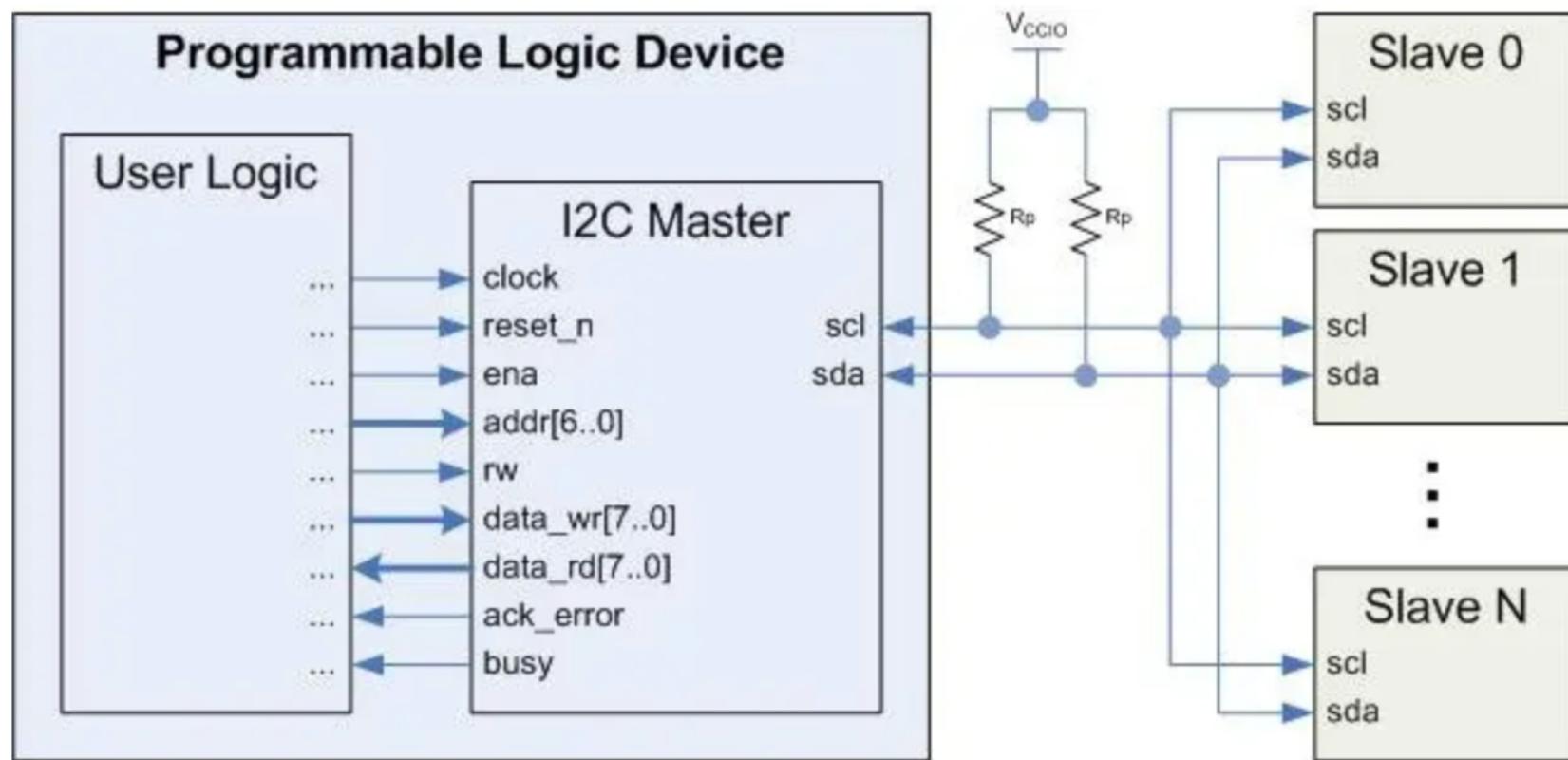


Lab 2 FSM/ASM for I2C

K. Chanon

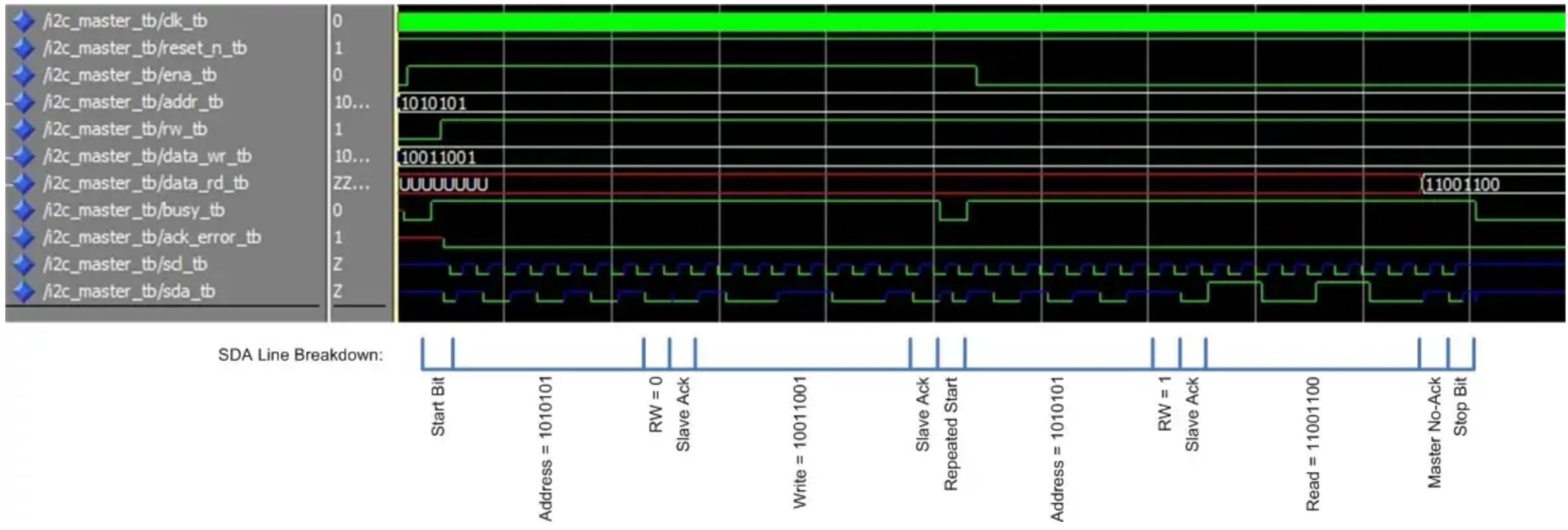
I2C: Overview



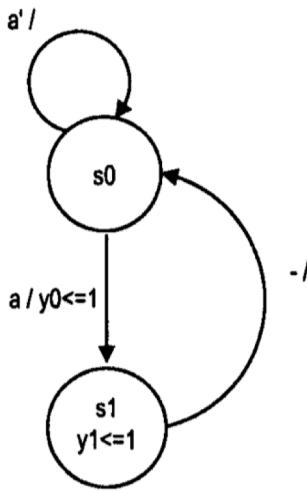
Port Description

Port	Width	Mode	Data Type	Interface	Description
clk	1	in	standard logic	user logic	System clock.
reset_n	1	in	standard logic	user logic	Asynchronous active low reset.
ena	1	in	standard logic	user logic	0: no transaction is initiated. 1: latches in addr, rw, and data_wr to initiate a transaction. If ena is high at the conclusion of a transaction (i.e. when busy goes low) then a new address, read/write command, and data are latched in to continue the transaction.
addr	7	in	standard logic vector	user logic	Address of target slave.
rw	1	in	standard logic	user logic	0: write command. 1: read command.
data_wr	8	in	standard logic vector	user logic	Data to transmit if rw = 0 (write).
data_rd	8	out	standard logic vector	user logic	Data received if rw = 1 (read).
busy	1	out	standard logic	user logic	0: I2C master is idle and last read data is available on data_rd. 1: command has been latched in and transaction is in progress.
ack_error	1	buffer	standard logic	user logic	0: no acknowledge errors. 1: at least one acknowledge error occurred during the transaction. ack_error clears itself at the beginning of each transaction.
sda	1	inout	standard logic	slave device(s)	Serial data line of I2C bus.
scl	1	inout	standard logic	slave device(s)	Serial clock line of I2C bus.

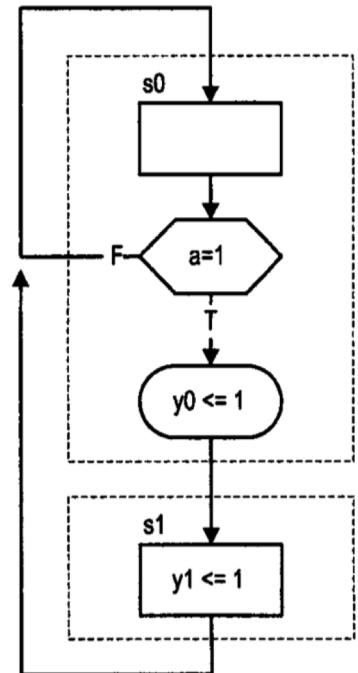
Example Transaction



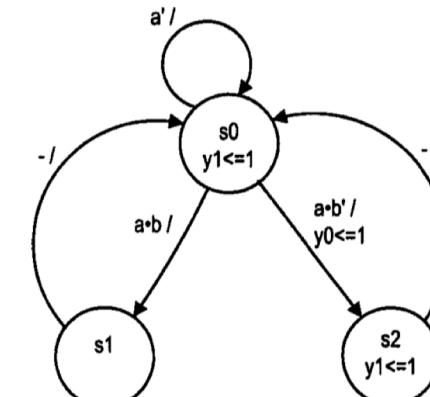
FSM vs ASM



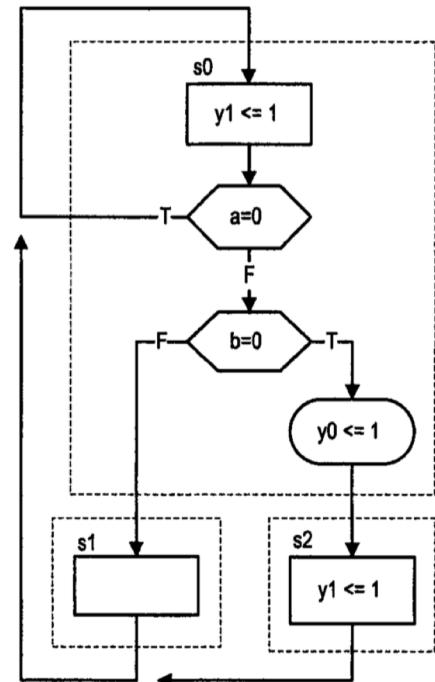
(a)



(b)



(a)



(b)

Lab Objective

- Design the user logic program which send and receive the data to/from i2c using ASM chart.
- Write the code from your design and test it with VHDL testbench.
- You need to design the condition which compatible with the i2c core.