

Bang (Max) Nguyen - Hoang Le

Part 0: Read material

Part 1a:

Serial Output

115200 baud ▾

Hyperload Flash Select Browse

Write Serial Data Here ... CR NL

```
-----  
peripherals_init(): Low level startup  
SD card mounted successfully  
  
I2C slave detected at address: 0x38  
I2C slave detected at address: 0x64  
I2C slave detected at address: 0x72  
  
entry_point(): Entering main()
```

Part 1b:

Output

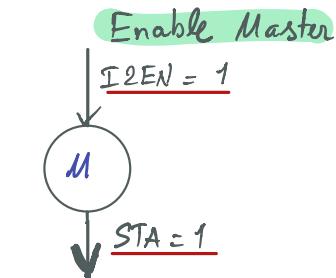
DTR RTS Options COM5 ▾

Write Serial Data Here ... CR NL

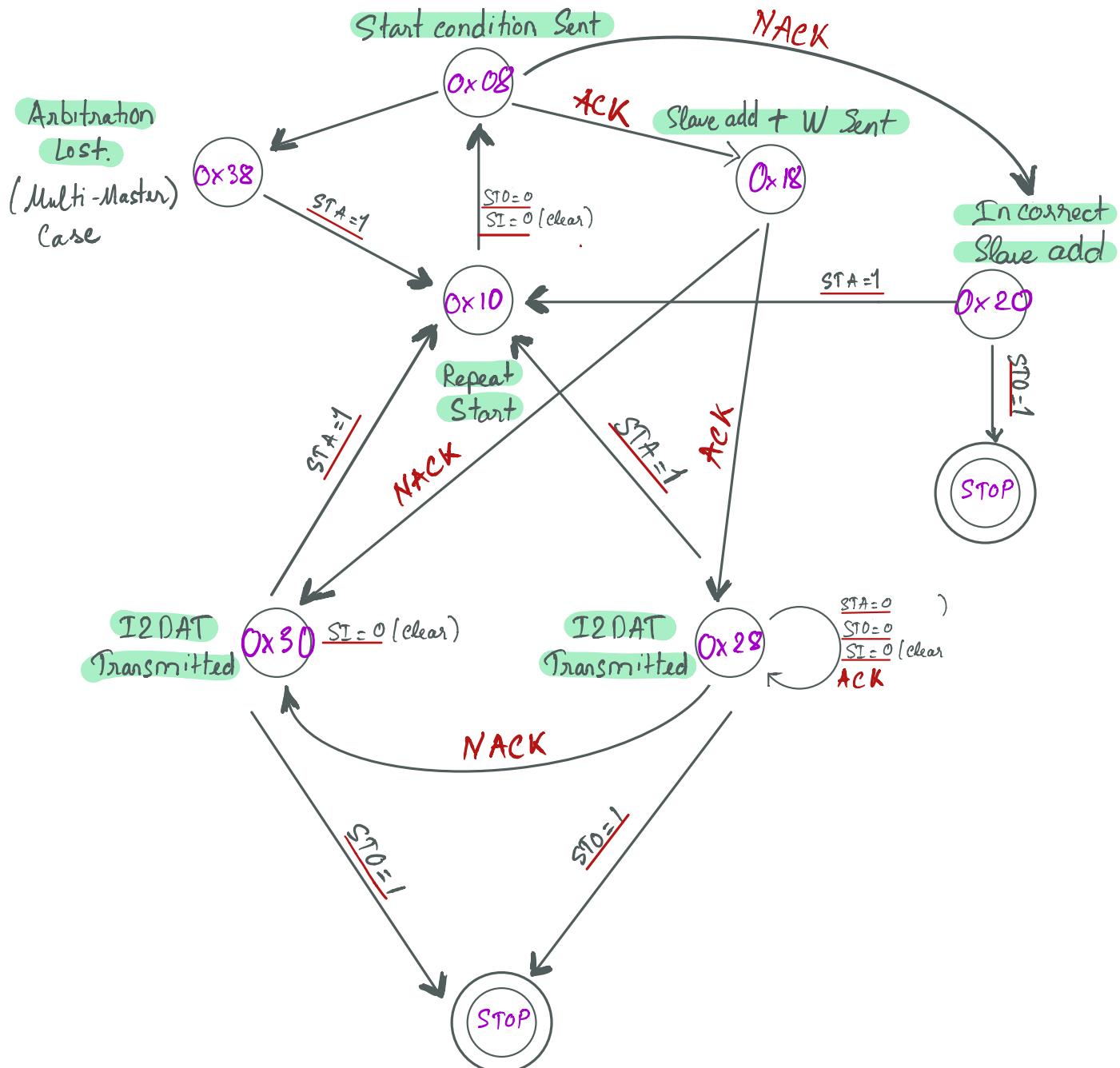
```
-----  
peripherals_init(): Low level startup  
SD card mounted successfully  
  
I2C:Starting transfer with device address: 0x02  
I2C: HW State: 0x08  
I2C: HW State: 0x20  
I2C: Finished with state: 0x20  
I2C:Starting transfer with device address: 0x04  
I2C: HW State: 0x08  
I2C: HW State: 0x20  
I2C: Finished with state: 0x20  
I2C:Starting transfer with device address: 0x06  
I2C: HW State: 0x08  
I2C: HW State: 0x20  
I2C: Finished with state: 0x20  
I2C:Starting transfer with device address: 0x08  
I2C: HW State: 0x08  
I2C: HW State: 0x20  
I2C: Finished with state: 0x20  
I2C:Starting transfer with device address: 0x0A  
I2C: HW State: 0x08  
I2C: HW State: 0x20  
I2C: Finished with state: 0x20  
I2C:Starting transfer with device address: 0x0C
```

I²C CONSET Master

[6]	[5]	[4]	[3]	[2]
I ² EN	STA	STO	SI	AA
1	X



Master Transmit



④ When Data is loaded (clear SI)

↳ I²DAT Load:

Slave address + R/W bit

↳ I²CONCLR

SIC = 1

Grey color: Input for next State

Red color: ACK or NACK

Slave Receive → Master

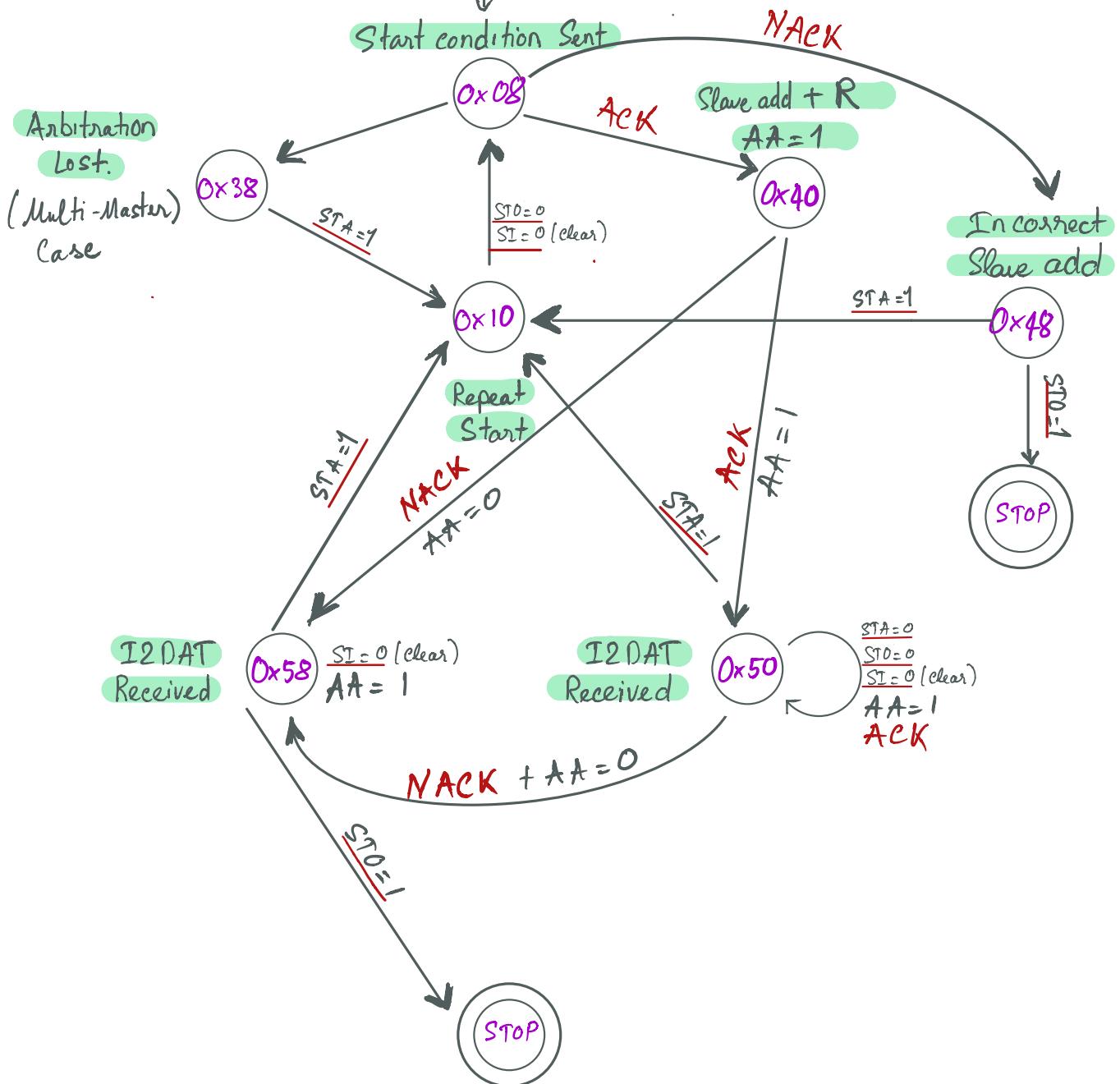
I2C CONSET Master

[s]	[s]	[4]	[3]	[2]
I2EN	STA	STO	SI	AA
1	...	---	---	X

Enable Master

I2EN = 1

Master Receiver

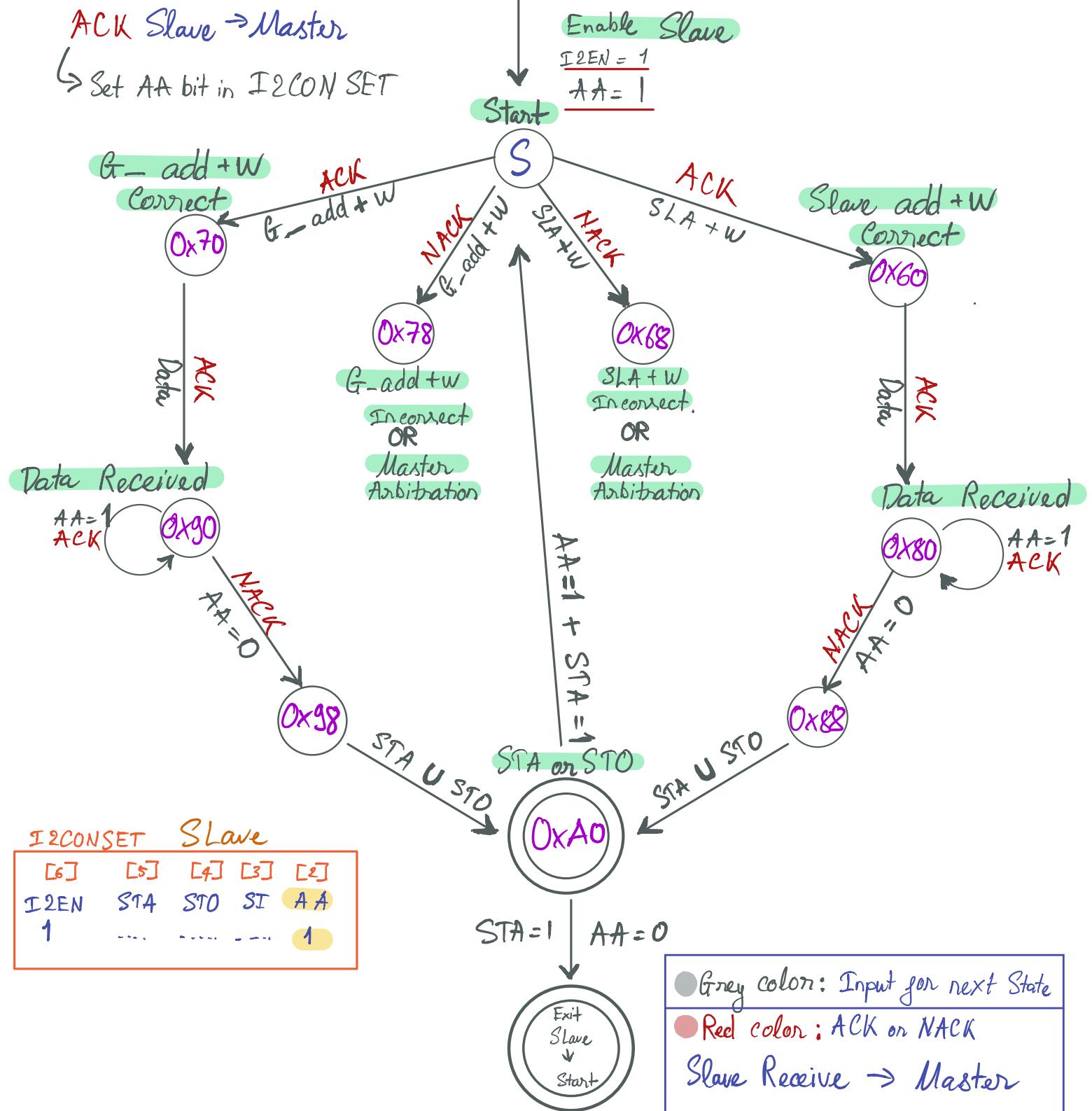


Grey color: Input for next State

Red color: ACK or NACK

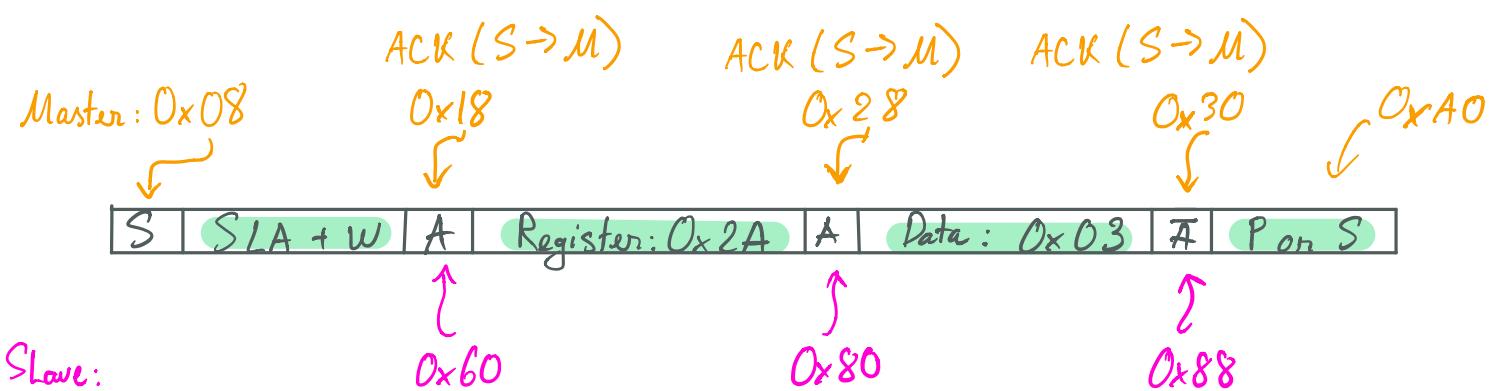
Master Receive → Slave

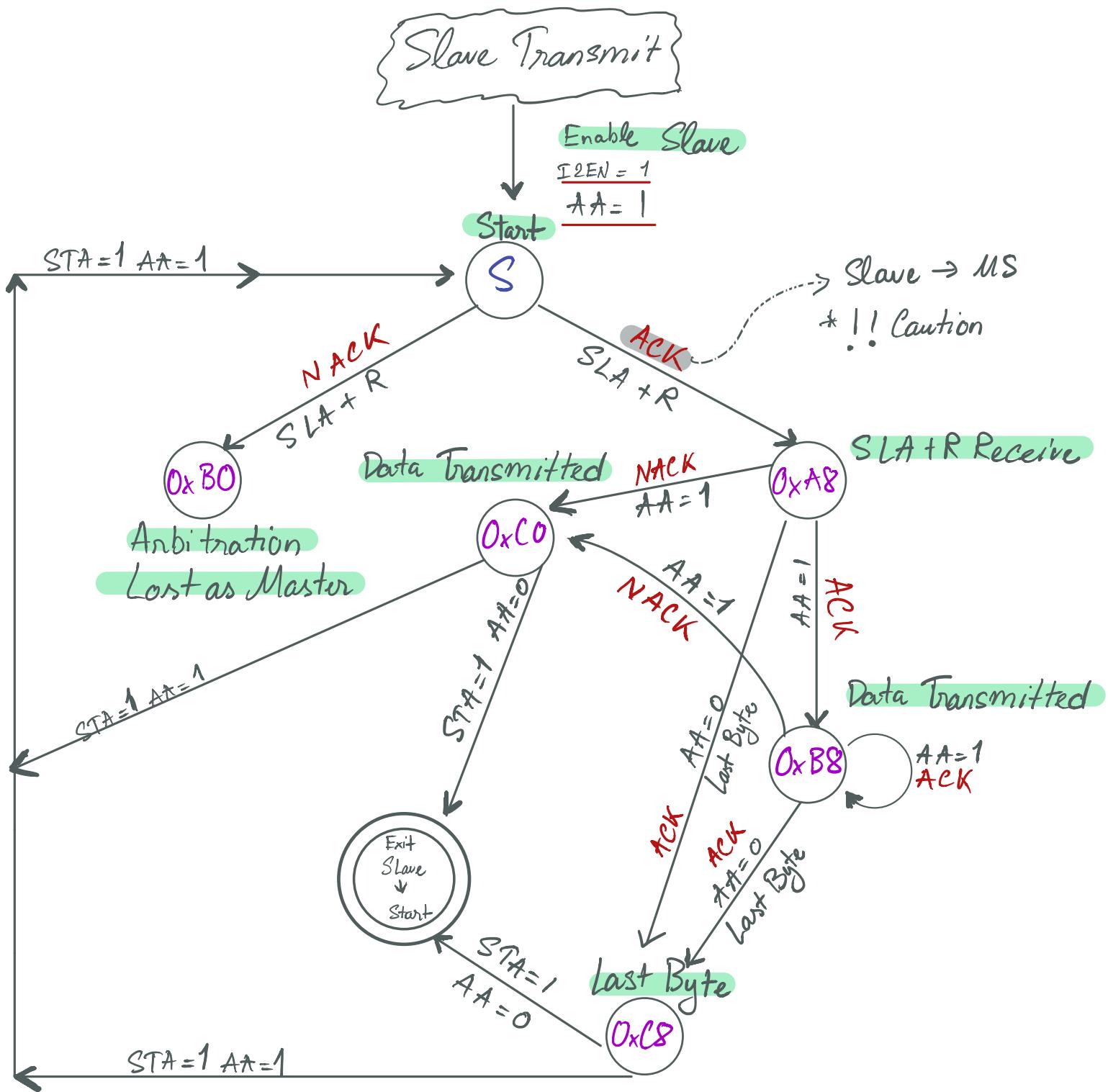
Slave Receive



Slave HW State	Why Slave got there	Next action Slave
0x60	SLA + W received ACK (From Slave)	* AA=1 Clear SI * Slave Receive mode * Initialize Slave Counter.
0x80	Data Byte received ACK (From Slave)	* Read Data from I2DAT ↳ <u>Last Byte:</u> Decrement the Slave Counter Clear SI and AA ↳ <u>n-Byte:</u> Set AA Clear SI Increase Slave buffer pointer
0x88	Data Byte received NACK (From Slave)	* SET AA bit * Clear SI bit
0xA0	STOP or START Received	* AA=1 Clear SI ↳ Slave Mode * AA=0 Clear SI ↳ Exit Slave mode

Example : SLA = 0x3A Register : 0x2A Data Byte = 0x03





I₂CONSET Slave

[5]	[5]	[4]	[3]	[2]	
I ₂ EN	STA	STO	SI	AA	
1	1	

● Grey color: Input for next State
 ● Red color: ACK or NACK
 Master Receive → Slave

Slave Transmit

Slave Hw State	Why Slave got there	Next action Slave
0x A8	SLA + W received ACK (From Slave)	* Load I2DAT Data (1 st Byte) * AA = 1 Clear SI * Initialize Slave Counter.
0x B8	Data Byte received ACK (From Master)	* Load I2DAT from Slave Transmit * Set AA Clear SI flag. * Increment Slave buffer pointer.
0xC0 or 0xC8	Data Byte received NACK (From Master)	* SET AA bit * Clear SI bit

Example : SLA = 0x 3A # Data Byte = 2

