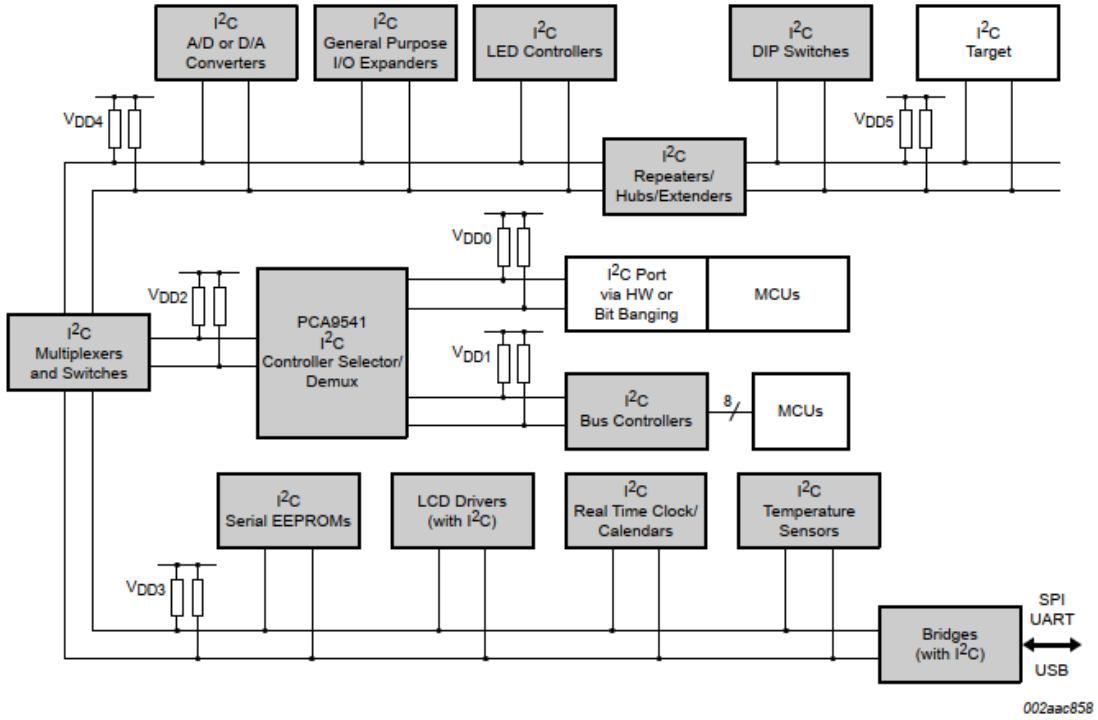
Inter-Integrated Circuit (I^2C)
Verilog Implementation

Outline

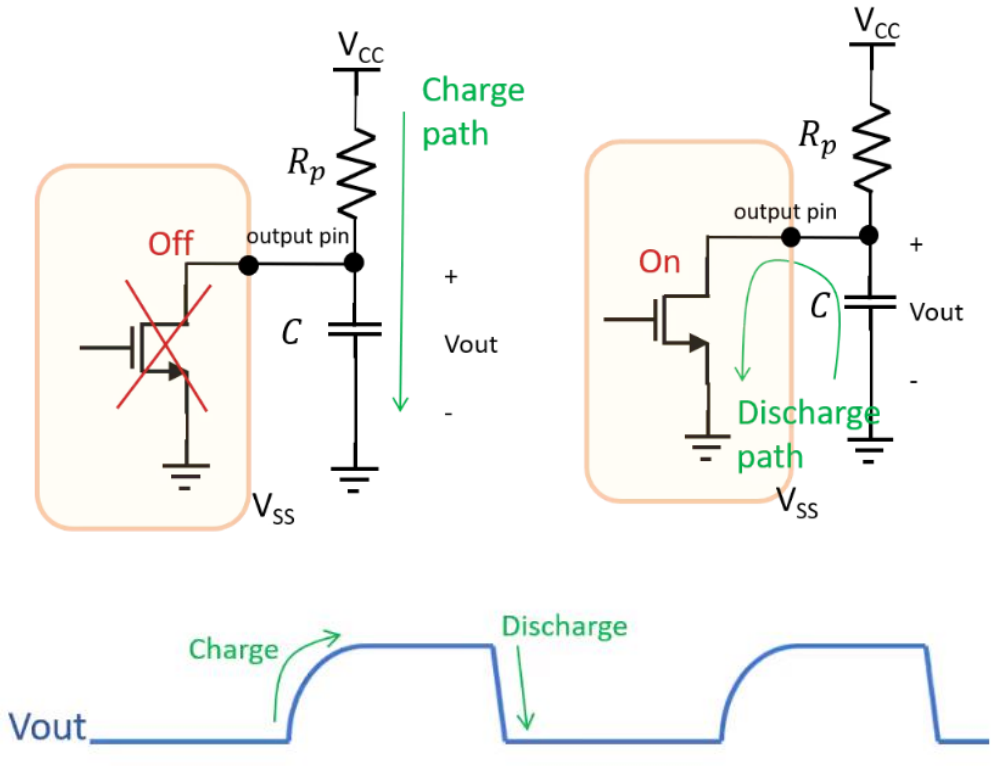
- IIC Introduction
- Data transmit format
- Implementation scope
- Design overview
- FSM
- Operation condition

About Inter-Integrated Circuit (I^2C)

- 1982, Philips. Now **NXP** Semiconductors
- Refer to I2C-bus specification and user manual, 2021
- Connect devices by two bus lines
  - Serial clock line (SCL)
  - Serial data line (SDA)
- Up to 100 kbits/s, standard-mode



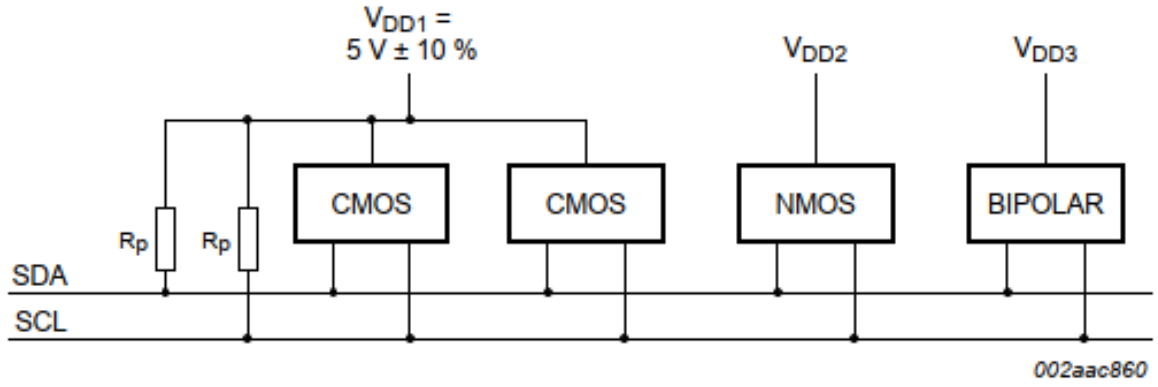
The Open Drain Output Pins



https://microchipsupport.force.com/s/article/Termination-of-Open-Drain-Output-Pins

The wired-AND connection

- SDA and SCL driven by pull-up resistor.
  - Bus line HIGH when it's free.
- Each device connect to an open-drain for the Wired-AND function
  - Wire-AND: LOW if any device put the bus low, otherwise HIGH



Transmit format – Start/ Stop

- Start and stop condition
  - Start condition: SCL high, SDA negedge
  - Stop condition: SCL high, SDA posedge
- Generate by the controller (Master)

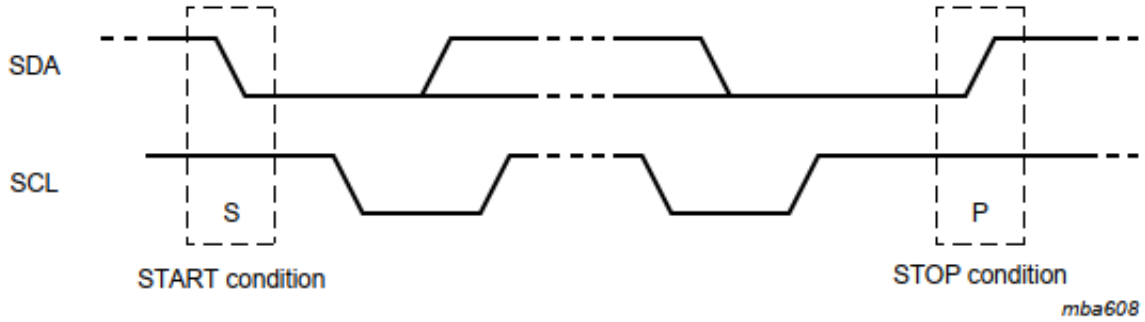


Figure 5. START and STOP conditions

Transmit format - Data validity

- SDA (data bus) must stable during SCL HIGH
  - SDA can only change when SCL LOW

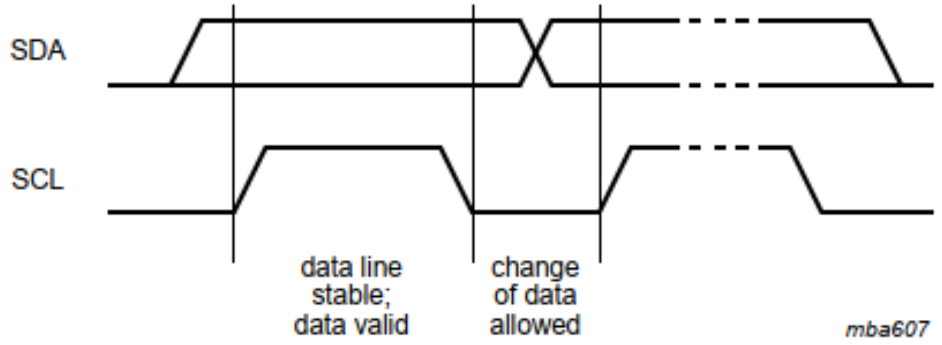


Figure 4. Bit transfer on the I^2C-bus

Transmit – Byte Format

- 1-byte data + acknowledge bit
  - Acknowledge send by data receiver
    - 0: Acknowledge (ACK), byte data correctly received
    - 1: Not acknowledge (NACK), (e.g. no receiver, receiver busy,…)

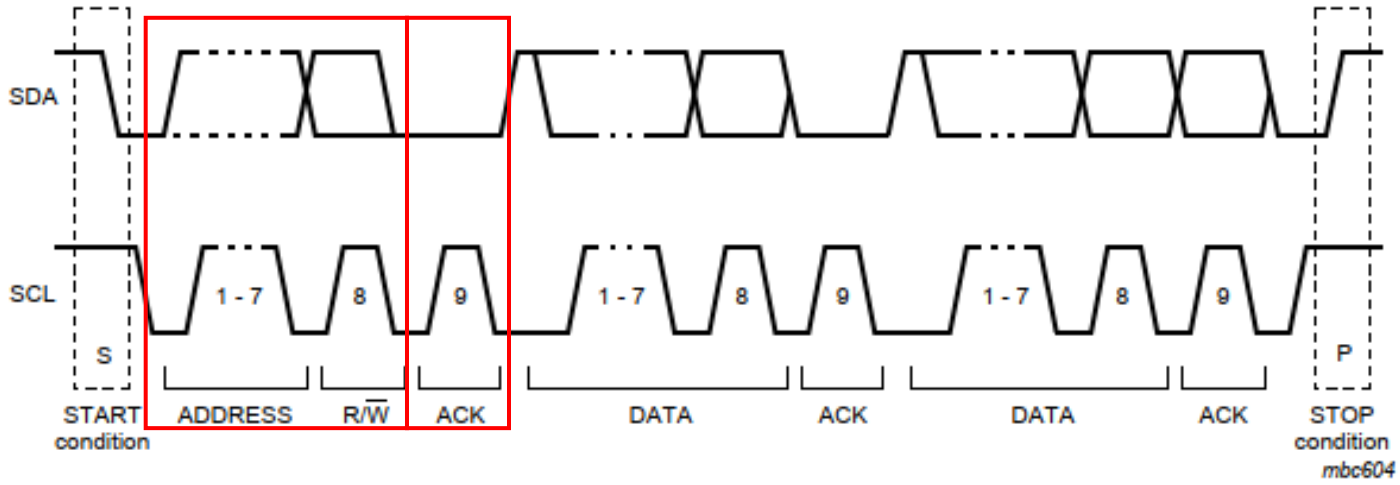


Figure 9. A complete data transfer

Transmit format

- First byte: device address (7-bit) + r/w operation(1-bit)
  - 0: Write (to device)
  - 1: Read (from device)

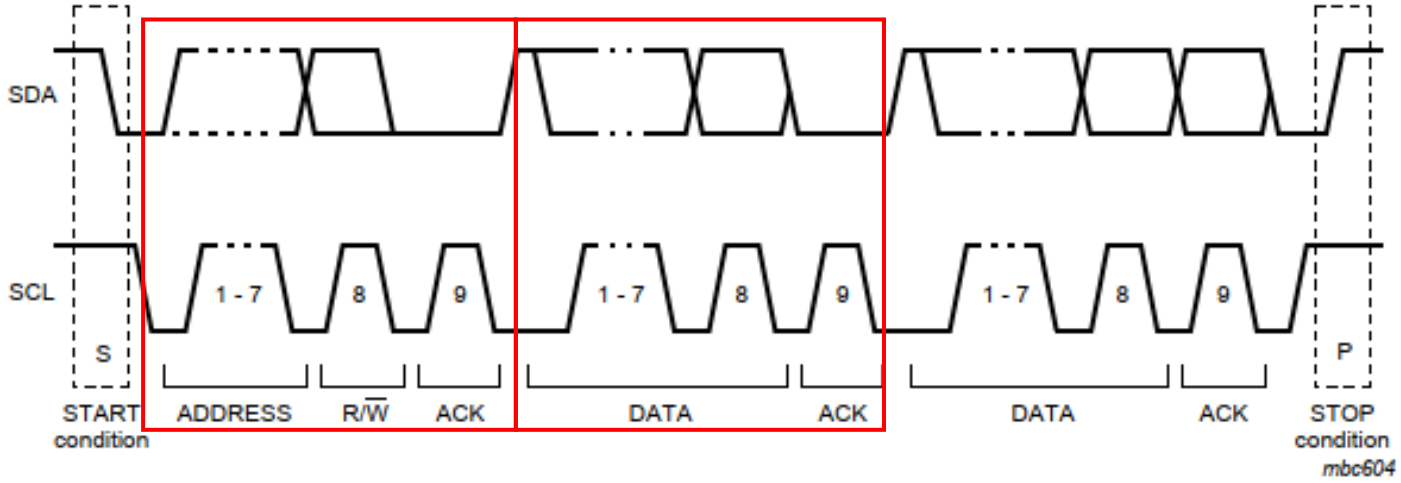


Figure 9. A complete data transfer

Operation

- Clock stretching
  - Only after acknowledge bit

3.1.9 Clock stretching

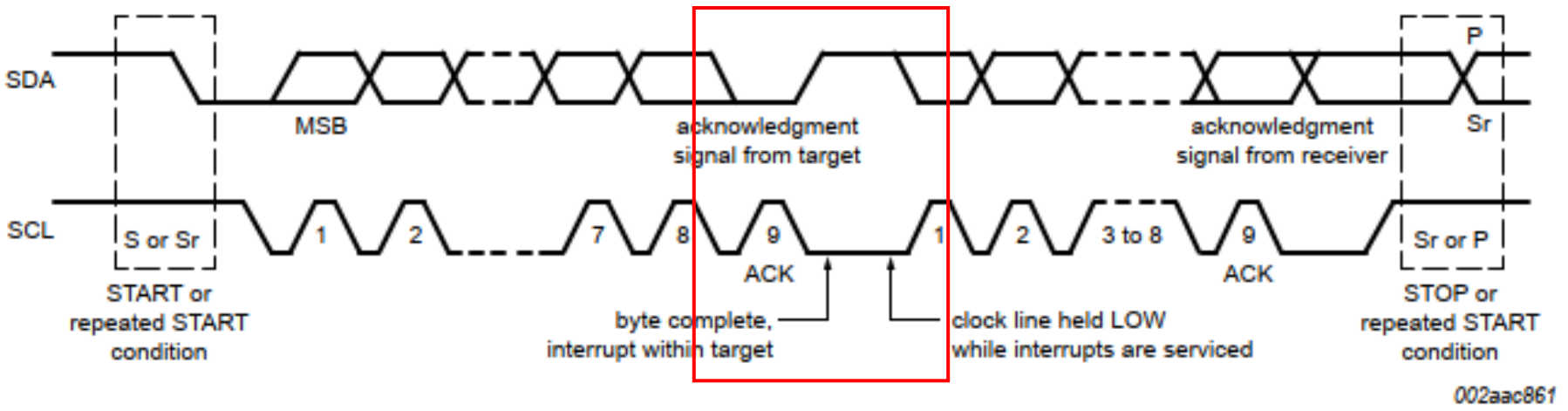


Figure 6. Data transfer on the I^2C-bus

Implementation Scope

- Single controller
  - Start/ Stop
  - Ack
  - Clock stretching
  - 7-bit address

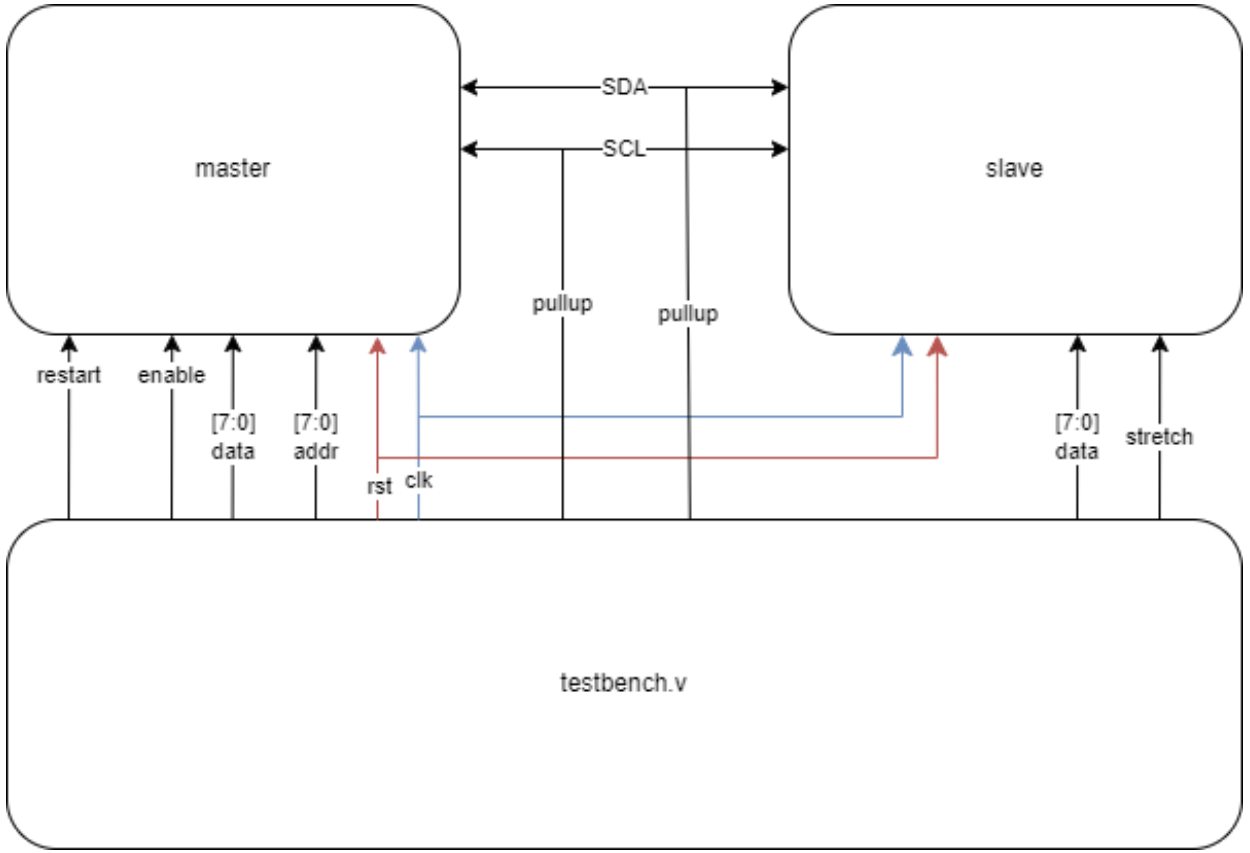
Table 3. Applicability of I²C-bus protocol features

M = mandatory; O = optional; n/a = not applicable.

Feature	Configuration		
	Single controller	Multi-controller	Target[1]
START condition	M	M	M
STOP condition	M	M	M
Acknowledge	M	M	M
Synchronization	n/a	M	n/a
Arbitration	n/a	M	n/a
Clock stretching	O[2]	O[2]	O
7-bit target address	M	M	M
10-bit target address	O	O	O
General Call address	O	O	O
Software Reset	O	O	O
START byte	n/a	O[3]	n/a

Design overview

- System clock scl
- Asynchronous reset rst
- Master
  - Enable: Idle state -> start, if high
  - Restart: Regenerate start signal, if high
  - Data: Write data
  - Addr: Write address (+1 w/r bit)
- Slave
  - Data: Read data
  - Stretch: Stretch the SCL bus after ack



Bidirectional bus control

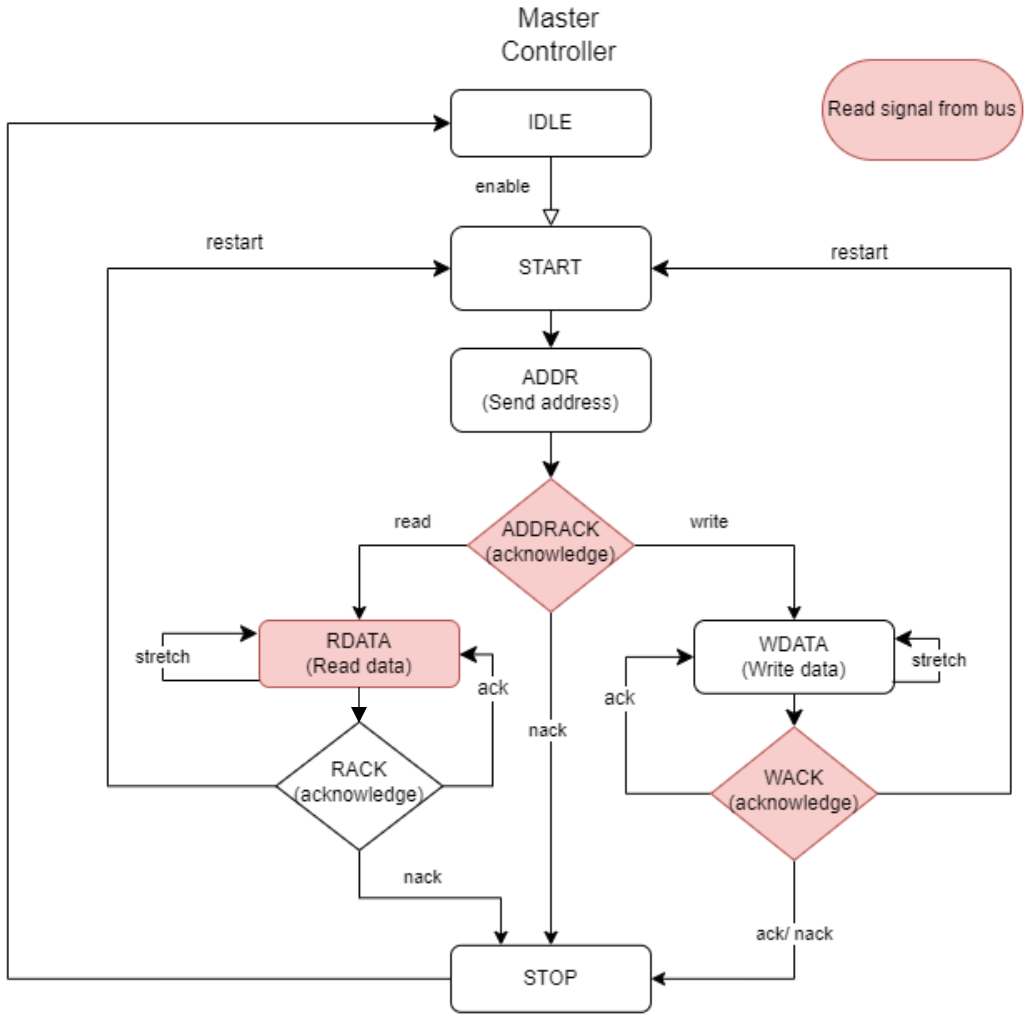
SCL/ SDA bus line truth table

Enable	Output value	out
0	0	1’bz
0	1	1’bz
1	0	0
1	1	1’bz

```
//I2c bidirectional bus control
assign i2c_scl = i2c_scl_enable & (!i2c_clk )?  0 : 1'bz;
assign i2c_sda = wen & (!sda_out) ? 0 : 1'bz;
```

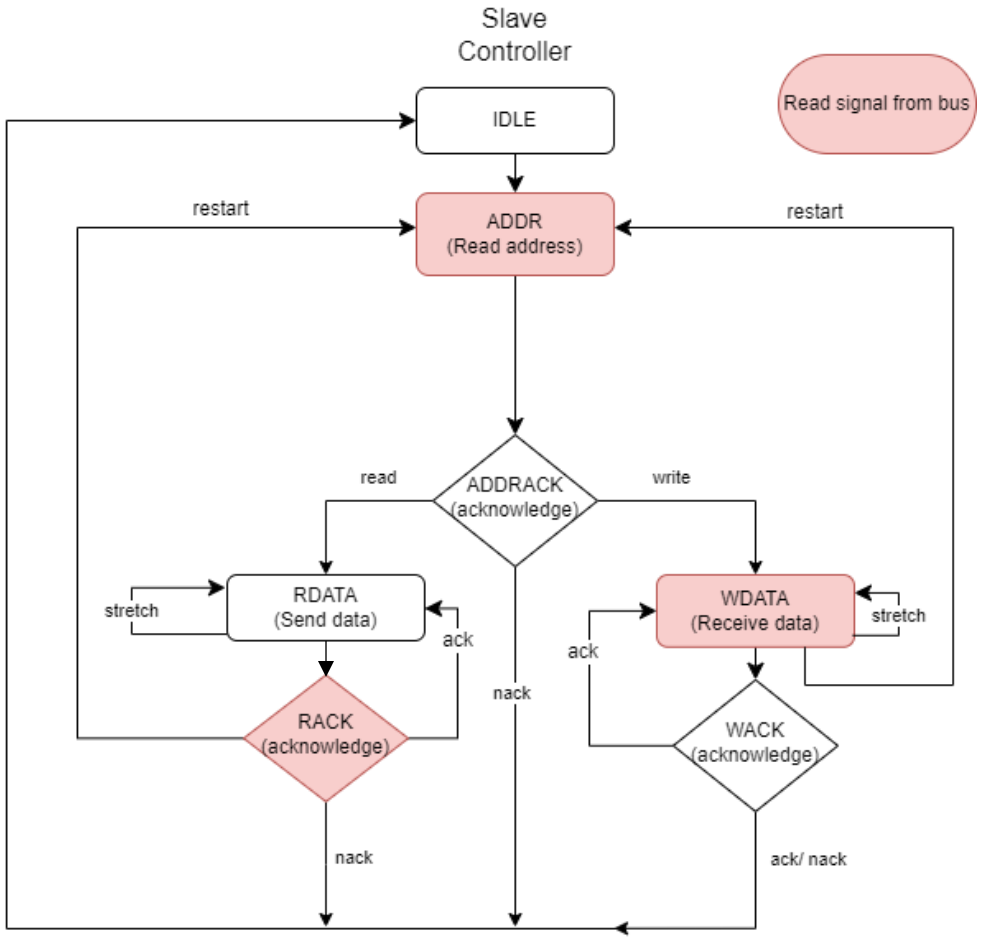
FSM- Master controller

1. IDLE
2. START: generate start signal
3. ADDR: Write address an r/w bit
4. ADDRACK: Check the response from device
5. WDATA: Write byte data
6. WACK: Check the response
7. RDATA: Read byte data
8. RWACK: Respond to device (ACK: continue, NACK stop transfer)
9. STOP: Generate stop signal



FSM- Slave controller

1. IDLE
2. ADDR: Read address from the bus
3. ADDRACK: Respond to controller
4. WDATA: Read data from the bus
5. WACK: Respond to controller
6. RDATA: Write to the bus
7. RWACK: Response from controller (ACK: continue, NACK stop transfer)



Operation

- Start -> write 1 byte -> ack/ nack -> stop

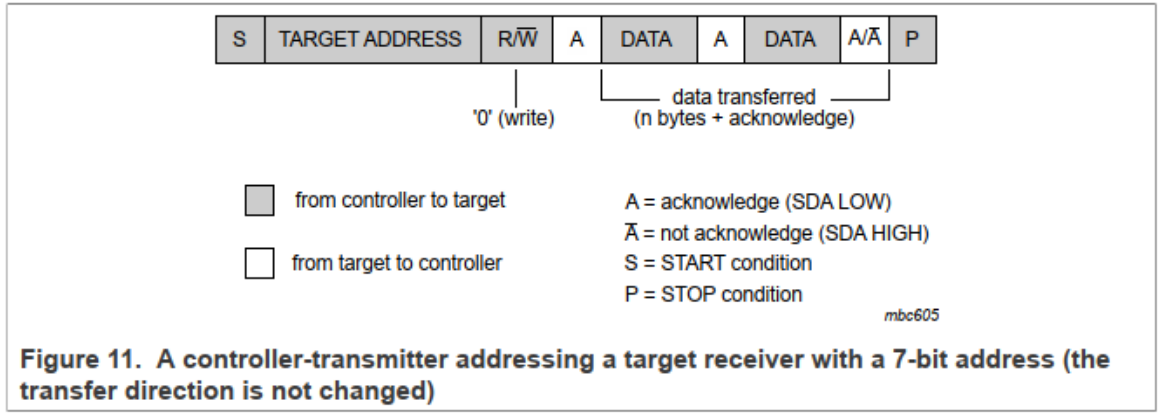
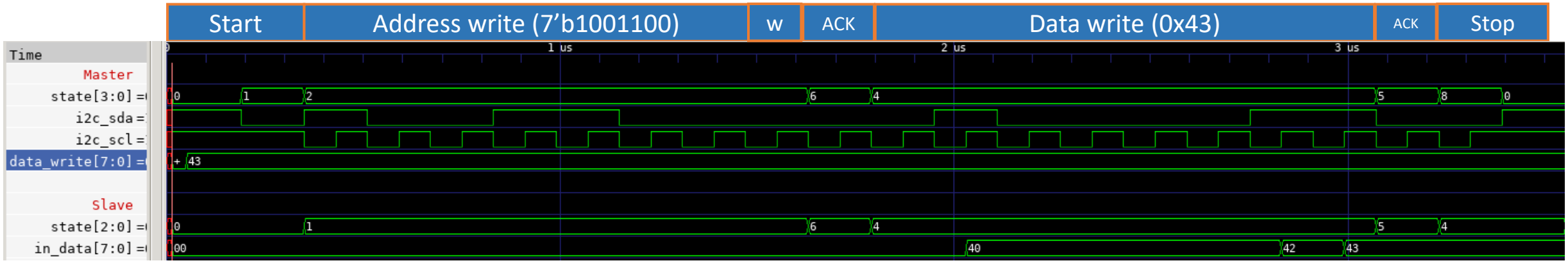


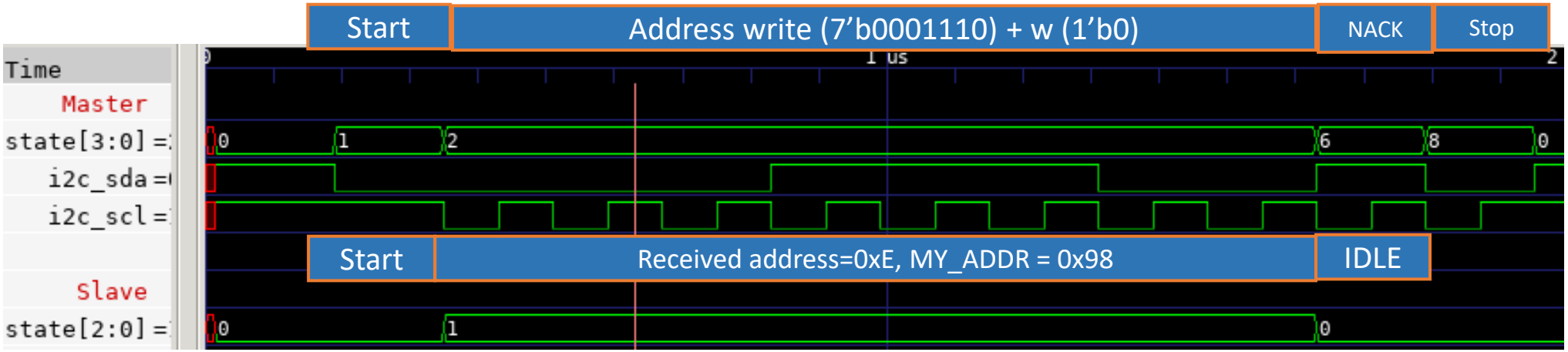
Figure 11. A controller-transmitter addressing a target receiver with a 7-bit address (the transfer direction is not changed)

Write 1 byte then stop



Operation

- Address NACK
- Address != Device address
  - Slave device to idle state



Operation

- Read byte -> nack -> stop

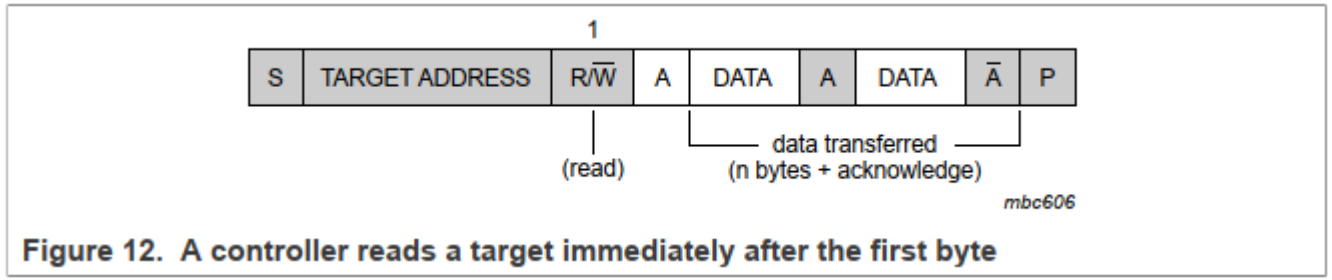
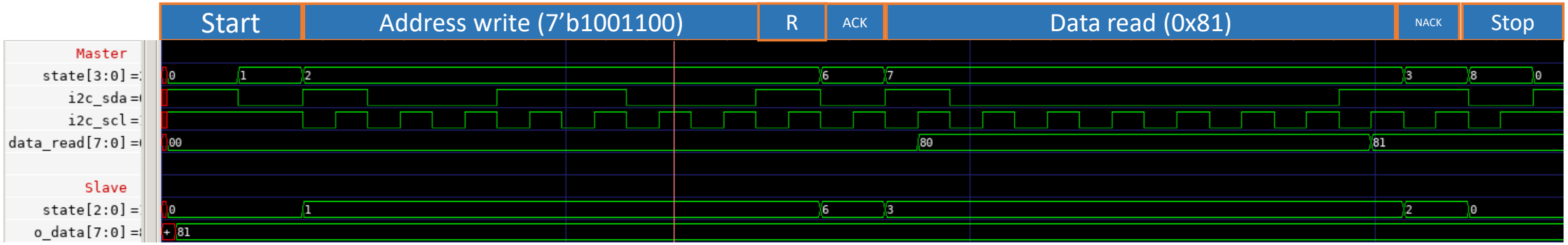


Figure 12. A controller reads a target immediately after the first byte

Read 1 byte then stop



Operation

- Start -> r/w 1 byte -> restart

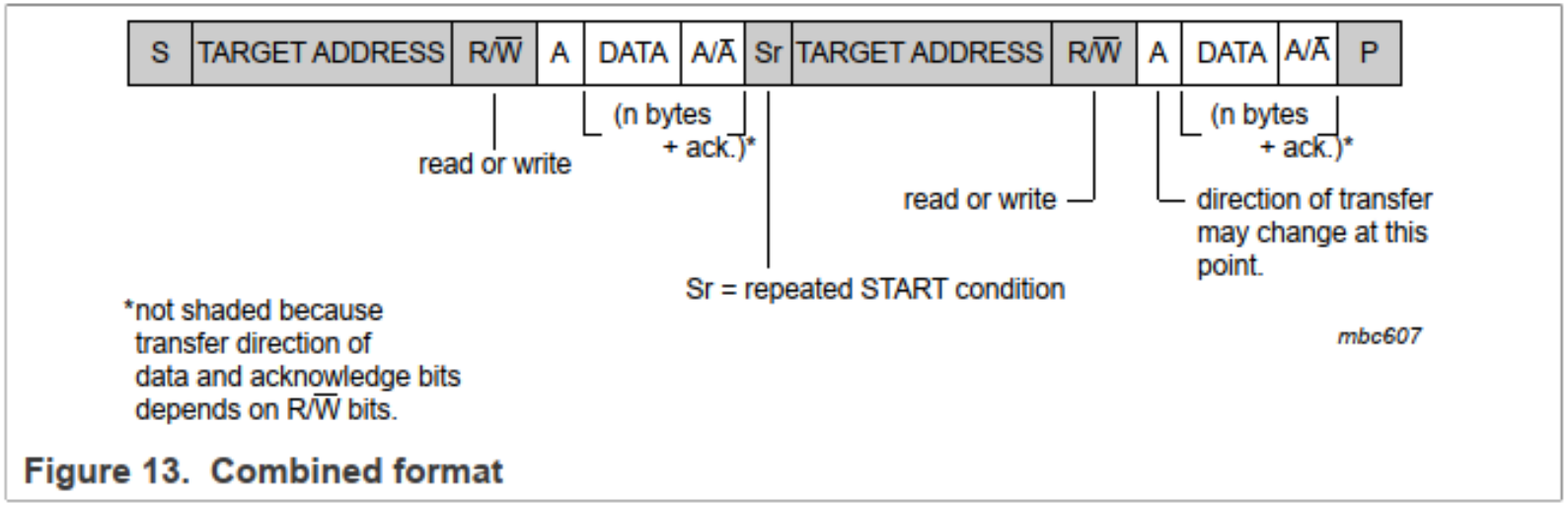
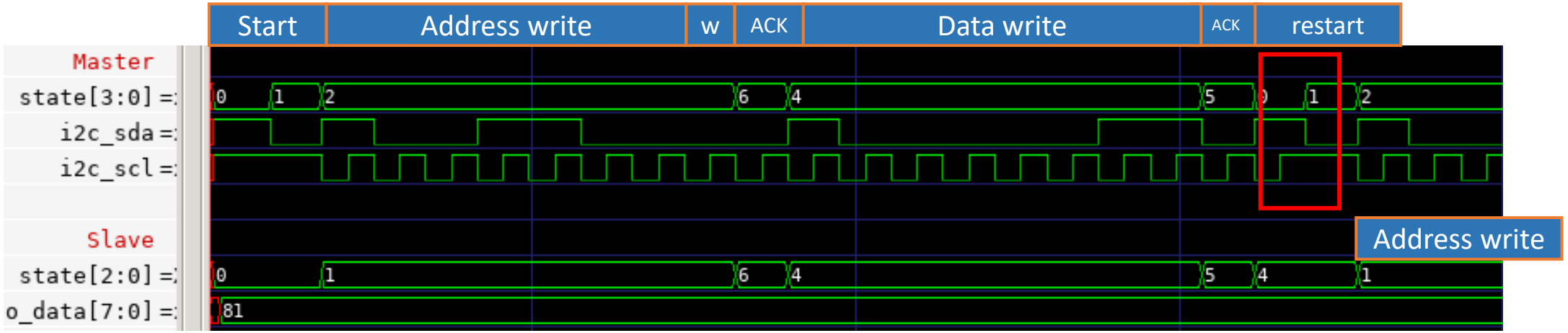
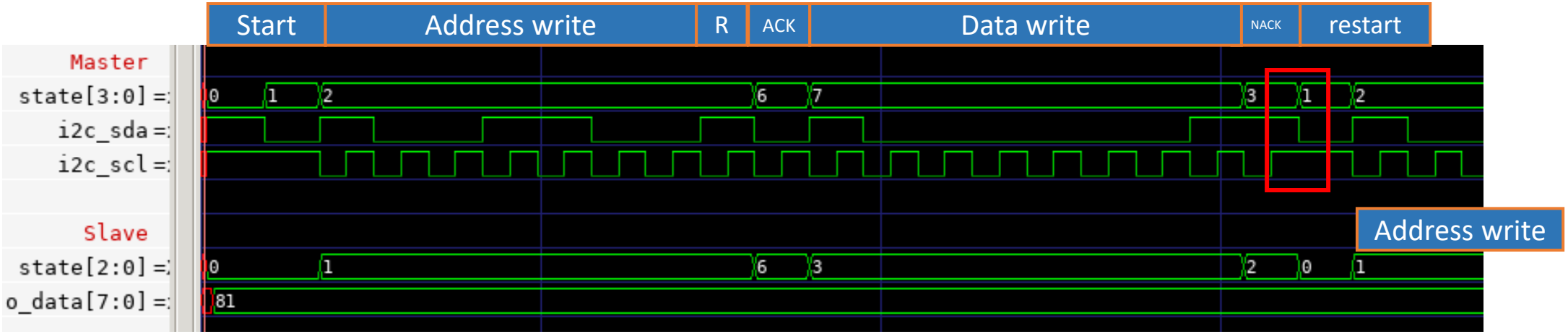


Figure 13. Combined format

Write 1 byte then restart

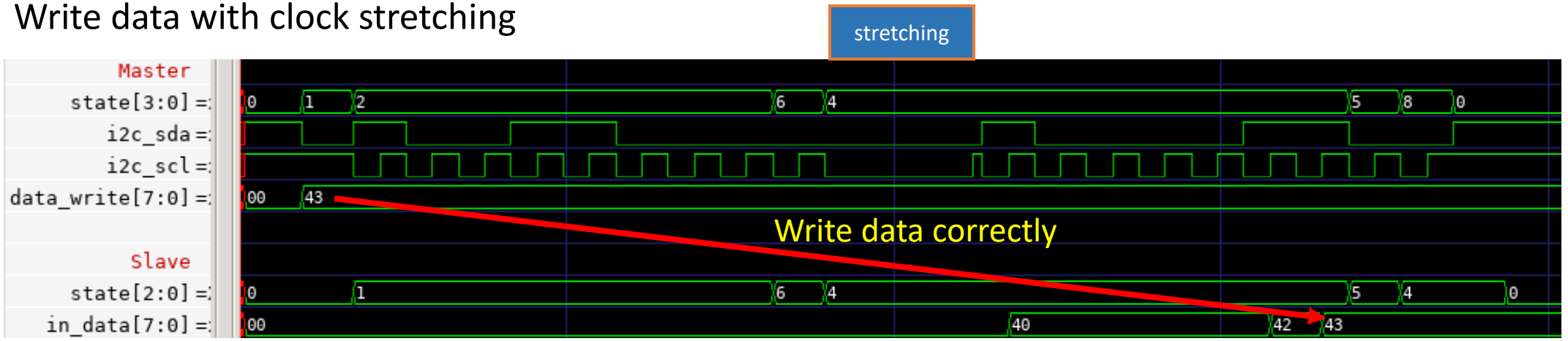


Read 1 byte then restart



Clock Stretching

Write data with clock stretching



Read data with clock stretching