



UNIVERSIDAD
COMPLUTENSE
MADRID

I2C in ESP32

HTS221

IoT Node Architecture

❑ Documentation available at

- <https://docs.espressif.com/projects/esp-idf/en/latest/esp32/api-reference/peripherals/i2c.html>

1. Configuring the connection
2. Installing the driver
3. Write / Read (as *master*)

- ❑ Choose one of the two available controllers (*port*): 0 or 1
- ❑ Choose pins for SDA and SCL
- ❑ Activate *pull-up*
- ❑ Choose clock frequency (max. 1 MHz)
 - And maybe some clock flags (DFS, *light sleep*)

```
int i2c_master_port = I2C_MASTER_NUM;
i2c_config_t conf;
conf.mode = I2C_MODE_MASTER;
conf.sda_io_num = I2C_MASTER_SDA_IO;
conf.sda_pullup_en = GPIO_PULLUP_ENABLE;
conf.scl_io_num = I2C_MASTER_SCL_IO;
conf.scl_pullup_en = GPIO_PULLUP_ENABLE;
conf.master.clk_speed = I2C_MASTER_FREQ_HZ;
i2c_param_config(i2c_master_port, &conf);
return i2c_driver_install(i2c_master_port, conf.mode, I2C_MASTER_RX_BUF_DISABLE, I2C_MASTER_TX_BUF_DISABLE, 0);
```

1. START command
2. Slave address
 - Slave address (7 bits)
 - Operation type: R/W bit: Read:1, Write:0

ACK sent by the selected slave

3. One or more data bytes
4. STOP command or repeated START

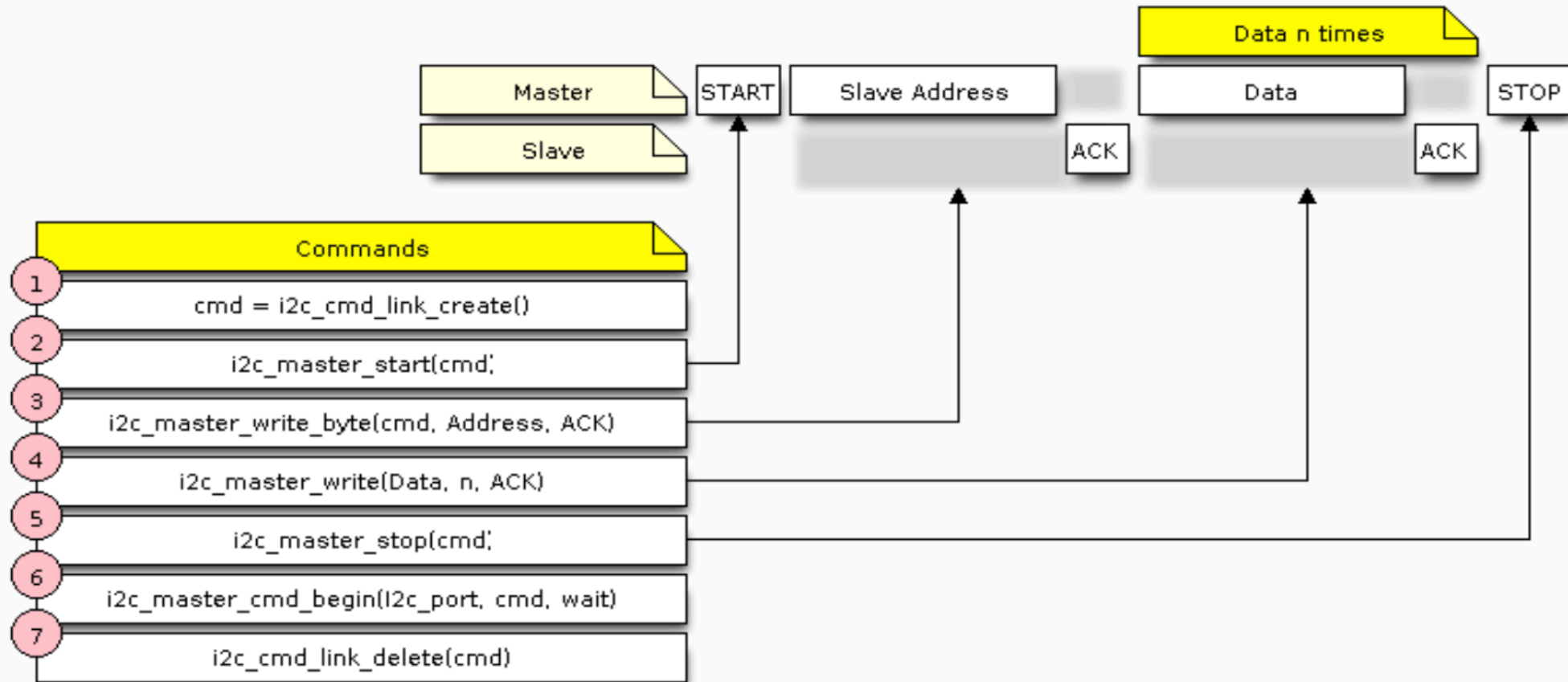


The 7 bits



Only one device will acknowledge

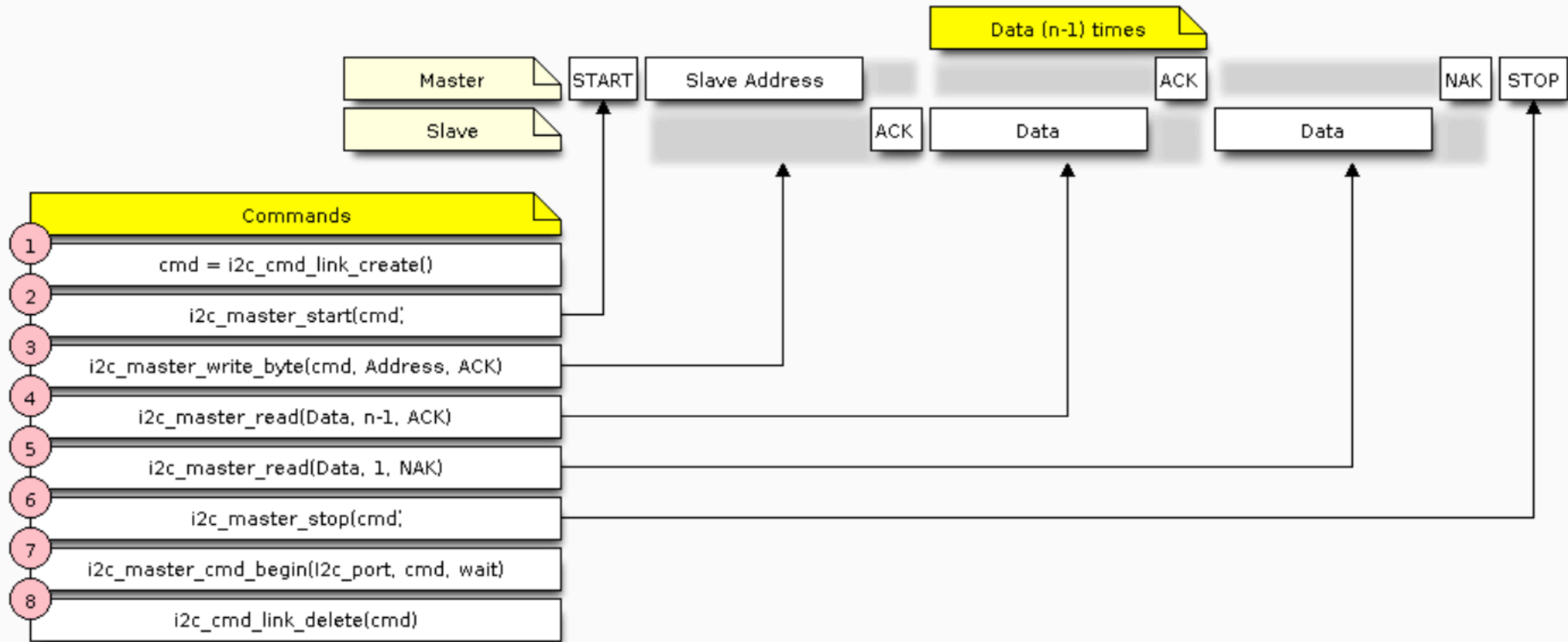
Write operation (master asks for data)



I2C command link - master write example 🔗

Source: <https://docs.espressif.com/projects/esp-idf/en/latest/esp32/api-reference/peripherals/i2c.html>

Read operation (master receives data)



I2C command link - master read example

Source: <https://docs.espressif.com/projects/esp-idf/en/latest/esp32/api-reference/peripherals/i2c.html>

- ❑ Temperature and humidity sensor (data sheet available)
 - What is its address? (*slave address*)
 - How will be the first call to `i2c_master_write_byte()`?
 - How many bytes need to be read in order to obtain raw temperature?