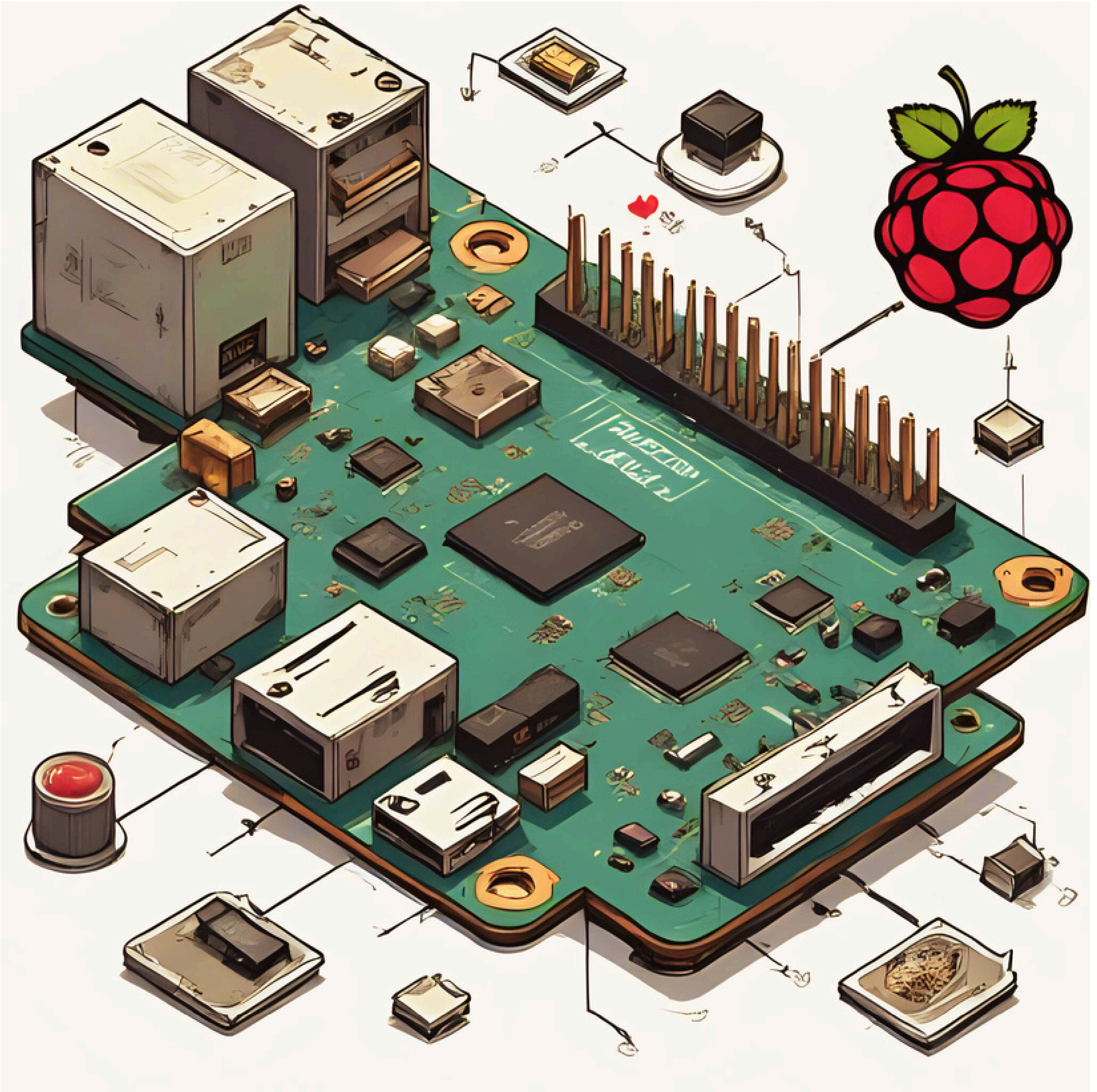
REVIEW 2

Edge Based Hybrid System Implementation for Long Range Safety and Healthcare IoT Applications

~ Research Authors: Fan Wu,Taiyang Wu and Mehmet Rasit Yuce



TEAM 27

Reenu L K	CB.SC.U4CSE23440
Yogini Aishwaryaa P T S	CB.SC.U4CSE23557
Bhuvaneshwari D	CB.SC.U4CSE23612
Rama Roshinee S V	CB.SC.U4CSE23645
Shruthilaya A V	CB.SC.U4CSE23653
Sharumathi P S	CB.SC.U4CSE23665

PROBLEM STATEMENT RECAP

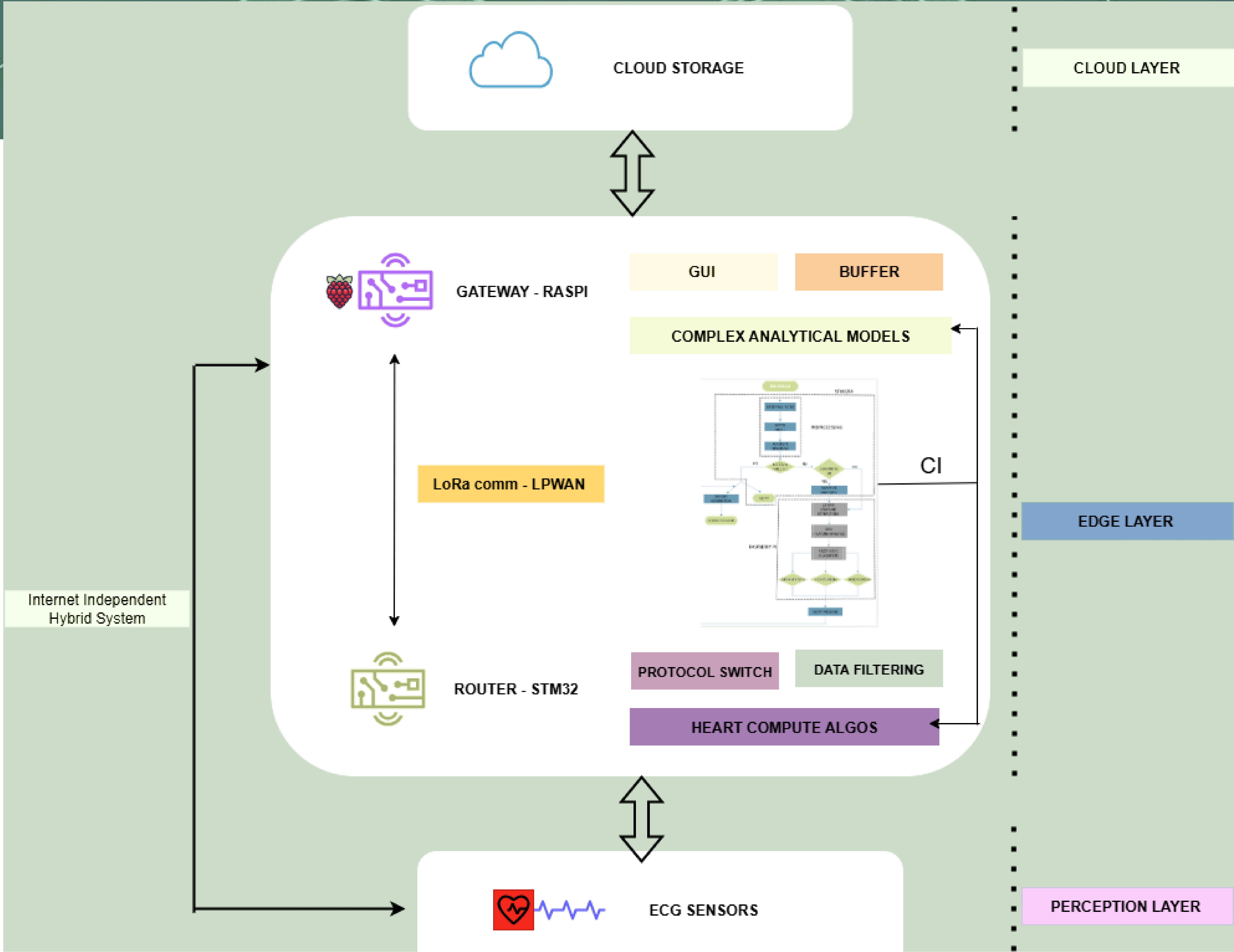
- High latency in cloud computing.
- Short-range protocols have limited coverage. (BLE)
- Real-time feedback is required.
- Low-Latency Edge Computing - reduces latency to as little as 11.5 ms (hybrid system).

Specific ECG data pipeline for edge task allocation.

Explicit Resource Management.

CI Model Integration.

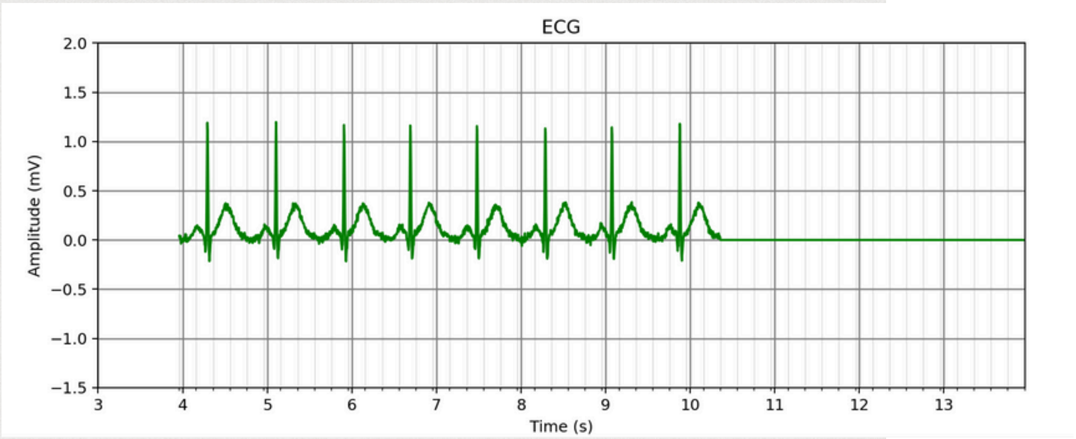
Scalable System Architecture



CODE WALKTHROUGH

ECG Sim

```python
import numpy as np
import matplotlib
matplotlib.use("TkAgg")
import matplotlib.pyplot as plt
import time

fs = 250
dt = 1.0 / fs
mean_hr = 75
hrv_std = 0.02
display_seconds = 10

def ecg_template(t):
    return (
        0.1 * np.exp(-((t-0.2)/0.05)**2) +
        -0.15 * np.exp(-((t-0.3)/0.015)**2) +
        1.2 * np.exp(-((t-0.32)/0.01)**2) +
        -0.25 * np.exp(-((t-0.34)/0.015)**2) +
        0.3 * np.exp(-((t-0.55)/0.1)**2)
    )
```

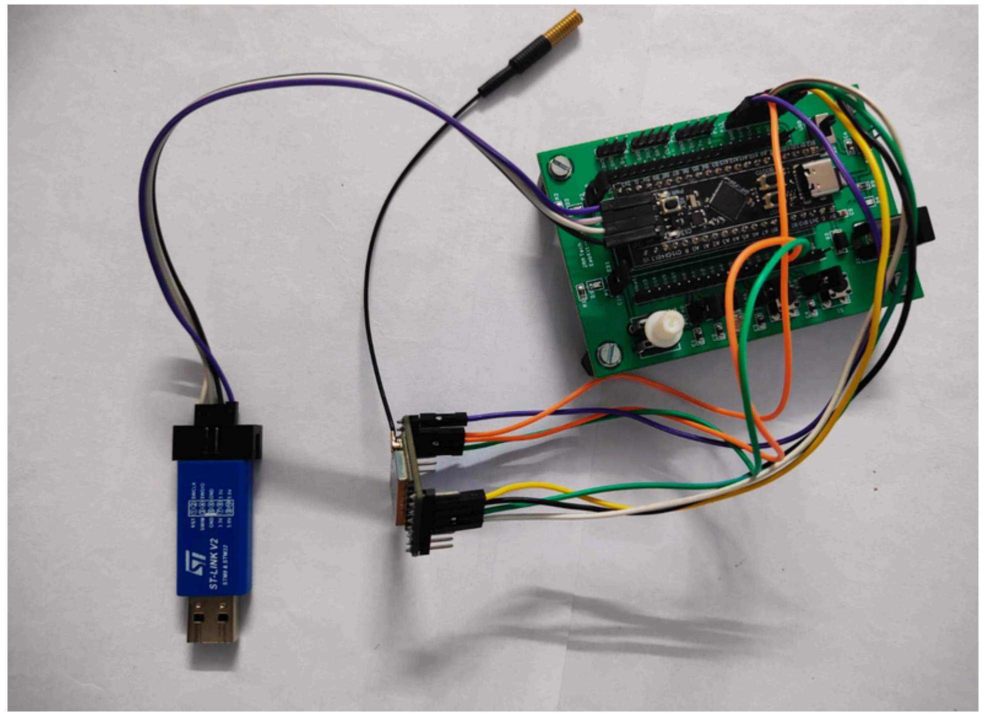


LoRa Config

```c
#include "stm32f4xx.h"
#define LORA_REG_VERSION   0x42
void SPI1_Init(void) {
    RCC->APB2ENR |= RCC_APB2ENR_SPI1EN;
    RCC->AHB1ENR |= RCC_AHB1ENR_GPIOAEN;

    GPIOA->MODER &= ~(0xFF << 8);
    GPIOA->MODER |= (0xAA << 8);
    GPIOA->AFR[0] &= ~(0xFFFF0000);
    GPIOA->AFR[0] |= (0x55550000);
}
```

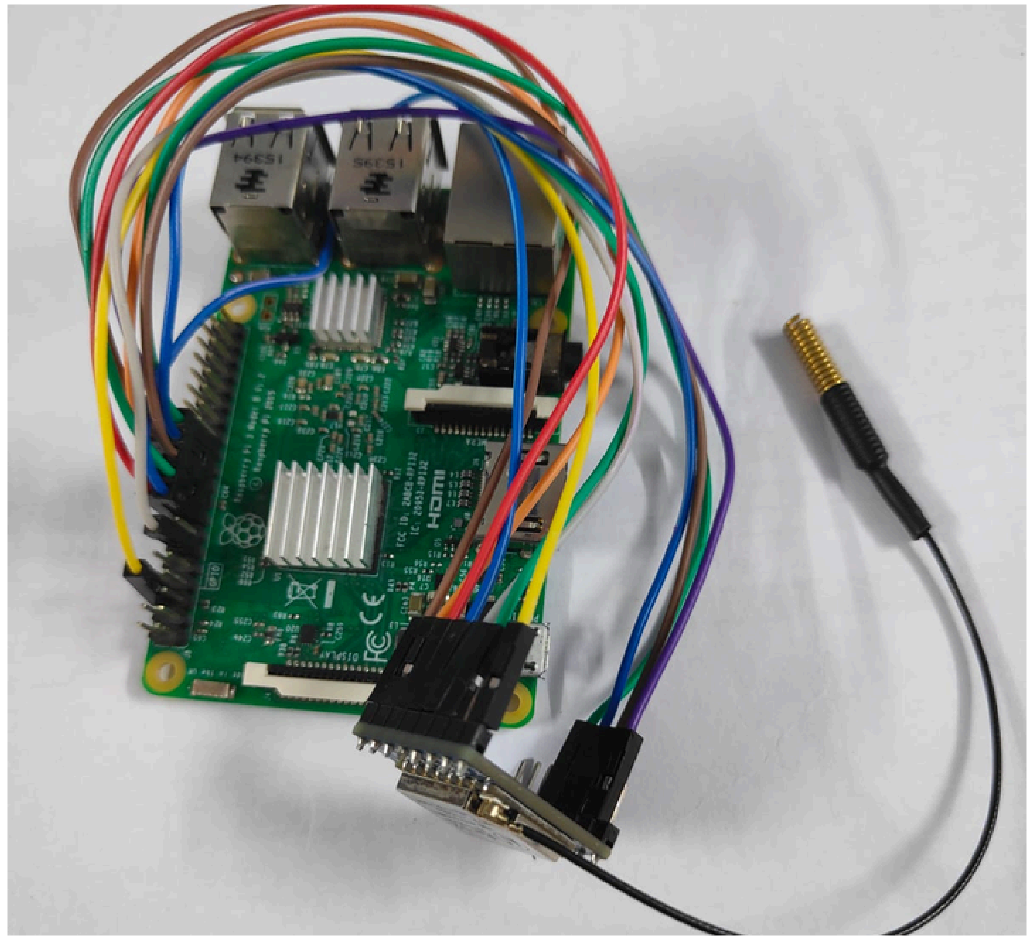
LoRa Comm - Sender

```c
#include "stm32f4xx.h"
#include <string.h>

#define LORA_NSS_PIN 8

void SPI1_Init(void){
    RCC->APB2ENR |= (1U<<12);
    RCC->AHB1ENR |= (1U<<0);
    GPIOA->MODER &= ~((3U<<10)|(3U<<12)|(3U<<14));
    GPIOA->MODER |=  ((2U<<10)|(2U<<12)|(2U<<14));
    GPIOA->AFR[0] &= ~((0xF<<20)|(0xF<<24)|(0xF<<28));
    GPIOA->AFR[0] |=  ((5<<20)|(5<<24)|(5<<28));
    SPI1->CR1 = (1U<<2)|(1U<<1)|(1U<<6);
}
void LoRa_WriteReg(uint8_t addr, uint8_t val){
    GPIOB->ODR &= ~(1U<<LORA_NSS_PIN);
    while(!(SPI1->SR & (1U<<1)));
    SPI1->DR = addr | 0x80;
```



RPi Config

```
from SX127x.LoRa import LoRa
from SX127x.board_config import BOARD

BOARD.setup()

lora = LoRa(verbose=True)
lora.set_mode(lora.MODE_STDBY)

version = lora.get_version()
print("LoRa chip version: 0x{:02X}".format(version))

BOARD.teardown()
```



LoRa Comm - Receiver

```
import spidev
import time

spi = spidev.SpiDev()
spi.open(0, 0)
spi.max_speed_hz = 500000

print("SPI loopback test — waiting for 0x55 bytes")

try:
 while True:
 byte = spi.xfer2([0x00])[0] # Send dummy byte to read
 if byte == 0x55:
 print("Received 0x55!")
 time.sleep(0.1)
except KeyboardInterrupt:
 spi.close()
 print("Test stopped")
```

Processing data using Computational Intelligence approaches

- **Preprocessing**

  Using bandpass filter (removes frequencies noise), notch filter (cancels powerline interference), wavelet denoising (reduces disturbances)

- **Adaptive sampling**

  Adjusting sampling rate of the signal when there is a abnormality.

- **1-Dimensional Convolutional Neural Network**

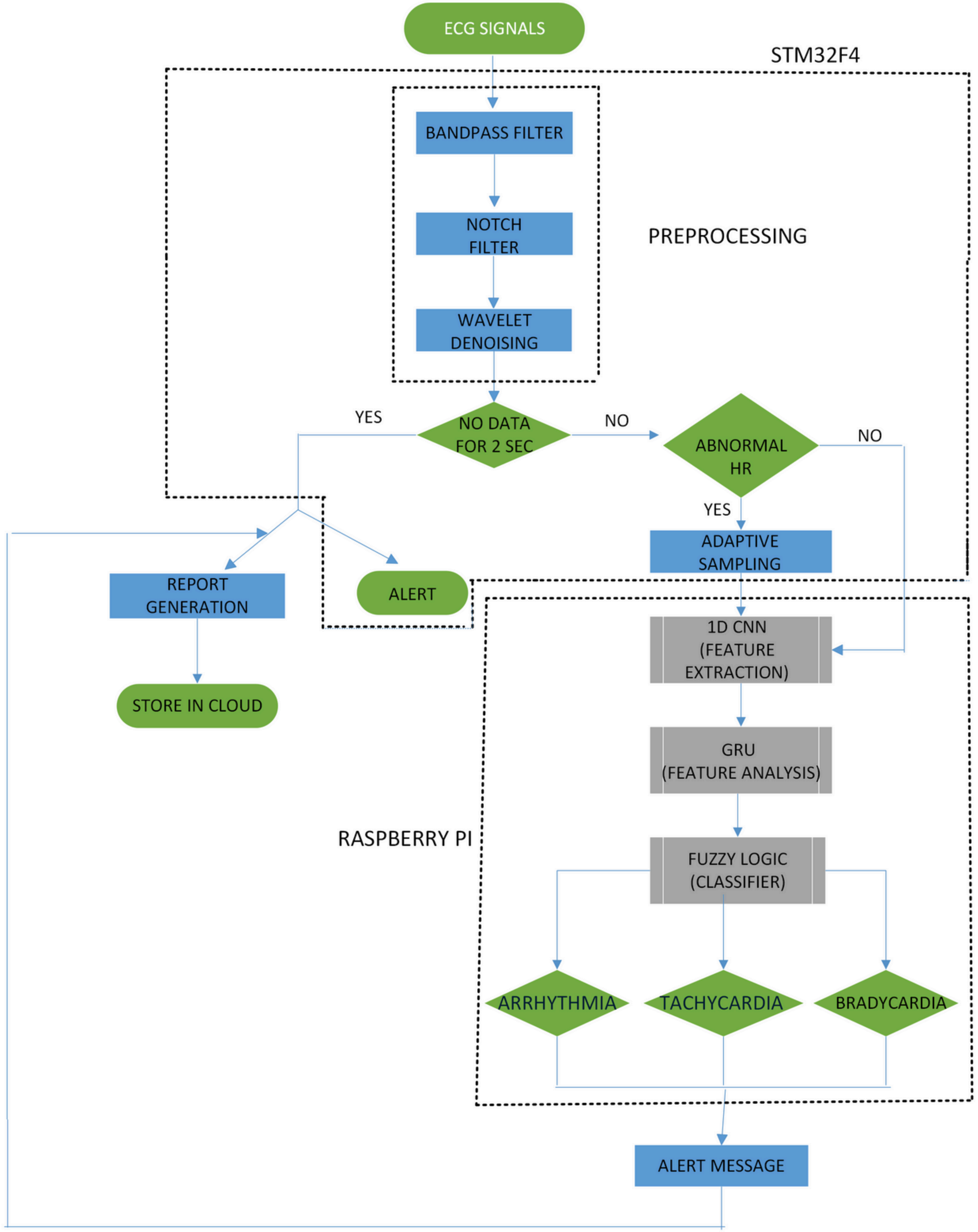
  Used for extracting important features from the ECG waveform.

- **Gated Recurrent Network**

  Used for the final evaluation of the data (to verify whether the data is normal or abnormal)

- **Fuzzy Logic**

  Final decision layer, combines CNN+GRU outputs with expert rules.



Why this CI technique?

Compared to ANN-only Models

🔴 ANN struggles with time-sequence patterns

🟢 Our model: GRU handles temporal rhythm + CNN learns features automatically

Compared to LSTM Models

🔴 LSTM is accurate but heavier (more memory, slower)

🟢 Our model: GRU is lighter, faster, better for Raspberry Pi

Compared to Transformer Models

🔴 Very powerful but computationally expensive

🟢 Our model: lower power, real-time friendly, suitable for embedded devices

Compared to Models without Fuzzy Logic

🔴 Pure AI models may misclassify borderline cases

🟢 Our model: Fuzzy logic adds expert rules → fewer false alarms, more reliable decisions

Comparison between current model and previous model

- GRU is chosen over LSTM because it uses fewer parameters , lower memory usage and trains faster

- 1-D CNN is preferred over Neuro-fuzzy system + IICA(Improved Imperialist Competitive Algorithm) because of its adaptability in feature extraction, runs faster , robust to noise and hardware efficient

- Fuzzy logic is retained because it handles borderline/ambiguous heart rates with human-like reasoning, making final decisions more reliable than a hard threshold.

THANK YOU