ABSTRACT:

- Vehicle windshield wipers play a important role during extreme weather conditions by wiping the rain water, dust, fog, mist continuously over the course of time and they help us give a clear vision to the driver. Wipers are designed and made to clear the unwanted dust, water etc. from a windscreen. It is Prime duty to improve the safety facilities in all automobiles and vehicles. The system is used to activate the Wipers manually. Due to this issue many engineers and designers and developers have contributed to the research and development of Automatic Windshield Wiper Functionality in automobiles. The Windshield Wiper Functionality of domestic cars primarily consists of three systems
- Two wipers and theist respective arms A mechanism, The concept of this proposed wiper system is similar to that of all the other existing wiper systems. A wiper normally consists of metal arm, one end pivots and the other end has a long rubber blade-attached to it. The purpose of the project is to design the windshield wiper control system for the safeties of the people who drive heavy automobiles and domestic four-wheeler vehicles.

INTRODUCTION:

- Wiper is an essential component that is used to wipe raindrops or any water from the windscreen. Wipers are designed and made to clear the water from a windscreen. Most cars have two wipers on the windscreen, one on the rear window and the other on each headlight. The wiper parts visible from outside the car are the rubber blade, the wiper arm holding the blade, a spring linkage, and parts of the wiper pivots. The wiper itself has about six parts called pressure points or claws that are small arms under the wiper.
- The existing system uses a control stalk to activate the wiper and the process of pulling up the wiper is difficult. needs to switch on and off the control stalk and it will reduce the driver's concentration during the driving. Thus, this system is proposed to solve all these problems. The concept of this wiper system is similar to other conventional wipers, yet this system will be upgraded to an automatic control system by using a controller. When water hits a dedicated sensor located on the windscreen, it triggers the wiper motor to move. is not detected by sensor, the wiper will automatically stop. This will help the driver to give more concentration and reduce the car accident probability.
- The operational speed of a wiper is controlled by a wiper speed control system in accordance with frequencies. The pulse signal is digitally processed to provide a control signal. A wiper driver circuit receives the control signal and adjusts the operational speed or timing in line with it.
- In this project, there were two innovations reviewed as the literature review. The two were designed with different concepts and operating mechanism however with same objective of working principle of the car wiper. The rain sensor was a highly versatile device for automatic wiping of vehicle windscreen when it is wet due to moisture, raindrops or even mud. It worked by reflecting harmonious light beams within the windscreen. When raindrops fall onto the windscreen, this harmony light is disturbed and creating a drop in the light beam intensity.

- The system then activated the wipers to be operated in full automatic mode. It has a response time of 0.1 seconds. It allowed for a quick reaction when it is a sudden splashes of water that will make the driver totally 'blinds' when the situation happened. With the automatic wiper, the driver can avert the risk of an accident. The automatic wiper is important during heavy traffic, e.g. in town, city, school zone and other public places. A driver may be subjected to many distractions with bad weather, dangerous road conditions and fatigue. The Rain Sensor reduced the driver's burden by making driving more comfortable.

COMPONENTS AND SUPPLIES

A. STM32F407 Discovery Board
B. Push Button
C. LEDs
D. Resistors
E. Power Supply

ADVANTAGES:

a) To save money during wet seasons, turn off the irrigation system. Electricity bills are lowered as a consequence.
b) Rain sensors store water during rain events, allowing it to be available throughout the summer and winter.
c) As a consequence, rain sensor-based equipment like vehicle wipers and irrigation systems last longer since they only work when needed.
d) It is quite simple to use.
e) As a consequence, less energy is consumed.
f) Rain sensor-based systems are extremely simple to install.
g) Individual rain sensors are fairly inexpensive.

Disadvantages:

a) When water falls squarely on the rain sensor, the mechanism activates.
b) The entire system cost rises when more components, including a rain sensor, are required.
c) Rain sensors must make a decision within a few minutes to avoid erroneous detection of rain

SOFTWARE REQUIREMENTS:

STM32 CUBE IDE

COMPONENTS:

STM32F4O7VG MICROCONTROLLER BOARD

The STM32F407 Kit takes advantage of the high-performance STM32F407 microcontrollers' capabilities to make it simple for users to create audio-based applications. It comes with an ST-LINK embedded debug tool, an ST-MEMS digital accelerometer, a digital microphone, an audio DAC with integrated class D speaker driver, LEDs, pushbuttons, and a USB OTG micro-AB connector. Ethernet connectivity, an LCD display, and other features have been added to the STM32F4 DISCOVERY kit. The STM32F405xx and STM32F407xx families are built around the high-performance Arm® Cortex®-M4 32-bit RISC core, which runs at up to 168 MHz.

REQUIREMENTS FOR THE PROJECTS ARE:

- STM32F407VG : Th STM32F405xx and STM32F407xx family is based on the high-performance Arm® Cortex®-M4 32-bit RISC core operating at a frequency of up to 168 MHz. The Cortex-M4 core features a Floating point unit (FPU) single precision which supports all Arm single-precision data-processing instructions and data types. It also implements a full set of DSP instructions and a memory protection unit (MPU) which enhances application security. The STM32F405xx and STM32F407xx family incorporates high-speed embedded

- Xpack Packages : Windows Build Tools: The xPack Windows Build Tools is a standalone Windows binary distribution of GNU make and a few of other tools required by the Eclipse Embedded CDT (formerly GNU MCU/ARM Eclipse) project, but the binaries can also be used in generic build environments.

- OpenOCD : Open On-Chip Debugger (OpenOCD) is a free, open-source project that aims to provide debugging, in-system programming, and boundary scan using a debug adapter. The adapter is a hardware module that provides the right signals for the target to understand.

- QEMU : The xPack QEMU Arm is a standalone cross-platform binary distribution of QEMU, with several extensions for Arm Cortex-M devices.

FEATURES

- Wiper system is easy to use.
- Three LEDs indicating the wiping operation.
- STM32F407VG microcontroller has Up to 1 Mbyte of Flash memory.
- STM32F407VG microcontroller has Up to 192+4 Kbytes of SRAM including 64-Kbyte of CCM (core coupled memory) data RAM. 512 bytes of OTP memory.
- USB ST-LINK with three separate interfaces and re-enumeration capability.
- Virtual Com port Debug port (with new order code only)
- Large-scale storage (with new order code only)
- Board power is supplied through USB or an external 5 V supply source.
- 3 V and 5 V external application power supply

WORKING PRINCIPLE:

Assume that the automobile is the microcontroller. If the button is hit, the first led (red) will turn on, Clicking again the wiper will start, and the second led (blue) will turn on for a desired rate. If the button is pressed again, the third led (green) will turn on, and the wiper's speed will be increased in comparison to the previous one. The fourth press will turn on the fourth led (orange), and the wiper speed will be increased in accordance with the previous one. The microcontroller (vehicle) is turned off after the fifth click.

4W & H (WHO,WHAT,WHEN,WHERE,HOW)

- WHO- Useful for drivers like car, buses and trucks for clear vision of the road
- WHAT- Aim is to implement automatic wiper system for vehicles
- WHEN- Helpful during rain
- WHERE- Installed at the front windshield
- HOW- Ignition is turned on, the wiper starts its operation.

SWOT ANALYSIS:

HIGH LEVEL REQUIREMENTS:

I'D	DESCRIPTION	STATUS
HR_01	Car in ACC mode	Implemented
HR_02	Car in ignition mode	Implemented
HR_03	Turning on the wiper	Implemented
HR_04	Turning off the wiper	Implemented

LOW LEVEL REQUIREMENTS:

I'D	DESCRIPTION	STATUS
LR_01	ON	Implemented
LR_02	Press wiper switch	Implemented
LR_03	Microcontroller supply	Implemented
LR_04	Activating wiper blades	Implemented
LR_05	OFF	Implemented

TEST PLAN:

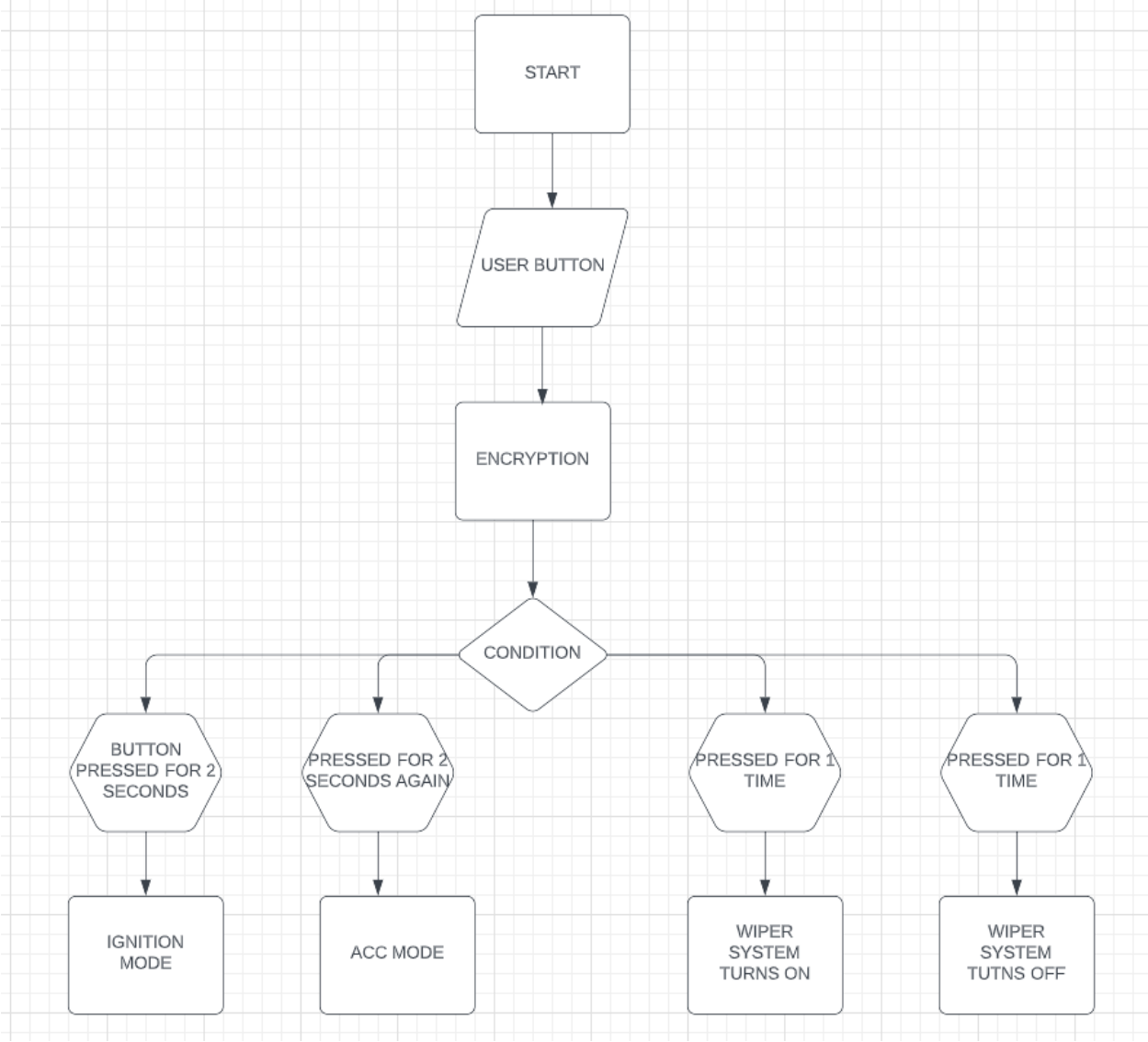
HIGH LEVEL TEST PLAN:

TEST I'D	DESCRIPTION	TEST TYPE	OUTPUT EXPECTED	FINAL STATUS
H_01	Moving the wiper along windshield	Secnario based test	PASS	IMPLEMENTED
H_02	wiper comes back to the rest position at the end	Boundary value based test	PASS	IMPLEMENTED

LOW LEVEL TEST PLAN:

TEST I'D	DESCRIPTION	TEST TYPE	OUTPUT EXPECTED	FINAL STATUS
L_01	Powering ON the Car	Secnario based test	PASS	IMPLEMENTED
L_02	Powering OFF the Car	Boundary value based test	PASS	IMPLEMENTED

WIPER CONTROL SYSTEM:



FINAL OUTPUTS:

1. EXECUTION WHEN BUTTON IS PRESSED ONCE FOR 2 SECONDS:

Ignition mode and Car in ON

```
$ qemu-system-gnuarmeclipse.exe -M STM32F4-Discovery -mcu STM32F407VG -kernel finalled.elf
[button:user down=1][button:user up=0][led:green on]
[led:orange on]
[led:red on]
[led:blue on]
```

2. EXECUTION WHEN BUTTON IS PRESSED ONCE AGAIN FOR 2 SECONDS:

ACC mode and Car is OFF

```
[led:blue on]
[button:user down=1][button:user up=0][button:user down=1][button:user up=0][led:green off]
[led:orange off]
[led:red off]
[led:blue off]
```

3. EXECUTION WHEN BUTTON IS PRESSED FOR SECOND TIME:

Turning ON the wiper

```
[button:user down=1][button:user up=0][button:user down=1][button:user up=0][button:user down=1][button:user up=0][led:green on]
[led:green off]
[led:orange on]
[led:orange off]
[led:red on]
[led:red off]
[led:blue on]
[led:blue off]
```

4. EXECUTION WHEN BUTTON IS PRESSED FOR THIRD TIME:

Turning OFF the wiper

```
[button:user down=1][button:user up=0][button:user down=1][button:user up=0][button:user down=1][button:user up=0][button:user down=1][button:user up=0][led:green on]
[led:green off]
[led:blue on]
[led:blue off]
[led:red on]
[led:red off]
[led:orange on]
[led:orange off]
```