**Smart Cradle Control and Monitoring System**

**Project Overview**

The goal of this project is to implement remote control and monitoring of a smart cradle using TTGO T1 and Raspberry Pi. The system includes the following core features:

1. **Sound Detection and Analysis**: Detect the baby’s crying and analyze the sound characteristics to trigger appropriate actions.

2. **Real-time Video Monitoring**: Monitor the baby’s status in real-time using a camera and stream the video to a web interface.

3. **Remote Cradle Control**: Control the cradle’s rocking motion remotely through a web interface.

**System Architecture**

The system is divided into the following main components:

1. **TTGO T1**: Responsible for sound detection and analysis, and controlling the cradle’s rocking motion.

2. **Raspberry Pi**: Handles real-time video monitoring and web server setup, receiving data from TTGO T1 and managing overall control.

3. **Web Interface**: Allows users to view real-time video, sound detection results, and remotely control the cradle’s rocking.

**System Components**

• **TTGO T1**:

• Sound Detection and Analysis Module (connected to a microphone module)

• Cradle Rocking Control Module (connected to a servo motor)

• LoRa or Wi-Fi Module (for communication with Raspberry Pi)

• **Raspberry Pi**:

• Camera Module (for video monitoring)

• Web Server (using Flask or Django framework)

• Data Processing and Storage (sound analysis results and video stream)

• **Web Interface**:

• Real-time video display

• Sound detection result display

• Cradle control buttons (start/stop rocking)

**Detailed Implementation Steps**

**1. TTGO T1 Configuration and Development**

**1.1 Sound Detection and Analysis**

• **Hardware Setup**:

• Connect a microphone module to TTGO T1’s I2S interface or ADC pins for capturing sound signals.

• **Software Implementation**:

• Use the ESP32’s I2S interface to read sound data.

• Implement simple threshold detection to identify the baby’s crying.

• Upon detecting crying, send a signal to the Raspberry Pi via LoRa or Wi-Fi.