

HW#5 – I2C Master Controller: Simulation, Submission, and Demo Information

This document outlines the required testing method, submission details, and demo information for the I2C master controller of HW#5. *Please be sure to read it carefully.*

SUBMISSION DEADLINE: Your completed problem is **due by 4:00 p.m. on Friday, November 18**. You will have both a hardcopy and electronic submission for this problem. In addition you will have a demo with Sumedh Attarde, Mithila Paryekar or George Faldamis on Friday 11/18, Monday 11/21, Tuesday 11/22 or Wednesday 11/23. *See below for details.*

Required Simulations: After modeling the I2C master controller in VHDL using the Quartus II CAD tool, you are to simulate your VHDL specification on a set of input sequences using the guidelines provided below. During the demo, you will be asked to show the output of these input sequences as well as a few other input sequences supplied by the TA. Thoroughly test your design in advance, so that all input patterns applied by the TA work correctly.

Clock Periods and Initialization Requirements. Note the clock period of `clk_hi` for your simulation should be 10ns. The following gives guidelines for the ‘required simulations’ below. However, see “Note” below: you still need to handle other cases that we may test as well!

At time $t=0$, assume the master is *inactive* and in the appropriate initial state, with all signals initialized to *correct initial values*. See p. 6 of Handout #29 and elsewhere for details on initial signal values. Also, at $t=0$, assume that `clk_hi` has a rising clock edge, starting the first local clock cycle.

Next, assume that the master is activated (i.e. wins arbitration) immediately afterwards. In particular, in the middle of the first local clock period (at $t=2\text{ns}$), the local counter input “`arb_win`” is asserted to 1 (for 1 local clock cycle), which begins the activation of the master controller. Follow p. 6 of the Handout #29 for details on the steps and timing of the resulting initialization.

Finally, the master should broadcast a START symbol as soon as possible, immediately followed by a slave’s address, for each of the simulations below.

The *SCL clock period* is determined as follows: assume that the local counter always counts 5 local clock cycles between a transition on its input “`cnt_enable`” to a transition on its “`SCL_toggle`” output.

IMPORTANT NOTE: Although you should follow the above guidelines for the required simulations below, you *still* must be able to correctly handle cases where the master *does not immediately get activated*, and other activity happens first on the bus.

You are asked to create the following input sequences for the following cases for master as transmitter:

(i) After receiving the signal to initiate transfer, the master sends a START symbol followed by the address “0010110” and the 8th bit (0) implying a write. After receiving ACK from a slave, the master sends a ***single byte of data***, which is immediately followed by the ACK and STOP symbols.

(ii) After receiving the signal to initiate transfer, the master sends a START symbol followed by the address “1011101” and the 8th bit (0) implying a write. After receiving ACK from a slave, the master then sends ***two or more bytes of data***. Each byte of data is acknowledged accordingly and the entire transaction is terminated with STOP symbol.

(iii) After receiving the signal to initiate transfer, the master sends a START symbol followed by the address “0010110” and the 8th bit (0) implying a write. After receiving ACK from a slave, the master sends a ***single byte of data***. Assume that the slave unit needs time to process the 3rd data bit and stretches the clock for just the following bit. At the end of the byte, the normal sequence of ACK and STOP symbols is observed. Model the clock stretching scenario appropriately and test if the master controller FSM handles it properly.

You are asked to create the following input sequences for the following cases for master as receiver (i.e. master in read mode):

(iv) After receiving the signal to initiate transfer, the master sends a START symbol followed by the address "1011011" and the 8th bit (1) indicates read. After receiving ACK from a slave, the master receives ***1 byte of data***. The transaction is then terminated appropriately.

(v) After receiving the signal to initiate transfer, the master sends a START symbol followed by the address "1101110" and the 8th bit (1) indicates read. After receiving ACK from a slave, the master receives ***2 bytes of data***. After receiving the second data byte, the master should terminate the transaction.

(vi) Design your own input sequence which test something unique in your design not previously test in (i)-(iv), for either master as receiver or transmitter. Do not just vary the address or the data bytes being sent but instead, come up with a creative input case to test something meaningful and the correctness of your design.

Remember, you are responsible for ALL assumptions, clarifications, and information posted on Handout #29d which is regularly updated online, as well as Handouts #29, 29a, 29b, 29c and 29d, and other required web handouts on I2C, as well as announced clarifications made in class, so be sure to read all postings carefully.

What to turn in?: Turn in the following written and electronic parts for HW#5:

- (a) The symbolic state diagram of the Moore FSM, neatly labeled, either handwritten or using a graphical editor;
IMPORTANT NOTE: If your drawing is messy, hard to read, or poorly labeled, it will be difficult for the TA's to follow, and therefore points might be deducted. The state diagram should be drawn as clear as possible (i.e. as if it were to be presented for management at a design review). Also, you should **label small groups of states (with a dotted box around each small group) to highlight which portion of your state diagram is detecting a 0 data input, a 1 data input, an address, an ACK symbol, a START symbol, and a STOP symbol.** That is, annotate small portions of your FSM diagram to clarify what part of the protocol is being handled by which states.
- (b) Printout of VHDL code for your Moore FSM (with comments inserted to clearly document the different portions of the state diagram, see 'important note' of (a) above);
- (c) Printout of waveforms that result from your simulations for your six test sequences (i)-(vi) above, as well as text details on each simulation (which data bytes used, etc.);
- (d) A short paragraph or two explaining your design experiences, testing methods, assumptions and challenges.
- (e) ****Electronic copies of your VHDL code and waveforms in a compressed .tar or .zip file named "CSEE4823_prob1_last_name_last_name" (fill in last_name with the last name of each group member). Email the .tar or .zip file to Sumedh Attarde at sumedh256@gmail.com with the mail subject at "CSEE4823 HW5 submission". Please be sure that all files needed to correctly simulate your FSM are included in the .tar or .zip file.****

Where to turn in the hardcopy parts of the assignment? (parts (a)-(d)). Hand these in either at start of class to Prof. Nowick on Thursday November 17, or under his office door before 4pm on Friday November 18.

Where to send the electronic part of the assignment? (part (e)) See above: mail to TA Sumedh Attarde, also by 4pm on Friday November 18.

Demo sign-up procedure: You must reserve a 25-minute demo time with a TA. An online 'doodle' URL sign-up will be made available shortly. *All group members must be present at the demo.* The demo will take place in the Embedded Systems Lab (1235 Mudd) on Friday November 18, Monday November 21, Tuesday November 22 and Wednesday November 23. The exact time slots will be disclosed shortly.

What to bring to the demo?: Bring a copy of your written parts (parts (a)-(d)) above. Of course, these written parts must already be handed in to Prof. Nowick by 4pm on Friday November 18.