

0 items in cart

\$ USD

You are **not** logged in.[log in](#)SparkFun Electronics 

Wireless

Buying Guide

by Nate | May 12, 2009 | [14 comments](#) [Home](#) | [Tutorials](#) | [Wireless Buying Guide](#)

Skill Level: ☆ Beginner

Wireless Buying Guide

There's a lot of different ways to get data from a remote unit to somewhere else. In this guide, we attempt to show you different technologies and the tradeoffs between those technologies. Remember, we are completely biased. We've used a lot of these modules, and we apologize to the companies that spent a lot of effort designing them. This is just what we think!

	Power	Distance	Data Rate	Data Delivery	Cost	Learning Curve	Size
Nordic	5	1	2	2	5	3	5
Cellular	1	5	4	5	1	5	1
Bluetooth	3	3	4	4	3	5	3
Zigbee	4	1	2	2	4	2	3
XBee	3	3	3	3	4	4	3
WiFi	2	3	5	5	2	3	3
FM	4	4	*	*	4	4	4
General	4	4	1	1	5	1	4

*FM radio does not do data

5 = Lowest	5 = Longest	5 = Highest	5 = Guaranteed delivery	5 = Cheapest	5 = Shortest time learning	5 = Smallest
1 = Highest	1 = Shortest	1 = Lowest	1 = May not get there	1 = Most expensive	1 = Lots of time learning	1 = Largest

Pictured above is our **quick and dirty breakdown**. Now let us explain ourselves...

Before you get started, think about your project goals. Then answer these questions:

- Is battery life a concern?
- How far does the wireless link need to transmit?
- How much data does your system need to transmit?
- Is it ok if some data gets lost? Or do all packets absolutely have to arrive at the base?
- How much are you able to spend?

- How much time do you have to play?
- How small does it need to be?

Answering these questions will help you significantly narrow down the search for the RF solution that best fits your project.

Features that are addressed by the questions:

- Power
- Distance
- Data rate
- Data delivery
- Cost
- Learning curve
- Size

Potential RF Solutions:

- Nordic
 - Cellular
 - Bluetooth
 - Zigbee
 - XBee
 - WiFi
 - FM
 - General
-

Power:

Depending on the type of solution, power consumption can range from micro amps to amps (6 orders of magnitude). Historically, RF requires a lot of power. If you're trying to broadcast pirate radio, expect to not be able to run your station off batteries. However, if you need to chirp some data every few minutes or on the hour, you can significantly reduce your power and do some fun sleep mode tricks.

Connecting a device to the power grid will open up many RF options. On the other hand, if you're planning on making your RF device portable, consider how much mAh (mili-amp hour) time your battery has and how often you need to transmit your data. If you've got a 50mAh coin cell and your RF device uses 12mA when receiving, you've got $50/12 = \sim 4$ hours of run time. Now if you power cycle the RF device (for example, in receive mode 5% of the time), you can significantly increase your battery life ($50/0.6 = \sim 83$ hours = ~ 3.5 days run time).

Distance:

Distance is related to power. If you need to transmit ~10-50 meters (32 to 164ft), there are lots of options open. If you need to broadcast over the **African Sahara**, you may need a bit more **umph**. Some of the **XBees** claim miles of range, but this is usually with very directional antennas and with a bit of power behind it (210mA at 3.3V). In embedded electronics land, we deal with mW (mili-Watts). If you need watts of power for your pirate radio station, you are probably talking about miles of transmission range, but you are not going to get it with a **coincell battery**.

Data Rate:

Do you need to download torrents? Or do you need to receive a 4-byte analog to digital conversion? Data rate is everything. Some of the **simple data links** are limited to ~300 bytes per second (2400 bps). You can achieve much higher datarates (WiFi being the forerunner) but this adds overhead, complexity, and power consumption. Low power goes hand in hand with low bandwidth. A small battery usually means you will be able to broadcast a handful of bytes (hundreds) every few seconds. If you're looking to stream audio data off **Pandora**, you are probably looking at the wrong tutorial.

Data Delivery:

Do I really care if my data gets to where it needs to be? Sounds funny, no? Well it's actually quite applicable. RF is a finicky... thing. One second I can get 100 bytes where it needs to be and the next minute one of the bytes get corrupted.

Because we cannot see high frequency RF does not mean it is not there. There is a lot of radiation coming from normal house hold wiring, computer monitors, the sun, even static electricity. There are a ton of EMI (electromagnetic interference) sources. And we really cannot predict where the interference will come from. Therefore, when we broadcast 'Hello world', it may get garbled on its way to the **lunar lander**. One of the ways to protect against this is verifying that the data was delivered correctly. Basically we wait for the receiver of the information to confirm 'yes, I heard you'. This systems of:

- Broadcast
 - Wait for the receiver to say 'I got it!', or if too much time passes
 - Re-broadcast

can add a lot of complexity to the overhead of the protocol (more power, less data bandwidth, etc). Do you

really care if your garage door opener fails to open your garage door? No, you just hit the button again. Do you care if your **cardiotocograph** gets recorded correctly to the nurses' station - um, ya.

Cost:

This correlates with power and complexity of use. If you've got a bunch of free time, you can probably spend a few days/weeks hacking at the really **cheap RF links**, creating your own protocol, and verifying **checksums**. However, if you simply need to plug it in and have it work - now!, expect to spend more money to have something work out of the box (Bluetooth and Cellular are the simplest solutions for unseasoned users).

Learning Curve:

This characteristic ties closely with cost - if you've got no money, expect to spend a bit more time wading through example code and datasheets. Once you get a basic RF link working, it's awesome! But it may take a few ferocious days of beating your head against the wall. On the other hand, Bluetooth may seem exorbitantly expensive, but you can open a serial pipe with a few mouse clicks. No matter which solution you choose, expect to spend many hours pouring over datasheets and/or example code. But once you really wrap your head around the wireless solution, it will seem trivial.

Size:

Regardless of what you may have seen in the movies, slipping someone an **RF pill** won't allow you to track them across a country. Help us with a good movie example! RF means a certain amount of size simply for the antenna and power source. There are very sneaky ways of miniaturizing the antenna but every time you fold the antenna smaller, you generally lose transmission/reception range (the smaller the antenna, the less distance of coverage you get). As the frequencies get higher (think 2.4GHz, 5.8GHz, etc), the antennas *do* get smaller. However, the penetration of walls and barriers gets worse. 5.8GHz likes to bounce off household walls, 2.4GHz might make it to the neighbors, 900MHz can get through walls, trees, kids, even livestock, 125kHz can work under water. Does **GPS** (1.5752GHz) work under water? No - because water is an awesome insulator (doesn't allow energy to pass through). Then how does my **dive computer** work under water? That's because a wireless dive computer works in the kHz range (much lower than MHz or GHz).

It's a trade off: the longer range and better reception (more penetration) you need, the lower the frequency (think kHz, not GHz), the lower the data rate, the bigger the antenna (2.4GHz = 3cm, 900MHz = 9cm, **32kHz = 243,212cm**). If you need a 32kHz antenna (giant 768ft antenna) to transmit 1Mbit/s of data, to power off a coin cell battery, you may run into problems.

Antenna selection is not trivial. You could have an amazingly powerful RF engine, but a crappy antenna and all of the features will suffer. A chip antenna might seem alluring, but it may not be the best fit with your project. If you need range, use a **directional biquad antenna**. If you need tiny size and only feet of range, use a trace or wire antenna. Chip antennas are a little better. Bigger **duck antennas** are incrementally better. It all depends on your application.



Nordic - how I love you so.

Pros:

- Low cost - ~\$2
- Low power - down to micro-amps
- Super flexible
- Great way to get a very small amount of data from one point to another
- Small footprint

Cons:

- You write your own protocol
- Shorter range (~50 to 100m)
- Support point-to-multiple point very well, but writing your own mesh network protocol is a bit daunting
- Doesn't do a great job of insuring data delivery like other technologies do

Other:

I really like Nordic hardware. The nRF24L01+ is the latest radio and is very flexible. If you need to get a couple bytes of data (like an analog to digital conversion of a sensor) from a remote unit to another remote unit (or computer), every second or so, these work pretty well. You can put some packets together and get units to twitter some data to each other.

The nRF24L01 has an SPI interface which makes it easy to control. You'll have to wrap your head around the datasheet (really, RTFM and you'll be fine), but this is nothing compared to the massive pile that is Zigbee.

Nordic is known for being low power. If you need to run a wireless node on a small battery for years, this is the

only way to do it. **Apple and Nike do it.** You can get wireless communication for a few dollars and for a few micro amps. You also get a bit more difficult radio to use out of the box, and range is really meant for near field - like 10-20 meters.



Cellular is everywhere! Well, sort of.

Pros:

-
- Range. Hard to define because cell networks are everywhere.
- Easy to use - it's nothing but simple serial AT commands. You could drop a node at a truck stop in Kansas and probably get data to the internet with an afternoon's worth of work.

Cons:

-
- Expensive: ~\$100 per radio
- Power hungry: 250mA average
- The connector is dense, unique, and hard to solder. So we built **breakout boards!**
- Pay per kilobyte
- Not small

Other:

Cellular is a great way to go if you need asset tracking across large distances. I used it so that we could get data from a weather station located in the middle of nowhere. The problem is that you have to rely on and pay for the cellular networks. You can get a SIM card for \$20-30 a month and transfer quite a lot of data, but it's no replacement for WiFi. These radios are really well designed and are very easy to setup and use - but you pay for it up front. Of course, if you go into an area without cell reception, the node will not be able to communicate. But you could take a cellular tracker to any country in the world and pipe data to a webpage! There's cellular all over the place!

Zigbee

Pros:

Cons:

- Bloated
- Expensive
- Hard to use

Other:

I'm mean. I have never actually used Zigbee because the dev kits are too expensive and overly complex. I am jaded because it was supposed to be the next greatest thing, and everyone jumped on board. It is cool to just say 'Zigbee'. Right. Last time I checked, the Zigbee 'stack' took up a so much memory that it requires heavy processors and co-processors that eat up mA worth of current. Zigbee is basically proprietary, the dev tools are expensive, and if you wish to buy an easy to use module, they are more expensive and worse than Bluetooth. I am not impressed.



Bluetooth - a classic.

Pros:

- Works pretty easily out of the box
- Certified interoperability
- Pretty good range (~50-100m)
- Sometimes supports audio links
- Pretty small footprint

Cons:

- Kind of expensive - \$10 is about the minimum per unit cost

- Power hungry - 10s of mA
- Multi-point is pretty much impossible

Other:

Bluetooth is pretty good. I really liked using it on the popular **BlueSMiRF**. It has been around for a few years now and the Serial Port Profile (SPP) works great for opening up a serial pipe between two points (what you pass in the RX pin, you get out at the TX pin on the other side of the wireless connection). Anything Bluetooth can pretty much talk to anything else Bluetooth. This makes it pretty flexible and easy to use, but you pay for it in cost, and in power consumption. Bluetooth modules with audio are one of the very few technologies that we've played with that support audio transmission.



XBee rocks. Don't let me confuse you! XBee sounds a lot like Zigbee. That's because this company called **Digi** (well, **Maxstream before they were bought**) took all the good stuff about Zigbee, and wrapped it up into an easy to use AT command set. I don't care how you get the data from point A to point B, XBee takes care of it all for me.

Pros:

- Simple to use interface
- Creates serial pipe between point A and point B out of the box
- Comes in many different footprints to support different frequencies, distances, and power ranges.
- Short range (100m) to long range (16 miles!)
- Same AT commands for all the different flavors (awesome)
- Great **configuration software** (windows only, sorry!)

Cons:

- Uses a slightly hard-to-find 2mm pitch connector. So we designed a bunch of breakout boards (**basic**, **USB**, **RS232**).

- Not low power (50mA)
- Series 2.5 supports multi-point and mesh-node-networking but does it with some really dense datasheets and nomenclature
- Not cheap. ~\$25(min) per radio.
- Not all that small. Pretty small, but not *really* small.

Other:

I like XBee. I wasn't a believer until I started using it for all my **wireless bootloading**. Now I love it. If power is not an issue, these work great for beginners to get data from one place to another with the least hassle, learning curve, and setup.



WiFi

Pros:

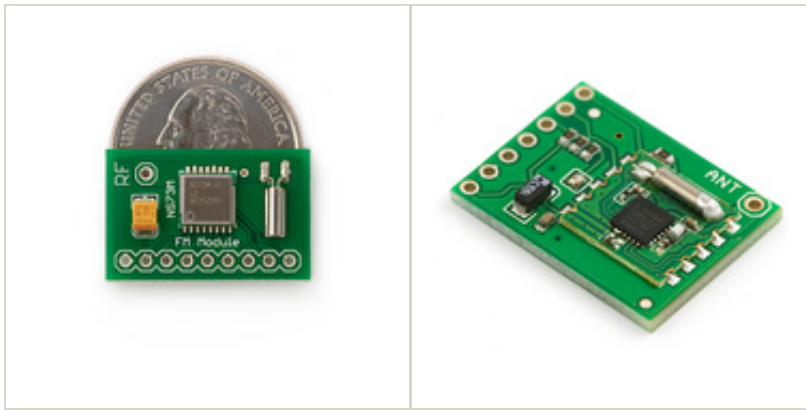
- Super common standard - pipe data to any router anywhere
- Good for moving large amounts of data (1Mb+)

Cons:

- Not low power - ~100mA
- Requires some serious processing power to handle 802.11 stack

Other:

I have not yet used WiFi. I usually don't need to move more than a few bytes over a wireless link so I haven't found a need for it, but many people have. There are some decent WiFi modules available that give us UART or SPI interface to WiFi, but it's still pretty complex in my opinion.



FM Radio **Transmitter** and **Receiver**

Pros:

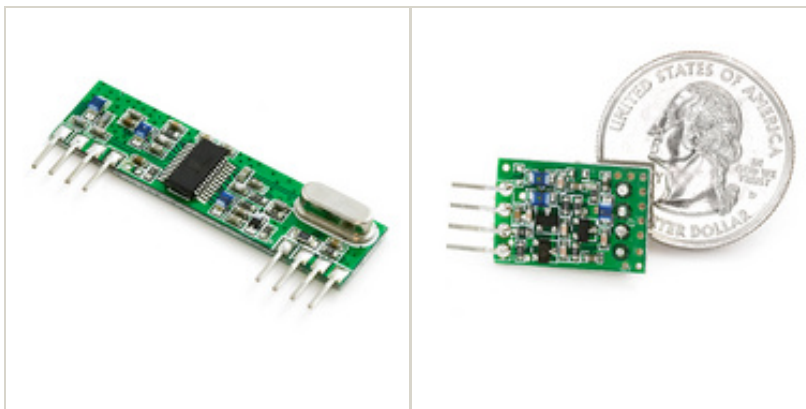
- A great way to get audio from one device to another
- Easy interface/control
- Small size

Cons:

- Only audio, no data
- Not directly FCC certified

Other:

The FM Transmitter module is easy enough to use, and it works surprisingly well, but it's limited to audio paths only. You can't transmit data (without being sneaky). The module can be FCC certified, but it is not certified off-the-shelf so you have to use it under the 'research only' umbrella. The FM receiver is very handy as well but with Pandora, iTunes, and the proliferation of MP3 - who listens to the radio anymore?



Pros:

- Cheap
- Basic interface
- Reasonable range (500ft)
- Large voltage range for transmitter (2-12V)

Cons:

- Big/long learning curve to get units functioning correctly
- Limited data rate (2400-4800bps)
- No data filtering or packetization
- Not directly FCC certified

Other:

The general RF transmitter and receiver pairs are commonly used for low-data rate consumer applications - like in your garage door, remote weather station, wireless window remote, the list goes on and on. These units are made cheap, and therefore have no brains. This leads to a headache from the users' perspective. It is possible to get bytes across this link, but it requires quite a bit of hacking, protocol creation, error checking, and patience.

So that's our take on the RF selection at SparkFun. We hope this gives you a bit of insight into what is, and what is not possible when using wireless devices. There are more options than what is listed here. If you know of a good solution, let us know!

▼ Comments 14 comments

Login or register to post comments.

-  daok | about 3 years ago 2

I do not understand why the Nordic say it costs 2\$ and when I click to see the price at SparkFun I see nothing under 20\$. It's an error?

-  tpag2000 | about 3 years ago 1

The price they're talking about is for the SMD IC itself. The parts in the wireless section are the breakout boards, protoboards etc. They also sell the IC all by itself, see:

http://www.sparkfun.com/commerce/product_info.php?products_id=690

-  Nate | about 3 years ago 1

tpag has got it. We're quoting volume prices. You can get the total BOM of a Nordic solution down into the \$2 if you build ~10,000 units. Many of the other solutions do not scale as well.

-  xyzy | about 3 years ago 2

Now I'm confused... In the above description of the General TX/RX devices, you say "It is possible to get bytes across this link, but it requires quite a bit of hacking, protocol creation, error checking, and patience."

In the product listing, however, you say "We have used these modules extensively and have been very impressed with their ease of use" and "This wireless data is the easiest to use, lowest cost RF link we have ever seen!"

Is it just compensating for the lack of error correction that makes these parts hard to use, or is there more to it?

-  hads | about 3 years ago 2

Though the Xbee aren't low power when active, they sleep at 1uA which is pretty low.

-  tj | about 3 years ago 1

Very useful tutorial. I was thinking of implementing some wireless in a project and was so confused with all the different possibilities. This tutorial is a really nice introduction. Thanks for posting.

-  waltr | about 3 years ago 1

With a firmware update the XBee modules can be fully ZigBee. These are great modules, not that cheap but not too expensive either. I did look into several ZigBee solutions and the XBee ZB was the best price and ease of use.

Oh yea, very nice overview on wireless solutions.

-  martin smith | about 3 years ago 1

Very helpful overview — I've started forwarding it to my colleagues. Thanks a bunch for your efforts and for relating your experiences so candidly.

-  3amsleep | about 3 years ago 1

NICE!, thanks a lot!

I was struggling to make up my mind on which protocol to use for my project

-  Mehon | about 2 years ago 1

I'm planning to build up a gps tracker. Thatfore I'm intersted in sendig ONLY the data related to

positioning (latitude and longitude). The question is, taking into account it's a low data, would it be enough a TX/RX to transmit it? What would be the best for that purpose?

Thanx

-  **Mehon** | about 2 years ago 1

I'm planning to build up a gps tracker. Thatfore I'm intersted in sendig ONLY the data related to positioning (latitude and longitude). The question is, taking into account it's a low data, would it be enough a TX/RX to transmit it? What would be the best for that purpose?

Thanx

-  **Allan** | about a year ago 1

Rf link is very easy or hard? the rf link and this page conflicts.

-  **Member #13000** | about a year ago 1

thanks for all of these helpful information on wireless connection components !
 i am a "newbee to electronic circuitry",
 did my googling & research on the picaxe (uk) forum(s), "instructable", "letsmakerobots",
 "makezine", "fritzing" sites and wonder,
 if you might even recommend additional links providing examples projects
 such as ... "how to code e.g. signals, wire rf components, breadboard & test them, ..."
 ... probably even for picaxe?
 eventhough i know that rf technolgies is not typically for "newbees" &
 this review already is much more than one could expect from a company
 selling these components (... thanks a lot for all this, again)
 best regards!

-  **Member #149271** | last year 1

Regarding the "General TX/RX" category, yes, these WERE hard to work with, but have now been tamed! This device: www.swiftEnterprises.biz should be listed in your list. I have spend a lot of time programming around all the negative aspects of these things and produced (IMHO) a really reliable and easy to use radio sharing solution. I'd love to see the Swift RD-11 listed in your table above and carried by SparkFun, but I hope you will at least note their existence here, via this comment. Thanks for listening!
 – Steven Swift, creator of the Swift RD-11 device.

SparkFun?

SparkFun is an online retail store that sells the bits and

Resources

- [Feeds](#)
- [Forum](#)

Feedback

What's on your mind?

SparkFun
 Electronics ® |
 Boulder, Colorado
 | [Customer Service](#)

pieces to make your electronics projects possible. Whether it's a robot that can cook your breakfast or a GPS cat tracking device, our products and resources are designed to make the world of electronics more accessible to the average person. In addition to products, SparkFun also offers classes and a number of online tutorials designed to help educate individuals in the wonderful world of embedded electronics.

- [SparkFun IRC Channel](#)
- [BatchPCB](#)
- [Open Circuits](#)
- [Ponoko - Your Personal Factory](#)
- [Jobs @ SparkFun](#)
- [Sell Your Widget on SparkFun](#)
- [Become a SparkFun Distributor](#)
- [Request SparkFun Sponsorship](#)
- [Take the SparkFun Quiz](#)
- [Tell us about a project](#)
- [SparkFun Kickstarter Projects](#)

For which department?

General

Please include your email address if you'd like us to respond to a specific question.