NEOCRUX
SYSTEMS

Receiving data via UART and read, write, erase flash memory

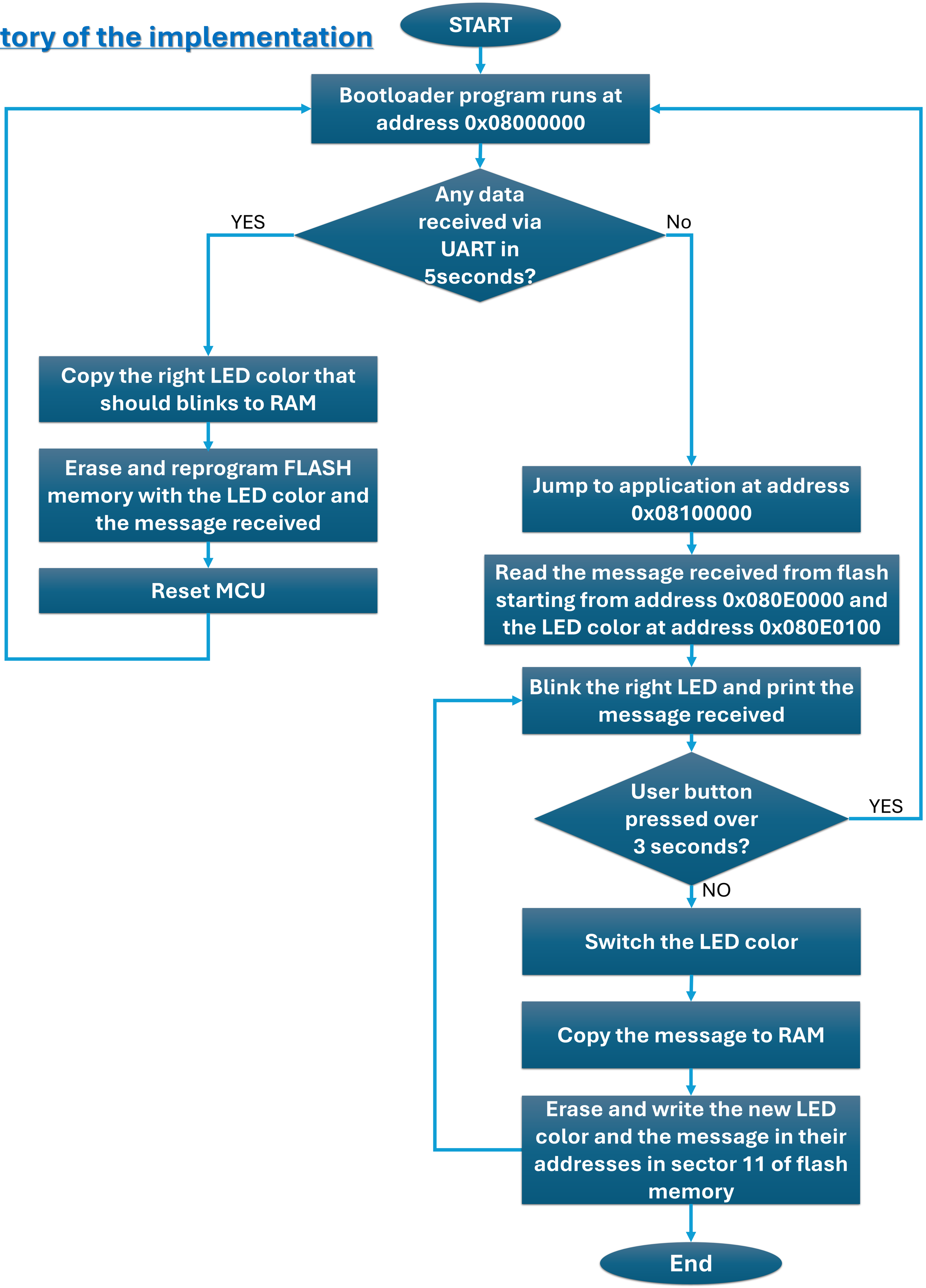
Case Study: STM32F767ZI

ONE MORE STEP TO BOOTLOADER

NEOCRUX SYSTEMS

NEOCRUX SYSTEMS

Story of the implementation



Results

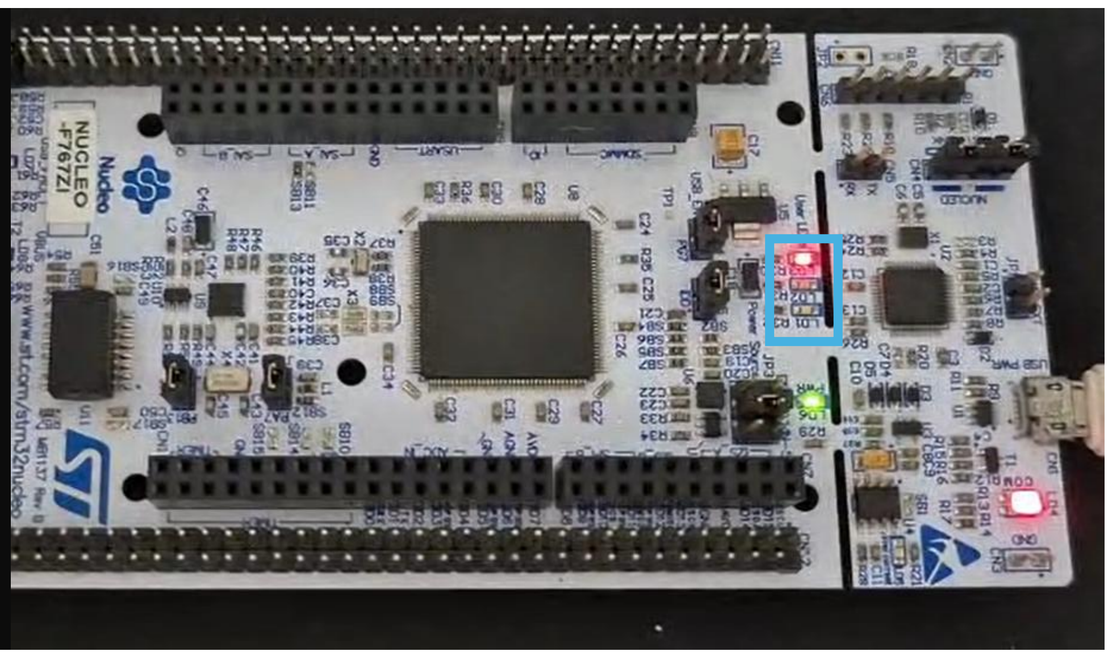
1st result: Power up without sending message

- After powering the MCU, the CPU goes to the first address of the Flash memory which is 0X08000000.
- Then the bootloader takes care of jumping to the address 0x08100000
- After that, It will read content of memories 0x080E0000 and 0x080E0100 which are the message already stored and the LED color

```
PS C:\Users\      > python -m serial.tools.miniterm COM6 115200
--- Miniterm on COM6  115200,8,N,1 ---
--- Quit: Ctrl+] | Menu: Ctrl+T | Help: Ctrl+T followed by Ctrl+H ---

Starting bootloader
Waiting for data (5s)...
No message received!

Jumping to application
Starting application
Blue led should blink
Stored sentence: Hello embedded community
```

2nd result: Switch led color and reset with user button without sending message

```
PS C:\Users\      > python -m serial.tools.miniterm COM6 115200
--- Miniterm on COM6  115200,8,N,1 ---
--- Quit: Ctrl+] | Menu: Ctrl+T | Help: Ctrl+T followed by Ctrl+H ---

Jumping to Bootloader
Starting bootloader
Waiting for data (5s)...
No message received!

Jumping to application
Starting application
Red led should blink
Stored sentence: Hello embedded community
```



3rd result: Reset with user button with sending new data

```
PS C:\Users\      > python -m serial.tools.miniterm COM6 115200
--- Miniterm on COM6  115200,8,N,1 ---
--- Quit: Ctrl+] | Menu: Ctrl+T | Help: Ctrl+T followed by Ctrl+H ---

Jumping to Bootloader
Starting bootloader
Waiting for data (5s)...
Data received. Writing to flash...
Flash write done.

Jumping to Bootloader
Starting bootloader
Waiting for data (5s)...
No message received!

Jumping to application
Starting application
Red led should blink
Stored sentence: Hello my embedded colleagues
```