## Embedded Development Roadmap for IoT Device Registration and Verification

### Overview

This system is designed for the registration and identity verification of smart home devices. The primary objectives are:

* Alleviate the performance issues caused by the centralization of the Internet of Things (IoT) system.
* Prevent unauthorized devices from connecting to the system.

### Milestone 1: Device Production and Initial Setup

**Timeline:** 1 week

**Tasks:**

**PUF extraction:**

* Each device is equipped with SRAM-based PUFs.
* A unique private ID is generated for the PUFs.
* Implement a mechanism to extract PUF

**Initial Device Setup:**

* Challenge the PUF of the device to obtain its response.
* This response is treated as the private ID of the device.
* Compute a cryptographically secure hash for the ID.

### Milestone 2: Device Registration in Local Blockchain

**Timeline:** 2-3 weeks

**Tasks:**

**Pre-registration Checks:**

* The local administrator checks if the device exists in the local blockchain dataset.
* If the device exists, the registration process is terminated.

**Key Generation and Encryption:**

* The administrator generates a random key and programs it into the device's OTP memory.
* This key is sent to the gateway of the local IoT system.
* The gateway encrypts the key using asymmetric encryption and returns it to the administrator.

**Blockchain Registration:**

* The device ID, hash value of the device's private ID, and the encrypted key are uploaded to the local blockchain.
* This completes the device registration process.

### Milestone 3: Regular Device Verification

**Timeline:** 2-3 weeks

**Tasks:**

**Data Retrieval from Blockchain:**

* The gateway queries the local blockchain dataset to obtain the device's public ID.
* The blockchain returns the hash value of the device's private ID and the encrypted key to the gateway.

**Key Decryption:**

* The gateway decrypts the encrypted key using its own secret key.

**Device Authentication:**

* The gateway reads the random key stored in the device.
* It compares this key with the decrypted key.
* If they match, the device is verified. Otherwise, it is marked as a "Failed device".

**Note:** The devices connected to the IoT network will undergo regular scans and verifications to ensure their authenticity and prevent unauthorized access.