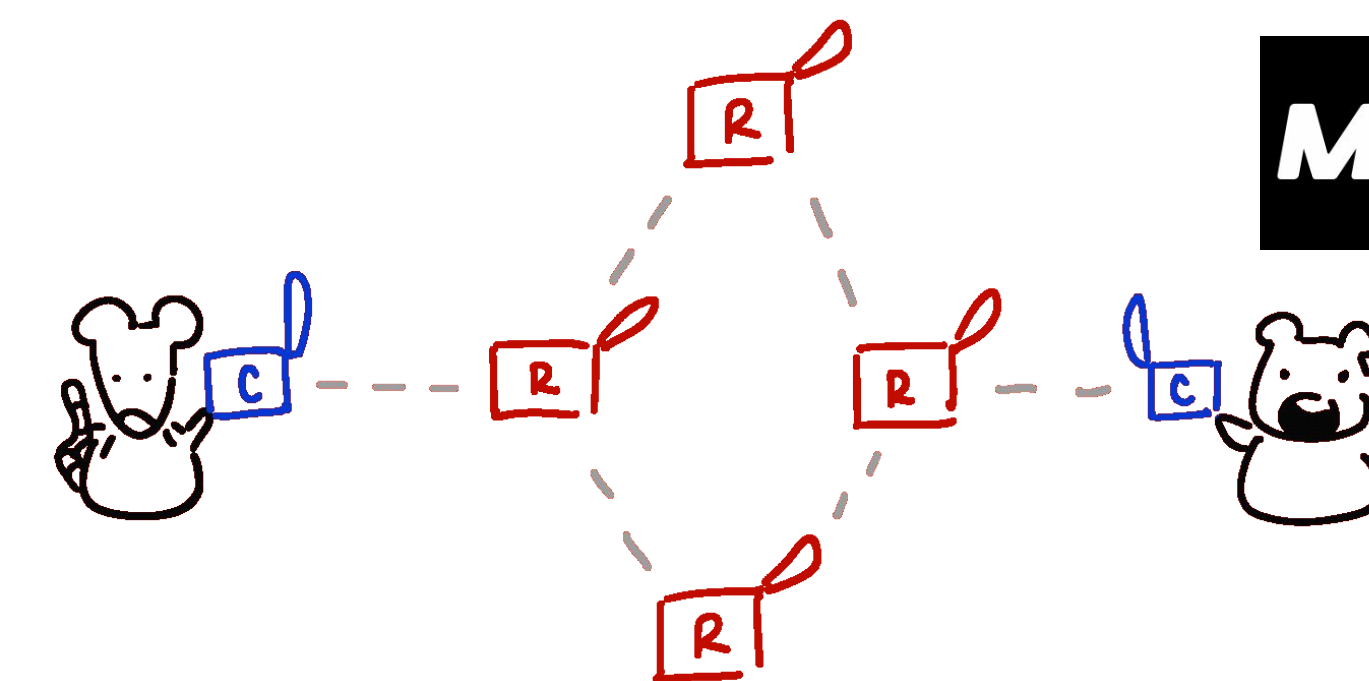
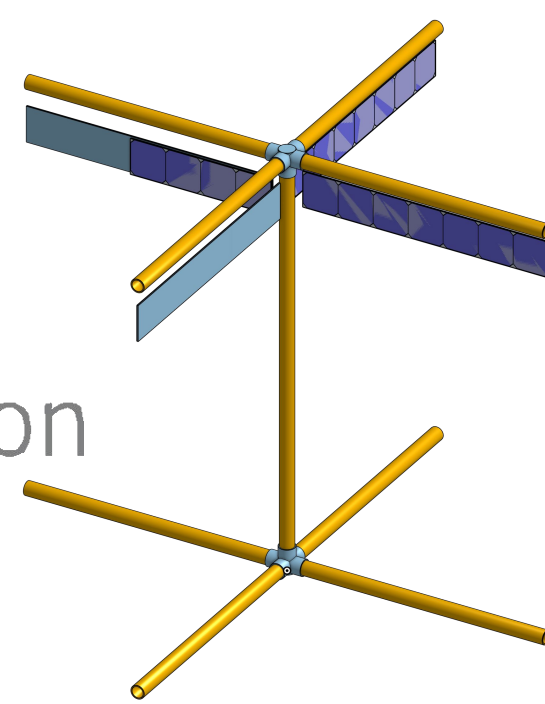




Introduction

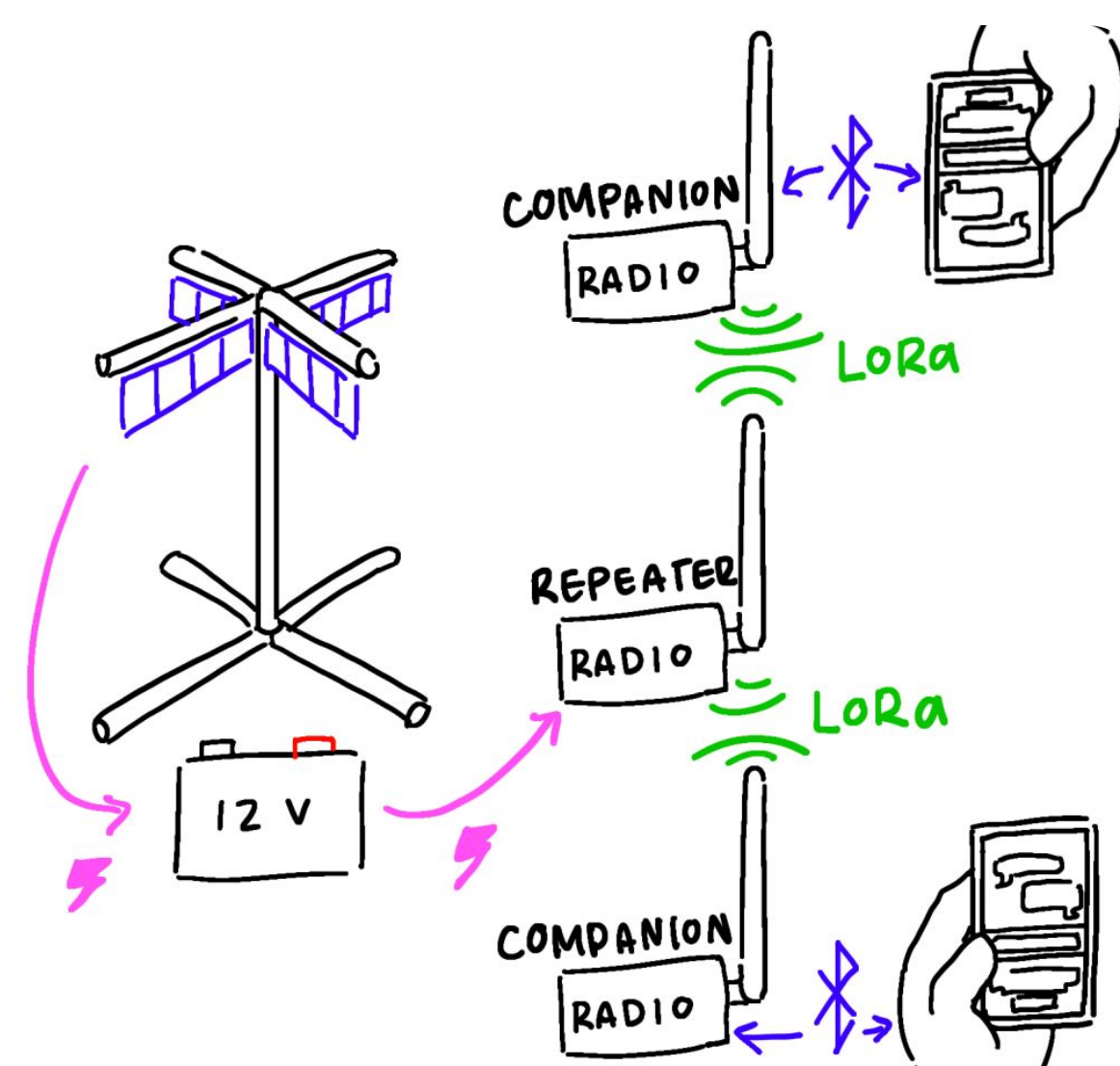
Modern cellular infrastructure and wide area networks (WAN) are expensive and gatekept by large corporations and governmental regulations. The average person cannot afford to build a \$100,000 to \$1,000,000 cell tower, nor can they afford to buy a spectrum. In addition, cellular networks tend to not cover rural areas, leaving individuals without communications. Censorship in war zones and areas under oppressive regimes also **reveal the need for a decentralized, accessible network.**

The Goals

The two main goals are (1) accessibility and (2) self-sufficiency via solar. (1) Our solution should scale down to underserved areas and be reasonably cheap to deploy. (2) The solution should not rely on existing infrastructure and can be deployed independently.

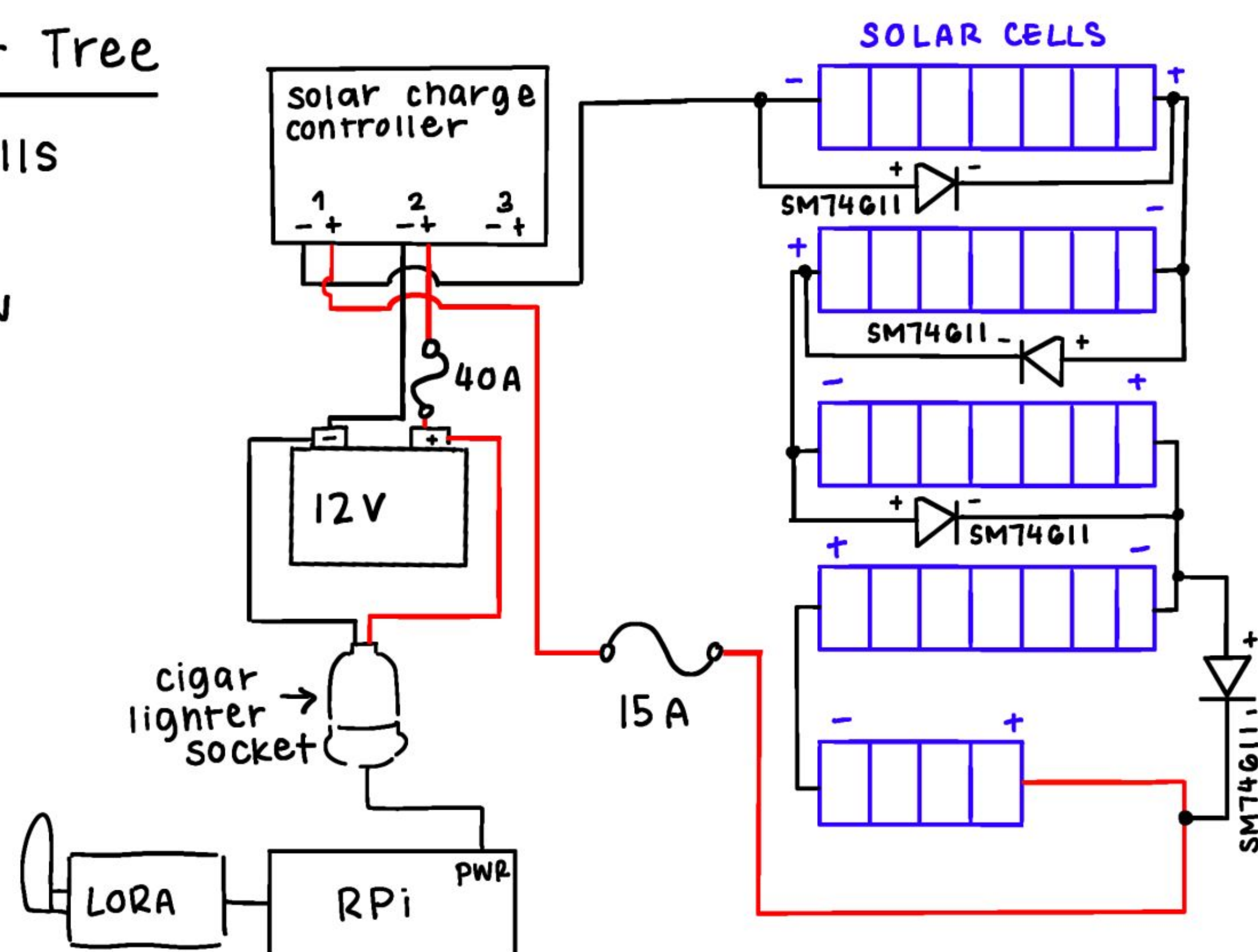
The Solution

A solar tree is to power long range (LoRa) radios using the MeshCore firmware to build a decentralized, self-healing mesh network. The solar tree is to be built of PVC pipes and solar cells to charge a 12V battery, which will power a LoRa radio. The solar panels are to be hung vertically to allow snow to fall through.



Solar Tree

32 cells
~19 V
~115 W



Methodology



3.6 W solar cells



12V 25Ah
LiFePO₄ battery



Solar charge
controller



Seed Studio
XIAO ESP32-S3
boards



Raspberry Pi

The Process

Solar panels. The solar cells were soldered together and sandwiched between two sheets of clear acrylic. They were sealed with 5200 marine sealant for its waterproof characteristics.

Wind-resistant panels. Paper clips were epoxied onto the panels to provide a hook from which they could be zip-tied to a pipe branch.

Software. Commands enable YOLOv8n object detection and SmolVLM vision-language inference over the mesh. Workloads spawn on-demand and idle-shutdown conserves power. This allows the user to send and receive on-scene updates from the tree.

On-device AI. Inference models run on Pi's 4GB RAM and CPU using quantization to fit in memory with zero cloud dependency.

Results

Object detection & alerting. Over a 46-minute session, the node caught 36 raw detections but only sent 11 mesh alerts, cutting redundant traffic by 69% while still catching sustained activity. Confidence ranged 55–93%.

On-device scene description. A `CP_PHOTO -D` command returned a generated description in 281 seconds. Text-only queries returned answers in seconds.

View the rest of our
results on our website!



Next Steps

The next step is to leave the solar tree outside for a year with periodic check-ups to evaluate its resistance to all four seasons.