



» FIRMWARE OVER THE AIR



Team Members:

Mahmoud Mostafa | Nardin Nader | Salma Ali | Mahmoud Saeed | Ahmed Ashry



» TABLE OF CONTENTS «

Introduction

1

Bootloader

4

Cloud Server

2

Video Demo

5

Raspberry Pi

3

Future Work &
References

6



FOTA Project



Introduction





EMBEDDED SYSTEMS TRACK



Why FOTA?



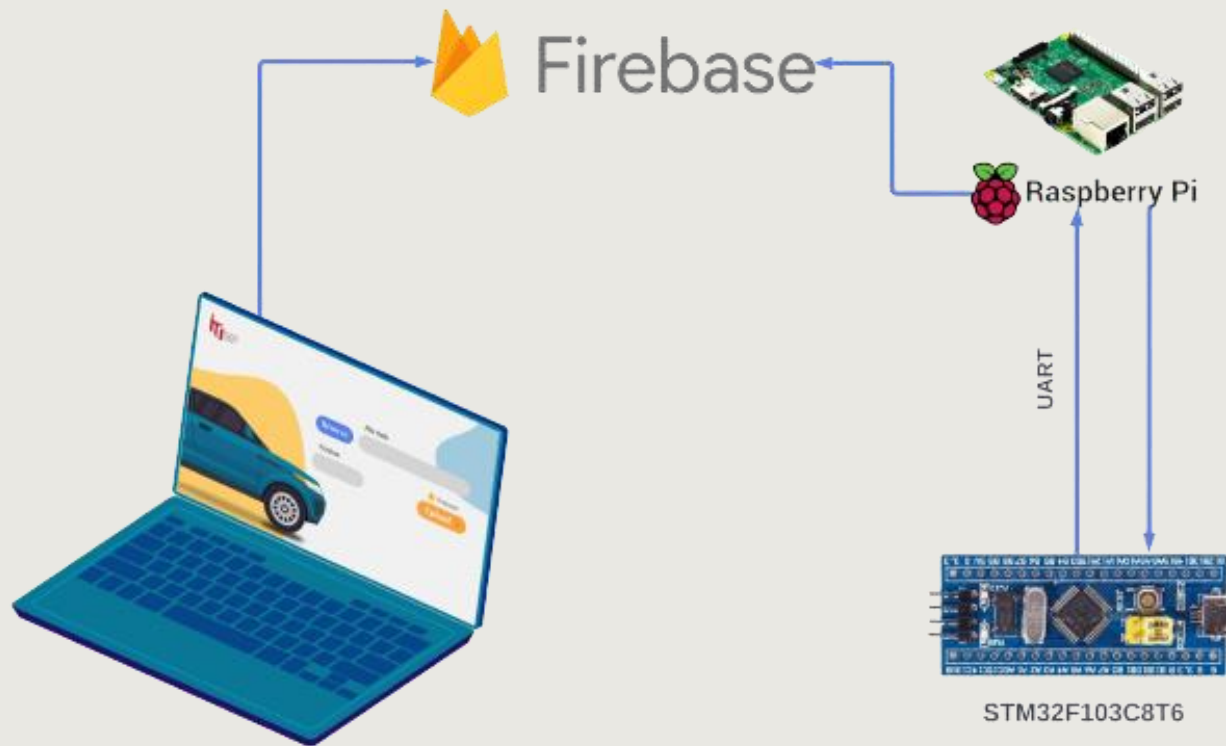
FOTA technology makes it possible to:

- Fix bugs and security patch updates.
- Improve system functionality.
- Update firmware versions with no physical contact.

Graduation Project



How does it work?





FOTA Project



Cloud Server



Server to link the car owner with
the company management





Cloud Server Roadmap

Choosing A Server



Creating A Project
On Server



Creating A Project
App



Copying App
Configurations
Into Python Code



Linking Python Code to
Cloud Server Project
App



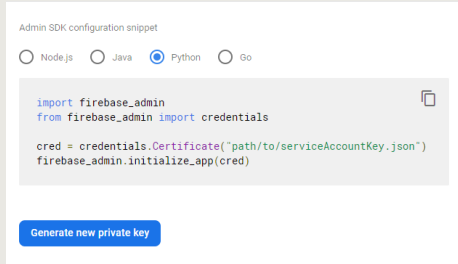
Creating A GUI For
Uploading Hex Files



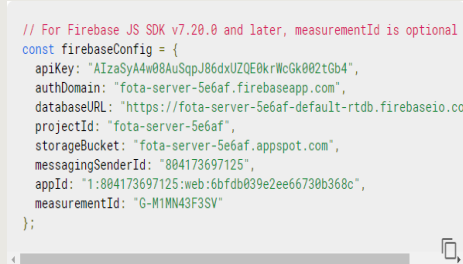
Uploading To Server



Why **Firebase?**



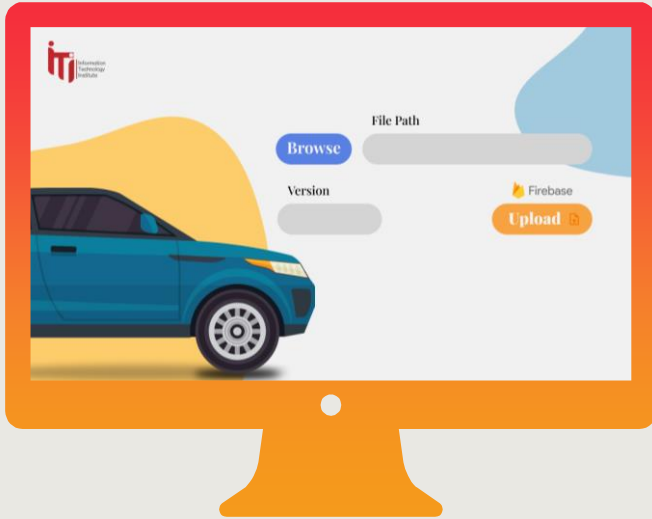
Cloud Security Key



Easy to handle with
python



Realtime Database



» DESKTOP GUI «

GUI based on Tkinter and Python.

FOTA Project



FOTA Project



Raspberry Pi



03





Raspberry Pi Work Sequence

AutoStart script to check for new updates on DB

If new update is detected, it fetches the new hex file from server

Ask the user whether to flash now or snooze it for later

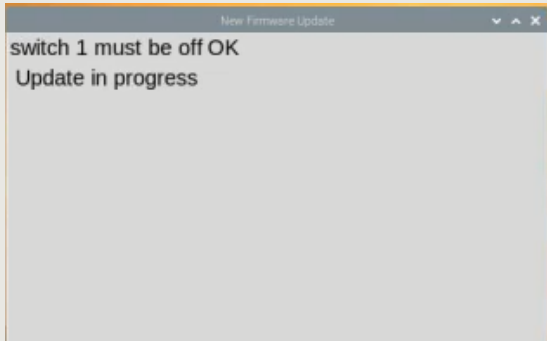
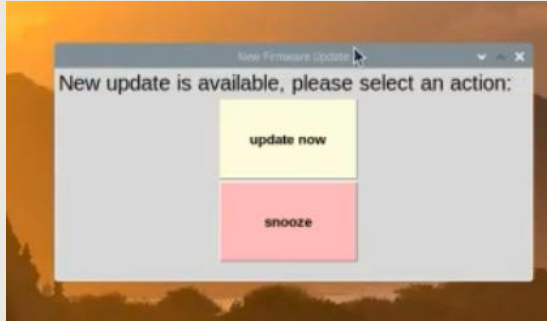
Checks on target STM32's safety switch (simulate engine running)

Status of the flashing is shown to the user till flashing is done

The downloaded file is then parsed into lines and sent to STM32 through UART

Once the safety switch is turned on, the GUI gives the user the option to start update





Why Raspberry Pi?

1. OS Based
2. Suitable memory and easily programmable
3. Background script
4. Open source



» BootLoader «

A large square with a vertical orange-to-red gradient, located on the right side of the slide.

04

**App Micro
controller**



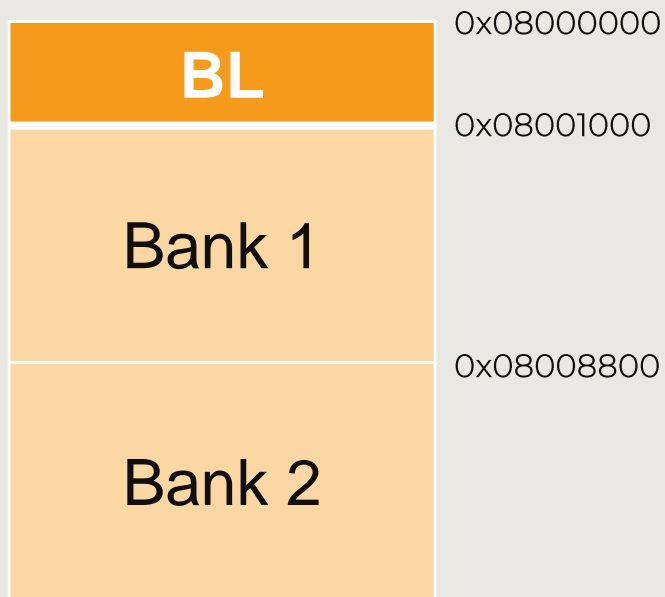
```
graph TD; A[App Micro controller] --- B[ ]; B --- C[Bootloader]; B --- D[Application];
```

Bootloader

Application

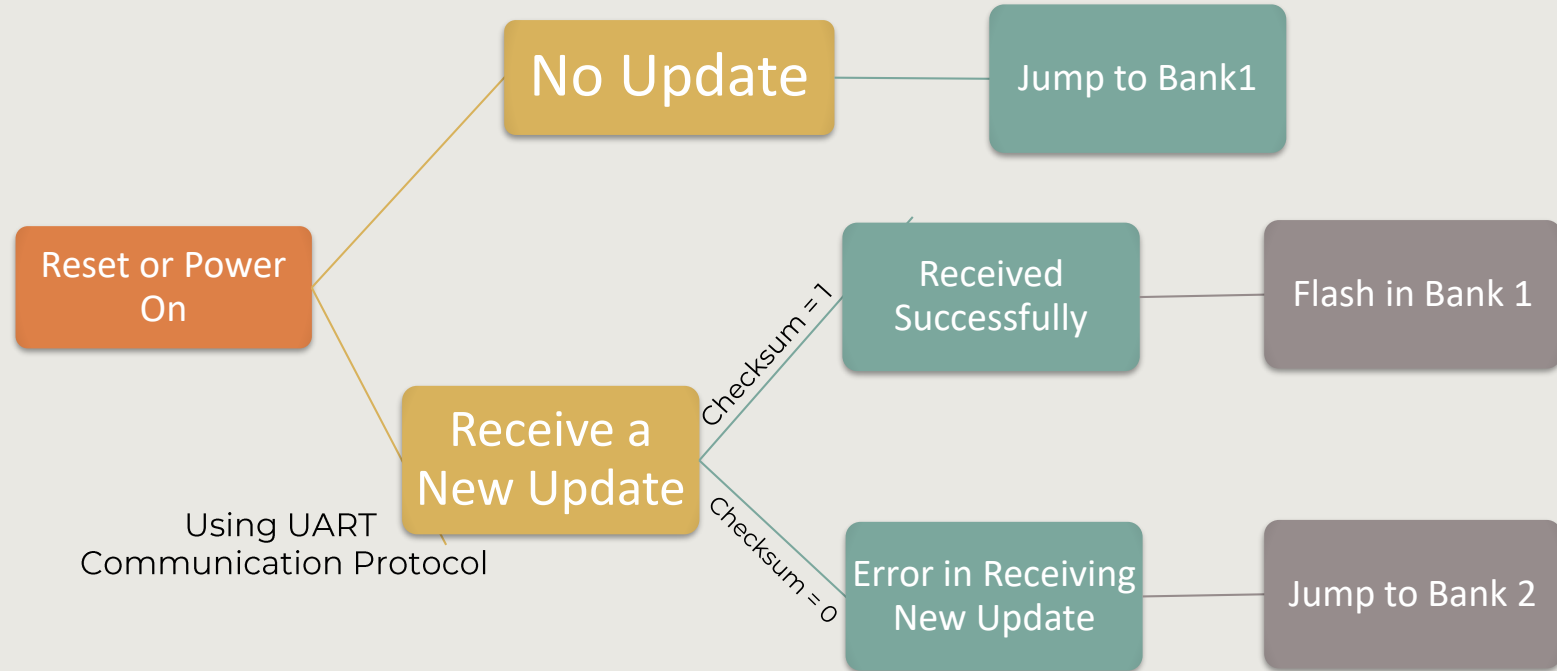


Flash Memory Arrangement



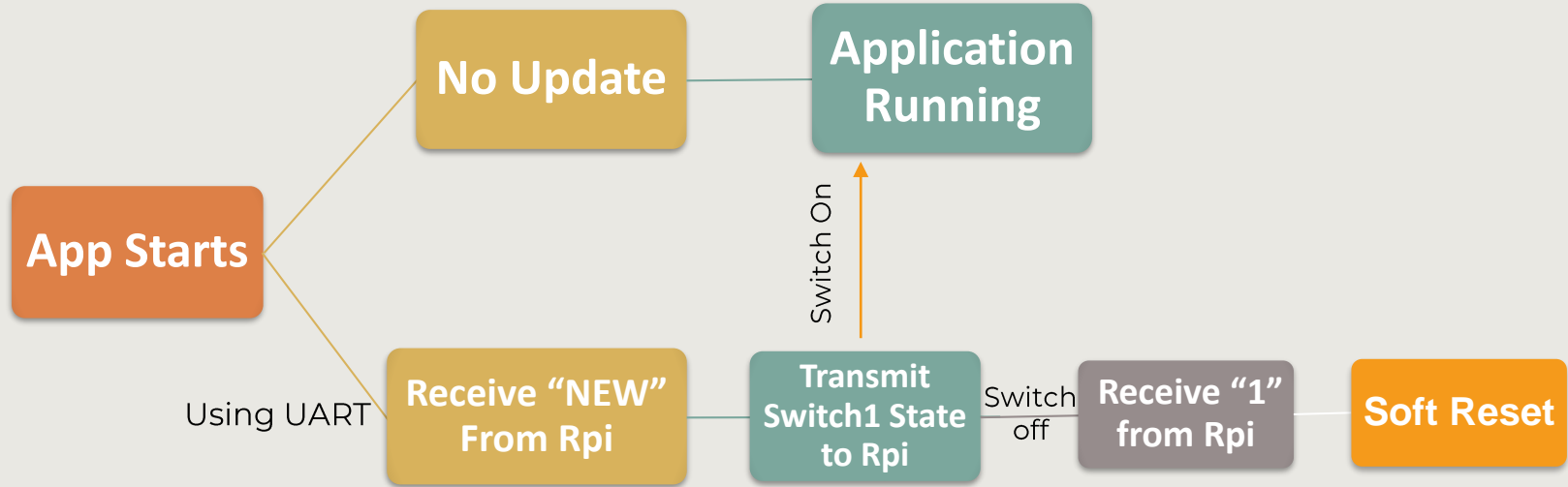
- Total Size : 64 KB
- Bootloader : 4 KB
- First Bank : 30 KB
- Second Bank : 30 KB

» Bootloader Possible Scenarios «





Application Scenarios





FOTA Project



» Video
Demo «







» Future Work «

X

A large square with a vertical orange-to-red gradient, located on the right side of the slide.

06



Future Work

1. Upgrade the communication protocol between the Raspberry Pi and target MCUs to a better communication protocol (e.g., CAN - LIN)
2. Implementing more advanced secure application
3. Implementing a Cybersecurity protocol
4. Creating our own cloud server





FOTA Project



» References





References

1. <https://www.electronicwings.com/raspberry-pi/raspberry-pi-uart-communication-using-python-and-c>
2. <https://github.com/AMHD/Connecting-Raspberry-Pi-with-Firebase-Database/blob/master/README.md>
3. <https://github.com/thisbejim/Pyrebase>
4. <https://www.youtube.com/watch?v=gLyaR3KPYt4>
5. <https://www.youtube.com/watch?v=WurCpmHtQc4>
6. <https://towardsdatascience.com/how-to-easily-convert-a-python-script-to-an-executable-file-exe-4966e253c7e9>



» THANK YOU «

Does anyone have any questions?

X