Wireless Presentation Remote

By: Abderraouf Adjal

What is a "presentation remote device"?
Definition: A a human to machine computer device that navigate a presentation remotely.

Design Consideration

Design Consideration

- <u>Computer interface for device host:</u> Universal Serial Bus (USB); Any standard data speed is accepted, we will use USB FS standard (12 Mbit/s) for less hardware requirements.

- <u>Remote-to-host wireless communication:</u> Radio Frequency (RF).

- <u>The RF desired frequency:</u> 2.4GHz (ISM band).

- <u>The RF standard:</u> **ESP-NOW**, is a fast, connectionless communication technology featuring short packet transmission (uses IEEE 802.11), using ESP8266 SoC will make it easy for RF regulations.

- <u>The remote battery:</u> One usage Alkaline AA Battery for the price and avoiding the expensive regulation certificate on lithium batteries.

Design Consideration

- <u>The host part computing:</u> STM32F103 microcontroller (ARM Cortex-M3 core) for USB communication, and ESP8266 SoC by Espressif Systems for the RF communication with the remote. And they communicate together using one wire for a unidirectional UART (9600bps, 8-bit frame, no parity).
- <u>The remote (hand) part computing:</u> Only the ESP8266 SoC.
- <u>The ESP8266 type:</u> A ESP-12F Shielded module, and FCC pre-certified with onboard PCB patch antenna.
- <u>Extra hardware:</u> Push-buttons, 3.3v voltage regulator for the host microcontrollers.

The firmwares

The firmwares

STM32F103: ~500 LoC using ST's official HAL library, <u>the task:</u> Receive a UART frame from ESP8266, then do keyboard arrow-key press relevant to it.

ESP8266: ~120 LoC using Arduino's framework, <u>the task:</u> Receive a RF signal by ESP-NOW protocol from the remote side, then send its content to STM32 via UART.

The Remote Side

ESP8266: ~200 LoC using Arduino's framework, <u>the task:</u> wait for a GPIO interrupt from the push-buttons, then send a RF signal corresponding to what button was pressed by ESP-NOW protocol to the host side.

The firmwares

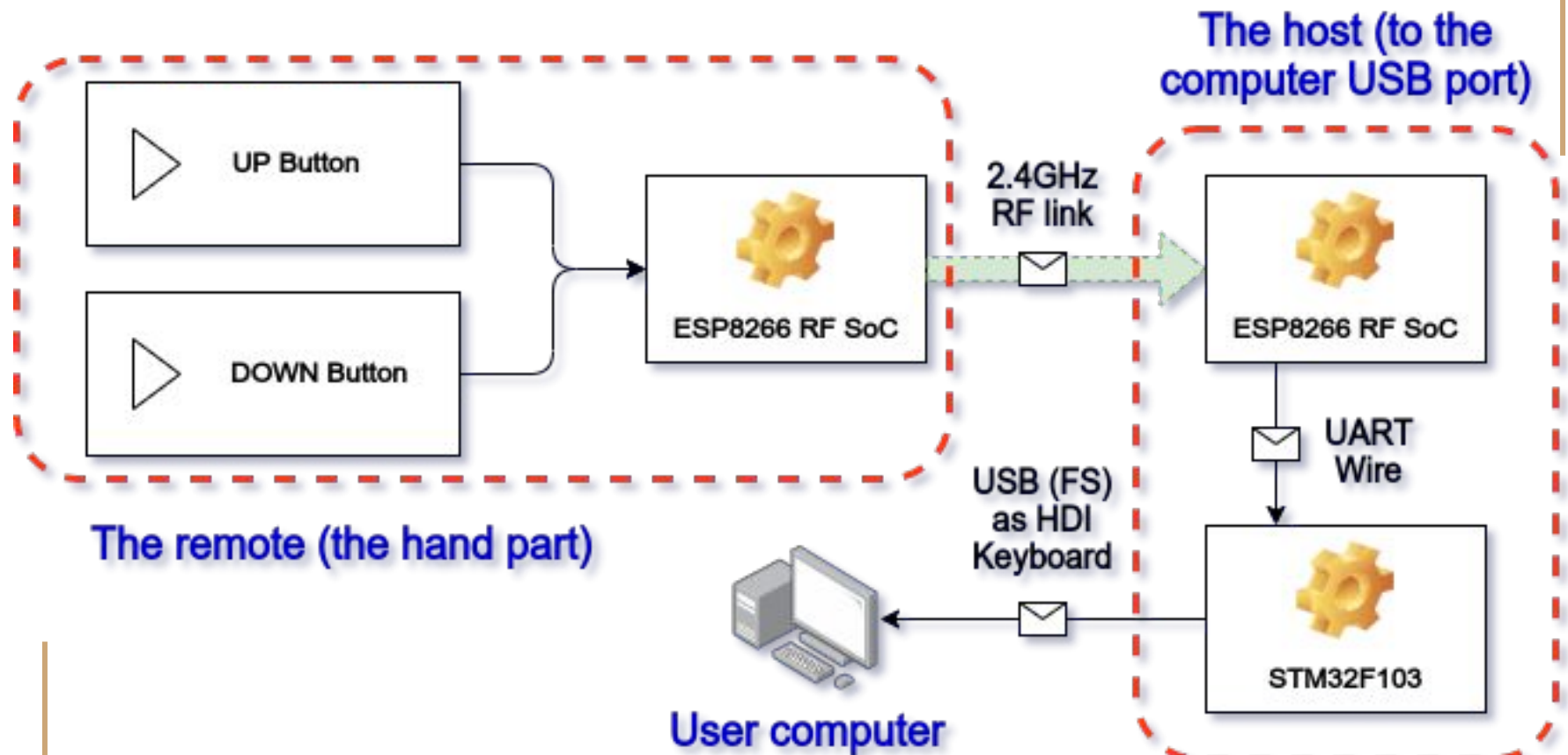**Extra futures had to be made:**

- Software button anti-debounce.

- No signal-repeating on push-button holding to avoid unwanted navigation.

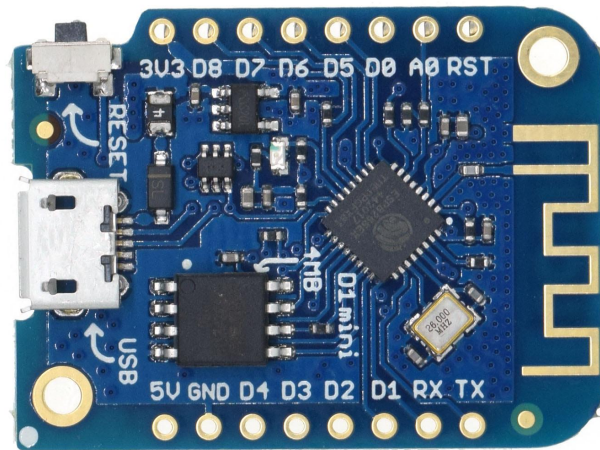- Defining a IEEE 48-bit MAC address for each pair.
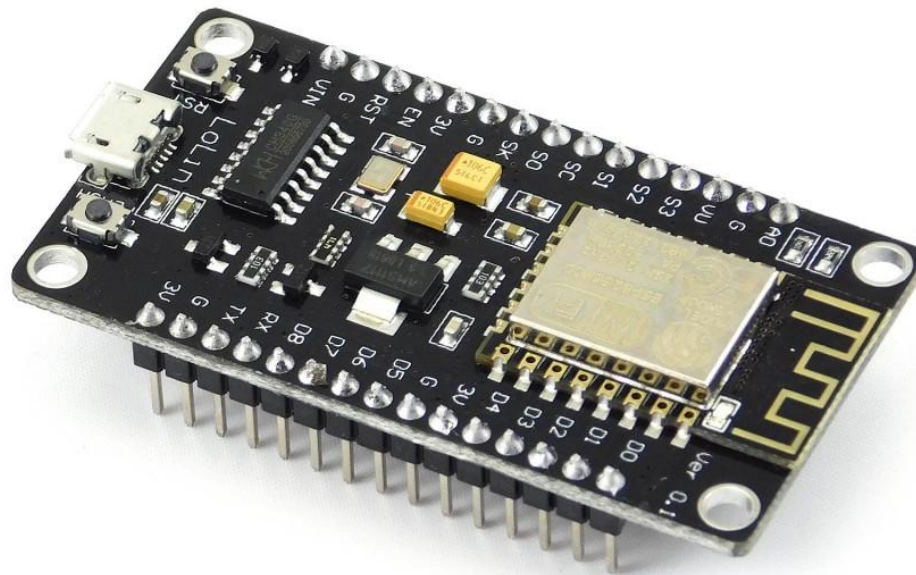
Pictures

Block Diagram:

The host (to the computer USB port)

The remote (the hand part)

UP Button

DOWN Button

ESP8266 RF SoC

2.4GHz RF link

ESP8266 RF SoC

UART Wire

USB (FS) as HDI Keyboard

STM32F103

User computer

Pictures

Espressif ESP8266 development boards: LOLIN D1 mini.
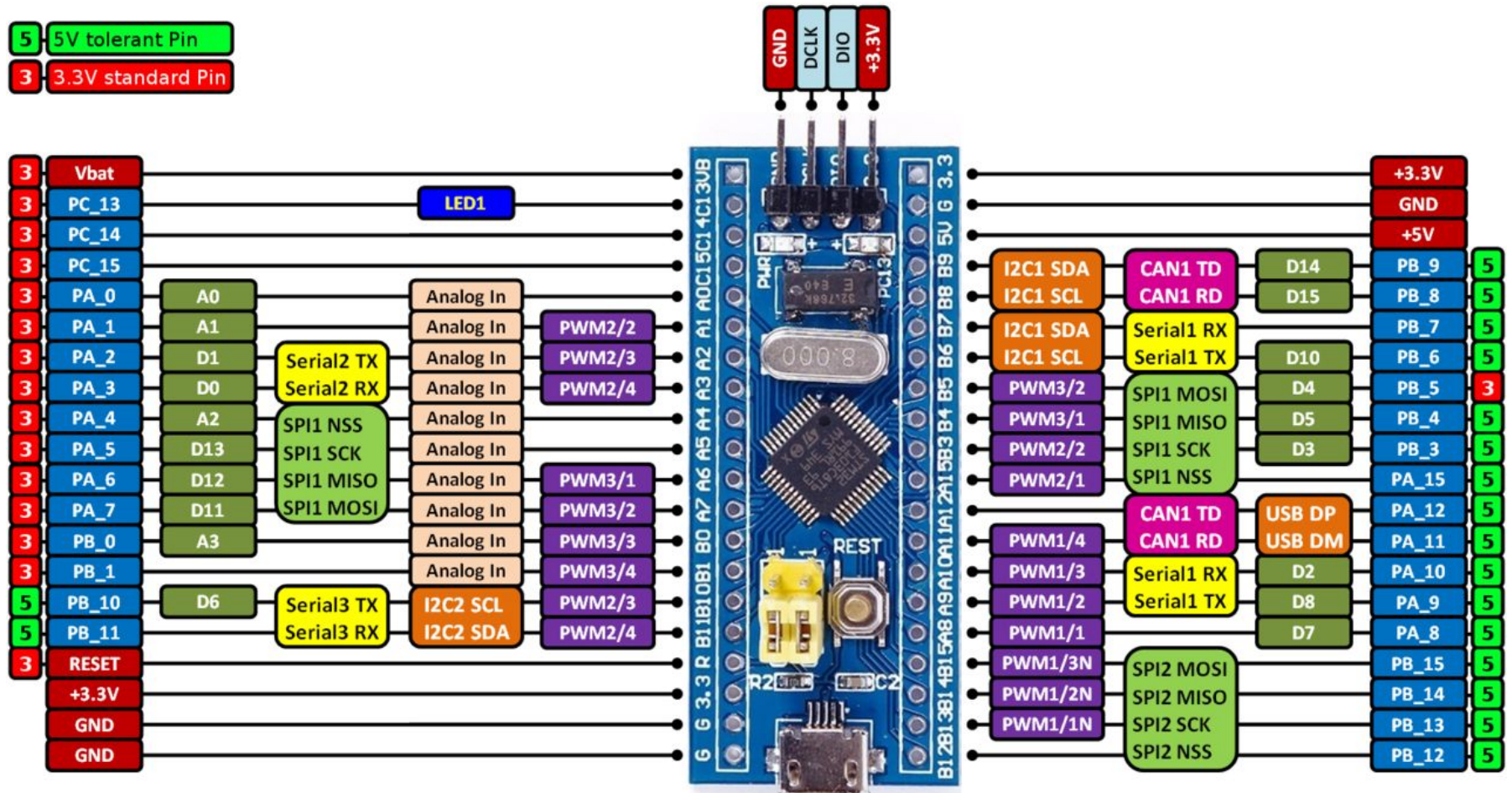
Pictures

Espressif ESP8266 development boards: NodeMCU V3.

Pictures

ST STM32F103C8T6 development boards (ARM Cortex-M core).

Conclusion

Implementing a low-cost wireless presentation remote needs a good resources and components consideration.

And the most difficult part will be the RF regulations, and the USB certification.

Thank You!