Semiconductors

AN10025_1

Interfacing ISP1160x to Intel® StrongARM® SA1110 Processor

January 2003

Application Note

Rev. 1.0

Revision History:

Rev.	Date	Descriptions	Author
1.0	Jan 2003	First release	Jason Ong

Note: ISP1160x denotes any Philips embedded USB host controller whose name starts with 'ISP1160'; this includes ISP1160A, ISP1160A1, and any future derivatives.

We welcome your feedback. Send it to wired.support@philips.com.

Philips Semiconductors - Asia Product Innovation Centre
Visit www.semiconductors.philips.com/buses/usb or www.flexiusb.com

PHILIPS

This is a legal agreement between you (either an individual or an entity) and Philips Semiconductors. By accepting this product, you indicate your agreement to the disclaimer specified as follows:

DISCLAIMER

PRODUCT IS DEEMED ACCEPTED BY RECIPIENT. THE PRODUCT IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND. TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, PHILIPS SEMICONDUCTORS FURTHER DISCLAIMS ALL WARRANTIES, INCLUDING WITHOUT LIMITATION ANY IMPLIED WARRANTIES OF MERCHANT ABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NONINFRINGEMENT. THE ENTIRE RISK ARISING OUT OF THE USE OR PERFORMANCE OF THE PRODUCT AND DOCUMENTATION REMAINS WITH THE RECIPIENT. TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, IN NO EVENT SHALL PHILIPS SEMICONDUCTORS OR ITS SUPPLIERS BE LIABLE FOR ANY CONSEQUENTIAL, INCIDENTAL, DIRECT, INDIRECT, SPECIAL, PUNITIVE, OR OTHER DAMAGES WHATSOEVER (INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF BUSINESS PROFITS, BUSINESS INTERRUPTION, LOSS OF BUSINESS INFORMATION, OR OTHER PECUNIARY LOSS) ARISING OUT OF THIS AGREEMENT OR THE USE OF OR INABILITY TO USE THE PRODUCT, EVEN IF PHILIPS SEMICONDUCTORS HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

CONTENTS

Note: ISP1160x denotes any Philips embedded USB host controller whose name starts with 'ISP1160'; this includes ISP1160A, ISP1160A1, and any future derivatives.

1. Overview

When ISP1160x is integrated into a personal digital assistant (PDA) or handheld personal computer (HPC), it is usually connected to the external bus interface of a Reduced Instruction Set Computer (RISC) processor. This application note presents the critical issues in the ISP1160x embedded design, using the Intel® StrongARM® SA1110 processor as a concrete example.

The actual implementation of the SA1110-ISP1160x uses an Intel StrongARM-1110 Hardware Development Platform, a Philips ISP1160x evaluation kit, and a Philips ISP1160x Bridging Board for SA1110.

2. ISP1160x Interface Signals to a RISC Processor Bus

The processor bus interface of ISP1160x is designed for a simple direct connection with a RISC processor. The data transfer can be done in the programmed I/O (PIO) or direct memory access (DMA) mode. The estimated maximum data transfer rate on ISP1160x's generic processor bus is approximately 15 Mbyte/s. This is based on an ISP1160x internal clock frequency value of 48 MHz. To achieve the maximum data transfer rate on the host processor bus, ISP1160x contains ping pong structured RAM that allows alternative access from the RISC processor or from the internal Host Controller. The Host Controller uses 2 kbytes of the ping memory and 2 kbytes of the pong memory in its allocated memory.

The main ISP1160x signals to consider when connecting to a StrongARM SA1110 processor are:

- A 16-bit data bus (D[15:0]) for ISP1160x, which is "little endian" compatible.
- An address line (A0) for complete addressing of the ISP1160x internal registers:
  - **A0 = 0**—Selects the Data Port of the Host Controller
  - **A0 = 1**—Selects the Command Port of the Host Controller
- One $\overline{\text{CS}}$ line to select ISP1160x in a certain address range of the host system. This input signal is active LOW.
- $\overline{\text{RD}}$ and $\overline{\text{WR}}$ are common read and write signals. These signals are active LOW.
- An interrupt line, INT1, which is programmable as active on level or edge and HIGH or LOW.
- The $\overline{\text{RESET}}$ signal is active LOW.

3. Intel StrongARM SA1110 Processor

This section describes the main features of the Intel StrongARM SA1110 processor for connecting to ISP1160x.

The "Memory and PCMCIA Control Module" of the Intel StrongARM processor is responsible for generating all signals for interfacing with ISP1160x. The following features are useful for direct connection of ISP1160x:

- The "Memory Control Module" generates all the necessary signals to control different types of external devices:
  - DRAM (up to four banks of FPM, EDO and SDRAM)
  - Static memory (up to three banks of ROM, Flash, SRAM and SMAROM, selected by nCS0, nCS1 and nCS2 signals)
  - Static memory and variable latency I/O devices (up to three banks of ROM, Flash, SMROM and SRAM, such as variable latency I/O devices, selected by the nCS3, nCS4 and nCS5 signals).

Additional wait-states can be inserted by programming the internal registers of SA1110, if necessary.

- The data bus size of these memory areas can be set as 16-bit or 32-bit wide. ISP1160x uses the 16-bit wide data bus size.
- The memory access is defined as "little endian" or "big endian" types, according to the value of the "big endian bit" in the control register. By default, at power-on or after a reset, the SA1110 processor uses the "little endian" memory access scheme that corresponds to the ISP1160x requirement.

4. Considerations in Timing Diagrams and WAIT States

The following is a short study of the timing diagrams of the main bus cycles of ISP1160x and Intel SA1110:

The memory clock determines the timing diagram of the external bus cycle of the Intel StrongARM SA1110 processor, which is equal to two CPU clock cycles. Timing during $\overline{RD}/\overline{WR}$ accesses is determined by the settings of the MSC0, MSC1 and MSC2 registers that correspond to the chip select pairs nCS(5,4), nCS(3,2) and nCS(1,0), respectively. All timing fields are specified as numbers of memory clock cycles. Each register contains two identical configuration fields corresponding to each nCS within one of the pairs mentioned earlier. By programming the MSC0, MSC1 and MSC2 registers, you can modify the assert time of each beat of a burst RD/WR, the deassert time between each beat of a burst RD/WR, and the hold-off time after a write to subsequent accesses.

According to the ISP1160x datasheet specifications, a read operation requires the following timing parameters (the write operation is similar); see Figure 4-1:

- t_{RL} = 33 ns ($\overline{RD}$ LOW pulse width—minimal value required by ISP1160x)
- t_{RHRL} = 110 ns ($\overline{RD}$ HIGH to next $\overline{RD}$ LOW—minimal value required by ISP1160x)
- t_{RHDZ} = 3 ns ($\overline{RD}$ hold time, minimal value that can be expected from ISP1160x)
- t_{RC} = 143 ns (will result as a sum of t_{RL} and t_{RHRL})
- t_{SHSL} = 300 ns (first $\overline{RD}/\overline{WR}$ after command).

For a detailed analysis of a timing diagram, consider the access of an ISP1160x internal register (for example, the Control Register of the Host Controller). The access requires two phases: writing the address (index) of the selected register into the Command Port, and then only data transfer access (RD/WR) may take place.

Note: the index of each register is different, according to whether it is a RD or WR operation.

The timing diagram in Figure 4-1 describes the two phases of accessing ISP1160x:

- The first phase is accessing the Command (control) Port of ISP1160x to write the address (index) of the data port that will be accessed. In this phase, $\overline{CS}$ is active. The data lines D[15:0] contain the desired address. The $\overline{WR}$ pulse will be activated and will latch the data. Note the value of t_{SHSL} that represents the minimum time required between occurrence of the first phase and the second phase. As an example of the Host Controller "Control Register", a value of 01H will be transferred during a $\overline{RD}$ operation and 81H during a $\overline{WR}$ operation.
- The second phase consists of access (read or write) to the data port selected by the address latched in the previous phase. Two timing diagrams are combined again in this second phase: one for read access and one for write access. A series of $\overline{RD}$ and $\overline{WR}$ pulses are shown in the diagram to define the timing requirements between two consecutive accesses to ISP1160x: t_{RHRL}, t_{WHWL}, t_{RC}, t_{WC}, t_{RLDV}, as specified in the datasheet.

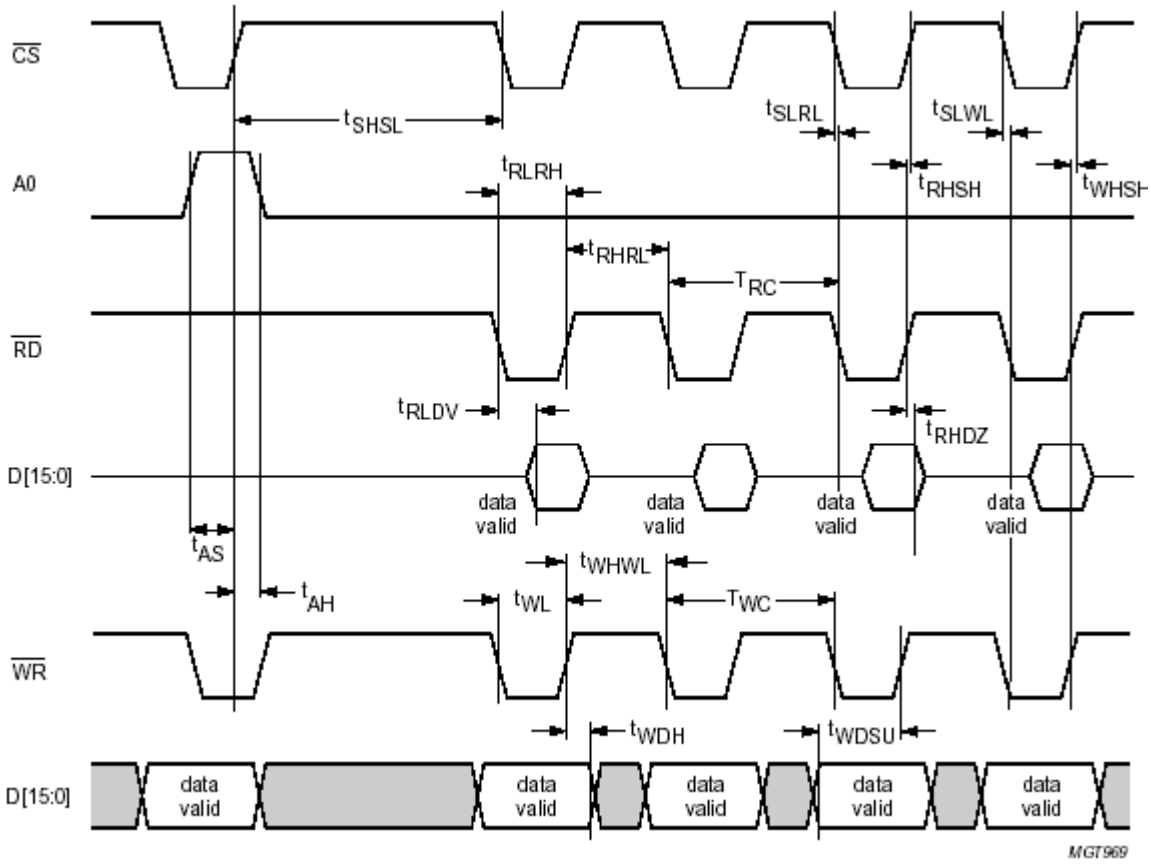


Figure 4-1: Programmed Interface Timing (16-bit Read/Write)

5. Using interrupts

ISP1160x generates an interrupt on the INT pin. ISP1160x's INT can be connected to one of the 28 GPIO lines of the SA1110 processor. Each GPIO line of the SA1110 can be programmed to detect a rising or falling edge and generate an interrupt. The type of edge detection is programmed through the GPIO rising-edge detect register (GRER) and GPIO falling-edge detect register (GFER). The state of the edge detect can be determined by reading the edge-detect status register (GEDR).

ISP1160x's INT is programmable as active on level or edge and HIGH or LOW, as specified in the *HcHardwareConfiguration register*. You must match the settings of the ISP1160x interrupt line—INT—with the settings of the GRER and GFER registers of SA1110 used for programming the type of edge detection.

You can use ISP1160x's INT output signal to wake up the host system's processor (in this case SA1110) from the idle or sleep mode. The GPIO pins are defined as category three signals and are actively sampled by SA1110 even during the sleep mode. SA1110's sleep mode offers the greatest power savings. In this mode, the internal sleep state machine of SA1110 is running off the 32.768 kHz crystal oscillator and watches for a preprogrammed event to occur, which will initiate the wake-up sequence. The V_{DDX} I/O voltage supply of SA1110 must be present during the sleep mode to enable this wake-up method.

6. Suspend and Resume

You can enable ISP1160x to enter the Reset, Resume, Operational and Suspend functional states by programming the *HcControl register*.

Another way to wake up ISP1160x from the suspend mode is to use the input signal H_WAKEUP. This signal may be connected to any available GP I/O line of SA1110.

Monitoring the H_SUSPEND pin can determine ISP1160x's actual status, without having to access internal status registers. Connecting this signal to any available GP I/O port of the SA1110 is an easy way to determine ISP1160x's status.

ISP1160x may wake up when its $\overline{CS}$ input signal becomes active.

7. Schematic

The schematic in Figure 7-1 shows the connection of ISP1160x to a StrongARM SA1110 processor in a minimal hardware configuration.

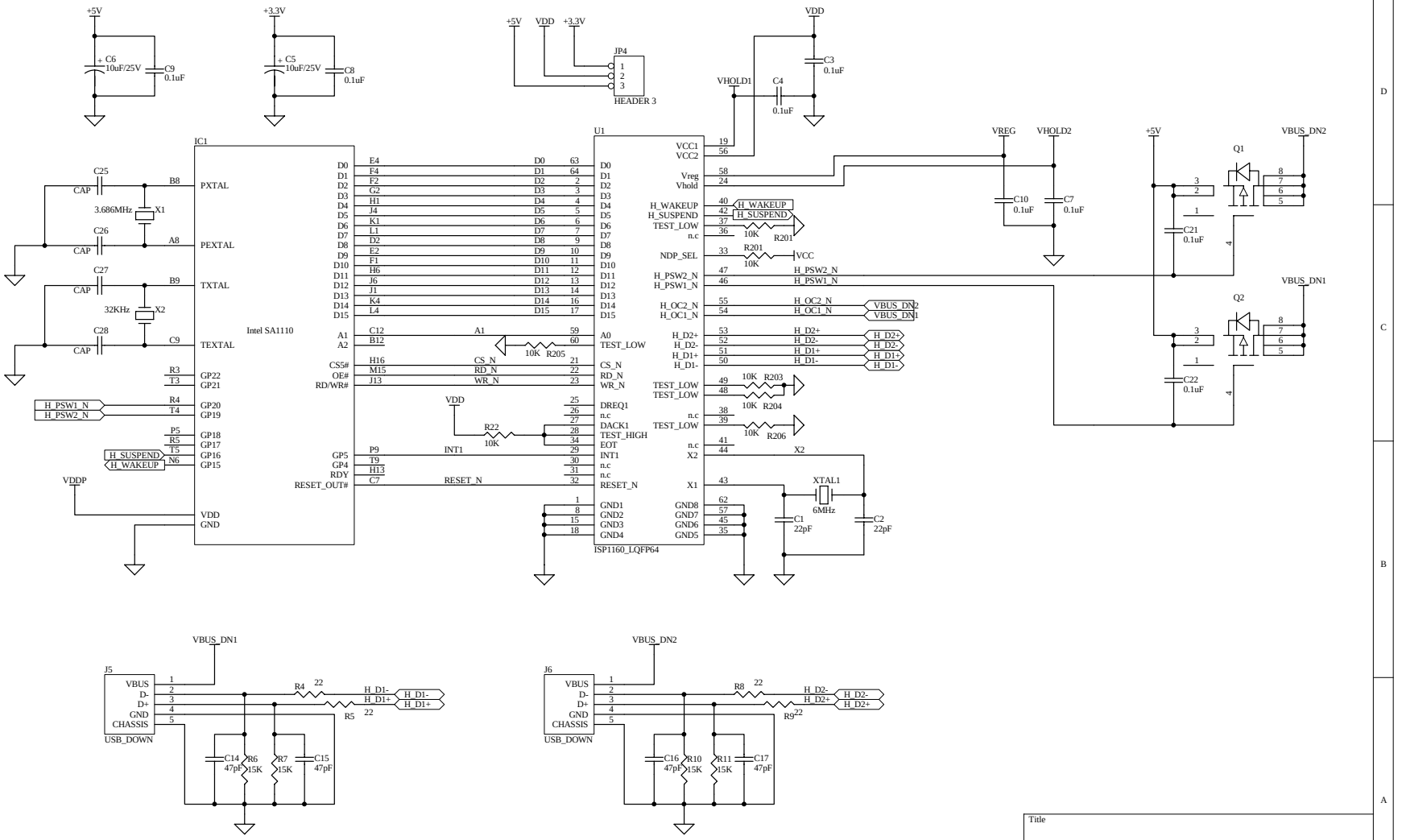
In this example schematic, ISP1160x is simply selected by nCS5. To correctly access ISP1160x, it is assumed that the memory space selected by nCS5 is programmed for 16-bit access and "little endian".

ISP1160x's $\overline{\text{RESET}}$ input signal is generated by the RESET_OUT# signal of the SA1110 processor, when its RESET_IN signal is active. The RESET_OUT# signal is also asserted for soft reset events (sleep and watchdog). Connecting ISP1160x's $\overline{\text{RESET}}$ input to RESET_IN of SA1110 may be a better solution if INT is used to wake up SA1110.

Pins $\overline{\text{H\_PSW1}}$ and $\overline{\text{H\_PSW2}}$ are connected to lines 4 and 5 of the same I/O port. This connection creates an alternative way to determine the power status of each downstream port.

Input signals $\overline{\text{H_OC1}}$and $\overline{\text{H\_OC2}}$ are used by ISP1160x to detect an overcurrent on the downstream facing ports. Because ISP1160x implements separate overcurrent detection and protection circuits are for each downstream facing port, so when overcurrent is detected on a downstream port, power to that port will be turned off. Connecting the voltages of the two downstream ports VBUS_DN1 and VBUS_DN2 to the $\overline{\text{H\_OC1}}$ and $\overline{\text{H\_OC2}}$ pins allow detection of the current value by sensing voltage drop on Q1 and Q2, where Q1 and Q2 are PMOS transistors with very low switch-on resistance Rds(on). Select Q1 or Q2 based on the maximum current you want, which determines the value of Rds(on). For example, if the allowed maximum current is approximately 0.5 A, a voltage drop of 75 mV will trigger the overcurrent circuitry and Rds(on) of approximately 150 MΩ will result. Connecting the ISP1160x input pins $\overline{\text{H\_OC1}}$ and $\overline{\text{H\_OC2}}$ to +5 V will disable ISP1160x's internal overcurrent protection. You may also opt for an external overcurrent protection circuit.

Figure 7-1: ISP1160x Interfaces to Intel StrongARM SA1110



Title		
Size: B	Number	Revision
Date: 18-Dec-2002	Sheet of	
File: I:\Jason Backup 030701\Philips\ISP1161\Documentation\App Note Schematic\1160 SA2.DDB	Drawn By:	

Appendix A. Hardware Installation

To install ISP1160x to Intel StrongARM SA1110:

1. Connect the ISP1160x Evaluation Board to the StrongBridge Board.
2. Connect the StrongARM Development Kit to the StrongBridge Board.
3. Connect the StrongARM Companion Chip Development Kit to the StrongARM Development Kit Board.

The ISP1160x interface to Intel StrongARM SA1110 Evaluation Kit is ready for testing!

For a clearer graphical installation, see the following figures:

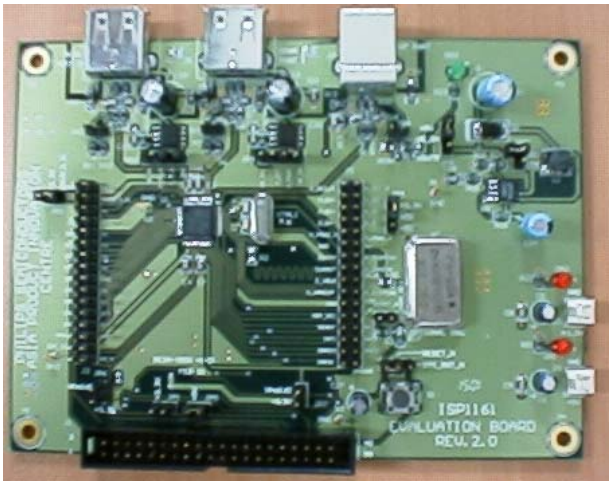


Figure A-1: ISP1160x Evaluation Board

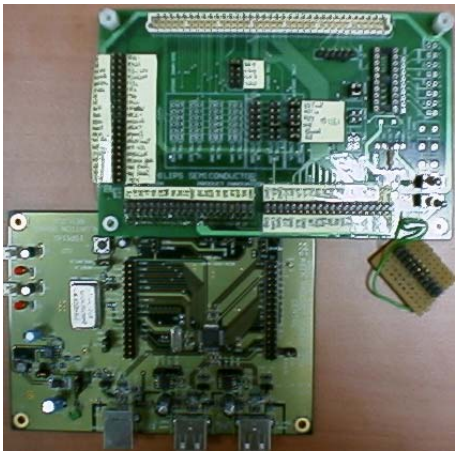


Figure A-2: ISP1160x Evaluation Board with StrongBridge Board

Figure A-3: ISP1160x Evaluation, StrongBridge and StrongARM Development Kit

Figure A-4: ISP1160x Evaluation, StrongBridge, StrongARM Development Kit and StrongARM Companion Chip Development Kit

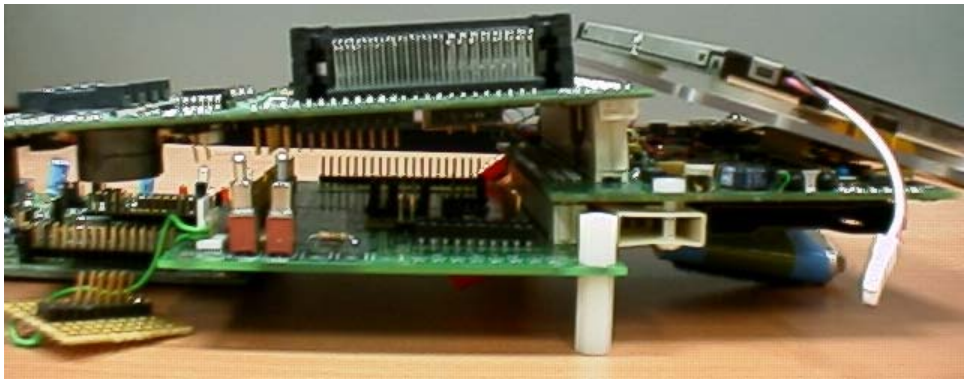


Figure A-5: Side View of ISP1160x Interface to Intel StrongARM SA1110 Evaluation Kit

Appendix B. Software Installation

Host Requirements

To develop on the Microsoft® Windows® CE platform:

1. Install Platform Builder on a PC running on Windows NT® 4.0 or Windows 2000. The current version for Platform Builder is 3.0
2. After the destination location has been chosen, the CESH Update Screen appears. Determine whether this is a new install or update of Microsoft Windows CE Platform Builder:
    - For new installs, choose **Use Ethernet**.
    - For updates, choose **Do not change Current CESH settings**.

Network Requirements

To use the Ethernet boot application to download the Windows CE image or to perform Ethernet debugging, the host and target systems must choose one of these network configurations:

- A network HUB with a DHCP server.
- A 10BASE-T unshielded twisted pair (UTP) crossover Ethernet cable between the host and SA1110 development board (provided with the SA1110 development board kit).

Often, the amount of network traffic present on a corporate network HUB can impede the downloading efforts. To avoid network traffic, it is recommended that you use this 10BASE-T UTP crossover Ethernet cable.

Appendix C. References

- *Universal Serial Bus Specification Rev. 2.0*
- *ISP1160 Embedded Universal Serial Bus Host Controller* datasheet.
- *ISP1160AI Embedded Universal Serial Bus Host Controller* datasheet.

Philips Semiconductors

Philips Semiconductors is a worldwide company with over 100 sales offices in more than 50 countries. For a complete up-to-date list of our sales offices please e-mail sales.addresses@www.semiconductors.philips.com.
A complete list will be sent to you automatically.
You can also visit our website
http://www.semiconductors.philips.com/sales/

www.semiconductors.philips.com

PHILIPS