ANALOG DEVICES

Precision Analog Microcontroller, 12-Bit Analog I/O, ARM7TDMI MCU

Silicon Anomaly

ADuC7121

This anomaly list describes the known bugs, anomalies, and workarounds for the ADuC7121 MicroConverter® Revision C silicon.

First Line   ADuC7121
Second Line   C5V

Analog Devices, Inc., is committed, through future silicon revisions, to continuously improve silicon functionality. Analog Devices tries to ensure that these future silicon revisions remain compatible with your present software/systems by implementing the recommended workarounds outlined here.

ADuC7121 FUNCTIONALITY ISSUES

Silicon Revision Identifier	Kernel Revision Identifier	Chip Marking	Silicon Status	Anomaly Sheet	No. of Reported Anomalies
C	5V	All silicon branded C5	Release	Rev. A	5

FUNCTIONALITY ISSUES

Table 1. External IRQ When Configured as Level Sensitive [er001]

Background	There are six external interrupt sources on the ADuC7121. These can be configured as edge triggered (rising or falling) or level triggered (active high or active low).
Issue	When any of the external interrupt sources are configured as level triggered, either active high or active low, the external pin (IRQx) must remain at the active level until the program vectors to the interrupt vector handler for that external interrupt.
	If the external pin is activated, triggering an interrupt, but subsequently goes to an inactive level before the program vectors to the interrupt handler, the appropriate bit in the IRQSTA register for the external interrupt may not be set. This results in the interrupt handler not knowing which interrupt source caused the part to vector to the interrupt vector.
Workaround	Edge triggered interrupts do not have this issue. This issue will be addressed in a future revision of the silicon.
Related Issues	None.

Table 2. Disabling I²C Interface in Slave Mode When a Transfer Is in Progress [er002]

Background	The I2CSEN bit (Bit 0 in the I2CxSCTL register) enables/disables the I²C slave interface. The I2CSBUSY bit (Bit 6 in the I2CxSSTA register) indicates whether the I²C slave interface is busy.
Issue	If I²C slave mode is enabled (I2CxSCTL Bit 0 = 1) and a transfer is in progress with the master, do not clear I2CxSCTL Bit 0 to 0 to disable the I²C slave interface until the I²C busy bit, I2CSBUSY (Bit 6 of I2CxSSTA), is cleared.
	When I2CxSCTL Bit 0 is cleared to 0 and I2CSBUSY is still set, the ADuC7121 may drive the SDAx pins low indefinitely. When this condition occurs, the ADuC7121 does not release the SDAx pins unless a hardware reset condition occurs.
Workaround	When disabling I²C slave mode by writing to the I2CSEN bit (Bit 0 in the I2CxSCTL register), first set the I2CMEN bit (Bit 0 in the I2CxMCTL register) = 1 to enable master mode. Then disable the slave mode by clearing the I2CSEN bit. Finally, clear the I2CMEN bit.
Related Issues	None.

Table 3. Operation of SPI in Slave Mode [er003]

Background	When in SPI slave mode, the ADuC7121 expects the number of clock pulses from the master to be divisible by 8 when the chip select ($\overline{CS}$) pin is low. The internal bit shift counter within the ADuC7121 is not reset when the chip select pin is deasserted.
Issue	If the number of clocks from the master is not divisible by 8 when the chip select is active, incorrect data can be received or transmitted by the ADuC7121 because the bit shift counter will not be at 0 for future transfers. The internal bit shift counter for the transmit or receive buffers can only be reset by a hardware, software, or watchdog reset.
Workaround	Always ensure that the number of SPI clocks is divisible by 8 when the ADuC7121 chip select is active.
Related Issues	None.

Table **4. I²C Slave Not Releasing the Bus [er004]**

Background	When an I²C read request happens, if the Tx FIFO of the slave is empty, the slave must NACK the request from the master. Then it must release the bus, allowing the master to generate a STOP condition.
Issue	If the Tx FIFO of the slave is loaded with a byte with an MSB of 0, just on the rising edge of SCL for the ACK/NACK, the slave will pull the SDA low and hold the line until the device is reset.
Workaround	Make sure the Tx FIFO is always loaded on time by preloading the Tx FIFO in the preceding Rx interrupt.
Related Issues	None.

Table **5. I²C Clock Stretch Issue [er005]**

Background	Clock stretching is a feature that allows a device to halt the I²C bus temporarily by holding SCL low.
	The I2CxSCON register Bit 6 enables clock stretching in slave mode.
	The I2CxMCON register Bit 3 enables clock stretching in master mode.
Issue	Writing to I2CxSCON Bit 6 or to I2CxMCON Bit 3 on the rising edge of SCL can cause a glitch that can be interpreted by other devices as a real clock edge and can hang the bus.
Workaround	Do not enable clock stretching.
Related Issues	None.

SECTION 1. ADuC7121 FUNCTIONALITY ISSUES

Reference Number	Description	Status
er001	External IRQ when configured as level sensitive	Open
er002	Disabling I²C interface in slave mode when a transfer is in progress	Open
er003	Operation of SPI in slave mode	Open
er004	I²C slave not releasing the bus	Open
er005	I²C clock stretch issue	Open

I²C refers to a communications protocol originally developed by Philips Semiconductors (now NXP Semiconductors).

www.analog.com