Élan™SC300 and Élan™SC310 Devices' ISA Bus Anomalies

Application Note

The Élan™SC300 and ÉlanSC310 microcontrollers' ISA bus has several anomalies that should be accounted for when designing a system with ISA devices. It is fairly simple to design around these issues, and none of them should stop the designer from using the ISA bus. ROM access using $\overline{\text{ROMCS}}$ and $\overline{\text{DOSCS}}$ are also discussed here since they are affected by some of the ISA anomalies.

ISA CLOCK SPEED

The ISA cycles are timed from a 9.2 MHz clock, which is slightly faster than a typical ISA bus clock (about 8.3 MHz). Therefore, the ISA cycles generated by the ÉlanSC300 and SC310 microcontrollers will be slightly faster than typical ISA cycles.

FAST ROM CYCLE ISSUES

The ROM can be accessed either at slow (ISA bus) speeds or fast (CPU clock) speeds by setting the bits at Index registers B3H and B8H. The slow cycles use the 9.2 MHz ISA clock and are compatible with ISA timings. The fast cycles use the CPU clock, which can be as fast as 33 MHz. These fast ROM cycles violate the ISA timing specification. This faster-than-ISA cycle is beneficial for system performance and is acceptable to the ROM device since the designer can use very fast ROM and Flash chips (70 nsec or faster parts).

A problem can occur with fast ROM cycles since the ROM shares signals with the ISA bus. The Address Bus (SA23:0), Data Bus (SD15:0), Commands ($\overline{\text{MEMR}}$ and $\overline{\text{MEMW}}$), and 16-bit signal ($\overline{\text{MEMCS16}}$) are used by both the ROM and ISA devices.

Designers should be aware of the following issues and avoid them in their system designs.

Fast Commands Seen On The ISA Bus

ROM uses the ISA bus commands ($\overline{\text{MEMR}}$ and $\overline{\text{MEMW}}$). In Fast mode, they occur faster than ISA timings allow. The designer should take care that any ISA devices in the design will not have a problem with these fast commands. Note that the ISA device will not be accessed with the fast commands, it will merely see them occur during the ROM access.

$\overline{\text{MEMCS16}}$ Return Time

$\overline{\text{MEMCS16}}$ is the signal that an ISA device can assert to signal to the ÉlanSC300 and SC310 devices that it can be accessed with a 16-bit data bus instead of the default 8-bit one. The ÉlanSC300 and SC310 microcontrollers use $\overline{\text{MEMCS16}}$ for ISA cycles as well as for $\overline{\text{ROMCS}}$ and $\overline{\text{DOSCS}}$ accesses. When $\overline{\text{MEMCS16}}$ is generated, the ÉlanSC300 and SC310 devices will allow a 16-bit cycle. $\overline{\text{SBHE}}$ and SA0 are decoded by the target device to determine which bytes of the data bus are active.

For $\overline{\text{DOSCS}}$, there is an alternate way to access the ROM with a 16-bit cycle. Setting index register 51H bit 1 will force all accesses to the $\overline{\text{DOSCS}}$ signal to occur with a 16-bit data bus, and $\overline{\text{MEMCS16}}$ does not need to be asserted. There is no alternate method of accessing the $\overline{\text{ROMCS}}$ signal at 16 bit.

When operating at a fast speed, both the $\overline{\text{ROMCS}}$ and $\overline{\text{DOSCS}}$ signals need $\overline{\text{MEMCS16}}$ to return within 10 nsec after the chip selects are generated inside the ÉlanSC300 and SC310 devices. Due to pin delay times, it may be necessary to return $\overline{\text{MEMCS16}}$ before the SA signals come out of the chip. Since this is not possible, it is recommended not to allow 16-bit accesses to $\overline{\text{ROMCS}}$ if it is programmed for fast access. Fast cycles to $\overline{\text{DOSCS}}$ can still be used as a 16-bit access if they are set with the register (Index 51H bit 1) and not by $\overline{\text{MEMCS16}}$.

Summary: Do not do fast $\overline{\text{ROMCS}}$ or $\overline{\text{DOSCS}}$ accesses at 16 bit if relying on the $\overline{\text{MEMCS16}}$ signal being returned. Only use slow $\overline{\text{ROMCS}}$ or $\overline{\text{DOSCS}}$ with $\overline{\text{MEMCS16}}$. The design can still use fast $\overline{\text{DOSCS}}$ at 16 bit if it is implemented using the Index register instead of $\overline{\text{MEMCS16}}$.

$\overline{\text{MEMCS16}}$ and $\overline{\text{IOCS16}}$ Hold Time

The $\overline{\text{MEMCS16}}$ and/or $\overline{\text{IOCS16}}$ hold time becomes an issue when using an ISA device that does 16-bit memory or I/O transfers and the system does 8-bit fast ROM cycles. The problem is that the $\overline{\text{MEMCS16}}$ and/or $\overline{\text{IOCS16}}$ signal may still be asserted by the ISA device at the beginning of the ROM access, and the chip will try to do a 16-bit access to a ROM that should be 8-bit.

Since the ROM access is timed off the CPU clock, it can start one CPU clock after the end of a valid 16-bit ISA access. At a 33 MHz CPU clock rate, this is less than 30 nsec (because of pad delays getting the signals off chip), which is less than the ISA spec. If $\overline{\text{MEMCS16}}$ and/or $\overline{\text{IOCS16}}$ is still asserted at a low level by the ISA device when the ROM access starts, the ROM access will be 16 bit.

This document contains information on a product under development at Advanced Micro Devices. The information is intended to help you evaluate this product. AMD reserves the right to change or discontinue work on this proposed product without notice.

Publication# **20747** Rev: **A** Amendment/**0**
Issue Date: October **1996**

Summary: If the system design demands a 16-bit ISA device and fast 8-bit ROM accesses, the $\overline{\text{MEMCS16}}$ and/or $\overline{\text{IOCS16}}$ signal must become inactive less than 30 nsec after the end of the ISA cycle. This will not be an issue if the system design does not use a 16-bit ISA device, or if the system design does not implement fast ROM cycles.

BALE

BALE is the ISA Address Latch Enable signal. It is used by devices to latch in the LA23:17 address lines. These LA address lines are not latched in the ÉlanSC300 and SC310 devices and so are not held active for the entire ISA cycle. BALE and the LA23:17 signals are available on the ÉlanSC300 and SC310 microcontroller pins in Maximum ISA mode. Since the SA23:0 address lines are already latched in the ÉlanSC300 and SC310 microcontrollers, devices that use them should not need a BALE.

The BALE signal is asserted for each ISA cycle to allow an ISA device to latch in the LA signals and use them to determine if the cycle is for that device. The problem encountered is that BALE does not get asserted for fast ROM cycles (it is asserted for slow ROM cycles, so these are not at issue). This is not a problem for the ROM, but if an ISA device that uses BALE is accessed, and the following access is a fast access to the ROM, since the fast ROM access does not generate a BALE, the ISA device may acknowledge the ROM cycle as its own (the ISA device is using the latched address from the previous cycle). This will result in either a conflict on the data bus in the case of a Read, or in faulty data being written to the ISA device in the case of a Write.

Summary: It is not recommended to design a system with ISA devices that need to use BALE if the design will also implement fast ROM accesses. If it is necessary to use an ISA device that needs BALE, the following board and software changes can be done:

Externally qualify the ISA memory commands ($\overline{\text{MEMR}}$ and $\overline{\text{MEMW}}$) with the ROM chip selects ($\overline{\text{ROMCS}}$ and/or $\overline{\text{DOSCS}}$) that are running at a fast rate. Route the new qualified memory commands to any ISA devices that use BALE. The qualifier logic will disable the commands to the ISA devices during the ROM cycles so that the ISA devices will not respond to ROM cycles. In addition to this hardware, software should program the fast ROM chip selects as address decodes only, not qualified with the command (this is done at Index registers B3H and B8H). ISA devices that use BALE should not generate $\overline{\text{MEMCS16}}$ and/or $\overline{\text{IOCS16}}$ even if the above workaround is used, unless the conditions in the sections above are also met.

ISA MASTER DEVICE ISSUE

ISA master is used by ISA devices that are capable of taking over the bus and transferring data to and from other devices without the CPU's involvement. Not all ISA devices are capable of Master mode.

The ÉlanSC300 and SC310 devices do not support ISA master devices because there is no $\overline{\text{MASTER}}$ pin on the chips. Therefore, only ISA devices that do not need Master mode to operate should be selected for system designs.

ISA REFRESH CYCLE ISSUE

The ÉlanSC300 and SC310 microcontrollers do not support refresh on the ISA bus. There is no $\overline{\text{REF}}$ signal.

The lack of the $\overline{\text{REF}}$ signal could potentially cause two problems: devices that need the refresh function to refresh their own memory, and devices that need the refresh function to know which ISA cycles to ignore.

The first case–devices that use the ISA $\overline{\text{REF}}$ signal to do their own memory refreshes—is rare (perhaps nonexistent). If a device has a built-in memory controller, it will likely also have refresh timing logic.

In the case of devices that use the $\overline{\text{REF}}$ signal to ignore the ISA cycle because it is a refresh, the lack of a $\overline{\text{REF}}$ signal is not a problem. The ÉlanSC300 and SC310 microcontrollers do not do refresh cycles on the ISA bus, so devices do not need to qualify $\overline{\text{MEMR}}$ with the $\overline{\text{REF}}$ signal. Any access the device decodes in its address range is a real access to it and should be acknowledged.

ZERO WAIT STATE TIMING ISSUE

The ISA Zero Wait State signal allows an ISA device to force commands to occur at zero wait states, thus speeding up the interface.

However, the ÉlanSC300 and SC310 microcontrollers require the Zero Wait State signal to be returned faster than the ISA specification allows. The ISA spec allows 40 nsec from the falling edge of the command to return the Zero Wait State signal. The ÉlanSC300 and SC310 devices require the signal returned in 20 nsec when the CPU is clocked at 33 MHz. If the Zero Wait State signal is missed by the ÉlanSC300 and SC310 devices, the ISA device will still work but will not be operating at zero wait speed.

Summary: Select ISA devices that will return Zero Wait State in time, or do not use the Zero Wait State feature.

ISA ADDRESS HOLD TIME ISSUE (B3 SILICON)

The ISA address signals (SA23:0) have no hold time from the Read command going inactive. This can be seen in the ÉlanSC300 and SC310 microcontroller data sheet in any of the ISA bus timing diagrams for the $\overline{\text{IOR}}$ and $\overline{\text{MEMR}}$ accesses to 8- and 16-bit devices.

This issue will not affect the data being read. The ÉlanSC300 and SC310 microcontrollers latch in the data from the data bus shortly before deasserting the command. This could potentially cause problems with ISA devices that have read destructive registers (i.e., registers that are cleared on reading). There is a small possibility that, when reading a register in an ISA device, the device will see an alternate register being read for a very short time at the end of the Read cycle. If this alternate register is read destructive, it could possibly be changed or cleared.

Note: *This is documented as Errata #EB41 which is corrected on Rev-B4 silicon of the* Élan*SC300 and* Élan*SC310 microcontrollers.*

$\overline{\text{LMEG}}$ ISSUE

On a standard ISA bus there are two sets of memory commands: $\overline{\text{MEMR}}$ and $\overline{\text{MEMW}}$ that toggle for all memory accesses and $\overline{\text{SMEMR}}$ and $\overline{\text{SMEMW}}$ that toggle only for memory accesses in the lowest one megabyte of memory (address SA23:20 are all zero).

The $\overline{\text{MEMR}}$ and $\overline{\text{MEMW}}$ signals that come out of the ÉlanSC300 and SC310 devices are $\overline{\text{MEMR}}$ and $\overline{\text{MEMW}}$, not $\overline{\text{SMEMR}}$ and $\overline{\text{SMEMW}}$. In Full ISA mode the $\overline{\text{LMEG}}$ (Low MEGabyte) signal is available, and it will assert for accesses in the low megabyte. $\overline{\text{LMEG}}$ can be qualified on the system board with $\overline{\text{MEMR}}$ and $\overline{\text{MEMW}}$ to generate $\overline{\text{SMEMR}}$ and $\overline{\text{SMEMW}}$ if they are necessary for the design.

There is an anomaly associated with the $\overline{\text{LMEG}}$ signal that makes it potentially unusable. When the MMS logic is used in the ÉlanSC300 and SC310 microcontrollers, the $\overline{\text{LMEG}}$ signal will assert on an MMS hit, even though external to the ÉlanSC300 and SC310 device the access is to memory above the lowest megabyte. This could result in the ISA device that uses $\overline{\text{SMEMR}}$ and $\overline{\text{SMEMW}}$ responding to cycles that are intended for another device.

Summary: $\overline{\text{LMEG}}$ should not be used to generate $\overline{\text{SMEMR}}$ or $\overline{\text{SMEMW}}$. Instead, the address lines SA23:20 should be decoded.