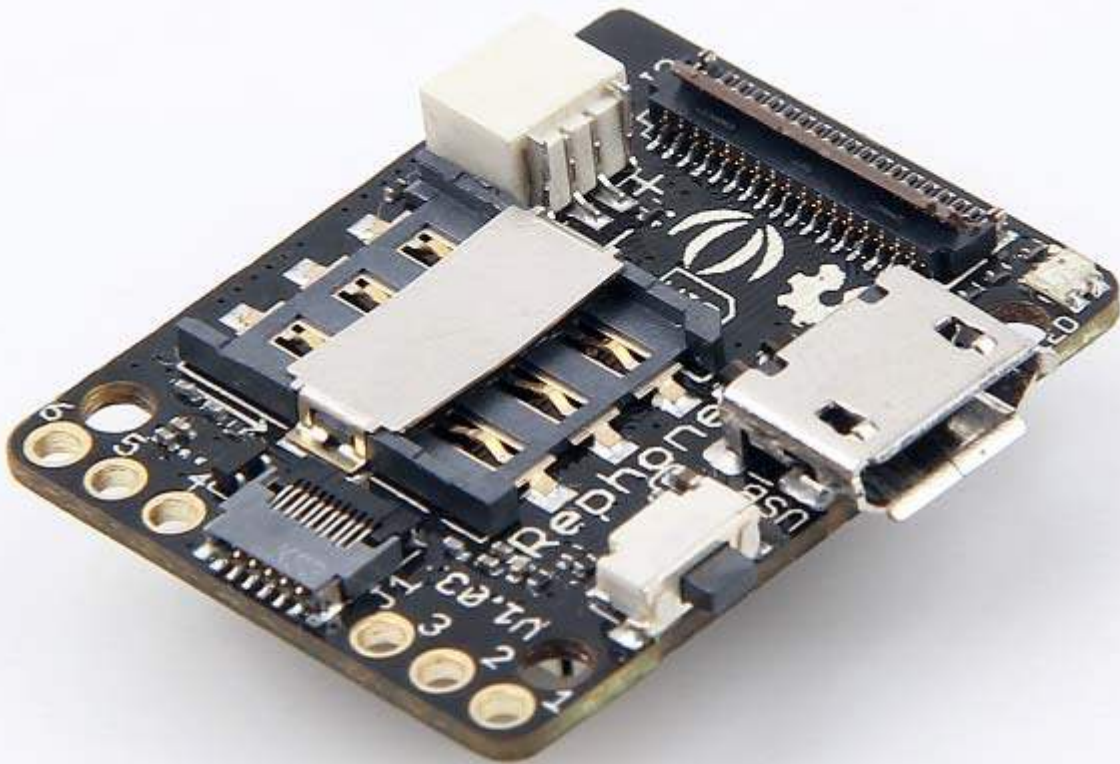


Xadow - GSM+BLE



Whether making and receiving telephone calls over a radio link using external speakers and microphone or exchanging data in short distances with Bluetooth, you can do it with Xadow GSM+BLE.

As the heart of RePhone kit Create, the Xadow GSM+BLE is built around the powerful System-On-Chip (SOC) MT2502, offering a rich range of communication protocols - GSM, GPRS and Bluetooth (v4.0 and 2.1 dual mode). It supports quad-band 850/900/1800/1900MHz that covers any GSM network in the world. Just insert a 2G Nano SIM card, and then you can enchant things with cellular connectivity.

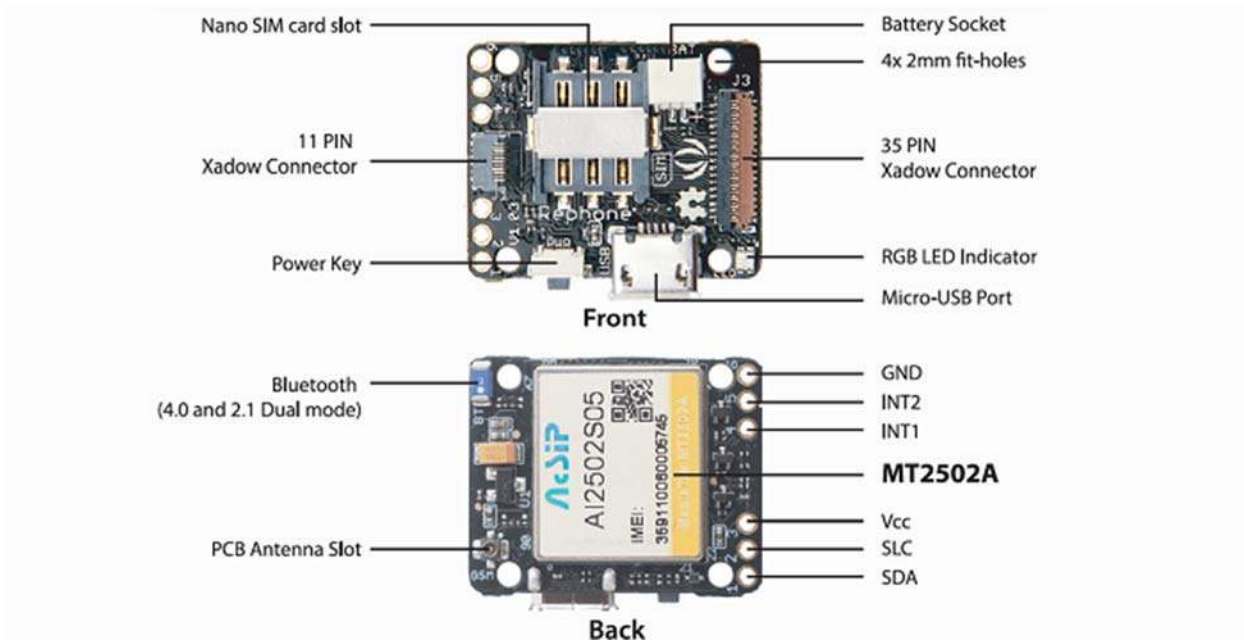
Features

- Based on the smallest commercial System-on-Chip
- (5.4mm x 6.2mm) currently on the market
- Open source and modular design
- Slim and small
- Built-in Xadow Connectors for easy pluggable FPC cables
- Stackable, chain-able and sew-able with other Xadow Modules
- Core module for most of RePhone Kits
- Perfect choice for long distance and short-range communication

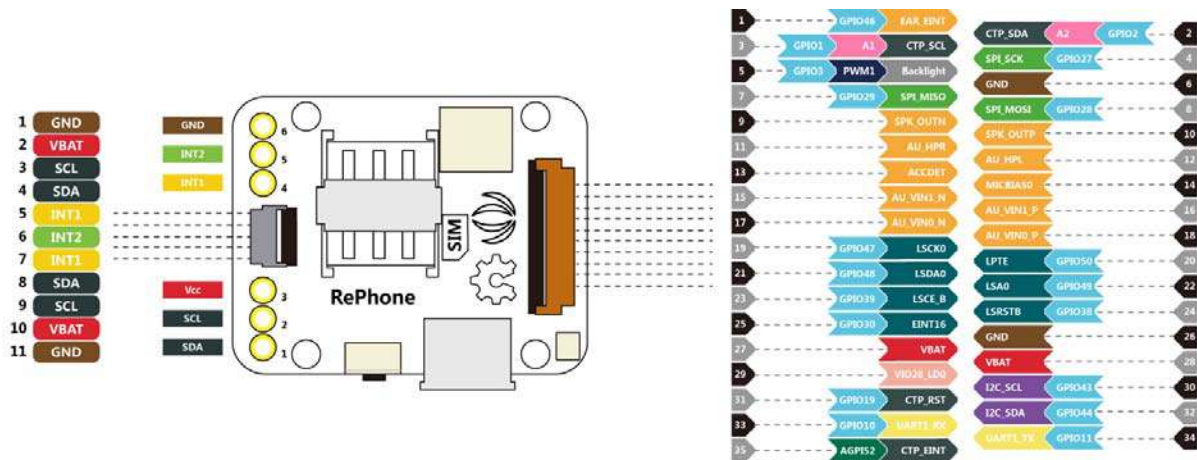
Specification

Microcontroller	MT2502
MCU Core	32-bit ARM7EJ-STM RISC processor
RAM	4 MB
Flash Memory	16 MB
Power Supply	3.3 ~ 4.2V(no SIM)/3.5 ~ 4.2V(with SIM)
Power Consumption	20mW/30mW/52mW @ standby(no radio)/standby(GSM)/standby(BT)
Quad-band	850/900/1800/1900 MHz
GPRS	Class 12 modem
Clock Speed	260 MHz
Connectors	35 PIN Connector & 11 PIN Connector for Xadow Modules; JST 1.0 Connector for the battery
Interfaces	LCD, Audio, I2C, SPI, UART, and GPIOs etc
Dimensions	25.37mm x 20.30mm / 1" x 0.8"

Hardware Overview



The image below illustrates the Pin Definitions of 11 Pin Xadow Connector, Solder-able Breakout Pins and the 35 Pin Xadow Connector in a sequence from **Left to Right**.



Power Up

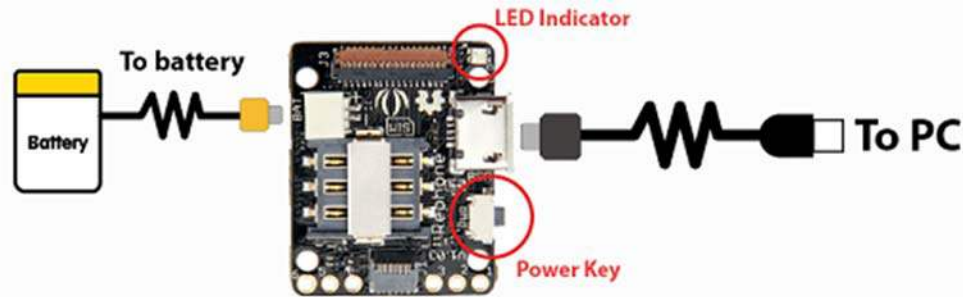
Power up the RePhone/Xadow GSM+BLE with a battery of **3.5V ~ 4.2V** and **JST 1.0 male connector**, or power it up through the breakout **PIN 3(VCC)** and breakout **PIN 6(GND)** as illustrated in the Hardwareview above.





Battery Charging

Xadow GSM+BLE uses the JST 1.0 female connector for the battery, you can charge the battery by simply connect your board with a USB cable.

Operating Mode

The Xadow GSM+BLE has two operating mode when you power it up and connect it with PC - **press and hold to power key for 2 seconds** to turn the module **ON** or **OFF** to access the **Mass Storage Mode** or **Flash/Debug Mode**.














	Status	LED Indicator Before Connect to PC	LED Indicator After Connect to PC
Mass Storage Mode	Turned OFF	OFF 	Red to Green 
Flash/Debug Mode	Turned ON	Green 	Green 

Mass Storage Mode

When the Xadow GSM+BLE is **OFF**, connect the board (with battery connected) to PC via Micro USB cable, you can access the 5MB 'Mass Storage Mode' on PC. All the applications (vxp files) and system settings are stored in this 5MB disk.

Name

-  @font
-  MRE
-  address_book.txt
-  autostart.txt
-  ifttt_book.txt
-  R2D2.mp3
-  result.txt
-  ringtone.mp3
-  settings.txt
-  sms_book.txt
-  sms_inbox.txt

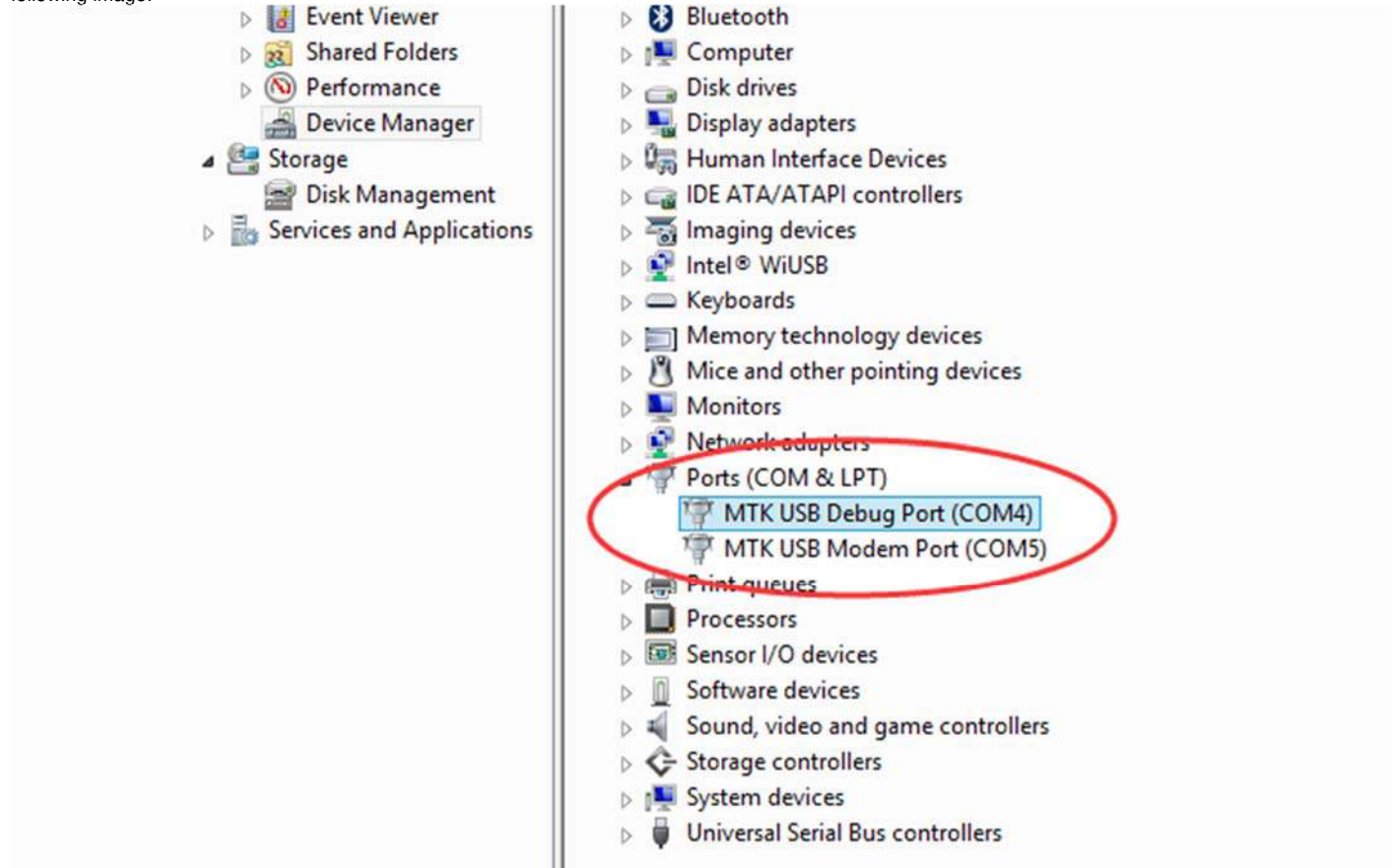
Flash/Debug Mode

When the Xadow GSM+BLE is **ON**, connect the board (with battery connected) to PC via Micro USB cable, you can find two **COM ports** on your 'Device Manager' :

- **MTK USB Debug Port(COM4)**
- **MTK USB Modem Port(COM5)**

The COM number might be different on your PC. Each COM port has different function according to the development environment you use, please refer to the Getting Started section for more details.

Open **Device Manager** by clicking the **Start** button, clicking **Control Panel**, clicking **System and Security**, and then, under **System**, clicking **Device Manager**. If you're prompted for an administrator password or confirmation, type the password or provide confirmation. See the following image:



Getting Started

We have developed rich libraries hooking into Arduino IDE, Lua and JavaScript, with detailed example sketches to help entry-level programmers develop with RePhone modules easily and quickly.

We also provide a powerful SDK based on Eclipse IDE for C/C++ developers to work with more intelligent applications.

Let's Start with ARDUINO IDE

Let's Start with ECLIPSE IDE

Let's Start with LUA Language

Let's Start with JAVASCRIPT

For more info you can refer to the RePhone Development Environment at RePhone Main Page:

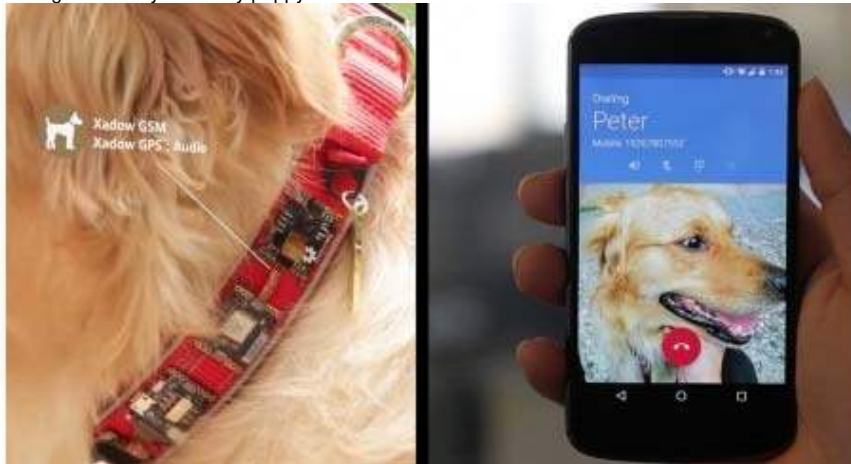
RePhone Development Environment

Related Projects

Check on awesome RePhone projects that has been achieved with RePhone.

A Traceable Dog Collar

5 steps to make a traceable dog collar for your lovely puppy.



RePhone Community



We've been looking for a better place where our backers (RePhone Users) can sit together, warmly and comfortably, have conversations about RePhone, discuss technical problems, share ideas/projects, and give feedback on the modules' development in the future. And then here we go, the RePhone Community.

Now join us in the RePhone Community! Together we seek answers, make interesting stuff, care about each other, and share our experiences.