Semiconductors

# AN10026_1

## Interfacing ISP1160x to Motorola® DragonBall™ EZ RISC Processor

January 2003

## Application Note

## Rev. 1.0

***Revision History:***

| Version | Date | Descriptions | Author |
|---|---|---|---|
| 1.0 | January 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.

# CONTENTS

# FIGURES

**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 RISC processor. This application note deals with the critical issues in ISP1160x embedded design, using the Motorola® DragonBall™ EZ RISC processor as a concrete example.

## 2. ISP1160x Processor Interface Signals

The ISP1160x processor bus interface 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 a 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 you should take into consideration for connecting to a Motorola DragonBall EZ RISC processor are:

- A 16-bit data bus (D[15:0]) for ISP1160x, which is "little endian" compatible.
- An address line (A0) necessary 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 used 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.
- A DMA channel standard control lines: DREQ, $\overline{\text{DACK}}$ and EOT. Since the DragonBall EZ processor does not contain a DMA controller, these signals will not be used in a minimal hardware implementation.
- An interrupt line INT1, which has programmable level/edge and polarity (active HIGH or LOW).
- The CLKOUT signal has a maximum value of 48 MHz.
- The $\overline{\text{RESET}}$ signal is active LOW.

## 3. Motorola DragonBall EZ

Motorola MC 68EZ328, also called DragonBall EZ, is a processor from the second generation of the DragonBall EZ family. The MC68EZ328 combines a Motorola MC68EC000 processor with intelligent peripheral modules and typical system interface logic. The operating frequency of this processor is 16 MHz, obtained by an internal PLL, which accepts as input a 32.768 kHz crystal, oscillator or a clock signal.

The main internal blocks of the DragonBall-EZ processor to consider when analyzing the bus interface, on which ISP1160x is connected, are as follows:

- The 8-bit or 16-bit 68000-bus interface block.
- The clock synthesizer and power control block.
- The chip-select generation block.
- The interrupt controller.

The DragonBall-EZ processor generates eight programmable general-purpose chip-select signals, which are arranged in four groups of two (CSA[1:0], CSB[1:0], CSC[1:0] and CSD[1:0]). Each chip-select block allows several features to be selected, which may be specific to the devices connected to each selected area:

- Bus size: 8 bits or 16 bits can be independently selected for each area. The default setting corresponds to a 16-bit bus size.
- Number of wait states inserted in a bus cycle can be set from zero to six.
- Each area selected by a CSn can be defined as read-only or read/write access.
- Different types of memory are supported for an area selected by a certain CSn: DRAM, ROM, SRAM and Flash memory.
- The size of any memory area selected by a Chip-Select signal can be selected from a set of predefined ranges (32 kbytes, 64 kbytes, 128 kbytes, 256 kbytes, 512 kbytes, 1 Mbytes, 2 Mbytes and 4 Mbytes).

For a CSn signal to operate correctly, you must program the “Chip-Select Group Base Address Registers” and the “Chip Select Registers”, accordingly.

## 4. Considerations in Timing Diagrams and WAIT states

The following is a short study of ISP1160x timing diagrams.

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

| | | |
|---|---|---|
| • $t_{RL}$ | = 33 ns | ($\overline{RD}$ LOW pulse width—minimal value ISP1160x requires), |
| • $t_{RHRL}$ | = 110 ns | ($\overline{RD}$ HIGH to next $\overline{RD}$ LOW—minimal value ISP1160x requires) and |
| • $t_{RHDZ}$ | = 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). It requires two phases: writing the address (index) of the selected register into the Command Port; then only data transfer access (RD/WR) may take place.

**Note**: The index of each register differs according to whether it is a RD or a WR operation.

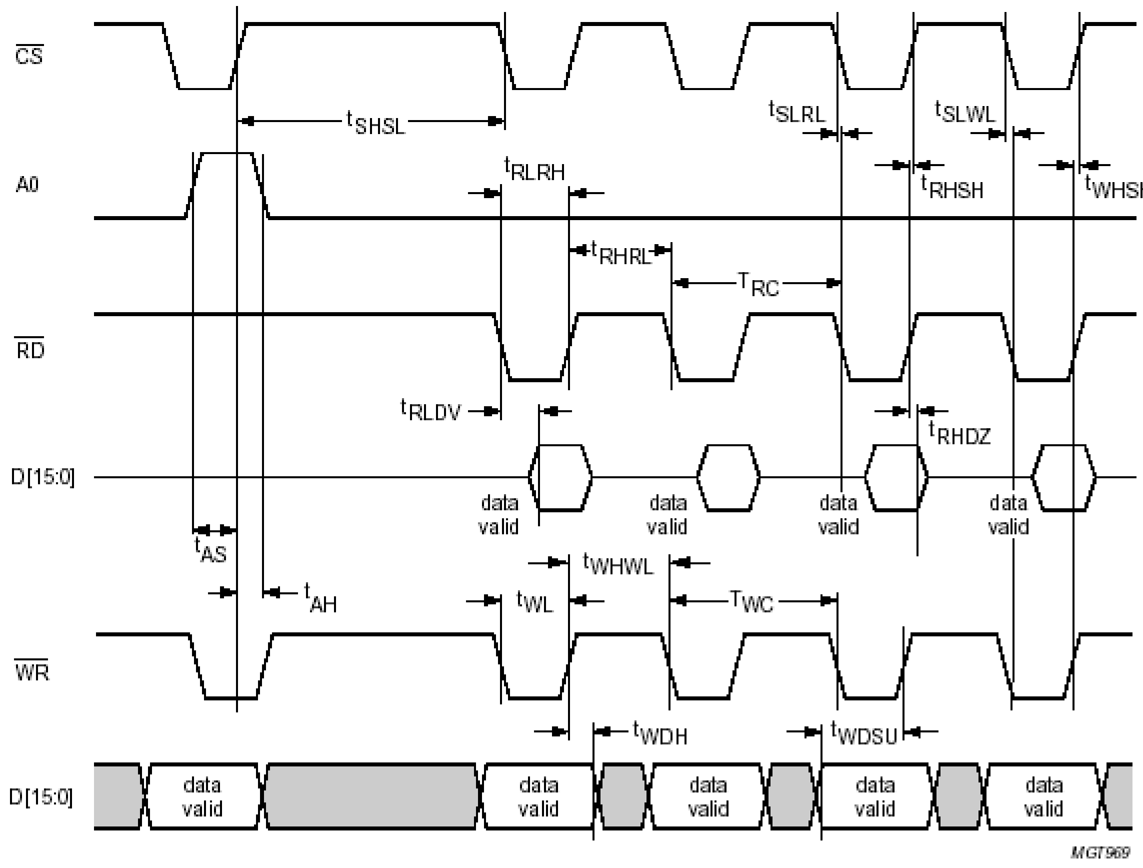


**Figure 4-1: Programmed Interface Timing (16-bit Read/Write)**

When the ISP1160x connection area is defined as SRAM, ISP1160x will operate correctly for a bus clock CKIO = 33 MHz. Timing measurements show that inserting wait-states in the standard bus cycles of DragonBall EZ is unnecessary. Nevertheless, we will describe wait-state insertion, to cater for cases when faster bus cycles are used for accessing ISP1160x.

Wait states insertion can be implemented through hardware or software. Both solutions will delay the rising edge of $\overline{RD}$ or $\overline{WR}$ to the next CKIO cycle and will determine an elongation of the $\overline{RD}$ or $\overline{WR}$ LOW pulse that can be calculated as:

$t_w$ = W x T(CKIO); where: (W) is the number of wait states desired

T(CKIO) is the cycle length of CKIO.

**Note**: the value of $t_{RHRL}$ will not be modified by the number of wait states inserted by any of the solutions mentioned earlier. The value of this parameter must be calculated and correctly adjusted according to the number and length of instructions executed by the DragonBall EZ processor between two successive accesses to ISP1160x. The "software solution" for wait-state insertion in a bus cycle is simple and is preferred in a minimal configuration, if additional wait states are necessary.

## 5. Using Interrupts

ISP1160x generates an interrupt on the INT pin. This INT signal can be directly connected to any available IRQ signal of the DragonBall processor. ISP1160x's INT can be programmed as active on level or edge and HIGH or LOW, as specified in the *HcHardwareConfiguration* register.

## 6. Suspend and Resume

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

Another way to wake up ISP1160x from the suspend mode is to use the input signals H_WAKEUP (for the Host Controller). Monitoring the H_SUSPEND pin for the host status can determine ISP1160x's actual status, without having to access the internal status registers.

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

## 7. Schematic Diagram

The schematic diagram on the following page shows ISP1160x connected to a Motorola DragonBall EZ processor, in a minimal hardware configuration. For a more detailed description of the connection of ISP1160x to a RISC processor and a study of each category of signals and timing diagrams, refer to the application note *Interfacing ISP1160x to Hitachi SH7709 RISC Processor*.

In this configuration, ISP1160x is selected by CSB0# signal, which is asserted according to the values programmed in the *Chip Select Group Base Address* and *Chip Select registers* corresponding to CSB0#.

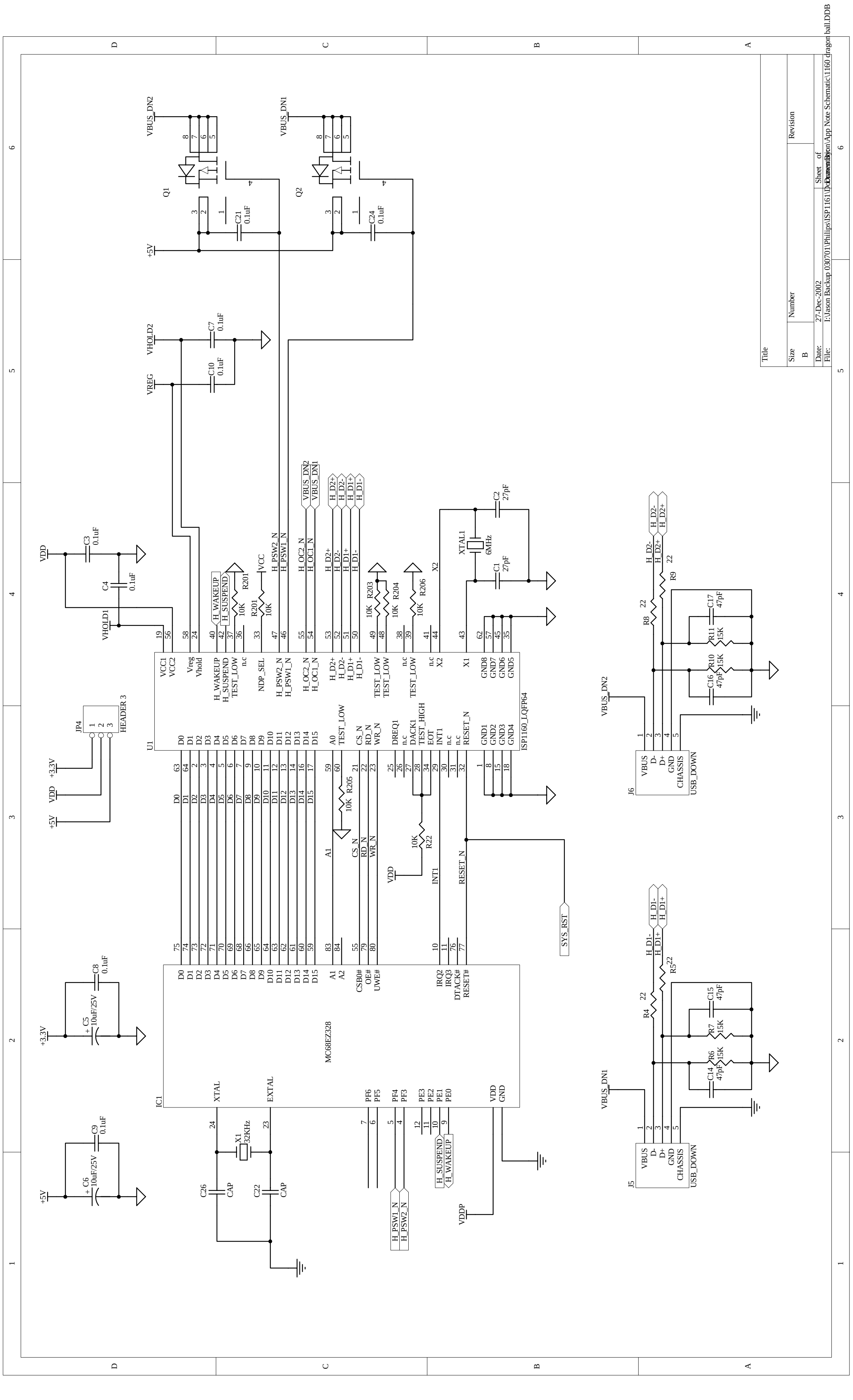
To correctly access ISP1160x, it is assumed that the area selected by CSB0# is programmed for the SRAM memory type and for 16-bit bus width accesses.

Interrupts INT are connected to the IRQ2 line of the DragonBall EZ processor. The interrupt inputs of the DragonBall EZ processor can be set as edge or level sensitive and of positive or negative polarity by programming its *Interrupt Control* register. ISP1160x also allows programming the polarity (LOW or HIGH) and the signaling mode (level or pulse) for generated interrupts INT, by setting the bits of the *HcHardwareConfiguration* register of the Host Controller. This ISP1160x feature allows a simple connection of the IRQ line, without any additional logic. The interrupts of the DragonBall processor can be masked in the *Interrupt Mask* register. Checking the status of an interrupt line can be done by monitoring the values of the *Interrupt Status* register and the *Interrupt Pending* register of the DragonBall-EZ processor.

Analysis of the timing diagrams of ISP1160x and the DragonBall-EZ processor, running at a standard frequency of 16 MHz, shows that inserting wait states in a standard bus cycle is unnecessary. If a similar RISC processor with a higher bus frequency is used, you can insert wait-states during a bus cycle (accessing a certain system resource selected by CSn) by specifying the number of wait states in the corresponding *(Chip-Select) n register* of the DragonBall processor.

ISP1160x uses the input signals $\overline{\text{H\_OC1}}$ and $\overline{\text{H\_OC2}}$ to detect an overcurrent on the downstream facing ports. Because separate overcurrent detection and protection circuits are implemented in ISP1160x, so when an overcurrent is detected on a downstream port, power will be turned off at that port only. 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 enables current to be detected by sensing the voltage drop on Q1 and Q2 which are MOSFET transistors with very low switch-on resistance Rds. Choose Q1 or Q2, depending on the maximum current you want, which in turn determines the value of Rds(on). For example, if a voltage drop of 75 mV will trigger the overcurrent circuitry and the allowed maximum current is about 0.5 A, 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 can opt for an external overcurrent protection circuit.

The $\overline{\text{RESET}}$ input signal of ISP1160x is connected to the system RESET signal, which also generates the RESET# input signal for the DragonBall-EZ processor.



ISP1160_LQFP64
U1
MC68EZ328
IC1
Q1
Q2
VBUS_DN1
VBUS_DN2
+5V
+3.3V
VDD
VDDP
VREG
VHOLD1
VHOLD2
C3 0.1uF
C4 0.1uF
C7 0.1uF
C10 0.1uF
C21 0.1uF
C24 0.1uF
C5 10uF/25V
C6 10uF/25V
C8 0.1uF
C9 0.1uF
C1 27pF
C2 27pF
XTAL1 6MHz
X1 32KHz
C22 CAP
C26 CAP
R201 10K
R203 10K
R204 10K
R205 10K
R206 10K
R22 10K
R4 22
R5 22
R8 22
R9 22
R6 15K
R7 15K
R10 15K
R11 15K
C14 47pF
C15 47pF
C16 47pF
C17 47pF
JP4 HEADER 3
J5 USB_DOWN
J6 USB_DOWN
VBUS
D-
D+
GND
CHASSIS
H_D1-
H_D1+
H_D2-
H_D2+
H_PSW1_N
H_PSW2_N
H_OC1_N
H_OC2_N
H_WAKEUP
H_SUSPEND
TEST_LOW
TEST_HIGH
NDP_SEL
CS_N
RD_N
WR_N
INT1
RESET_N
SYS_RST
DREQ1
DACK1
EOT
A0
A1
D0–D15
X1
X2
n.c
Title
Size B
Number
Revision
Date: 27-Dec-2002
Sheet of
File: I:\Jason Backup 030701\Philips\ISP1161\Documentation\App Note Schematic\1160 dragon ball.DDB
Drawn By:

## 8. References

- *Universal Serial Bus Specification Rev. 2.0*
- *ISP1160 Embedded Universal Serial Bus Host Controller* datasheet
- *ISP1160A1 Embedded Universal Serial Bus Host Controller* datasheet
- *Interfacing ISP1160x to Hitachi SH7709 RISC Processor* application note.

## 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