

ESP12-E programming tests

Using a breakout board and Arduino IDE 1.6.4

Breakout board has 10k PULL-UP resistor connected to ESP12E CH_PD pin.

Breakout board has 10k PULL-DOWN resistor connected to ESP12E GPIO2 pin.

Both resistors are surface mount parts soldered on the breakout board PCB.

All tests have FTDI / RS232 3V3 Level converter attached to Rx, Tx, and GND pins on ESP12E module.

Programming tests:

- 1) Just click upload - GPIO NOT pulled LOW via 10k resistor.

Does not work.

Arduino IDE Error(s) displayed:

warning: espcomm_sync failed

error: espcomm_open failed

- 2) Pull GPIO0 LOW with 10k resistor. (DO NOT RESET)

Does not work.

Arduino IDE Error(s) displayed:

warning: espcomm_sync failed

error: espcomm_open failed

- 3) Pull GPIO0 LOW with 10k resistor. (DO RESET)

Arduino IDE Error(s) displayed: (none)

Does work.

- 4) After programming, DO NOT disconnect the 10k pull down resistor from GPIO0

When Arduino IDE displays "Done Uploading." I tapped the RFID card.

First tap did not work.

Second tap did function normally.

Did the SPI buss activity wake the chip?

Pulled RESET pin LOW momentarily through 10k resistor.

Card taps on the RFID reader did not wake the ESP12E up. ESP12E is held in "ready to program" mode.

Power off, then on (power cycle).

Same results as reset: Card taps on the RFID reader did not wake the ESP12E up. It is held in "ready to program" mode.

- 5) After programming, remove GPIO0 pull-down resistor and power cycle or reset the ESP12-E module. Functions normally.

Observation:

When GPIO0 pin is held LOW when ESP12-E powers on (or device is reset) the module enters "programming" mode and is ready and able to be re-programmed from the Arduino IDE. For normal operation GPIO0 should not be LOW on device power on or reset.

John Soucy

Monkey Wrench Manufacturing

01JUL2015