**AWS IoT Demo Project Overview**

**1. System Introduction**

This project demonstrates an end-to-end IoT monitoring solution based on the **STM32MP157** platform. It integrates:

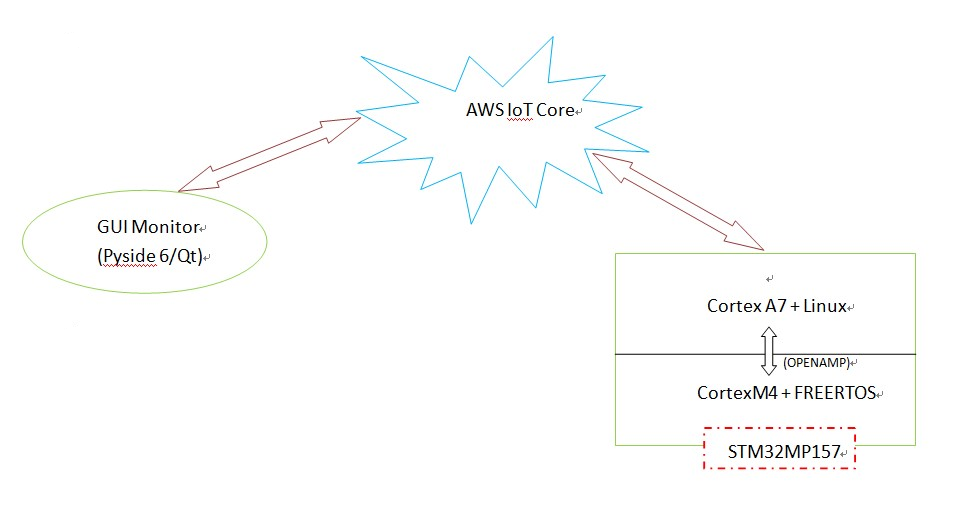
* Real-time sensor data acquisition (temperature, humidity, gyroscope, etc.)
* MQTT communication via **AWS IoT Core**
* A responsive **Python GUI** for data visualization and alerting

The system is designed as a hands-on showcase of embedded-cloud integration, highlighting practical skills across firmware, Linux, cloud messaging, and desktop interface design.

**2. Main Components**

* **M4 core (FreeRTOS):** Reads sensor data and communicates with A7 via OpenAMP
* **A7 core (Linux):** Uses the **AWS IoT Embedded C SDK** to securely publish sensor data via MQTT
* **AWS IoT Core:** Handles authentication and message routing
* **GUI Monitor (Python):** Subscribes to topics, visualizes multi-channel sensor data, and provides configurable alerts (sound, pop-up, email etc.)

The GUI also allows users to configure report intervals, start/stop monitoring, switch alert methods, and export historical sensor data to CSV.



**3. Technical Stacks**

**3.1 Desktop Application**

* **OS:** Ubuntu 22.04
* **Language:** Python 3
* **Libraries:** PySide6 (Qt), AWS IoT Device SDK for Python
* **Tools:** VSCode, venv, dotenv

**3.2 Device - STM32MP157**

* **M4 core:** FreeRTOS v10.2.1, STM32 Cube IDE, STM32 HAL
* **A7 core:** Linux 5.4, CMake
* **Middleware:** OpenAMP, AWS IoT Embedded C SDK
* **Protocol:** MQTT with TLS Mutual Authentication

**4. Further Enhancements**

* Add **email-based alerting**
* Improve **log file management** (rotation, compression)
* Extend to **cloud dashboard integration** (e.g., AWS Lambda + DynamoDB + CloudWatch)
* Add **OTA firmware update** functionality

**5. Note - Important**

* This demo project intentionally **adds small random noise** to the simulated sensor data (e.g., temperature, humidity, gyroscope) **for visualization purposes only**.
* In a real deployment scenario, this noise generation should be **disabled or removed** to ensure accurate and clean sensor readings.