# The 1% Project: Polka-Resilient-Actuator (Polka-RA)

You have a powerful and unique set of components. We will pivot the "Polka-DePIN-Bot" concept from a mobile robot fleet to a **Decentralized, Resilient, Off-Grid Actuation and Monitoring System**—a project that maximizes the use of your specific hardware and directly addresses the "Resilient Apps" theme in a way no other team is likely to attempt.

This project, **Polka-Resilient-Actuator (Polka-RA)**, focuses on verifiable, off-grid control, which is a critical need in regions with unstable infrastructure.

## 1. The "Woah" Factor: Off-Grid, Verifiable Actuation

The core innovation is using the **LilyGO TTGO T-Call ESP32 SIM800L** to establish a resilient, off-grid connection to the Polkadot network via GSM/GPRS. This proves that the system is truly **resilient** and not dependent on local Wi-Fi or fiber, which is a major win for the "Resilient Apps" theme.

### Project Name: Polka-Resilient-Actuator (Polka-RA)

**Hackathon Track:** User-centric Apps / Polkadot Tinkerers (Advanced)

**Core Function:** A decentralized, on-chain system for remotely controlling and verifying the execution of physical tasks (e.g., watering, monitoring) in an off-grid environment.

## 2. Hardware Mapping and Technical Implementation

We will use your available components to build a sophisticated, multi-functional prototype.

| Component | Role in Polka-RA | Your Expertise Used | Polkadot Integration |
|---|---|---|---|
| **LilyGO TTGO T-Call ESP32 SIM800L** | **Resilient Gateway.** Uses the SIM800L (GSM/GPRS) to connect to the Polkadot network via a lightweight client. | **Embedded Systems, Programming** | Submits transactions (Proof of Actuation) and listens for on-chain commands. |
| **ESP-CAM** | **Verifiable Sensor.** Captures images of the physical action (e.g., water flowing) before and after actuation. | **AI (Image Processing)** | Captures images, hashes them, and sends the hash to the Polkadot chain for immutable verification. |
| **Arduino Mega / Nano RP2040** | **Actuator Controller.** Controls the motors, pump, and relay based on commands received from the ESP32. | **Programming, Robotics** | Controls the **8 Servos** (e.g., for a multi-valve system), **Nema 17** (e.g., for linear movement), **Pump**, and **Relay**. |
| **Relay, Pump, Servos, Nema 17** | **Physical Actuators.** The devices that perform the "useful" work. | **Robotics, Embedded Systems** | The physical output that is controlled by the decentralized command. |

## A. Polkadot Integration: The Resilient Command Pallet

We will use a custom **Substrate Pallet** (or a smart contract on a Polkadot Parachain like Moonbeam) to manage the resilient command flow:

1. **Command Submission:** A user submits a command (e.g., `ActuatePump(duration)`) via a dApp. This command is stored on-chain.
2. **Off-Grid Polling:** The **LilyGO TTGO T-Call** periodically wakes up, connects via GPRS, and queries the Polkadot chain for new commands addressed to its unique device ID.
3. **Verifiable Actuation:**
   - The device captures a **"Pre-Actuation"** image hash via the ESP-CAM and sends it to the chain.
   - It executes the command (e.g., turns on the pump via the Relay).

- It captures a **"Post-Actuation"** image hash and sends it to the chain, along with a **Proof of Actuation** (a signed transaction confirming the action was taken).

4. **Automated Payment/Reward:** The Polkadot chain verifies the Proof of Actuation and automatically releases a small reward to the device's wallet for completing the task.

### B. The "Wow" Factor: Demonstrating Resilience

The key to winning is the narrative around the **LilyGO TTGO T-Call**.

- **Scenario:** Demonstrate the system operating in a location with **NO Wi-Fi**. Show the GPRS connection status and the successful submission of a transaction to the Polkadot chain.
- **The Proof:** The dApp will display the two image hashes (**Pre** and **Post** actuation) stored immutably on the Polkadot chain, proving that the physical action was not only commanded but **verifiably executed** in an off-grid, resilient manner.

## 3. Judging Criteria Optimization

This adapted project is now hyper-optimized for the hackathon's theme and your resources.

| Criteria | Polka-RA's Edge |
|---|---|
| **Technological Implementation** | **Highest Score.** Requires complex, low-level programming for the **ESP32 SIM800L (GPRS)**, integration of multiple microcontrollers (**Mega/Nano**), and the use of **Rust/Substrate** for the on-chain logic. This is a highly advanced technical stack. |
| **Resilience** | **Perfect Alignment.** The use of **GPRS/GSM** for off-grid communication is the ultimate demonstration of a "Resilient App" in a challenging environment like Burundi. |
| **Creativity** | **High.** It moves beyond simple data logging to **Decentralized Actuation**, using the blockchain not just for data, but for resilient, verifiable **command and control** of physical hardware. |
| **Potential Impact** | **High.** The model is immediately applicable to off-grid infrastructure, remote monitoring, and automated farming in developing regions, directly addressing the "radically useful" motto. |

# 4. Action Plan for the Hackathon

1. **Phase 1: Resilient Communication (LilyGO)**
    - Get the LilyGO TTGO T-Call to successfully connect to the internet via GPRS.
    - Implement a lightweight Polkadot client (e.g., using a library or raw HTTP/WebSocket calls) to submit a simple transaction.
2. **Phase 2: Verifiable Actuation (Mega/Nano/ESP-CAM)**
    - Write the firmware to receive a command from the ESP32 and actuate the **Pump** and **Relay**.
    - Integrate the **ESP-CAM** to capture and hash images before and after actuation.
3. **Phase 3: Polkadot Pallet & Demo**
    - Develop the `Command-Pallet` in Substrate to store the command and verify the image hashes.
    - Create a compelling video that shows the entire loop: **On-chain Command -> Off-Grid GPRS Transmission -> Physical Actuation -> On-chain Verification (Image Hashes)**.

This project is a guaranteed showstopper and perfectly tailored to your unique strengths and resources.