide an interrupt to the CS5954AM engine. The timers decrement on every microsecond clock tick. An interrupt occurs when the timer reaches zero.

4.14.1 Timer 0 Count Register (0xC010: R/W)

The CS5954AM BIOS uses the timer 0 for time-out function and power-down mode. At the end of the power-up, the CS5954AM BIOS disables the timer 0 interrupt. If you wish to use timer 0 for power-down function, see the [Ref. 1] SL11R_BIOS for more information.

D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
T15	T14	T13	T12	T11	T10	T9	T8	T7	T6	T5	T4	T3	T2	T1	T0

D15–0 T15–0 Timer Count value.

4.14.2 Timer 1 Count Register (0xC012: R/W)

The CS5954AM timer 1 is for user applications. The CS5954AM BIOS does not use this timer.

D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
T15	T14	T13	T12	T11	T10	T9	T8	T7	T6	T5	T4	T3	T2	T1	T0

D15–0 T15–0 Timer Count value.

4.14.3 Watchdog Timer Count and Control Register (0xC00C: R/W)

The CS5954AM provides a Watchdog timer to monitor certain activities. The Watchdog timer can also interrupt the CS5954AM processor. The default value of this register is 0x0000.[21, 22, 23]

D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
0	0	0	0	0	0	0	0	0	0	WT	TO1	TO0	ENB	EP	RC

D5	WT	Watchdog Time-out occurred.		
D4–3	TO1–0	Time-out Count:	00	01 milliseconds
			01	04 milliseconds
			10	16 milliseconds
			11	64 milliseconds
D2	EP	Enable Permanent WD timer. If set =”1”		
		WD timer is always enabled. Cleared only on Reset.		
D1	ENB	Enable WD Timer operation when =”1.”		
D0	RC	Reset Count. When set =”1.”		

4.15 Special GPIO Function for Suspend, Resume and Low-power Modes

The CS5954AM CPU supports suspend, resume, and CPU low-power modes. The CS5954AM BIOS assigns GPIO29 for the USB DATA+ line pull-up (this pin can simulate USB cable removal or insertion while USB power is still applied to the circuit) and the GPIO20 for controlling power-off function. The GPIO20 can be used for device low-power mode: it will remove power from the peripherals in suspend mode. Once USB power is restored, the power to the peripherals may be enabled. The CS5954AM BIOS will execute the pull-up interrupt upon power-up. To use this feature, the GPIO29 pin must be connected to the DATA+ line of the USB connector (see *Figure 4-5* below). For more information about this function, see [Ref. 1] SL11R_BIOS.

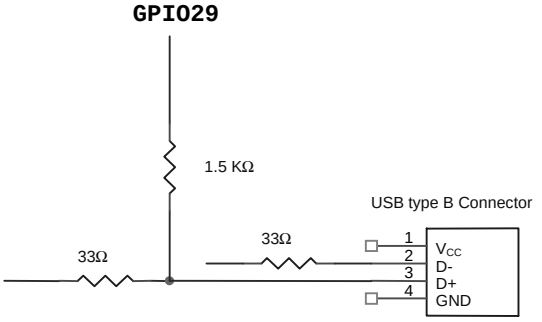


Figure 4-5. Special GPIO Pull-up Connection Example

5.0 CS5954AM Interface Modes

The CS5954AM has a general-purpose I/O mode.[24]

5.1 General-purpose I/O Mode (GPIO)

In GPIO mode, the CS5954AM has up to 32 general-purpose I/O signals available. However, there are four pins used by the 2-wire serial interface that cannot be used as GPIO pins. The CS5954AM executes at 48MHz. Other available general-purpose I/O pins can be programmed for peripheral control and/or status, etc.

The following registers are used for all pins configured as GPIO. The outputs are enabled in the I/0 Control registers. Note that the output Data can be read back via the Output Data Register even if the outputs are not enabled.

Notes:

21. In order to avoid Watchdog trigger, Reset Count (RC) must be asserted before time-out occurs.
22. The Watchdog Timer overflow causes an internal processor reset. The Processor can read the WT bit after exiting reset to determine if the WT bit is set. If it is set, a Watchdog time-out occurred.
23. The WT value will be cleared on the next external reset.
24. The 2-wire serial interface IO pins are fixed in all CS5954AM Interface modes.

5.1.1 I/O Control Register 0 (0xC022: R/W)

This register controls the I/O direction of the GPIO data pins from GPIO15 to GPIO0. When any bit of this register set to one, the corresponding GPIO data pin becomes an output pin. When any bit of this register is set to zero, the corresponding GPIO data pin becomes an input pin.

D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
E15	E14	E13	E12	E11	E10	E9	E8	E7	E6	E5	E4	E3	E2	E1	E0

D15–0 E15–0 Enable individual outputs, GPIO 15–0. Logic “1” enables.

5.1.2 I/O Control Register 1 (0xC028: R/W)

This register controls the I/O direction of the GPIO data pins from GPIO31 to GPIO16. When any bit of this register set to one, the corresponding GPIO data pin becomes an output pin. When any bit of this register is set to zero, the corresponding GPIO data pin becomes an input pin.

D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
E31	E30	E29	E28	E27	E26	E25	E24	E23	E22	E21	E20	E19	E18	E17	E16

D15–0 E31–16 Enable individual outputs, GPIO 31–16. Logic “1” enables.

5.1.3 Output Data Register 0 (0xC01E: R/W)

This register controls the output data of the GPIO data pins from GPIO15 to GPIO0.[25]

D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
O15	O14	O13	O12	O11	O10	O9	O8	O7	O6	O5	O4	O3	O2	O1	O0

D15–0 O15–0 Output Pin data

5.1.4 Output Data Register 1 (0xC024: R/W)

This register controls the output data of the GPIO data pins from GPIO31 to GPIO16.[25]

D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
O31 15	O30 14	O29 13	O28 12	O27 11	O26 10	O25 9	O24 8	O23 7	O22 6	O21 5	O20 4	O19 3	O18 2	O1 7	O16 0

D15–0 O31–16 Output Pin data

5.1.5 Input Data Register 0 (0xC020: Read only)

This register reads the input data of the GPIO data pins from GPIO15 to GPIO0.

D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
I15	I14	I13	I12	I11	I10	I9	I8	I7	I6	I5	I4	I3	I2	I1	I0

D15–0 I15–0 Input Pin data

5.1.6 Input Data Register 1 (0xC026: Read-only)

This register reads the input data of the GPIO data pins from GPIO31 to GPIO16.

D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
I31	I30	I29	I28	I27	I26	I25	I24	I23	I22	I21	I20	I19	I18	I17	I16

D15–0 I31–16 Input Pin data

Note:

25. A Read of this register reads back the last data written, not the data on pins configured as input (see below).

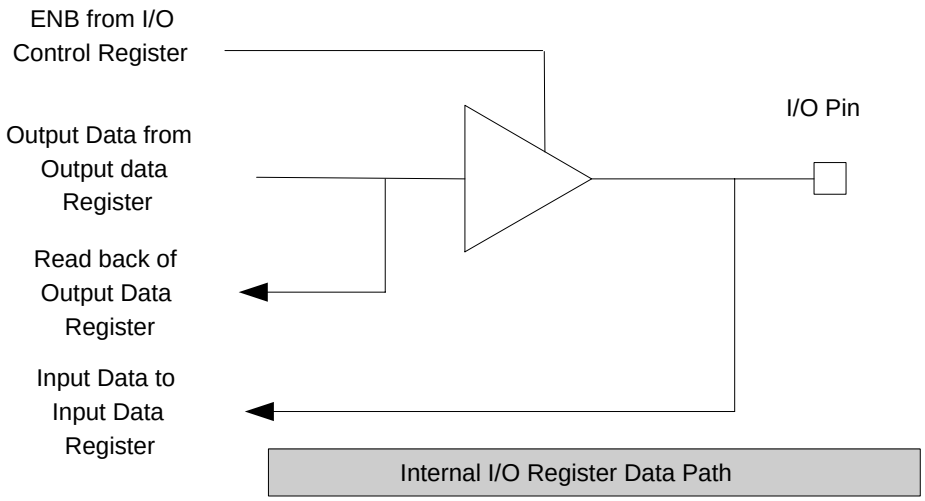


Figure 5-1. GPIO Mode Block Diagram

5.1.7 I/O Address Map

Table 5-1. I/O Address Map

Function	Address	Mode
USB Endpoint 0 Address Register	0x0120	R/W
USB Endpoint 0 Count Register	0x0122	R/W
USB Endpoint 1 Address Register	0x0124	R/W
USB Endpoint 1 Count Register	0x0126	R/W
USB Endpoint 2 Address Register	0x0128	R/W
USB Endpoint 2 Count Register	0x012A	R/W
USB Endpoint 3 Address Register	0x012C	R/W
USB Endpoint 3 Count Register	0x012E	R/W
Configuration Register	0xC006	R/W
Speed Control Register	0xC008	R/W
Power-down Control Register	0xC00A	R/W
Watchdog Timer Count and Control Register	0xC00C	R/W
Interrupt Enable Register	0xC00E	R/W
Timer 0 Count Register	0xC010	R/W
Timer 1 Count Register	0xC012	R/W
Breakpoint Register	0xC014	R/W
Extended Page 1 Map Register	0xC018	R/W
Extended Page 2 Map Register	0xC01A	R/W
GPIO Interrupt Control Register	0xC01C	R/W
Output Data Register 0	0xC01E	R/W
Input Data Register 0	0xC020	Read Only
I/O Control Register 0	0xC022	R/W
Output Data Register 1	0xC024	R/W
Input Data Register 1	0xC026	Read Only
I/O Control Register 1	0xC028	R/W
Extended Memory Control Register	0xC03A	R/W

Table 5-1. I/O Address Map (continued)

Function	Address	Mode
Memory Control Register	0xC03E	R/W
Serial Interface Control and Status Register	0xC068	R/W
Serial Interface Address Register	0xC06A	Write Only
Serial Interface Data Write Register	0xC06C	Write Only
Serial Interface Data Read Register	0xC06C	Read Only
USB Global Control and Status Register	0xC080	R/W
USB Frame Number Register	0xC082	Read Only
USB Address Register	0xC084	R/W
USB Command Done Register	0xC086	Write Only
USB Endpoint 0 Control and Status Register	0xC090	R/W
USB Endpoint 1 Control and Status Register	0xC092	R/W
USB Endpoint 2 Control and Status Register	0xC094	R/W
USB Endpoint 3 Control and Status Register	0xC096	R/W
STATUS Register	0xC0C2	Read Only

6.0 Pin Assignments

6.1 Pin Diagram

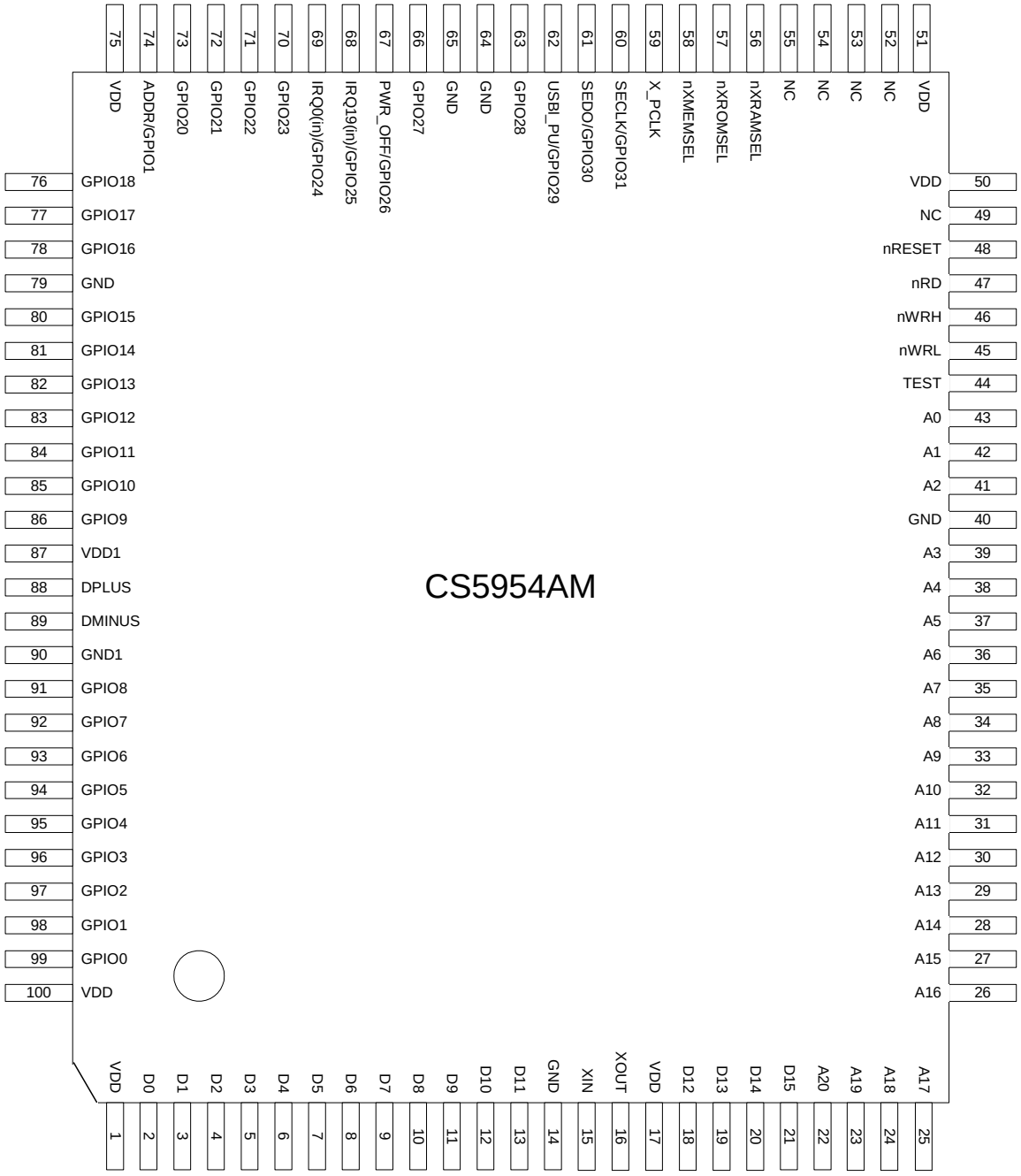


Figure 6-1. 100-pin PQFP

7.0 Physical Connection

7.1 Package Type

100 PQFP.

7.2 Pin Assignment and Description

Table 7-1. Pin Assignment and Description

Pin No.	Pin Name	Pin Type	Description
1	VDD	Power	+3.3 VDC Supply.
2	D0	I/O	External Memory Data Bus, Data0.
3	D1	I/O	External Memory Data Bus, Data1.
4	D2	I/O	External Memory Data Bus, Data2.
5	D3	I/O	External Memory Data Bus, Data3.
6	D4	I/O	External Memory Data Bus, Data4.
7	D5	I/O	External Memory Data Bus, Data5.
8	D6	I/O	External Memory Data Bus, Data6.
9	D7	I/O	External Memory Data Bus, Data7.
10	D8	I/O	External Memory Data Bus, Data8.
11	D9	I/O	External Memory Data Bus, Data9.
12	D10	I/O	External Memory Data Bus, Data10.
13	D11	I/O	External Memory Data Bus, Data11.
14	GND	GND	Digital ground.
15	XIN	Input	External 48-MHz Crystal or Clock Input.
16	XOUT	Output	External crystal output. No connection when XIN is used for clock input.
17	VDD	Power	+3.3 VDC Supply.
18	D12	I/O	External Memory Data Bus, Data12.
19	D13	I/O	External Memory Data Bus, Data13.
20	D14	I/O	External Memory Data Bus, Data14.
21	D15	I/O	External Memory Data Bus, Data15.
22	A20	Output	External Memory Address Bus, A20.
23	A19	Output	External Memory Address Bus, A19.
24	A18	Output	External Memory Address Bus, A18.
25	A17	Output	External Memory Address Bus, A17.
26	A16	Output	External Memory Address Bus, A16.
27	A15	Output	External Memory Address Bus, A15.
28	A14	Output	External Memory Address Bus, A14.
29	A13	Output	External Memory Address Bus, A13.
30	A12	Output	External Memory Address Bus, A12.
31	A11	Output	External Memory Address Bus, A11.
32	A10	Output	External Memory Address Bus, A10.
33	A9	Output	External Memory Address Bus, A9.
34	A8	Output	External Memory Address Bus, A8.
35	A7	Output	External Memory Address Bus, A7.
36	A6	Output	External Memory Address Bus, A6.
37	A5	Output	External Memory Address Bus, A5.
38	A4	Output	External Memory Address Bus, A4.
39	A3	Output	External Memory Address Bus, A3.
40	GND	GND	Digital ground.
41	A2	Output	External Memory Address Bus, A2.

Table 7-1. Pin Assignment and Description (continued)

Pin No.	Pin Name	Pin Type	Description
42	A1	Output	External Memory Address Bus, A1.
43	A0	Output	External Memory Address Bus, A0.
44	TEST	Input	No Connection, MFG test only Note: CS5954AM NC = 48 MHz, GND = 12 MHz.
45	nWRL	Output	Active LOW, Write to lower bank of External SRAM.
46	nWRH	Output	Active LOW, Write to upper bank of External SRAM.
47	nRD	Output	Active LOW, Read from External SRAM or ROM.
48	nRESET	Input	Master Reset. CS5954AM Device active LOW reset input.
49	NC	Output	Reserved.
50	VDD	Power	+3.3 VDC Supply.
51	VDD	Power	+3.3 VDC Supply.
52-55	NC	Output	Reserved.
56	nXRAMSEL	Output	Active LOW, select external SRAM (16 bit).
57	nXROMSEL	Output	Active LOW, select external ROM.
58	nXMEMSEL	Output	Active LOW, select external Memory bus, external SRAM, DRAM, ROM or any memory mapped device.
59	X_PCLK	I/O	See register 0xC006 for more information.
60	SECLK or GPIO31	I/O	SECLK, Serial EEPROM clock, or GPIO31.
61	SEDO or GPIO30	I/O	SEDO, Serial flash EPROM Data, or GPIO30 This pin requires a 5-kΩ pull-up.
62	USB_PU or GPIO29	I/O	Turn on/off D+ Pull-up Resistor, or GPIO29.
63	GPIO38	Output	
64	GND	GND	Digital ground.
65	GND	GND	Digital ground.
66	GPIO27	Input	
67	PWR_OFF or GPIO26	I/O	This signal can be used for device low-power mode: it will turn off or disable external powers to the peripheral in suspend mode. Once USB power is resumed, external power can be enabled again.
68	IRQ1 (in) or GPIO25	I/O	GPIO25, or IRQ1 (in) interrupts the CS5954AM processor.
69	IRQ0 (in) or GPIO24	I/O	IRQ0 (in) interrupts the CS5954AM processor.
70	GPIO23	I/O	GPIO23
71	GPIO22	I/O	GPIO22
72	GPIO21	I/O	GPIO21
73	GPIO20	I/O	GPIO20
74	ADDR or GPIO19	I/O	ADDR = 1, Read/Write data from the INBUF/OUTBUFF, ADDR = 0 read data from the STATUS register, or GPIO19.
75	VDD	Power	+3.3 VDC Supply.
76	GPIO18	I/O	GPIO18
77	GPIO17	I/O	GPIO17
78	GPIO16	I/O	GPIO16
79	GND	GND	Digital ground.
80	GPIO15	I/O	GPIO15
81	GPIO14	I/O	GPIO14
82	GPIO13	I/O	GPIO13
83	GPIO12	I/O	GPIO12
84	GPIO11	I/O	GPIO11

Table 7-1. Pin Assignment and Description (continued)

Pin No.	Pin Name	Pin Type	Description
85	GPIO10	I/O	GPIO10
86	GPIO9	I/O	GPIO9
87	VDD1	Power	USB +3.3 VDC Supply.
88	DPLUS	I/O	USB Differential DATA Signal High Side.
89	DMINUS	I/O	USB Differential DATA Signal Low Side.
90	GND1	GND	USB Digital Ground.
91	GPIO8	I/O	GPIO8
92	GPIO7	I/O	GPIO7
93	GPIO6	I/O	GPIO6
94	GPIO5	I/O	GPIO5
95	GPIO4	I/O	GPIO4
96	GPIO3	I/O	GPIO3
97	GPIO2	I/O	GPIO2
98	GPIO1	I/O	GPIO1
99	GPIO0	I/O	GPIO0
100	VDD	Power	+3.3 VDC Supply.

8.0 CS5954AM CPU Programming Guide

This is the specification for the CS5954AM Processor Instruction set.

8.1 Instruction Set Overview

This document describes the CS5954AM CPU Instruction Set, Registers and Addressing modes, Instruction format, etc. The CS5954AM PROCESSOR uses a unified program and data memory space; although this RAM is also integrated into the CS5954AM core, provision has been made for external memory as well.

The CS5954AM PROCESSOR engine incorporates 38 registers: fifteen general-purpose registers, a stack pointer, sixteen registers mapped into RAM, a program counter, and a REGBANK register whose function will be described in a subsequent section.

The CS5954AM PROCESSOR engine supports byte and word addressing. Subsequent sections of this document will describe:

- **The CS5954AM PROCESSOR Engine (QT Engine) Register Set**
- **CS5954AM PROCESSOR Engine Instruction Format**
- **CS5954AM PROCESSOR Engine Addressing Modes**
- **CS5954AM PROCESSOR Engine Instruction Set.**

8.2 Reset Vector

On receiving hardware reset, the CS5954AM Processor jumps to address 0xFFF0, which is an internal ROM address.

8.3 Register Set

The CS5954AM Processor incorporates 16-bit general-purpose registers called R0..R15, a REGBANK register, and a program counter, along with various other registers. The function of each register is defined as follows.

Name	Function
R0-R14	General-purpose Registers
R15	Stack Pointer
PC	Program Counter
REGBANK	Forms base address for registers R0-R15
FLAGS	Contains flags: defined below
INTERRUPT ENABLE	Bit masks to enable/disable various interrupts

8.4 General-purpose Registers

The general-purpose registers can be used to store intermediate results, and to pass parameters to and return them from subroutine calls.

8.5 General-purpose/Address Registers

In addition to acting as general-purpose registers, registers R8-R14 can also serve as pointer registers. Instructions can access RAM locations by referring to any of these registers. In normal operation, register R15 is reserved for use as a stack pointer.

8.6 REGBANK Register (0xC002: R/W)

Registers R0..R15 are mapped into RAM via the REGBANK register. The REGBANK register is loaded with a base address, of which the 11 Most Significant Bits (MSBs) are used. A Read from or Write to one of the registers will generate a RAM address by:

- **Shifting the four Least Significant Bits (LSBs) of the register number left by 1.**
- **OR-ing the shifted bits of the register number with the upper 11 bits of the REGBANK register.**
- **Forcing the LSB to 0.**

For example, if the REGBANK register is left at its default value of 100 hex, a read of register R14 would read address 11C hex.[26]

Register	**Hex Value**	**Binary Value**															
REGBANK	0100	0	0	0	0	0	0	0	1	0	0	0	x	x	x	x	x
R14	000E << 1 = 001C	x	x	x	x	x	x	x	x	x	x	0	1	1	1	0	0
RAM Location	011C	0	0	0	0	0	0	0	1	0	0	0	1	1	1	0	0

8.7 Flags Register (0xC000: Read-only)

The CS5954AM Processor uses these flags.[27]

FLAG																
bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	0	0	0	0	0	0	0	0	0	0	0	I	S	O	C	Z

Z **Zero**: instruction execution resulted in a result of 0.

C **Carry/Borrow**: Arithmetic instruction resulted in a carry (for addition) or a borrow (for subtraction).

O **Overflow**: Arithmetic result was either larger than the destination operand size (for addition) or smaller than the destination operand should allow for subtraction.

S **Sign**: Set if MS result bit is “1.”

I **Global Interrupts**: Enabled if “1.”

8.8 Instruction Format

To understand addressing modes supported by the CS5954AM Processor, one must know how the instruction format is defined. In general, the instructions include four bits for the instruction ***opcode***, six bits for the source operand, and six bits for the destination operand.

Instr																
bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	opcode				source						destination					

Some instructions, especially single operand-operator and program control instructions, will not adhere strictly to this format. They will be discussed in detail in turn.

Notes:

26. Regardless of the value loaded into the REGBANK register, bits 0..4 will be ignored.
27. Flag behavior for each instruction will be described in the following section.

8.9 Addressing Modes

This section describes in detail the six-operand field bits referred to in the previous section as ***source*** and ***destination***. Bear in mind that although the discussion refers to bits 0 through 5, the same bit definitions apply to the "source" operand field, bits 6 through 11. These are the basic addressing modes in the CS5954AM processor.

Mode	5	4	3	2	1	0
Register	0	0	r	r	r	r
Immediate	0	1	1	1	1	1
Direct	1	0	b/w[28]	1	1	1
Indirect	0	1	b/w[28]	r	r	r
Indirect with Auto Increment[29]	1	0	b/w[28]	r	r	r
Indirect with Index	1	1	b/w[28]	r	r	r

8.10 Register Addressing

In register addressing, any one of registers R0–R15 can be selected using bits 0–3. If register addressing is used, operands are always 16-bit operands, since all registers are 16-bit registers. For example, an instruction using register R7 as an operand would fill the operand field as follows.

Bits	5	4	3	2	1	0
Register Operand	0	0	0	1	1	1

8.11 Immediate Addressing

In Immediate Addressing, the instruction word is immediately followed by the source operand. For example, the operand field would be filled as follows.[30]

Bits	5	4	3	2	1	0
Operand field	0	1	1	1	1	1

8.12 Direct Addressing

In Direct Addressing, the word following the instruction word is used as an address into RAM. Again, the operand can be either byte or word sized, depending on the state of bit 3 of the operand field. For example, to do a word-wide read from a direct address, the ***source*** operand field would be formed as follows.[31]

Bits	5	4	3	2	1	0
I/O operand	1	0	0	1	1	1

8.13 Indirect Addressing

Indirect addressing is accomplished using address registers R8–15. In Indirect addressing, the operand is found at the memory address pointed to by the register. Since only eight address registers exist, only three bits are required to select an address register. For example, register R10 (binary 1010) can be selected by ignoring bit 3, leaving the bits 010. Bit 3 of the operand field is then used as the byte/word bit, set to "0" to select word or "1" to select byte addressing. In this example, a byte-wide operand is selected at the memory location pointed to by register R10.[32]

Bits	5	4	3	2	1	0
Memory operand	0	1	1	0	1	0

Notes:

28. b/w: "1" for byte-wide access, "0" for word access.
29. Indirect with auto-increment and byte-wide Indirect addressing is illegal with the stack pointer (R15).
30. In immediate addressing, the source operand *must* be 16 bits wide, eliminating the need for a b/w bit.
31. For a memory-to-memory move, the instruction word would be followed by two words, the first being the ***source*** address and the second being the ***destination***.
32. For register R15, byte-wide operands are prohibited. If bit 3 is set high, the instruction is decoded differently, as explained at the top of this section.

8.14 Indirect Addressing with Auto Increment

Indirect Addressing with Auto Increment works identically to Indirect Addressing, except that at the end of the read or write cycle, the register is incremented by 1 or 2 (depending whether it is a byte-wide or word-wide access).

This mode is prohibited for register R15. If bits 0..2 are all high, the instruction is decoded differently, as explained at the top of this section.

8.15 Indirect Addressing with Offset

In Indirect Addressing with Offset, the instruction word is followed by a 16-bit word that is added to the contents of the address register to form the address for the operand. The offset is an unsigned 16-bit word, and will “wrap” to low memory addresses if the register and offset add up to a value greater than the size of the processor’s address space.

8.16 Stack Pointer (R15) Special Handling

Register R15 is designated as the Stack Pointer, and has these special behaviors:

- **If addressed in indirect mode, the register pre-decrements on a write instruction, and post-increments on a read instruction, emulating Push and Pop instructions.**
- **Byte-wide reads or writes are prohibited in indirect mode.**
- **If R15 is addressed in Indirect with Index mode, it does not auto-increment or auto-decrement.**

CS5954AM–CPU Instruction Set

The instruction set can be roughly divided into three classes of instructions:

- **Dual Operand Instructions (Instructions with two operands: a source and a destination)**
- **Program Control Instructions (Jump, Call, and Return)**
- **Single Operand Instructions (Instructions with only one operand: a destination).**

8.17 Dual Operand Instructions

Instructions with source and destination for ALL dual operand instructions–byte values are zero extended by default.

MOV																
bit:	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
	0000				source						Destination					

destination:= source

Flags Affected: none

ADD																
bit:	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
	0001				source						Destination					

destination:= destination + source

Flags Affected: Z, C, O, S

ADDC																
bit:	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
	0010				source						Destination					

destination:= destination + source + carry bit

Flags Affected: Z, C, O, S

SUB																
bit:	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
	0011				source						Destination					

destination:= destination – source

Flags Affected: Z, C, O, S

SUBB																
bit:	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
	0100				source						Destination					

destination:= destination – source – carry bit

Flags Affected: Z, C, O, S

CMP																
bit:	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
	0101				source						Destination					

[not saved] = destination – source

Flags Affected: Z, C, O, S

AND																
bit:	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
	0110				source						Destination					

destination:= destination and source

Flags Affected: Z, S

TEST																
bit:	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
	0111				source						Destination					

[not saved]:= destination and source

Flags Affected: Z, S

OR																
bit:	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
	1000				source						Destination					

destination:= destination | source

Flags Affected: Z, S

XOR																
bit:	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
	1001				source						Destination					

destination:= destination ^ source

Flags Affected: Z, S

8.18 Program Control Instructions

Jcc JUMP RELATIVE cccc																
bit:	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
	1100				cccc				0	Offset						

PC:= PC + (offset*2) (offset is a 7-bit *signed* number from –64..+63)

JccL JUMP ABSOLUTE cccc																
bit:	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
	1100				cccc				1	0	Destination					

PC:= [destination] (destination is computed in the normal fashion for operand fields)

Rcc RET cccc																
bit:	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
	1100				cccc				1	0	010111					

PC:= [R15]

R15++

Ccc CALL cccc																
bit:	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
	1010				cccc				1	0	Destination					

R15—

[R15]:= PC

PC = [destination]

INT																
bit:	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
	1010				0000				0	int vector						

[R15]:= PC

R15—

PC = [int vector * 2]

This instruction allows the programmer to implement software interrupts. *Int vector* is multiplied by two, and zero extended to 16 bits.[33]

The condition (cccc) bits for all of the above instructions are defined as follows.

Condition	cccc Bits	Description	JUMP Mnemonic[34]	CALL Mnemonic	RET Mnemonic
Z	0000	Z=1	JZ	CZ	RZ
NZ	0001	Z=0	JNZ	CNZ	RNZ
C / B	0010	C=1	JC	CC	RC
NC / AE	0011	C=0	JNC	RNC	RNC
S	0100	S=1	JS	CS	RS
NS	0101	S=0	JNS	CNS	RNS
O	0110	O=1	JO	CO	RO
NO	0111	O=0	JNO	CNO	RNO
A / NBE	1000	(Z=0 AND C=0)	JA	CA	RA
BE / NA	1001	(Z=1 OR C=1)	JBE	CBE	RBE
G / NLE	1010	(O= S AND Z=0)	JG	CG	RG
GE / NL	1011	(O=S)	JGE	CGE	RGE
L / NGE	1100	(O≠S)	JL	CL	RL
LE / NG	1101	(O≠S OR Z=1)	JLE	CLE	RLE
(not used)	1110				
Unconditional	1111	Unconditional	JMP	CALL	RET

8.19 Single Operand Operation Instructions

Since Single operand instructions do not require a source field, the format of the Single operand Operation instructions is slightly different.[35, 36]

Instruction																	
bit:	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
	1101***							[param]			destination					

Notice that the ***opcode*** field is expanded to seven bits wide. The four MSBs for all instructions of this class are “1101.”

Notes:

33. Interrupt vectors 0 through 31 *may* be reserved for hardware interrupts, depending on the application.
34. For the JUMP mnemonics, adding an “L” to the end indicates a long or absolute jump. Adding an “S” to the end indicates a short or relative jump. If nothing is added, the assembler will choose “S” or “L.”
35. For the SHR, SHL, ROR, ROL, ADDI and SUBI instructions, the three-bit *count* or *n* operand is incremented by 1 before it is used.
36. The CS5954AM QT assembler takes this into account.

In addition, there is space for an optional three bit immediate value, which is used in a manner appropriate to the instruction. The destination field functions exactly as it does in the dual operand operation instructions.

SHR																
bit:	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
	1101000							count-1			destination					

destination:= destination >> count

Flags Affected: Z, C, S

SHL																
bit:	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
	1101001							count-1			destination					

destination:= destination << count

Flags Affected: Z, C, S[37, 38]

ROR																
bit:	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
	1101010							count-1			destination					

Works identically to the SHR instruction, except that the LSB of *destination* is rotated into the MSB, as opposed to SHR, which discards that bit

Flags Affected: Z, C, S[39]

ROL																
bit:	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
	1101011							count-1			destination					

Works identically to the SHL instruction, except that the MSB of *destination* is rotated into the LSB, as opposed to SHL, which discards that bit

Flags Affected: Z, C, S

ADDI																
bit:	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
	1101100							n-1			destination					

destination:= destination + n

Flags Affected: Z, S

SUBI																
bit:	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
	1101101							n-1			destination					

destination:= destination – n

Flags Affected: Z, S

NOT																
bit:	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
	1101111							000			destination					

destination:= ~destination (bitwise 1's complement negation)

Flags Affected: Z, S

NEG																
bit:	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
	1101111							001			destination					

Notes:
37. The SHR instruction shifts in sign bits.
38. The C flag is set with last bit shifted out of LSB.
39. The C flag is set with last bit shifted out of MSB.

destination:= -destination (2's complement negation)

Flags Affected: Z, O, C, S

CBW																
bit:	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
	1101111							010			destination					

Sign-extends a byte in the lower eight bits of [destination] to a 16-bit signed word (integer).

Flags Affected: Z, S

8.20 Miscellaneous Instructions

STI																
bit:	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
	1101111							111			000000					

Sets interrupt enable flag[40]

Flags Affected: I

CLI																
bit:	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
	1101111							111			000001					

Clears interrupt enable flag

Flags Affected: I

STC																
bit:	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
	1101111							111			000010					

Set Carry bit.

Flags Affected: C

CLC																
bit:	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
	1101111							111			000011					

Clear Carry bit.

Flags Affected: C

8.21 Built-in Macros

For the programmer's convenience, the CS5954AM QT assembler implements several built-in macros. The table below shows the macros, and the mnemonics for the code that the assembler will generate for these macros.

Macro	Assembler will generate
INC X	ADDI X, 1
DEC X	SUBI X, 1
PUSH X	MOV [R15], X
POP X	MOV X, [R15]

Note:

40. The STI instruction takes effect 1 cycle after it is executed.

8.22 CS5954AM Processor Instruction Set Summary

Mnemonic	Operands	Description	Opcode MSb LSb	Flags Affected	Clock Cycles	Notes
MOV	s,d	Move s to d	0000 ssss ssdd dddd	None	5	41, 42
ADD	s,d	Add s to d	0001 ssss ssdd dddd	Z,C,O,S	5	41, 42
ADDC	s,d	Add s to d with carry	0010 ssss ssdd dddd	Z,C,O,S	5	41, 42
SUB	s,d	Subtract s from d	0011 ssss ssdd dddd	Z,C,O,S	5	41, 42
SUBB	s,d	Subtract s from d with carry	0100 ssss ssdd dddd	Z,C,O,S	5	41, 42
CMP	s,d	Compare d with s	0101 ssss ssdd dddd	Z,C,O,S	5	41, 42
AND	s,d	AND d with s	0110 ssss ssdd dddd	Z,S	5	41, 42
TEST	s,d	Bit test d with s	0111 ssss ssdd dddd	Z,S	5	41, 42
OR	s,d	OR d with s	1000 ssss ssdd dddd	Z,S	5	41, 42
XOR	s,d	XOR d with s	1001 ssss ssdd dddd	Z,S	5	41, 42
Jcc	c,v	Jump relative on condition ’c’	1100 cccc 0ooo oooo	None	3	42
JccL	c,d	Jump absolute on condition ’c’	1100 cccc 10dd dddd	None	4	42
Rcc	c	Return on condition ’c’	1100 cccc 1001 0111	None	7	42
Ccc	c,d	Call subroutine on condition ’c’	1010 cccc 10dd dddd	None	7	42
Int	v	Software interrupt	1010 0000 0vvv vvvv	None	7	42
SHR	n,d	Shift right out of carry	1101 000n nndd dddd	Z,C,S	4	41, 42, 43
SHL	n,d	Shift left into carry	1101 001n nndd dddd	Z,C,S	4	41, 42, 43
ROR	n,d	Rotate right	1101 010n nndd dddd	Z,C,S	4	41, 42, 43
ROL	n,d	Rotate left	1101 011n nndd dddd	Z,C,S	4	41, 42, 43
ADDI	n,d	Add immediate	1101 100n nndd dddd	Z,S	4	42
SUBI	n,d	Subtract immediate	1101 101n nndd dddd	Z,S	4	42
NOT	d	1’s complement	1101 1110 00dd dddd	Z,S	4	42
NEG	d	2’s complement	1101 1110 01dd dddd	Z,O,C,S	4	42
CBW	d	Sign-extend d(7:0) to d(15:0)	1101 1110 10dd dddd	Z,S	4	42
STI		Enable interrupts	1101 1111 1100 0000	None	3	42
CLI		Disable interrupts	1101 1111 1100 0001	None	3	42
STC		Set carry	1101 1111 1100 0010	C	3	42
CLC		Clear carry	1101 1111 1100 0011	C	3	42

Opcode field descriptions									Clock Adder
Field	Description	Addressing mode	5	4	3	2	1	0	
S	Source	Register	0	0	r	r	r	r	0
D	Destination	Immediate	0	1	1	1	1	1	0
C	Condition code	Direct	1	0	b/w	1	1	1	1
O	Signed offset	Indirect	0	1	b/w	r	r	r	1
V	Interrupt vector	Indirect with Auto Increment	1	0	b/w	r	r	r	2
N	Count value -1	Indirect with Index	1	1	b/w	r	r	r	3

b/w: “1” = byte access, “0” = word access.

Indirect with auto-increment and byte-wide indirect addressing is illegal with R15.

Notes:

41. The number in the “clock cycles” column reflects the number of clock cycles for register or immediate accesses. For each occurrence of other types of accesses, include the appropriate “clock adder” as listed in the Addressing Modes table below.
42. All clock cycle values assume zero wait-states.
43. A shift of one is done in four clock cycles, each additional shift adds two more clock cycles.

9.0 CS5954AM–Electrical Specifications

9.1 Absolute Maximum Ratings

This section lists the absolute maximum ratings of the CS5954AM. Stresses above those listed can cause permanent damage to the device. Exposure to maximum rated conditions for extended periods can affect device operation and reliability.

Parameter	Range
Storage Temperature	–40°C to 125°C
Voltage on Any Pin with Respect to Ground	–0.3V to 7.3V
Power Supply Voltage (V_{DD})	3.3V ±10%
Power Supply Voltage (V_{DD1})	3.3V ±10%
Lead Temperature (10 seconds)	180°C
Junction Temperature (Tjmax)	125°C

9.2 Recommended Operating Conditions

Parameter	Min.	Typical	Max.
Power Supply Voltage, V_{DD}	3.0V	3.3V	3.6V
Power Supply Voltage, V_{DD1}	3.0V		3.6V
Operating Temperature	0°C		65°C

9.3 Crystal Requirements (XIN, XOUT)

Crystal Requirements (XIN, XOUT)	Min.	Typical	Max.
Operating Temperature Range	0°C		65°C
Series Resonant Frequency		48 MHz	
Frequency Drift over Temperature			±20 PPM
Accuracy of Adjustment			±30 PPM
Series Resistance			100Ω
Shunt Capacitance	3 pF		6 pF
Load Capacitance		20 pF	
Driver Level	20 µW		5 mW
Mode of Vibration 3rd overtone			

9.4 External Clock Input Characteristics (XIN)

Parameter	Min.	Typical	Max.
Clock Input Voltage @ XIN (XOUT is Opened)	1.5V		
Clock Frequency		48 MHz	

9.5 CS5954AM DC Characteristics

Parameter	Description	Min.	Typical	Max.
V_{IL}	Input Voltage LOW	–0.5V		0.8V
V_{IH}	Input Voltage HIGH	2.0V		V_{DD} + 0.3V
V_{OL}	Output Voltage LOW (I_{OL}= 4 mA)			0.4V
V_{OH}	Output Voltage HIGH (I_{OH}= –4 mA)	2.4V		
I_{OH}	Output Current HIGH	4 mA		

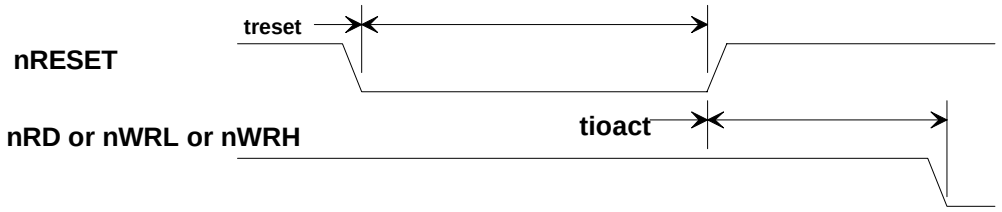
9.5 CS5954AM DC Characteristics (continued)

I_{OL}	Output Current LOW	4 mA		
C_{IN}	Input Capacitance			20 pF
I_{CC}	Supply Current (V_{DD})			< 30 mA
I_{USB}	Supply Current (V_{DD1})			< 10 mA
Pd	Power Dissipation			0.7W
$I_{CC\ +USB\ Susp.}$	Suspend Supply Current			< 220 µA

9.6 CS5954AM USB Transceiver Characteristics

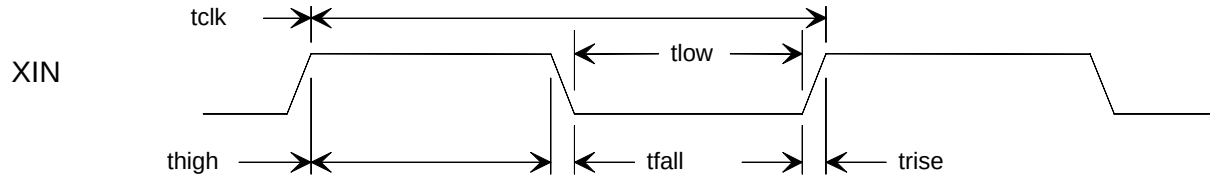
Parameter	Description	Min.	Typical[44]	Max.
V_{IHYS}	Hysteresis On Input (Data+, Data–)	0.1V		200 mV
V_{USBIH}	USB Input Voltage HIGH		1.5 V	2.0V
V_{USBIL}	USB Input Voltage LOW	0.8V	1.3 V	
V_{USBOH}	USB Output Voltage HIGH	2.2V		
V_{USBOL}	USB Output Voltage LOW			0.7V
Z_{USBH}[45]	Output Impedance HIGH State	28Ω		42Ω
Z_{USBL}[45]	Output Impedance LOW State	28Ω		42Ω

9.7 CS5954AM Reset Timing



Parameter	Description	Min.	Typical	Max.
treset	nRESET Pulse Width	100 ms		
tioact	nRESET HIGH to nRD or nWRx Active	100 ms		

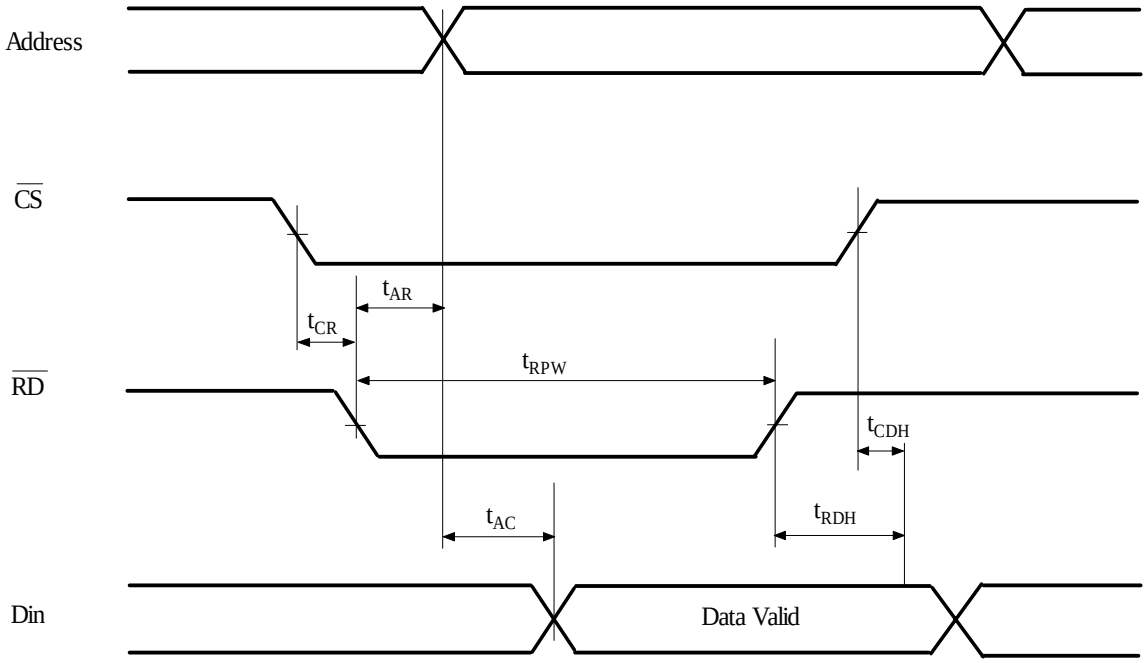
9.8 CS5954AM Clock Timing Specifications



Notes:
44. All typical values are V_{DDx} = 3.3 V and TAMB= 25°C.
45. Z_{USBX} Impedance Values includes an external resistor of 28–42Ω ± 1%.

Parameter	Description	Min.	Typical	Max.
tclk	Clock Period (48 MHz)	20.0 ns	20.8 ns	
thigh	Clock HIGH Time	9 ns		11 ns
tlow	Clock LOW Time	9 ns		11 ns
trise	Clock Rise Time			5.0 ns
tfall	Clock Fall Time			5.0 ns
	Duty Cycle	–5%		+5%

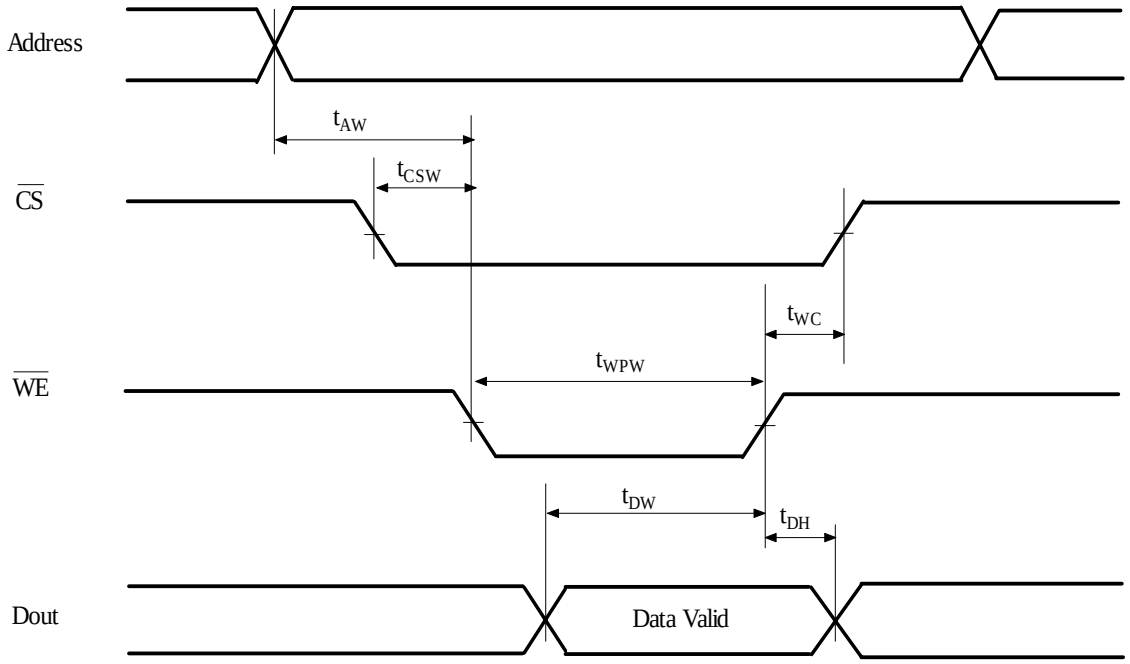
9.9 CS5954AM SRAM Read Cycle



Parameter	Description	Min.	Typical	Max.
t_{CR}	CS LOW to RD LOW	1 ns		
t_{RDH}	RD HIGH to data hold	5 ns		
t_{CDH}	CS HIGH to Data Hold	3 ns		
t_{RPW}[46]	RD LOW Time	28 ns		31 ns
t_{AR}	RD LOW to Address Valid	1 ns		3 ns
t_{AC}[47]	RAM Access to Data Valid			12 ns

Notes:
46. 0 wait state cycle.
47. t_{AC} means at 0 wait states, with PCLK = 2/3 RCLK, the SRAM access time should be 12 ns max. For a 1-wait state cycle, with PCLK = 2/3 RCLK, the SRAM access time should be at 12 + 31ns = 43 ns max. See register 0xC006 description for PCLK information.

9.10 CS5954AM SRAM Write Cycle



Parameter	Description	Min.	Typical	Max.
t_{AW}	Write Address Valid to WE LOW	13 ns		
t_{CSW}	CS LOW to WE LOW	13 ns		
t_{DW}	Data Valid to WE HIGH	25 ns		
t_{WPW}[48]	WE Pulse Width	28 ns		
t_{DH}	Data Hold from WE HIGH	5 ns		
t_{WC}	WE HIGH to CS HIGH	15 ns		

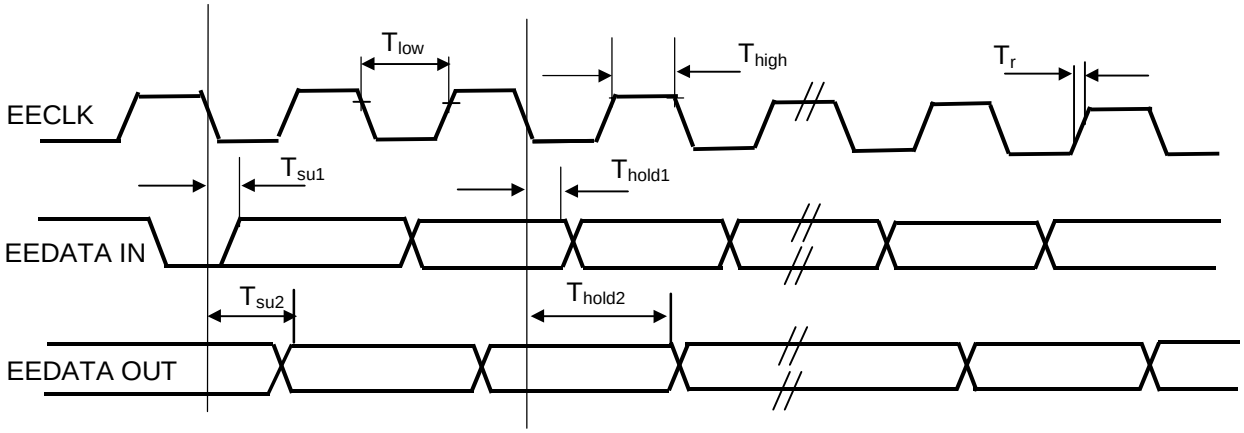
9.11 Thermal Specifications

Parameter	Min.	Typ.	Max.
Junction Temperature (T_{jmax})			125°C
Package thermal impedance (θ_{ja})	65°C/W		75°C/W
Dissipated power @ 65°C ambient (P_{max})			0.8W

Note:
48. This is at 1-wait state with PCLK = 2/3 RCLK. For 2-wait states, add 31 ns.

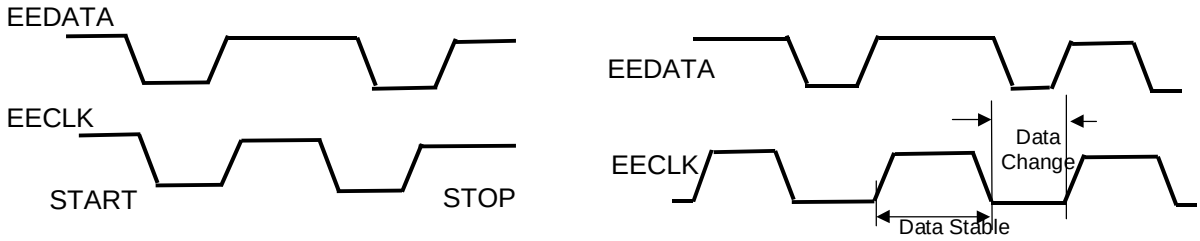
9.12 2-wire Serial Interface EEPROM Timing

1-EEPROM Bus Timing–Serial I/O[49]



2-Start and Stop Definition

3-Data Validity



Parameter	Min./Max. Timing	Notes
T_{low}	4.7 µs min.	See ATMEL Data Sheet for
T_{high}	4.0 µs min.	Complete Timing Detail
T_r	1.0 µs max.	
T_{su1}	200 ns max.	
T_{hold1}	0 ns	
T_{su2}	4.5 µs min.	
T_{hold2}	100 ns max.	

Note:
49. Timing will conform to standard as illustrated in ATMEL AT24COX data sheet

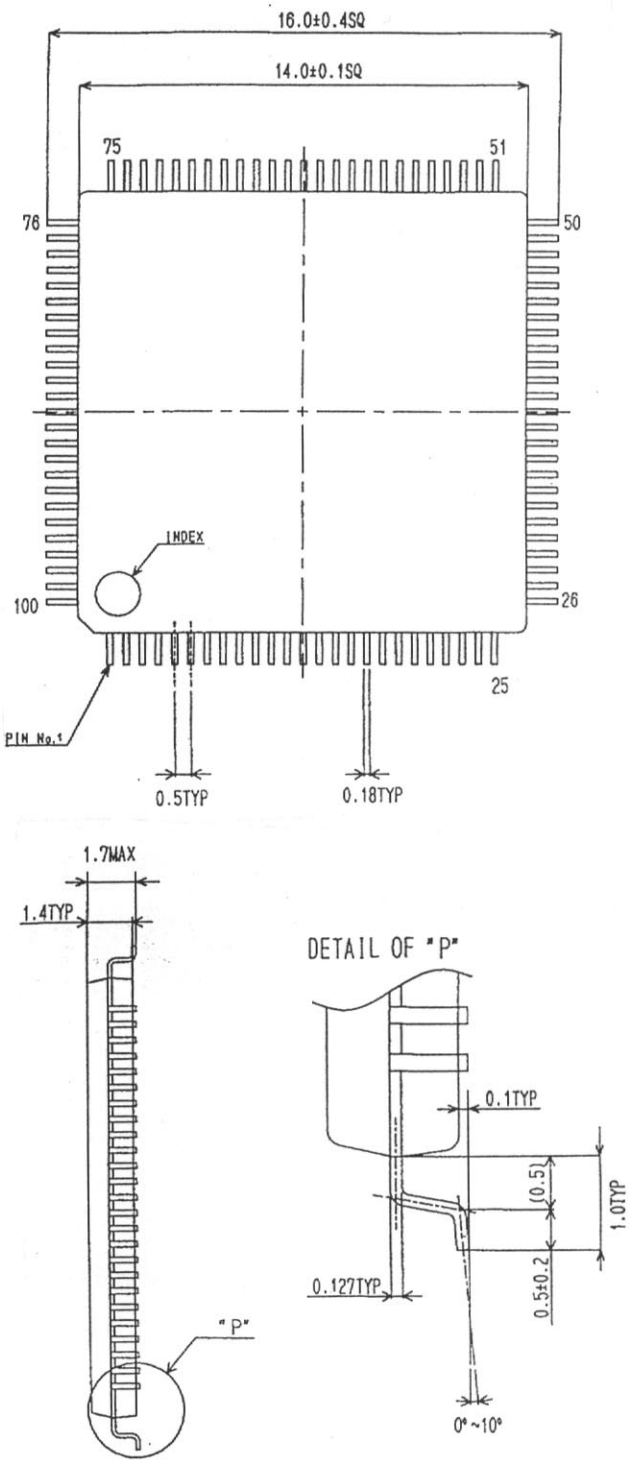
10.0 Package and Ordering Information

10.1 Ordering Information

Table 10-1. Ordering Information

Order Code	Package Type
CS5954AM	SQFP-100
CY4616	Flash Drive Reference Design Kit

10.2 Package Drawings and Dimensions



10.3 Package Markings

YYWW = Date code

XXXX = Product code

Z.ZZ = Revision code

11.0 Warranty Disclaimer and Limited Liability

Microsoft and Windows are registered trademarks of Microsoft Corporation. CS5954AM is a trademark of the Cypress Semiconductor. All product names are trademarks or registered trademarks of their respective owners.

12.0 Revision History

Document Title: CS5954AM USB Controller for NAND Flash Document Number: 38-08025				
REV.	**ECN NO.**	**ISSUE DATE**	**ORIG. OF CHANGE**	**DESCRIPTION OF CHANGE**
**	114563	06/10/02	BHA	New Data Sheet