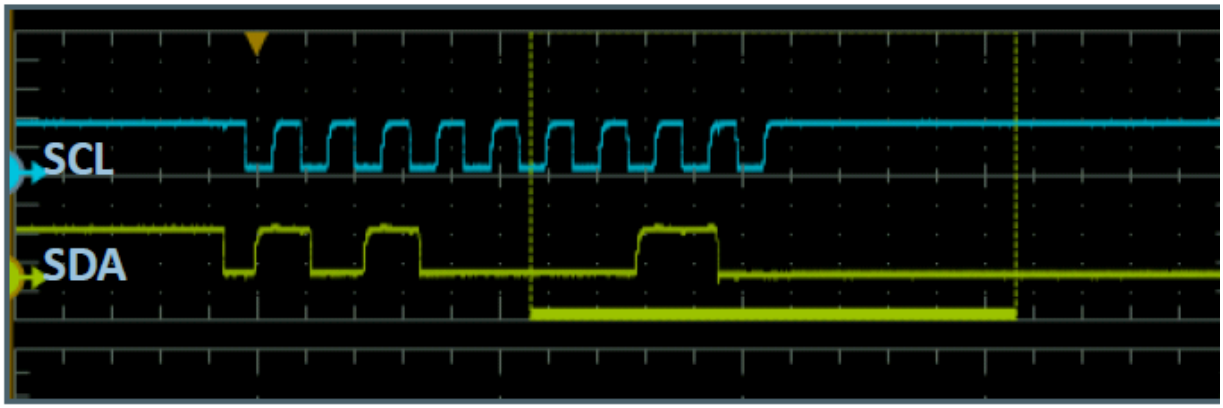
I2C Errata 1

A. The errata applies to ADuCM350 1p0, 1p1 and 1p2 silicon.

B. Anomaly: I2C bus hangs when configured as slave Tx

C. Description:



During a read from the master to slave, if the slave's FIFO is empty, the slave should NACK the master's request. Then it should release the bus allowing the master to generate a STOP condition. In the case of the slave's FIFO buffer being empty, if data is loaded when the ninth SCL clock is high, the slave generates an ACK to the master and the MSB of read data is sent on the following SCL positive edge. If the MSB is 0, the SDA line will be held low and the slave will wait for extra SCL clocks to transmit the remaining bits. When this happens, the I2C master detects a NACK and then generates a STOP condition. Master arbitration is lost due to the SDA line being indefinitely low.

D. Workarounds:

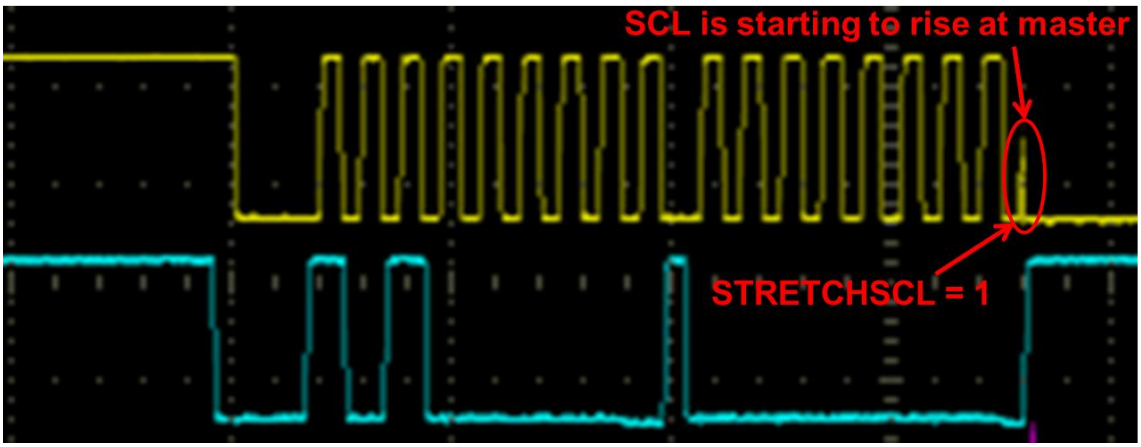
1. Placing valid data in the Slave Tx FIFO.
2. Set the BUS_CLR_EN bit in the I2CMCON register. If this bit is set, the master will initiate a Bus-Clear operation by sending up to 9 extra SCL cycles.
3. Set EARLYTXR bit in the I2CSCON register. Setting this bit enables a transmit request just after the positive edge on the SCL line during the read bit transmission.
4. Resetting the slave interface by disabling/enabling the slave.

I2C Errata 2

A. The errata applies to ADuCM350 1p0, 1p1 and 1p2 silicon.
B. Anomaly: I2C master fails to operate when the sum of the values in HIGH and LOW in the I2CDIV register has a value of less than 18 (0x12).
C. Description: In an attempt to minimize the power consumption of the I2C interface, it may be desirable to set the PCLK frequency to 1MHz and set the SCL frequency to 100kHz. Using the provided formula, this would lead to the following HIGH and LOW values in the I2CDIV register:
   HIGH = (REQD_HIGH_TIME / PCLK_PERIOD) – 2
   HIGH = ((10us / 2) / 1us) – 2
   HIGH = 3
   LOW = (REQD_LOW_TIME / PCLK_PERIOD) – 1
   LOW = ((10us / 2) / 1us) – 1
   LOW = 4
   When these values are programmed, the I2C master will not function.
D. Workarounds:
   1. Use the adi_I2C_SetMasterClock() function to set the SCL frequency. Ensure to check for a valid return code.
   2. Do not program values to the HIGH and LOW fields in the I2CDIV register which give a sum of less than 18 (0x12).

I2C Errata 3

A. The errata applies to ADuCM350 1p0, 1p1 and 1p2 silicon.
B. Anomaly: I2C slave hangs the I2C bus when clock stretching is enabled.
C. Description: Clock stretching is an I2C feature that allows a slave device to temporarily pause communication by holding SCL low (for example, a slave device receives a READ request from a Master but, its Transmit FIFO is empty).
   Clock stretching is enabled setting the STRETCHSCL bit in the I2CSCON register to 1. If this is done on the rising edge of SCL a glitch can occur. Other devices might interpret this as a real clock pulse and it could cause the bus to hang.



D. Workaround: It is recommended that the clock stretching feature is not used.