blues wireless

Building an IoT Remote Control Device

“No matter what device you are managing or where it is deployed, you can embed a cellular IoT solution for full remote control.”

Zachary J. Fields
Senior IoT Content & Projects Engineer

Instructions on Hackster

IoT has made remote work possible across industries, now businesses that want to stay ahead of the curve are seeking new ways to leverage the IoT to its full potential. One of the most powerful aspects of the technology is the ability to securely and remotely control devices at scale from anywhere in the world. IoT remote control devices reduce costs by giving you greater insight into your assets.

When building proof-of-concept or prototype IoT remote control devices it is important to spend most of your time on features that solve business problems, not utility functionality. To that end, Blues Wireless Notecard is the simplest, and most cost-effective way to add connectivity to IoT devices. Simply connect the Notecard to your device’s existing UART or I2C bus, and it will connect your device to the cellular network automatically, ready to transmit and receive data from its associated Notehub account.

Learn how to build a cellular IoT remote control prototype in a single day for around $200.

Cellular IoT for Remote Control Devices

No matter what device you are managing or where it is deployed, you can embed a cellular IoT solution for full remote control. Configuration updates and bug fixes can be sent to the device over the cloud, with cloud-based reporting to proactively monitor vitals. This allows you to take control of and optimize your operations from anywhere in the world, giving you the ability to:

- **Control Any Product From Anywhere** - Create an interconnected system to remotely control your fleet of devices from anywhere in the world.
- **Fix Bugs** - Avoid recalling devices that are deployed in the field with OTA firmware updates managed by the Blues Wireless Notehub service.
- **Predict Maintenance** - Maximize uptime of your connected devices by monitoring key sensor data and receiving alerts of potential failures.

Whether building new devices or retrofitting outdated technology or analog devices, the Blues Wireless Notecard can help. Notecard is a 30 by 35 mm SoM with a built-in eSIM and data package that provides cellular connectivity and data routing to the cloud. Developers can unbox the product, and start sending arbitrary data bidirectionally between the device and their preferred cloud application in less than 30 minutes. Take it one step further, and the Notecard can remotely update firmware and even securely send commands to an entire fleet of devices. Below you'll see the components of a device build including the Notecard.

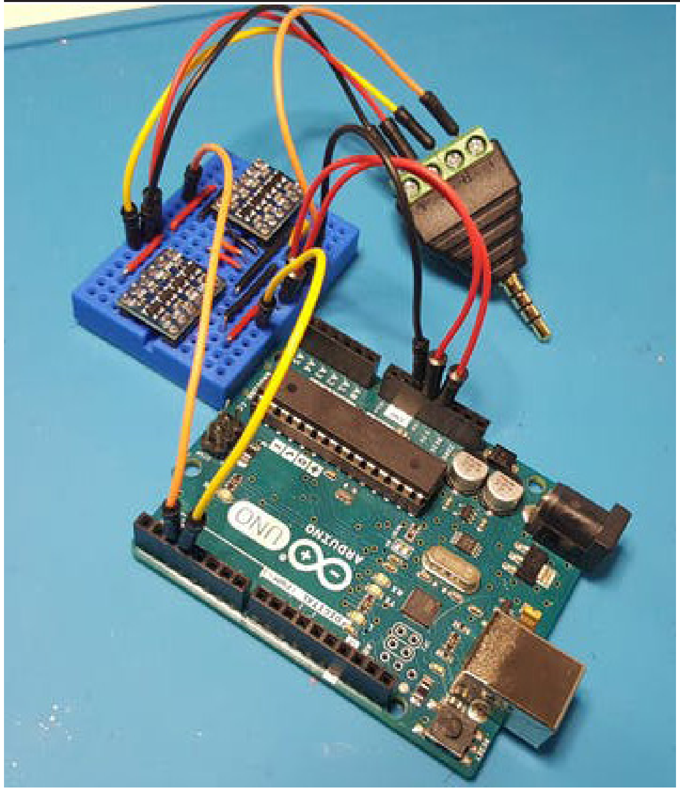
Behind the Remote Access IoT Device

This project demonstrates how to make older tech devices IoT-compatible with Blues Wireless hardware and firmware. In this case, it was updating Nintendo R.O.B. technology from the 1980s and enabling remote control through its own website. You can find the complete source code for the project at the GitHub repository linked below and complete project assembly instructions on Hackster.

GitHub: https://github.com/zfields/nes-rob

Hackster: https://www.hackster.io/zachary_fields/cellular-r-o-b-with-blues-wireless-38ac41

Price: $194.04	Languages: C++
Lines Of Code: 37	
Project Time: 8 Hours	



Hardware

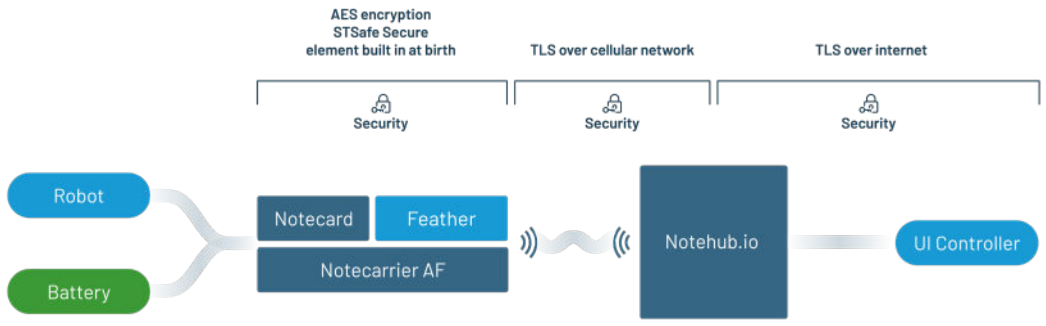
- Nintendo R.O.B.
- Blues Wireless Feather Starter Kit
  - Notecard SoM
  - Notecarrier-AF
  - 2 U.FL connector cables
  - Adafruit HUZZAH32 - ESP32 Feather
- SparkFun Logic Level Converter - Bi-Directional
- Capacitor 1000 µF
- Capacitor 100 µF
- Male/Male Jumper Wires
- Female Header 8 Position 1 Row (0.1")
- Generic Jumper (0.1")
- Board-To-Board Connector, 2 mm
- SparkFun Snappable Protoboard
- Battery Contact, Button
- Battery Contact, Plate
- Battery Contact, Spring
- TRRS 3.5 Panel Mount Jack
- TRRS Coiled Cable
- Wire-To-Board Terminal Block (4 position)

Software apps and online services

- Arduino IDE
- Blues Wireless Notehub.io
- Blues Wireless Arduino Library
- NesRob Arduino Library
- Autodesk Tinkercad.com

The main parts of the project are:

- Disassembling the old device and identifying components.
- Updating hardware and creating a Brainstem Access Port.
- Solving for voltage differences.
- Connecting Blues Wireless Feather Starter Kit for remote control.
- Setting up cloud-based access.



Building a Remote Access IoT Device Prototype

Building connectivity into an existing product increases the potential of the device and can often be a more prudent financial decision than upgrading equipment. But whether you're retrofitting a device or building a new one, adding cellular connectivity can be one of the most challenging and complicated pieces. Using Blues Wireless' hardware, firmware, and data routing service gets your device connected in less than 30 minutes.

Hardware

- The Blues Wireless Notecard is a tiny 30 by 35 mm SoM device-to-cloud data pump that provides instant connectivity, comes with 500 MB of data over 10 years, and is ready to embed in a custom project.
- The Notecarrier is an expansion board with features perfect for prototyping. The best board for this project is the Notecarrier-AF, because it has an Adafruit Feather socket for the ESP32, an M.2 connector for the Notecard, and handles all the power management and charging circuitry.

Firmware

- Notehub has the ability to manage the firmware running on your Notecards as well as your host MCU. You can easily choose between the available firmware and deploy to one or more devices through OTA DFU.

Cloud

- The Notecard is preconfigured to securely communicate with Notehub.io.
- Notehub.io enables secure device-to-cloud data flow and allows the Notecard to function as a bi-directional device that can receive data in addition to publishing data.

Notecards are assigned to a project in Notehub.io, then sync data into those projects for routing to your cloud application. After getting your project set up on Notehub.io, you're ready to write the Notecard configuration firmware that will enable you to send sensor readings from your device to Notehub. The Notecard uses a JSON-based API, taking JSON requests and returning JSON responses. You don't have to learn or use any AT commands to work with this device. If you're interested in learning more from Blues Wireless, there's a Quickstart on the developer portal that introduces the basic concepts and commands.

With the Notecard communicating with Notehub, you can send commands over the internet through Notehub.io to take control of your device. Notehub.io can be used to communicate with the Notecard, set environment variables, and more. By sending a note.add request, you can queue a message on Notehub.io that will be sent to the Notecard. In this build, once synchronized, the ESP32 retrieves the note and sends the commands the device needs to perform.

Applications of This Project

IoT remote control is useful for any situation in which you want to securely manage IoT devices at scale from a remote dashboard. You can monitor device sensor readings, system and network performance, hardware vitals, and proactively troubleshoot issues. Some applications include:

- Robotics
- Smart home products
- Industrial machine and equipment sensors
- Commercial pumps
- Environmental monitoring systems
  - Air quality monitors
  - Water quality monitors
- Patient monitoring devices
- Building management systems
- Interactive kiosks

Start building your prototype today using these instructions and a dev kit from Blues Wireless.