433Mhz Library [COMPLETED]

taylormade201	May '14

Is anyone working on porting a 433Mhz library such as VirtualWire?

% **What RF Transceiver will work with spark?**

% **Is there any french people here? Est ce qu'il y a des Français ici?**

peekay123 🛡	May '14
Elite	

taylormade201, I've thought about it but not yet gotten around to it. Is there a specific library you want ported? Perhaps you could include a link? 😄

taylormade201	May '14

Perhaps something like RadioHead?

Thanks

peekay123 🛡	May '14
Elite	

taylormade201, to simplify things a bit, can you tell me which 433MHz module you expect to use?

daedalusminos	May '14

I can definitely recommend the TI CC1101 based modules. A library for this can be found on the panStamp website.

wgbartley 🛡	May '14
Elite	

I have 433 Mhz transmitter and receiver modules (note: **not** transceiver) that I got from SparkFun a few years ago. I have the older versions. I know they went out of stock for quite a long time while they switched manufacturers or something, so the modules I have may be different from what they're selling now. I imagine they operate the same, though.

taylormade201   May '14

Most of the 433/315 Mhz modules are pretty generic such as the ones wgbartley mentioned.

Another good library to look at which may be even more applicable is the Manchester library. I believe it is a bit more flexible when the transmitter and receiver are running at different clocks, which is likely to be the case when using the core with something else like an Arduino.

Let me know if there are any ways I can help!

peekay123 🛡   May '14
Elite

I think I can port the Manchester library using the IntervalTimer library I created. I'm on it.

peekay123 🛡   May '14
Elite

@taylormade201, I have been working on porting the library and got stuck on some compile errors with my IntervalTimer library. I think we have that fixed now so I will finish the port and post it on my github for you when done. 😄

taylormade201   May '14

Thanks @peekay123 , this will be a great library to have!

peekay123 🛡   May '14
Elite

@taylormade201, I have the Manchester library completed and posted on my github. One thing I am not certain about is the timer interval as the documentation was not very clear. Hopefully, what I understood and how I coded the timer interval is correct. Let me know how it goes! 😄

taylormade201   May '14

Great! I will try and test it tomorrow or Friday.

taylormade201   May '14

Tested the library today, works great.
Thanks @peekay123

peekay123 🛡️
Elite

May '14

@taylormade201 , my pleasure. Since it's working, I will post it as a library when that feature comes out soon 😄

---

d82k

Jun '14

Hello,

I am wondering, can I use this library to sniff the code from a transmitter? I have a couple of remote 433, which I would like to use to interact with the spark.
Can I use it to display the raw message and tweak it accordingly with the lenght and the time interval?
Thank you
dk

---

peekay123 🛡️
Elite

Jun '14

@d82k , I don't see why not. You can use the receive examples as your foundation and go from there. 😄

---

d82k

Jun '14

Thank you @peekay123 , I have tried modifying ManchesterRX_Basic.ino from the examples, but I don't understand: how do I set the length in bit of the message and the frequency?
Is the message^"m" than printed as an int?
Thank you,
dk

---

peekay123 🛡️
Elite

Jun '14

@d82k , I did not write the library, I only ported it from an Arduino library. You will need to read the descriptions in both Manchester.cpp and Manchester.h to get an idea of the timing for the receiver. You can set a baud rate (MAN_1200) which is set when you call `man.setupReceive(RX_PIN, MAN_1200);` in the test code. The bit timing for the receiver is entirely dependent on the baud rate which sets the frequency of the receiver interrupt service routine.

Hope that helps! 😄

---

peanutsguy

Jun '14

Hi! Any plans on uploading your port now that the library feature is available??? Thanks for your work!

peekay123 🛡️
Elite

Jun '14

Yup, just need to prep it. Thanks for reminding me 🙂

---

peekay123 🛡️
Elite

Jun '14

@**peanutsguy** , I have published the Manchester library to the IDE. It needs the SparkIntervalTimer library to be added as well. 😄

---

peanutsguy

Jun '14

Great work! Thanks! I'll check it right now 😃

---

krvarma
Level 4 | Dev Guru

Aug '14

@**peekay123** , I am having some issues using this library. I have one Core with 433Mhz Transmitter and TX_Basic demo and another with Receiver and RX_Basic demo running. The TX demo Core' LED is blinking, but the RX demo is not working. The below if condition never becomes TRUE.

```
if (man.receiveComplete()){
...
}
```

Can you give some tips to debug and correc this?

---

wyealex

May '15

I'm interested in using the Manchester library to decode the output from an Oregon Scientific temperature sensor. The decoding guidelines here mention that the bitrate is 1024Hz for the sensor I have. Can someone clarify if this is the rate I need to set in my receiver, or whether the MAN_1200 option will suffice since it is above the 1024Hz rate?

---

peekay123 🛡️
Elite

May '15

@**wyealex** , the library was a port I did originating HERE. A quick search on google led me to some Arduino code for your sensor. It may be best to port a library specifically written for the sensor.

Arduino Wireless Weather Station & Web Server

Objective Replace my PC (which has been on 24-7 for 6 years) and weather station software with an Arduino – Lower power consumption,

less maintenance and...

Arduino Intercepting Oregon Scientific Weather Station Data

The Experiment: Arduino Intercepting Oregon Scientific 433Mhz Wireless Weather Station Data. Build a 433Mhz wireless weather station receiver around an Arduino UNO micro-controller, to recei…

Find the best Arduino library for your application and I can help port it if needed. 😃

---

wyealex   May '15

Thanks for pointing me in the right direction peekay. My sensor is certainly on the list (THN132N) so this should be applicable. I am actually prototyping this project on an Arduino Mega before my Spark Photon arrives (hopefully later this month), so I will try to test it out directly before attempting to port it. Thanks for the offer of help with the port - I might take you up on it before too long though this is my first experience with code this dense so it might be a while before I understand it fully!

---

smogs   Jul '15

I would be really keen to use this library but it wont let me compile any of the examples for my photon 😕

I was hoping you may be able to help @peekay123 given you did the first port… this is way beyond my still new skills
I get the following errors:

```
manchesterrx_array.cpp: In function 'void loop()':
manchesterrx_array.cpp:47:20: warning: operation on 'moo' may be undefined [-Wsequence-point]
if (man.receiveComplete()) {
                    ^
manchesterrx_array.cpp:43:13: warning: unused variable 'receivedSize' [-Wunused-variable]
man.beginReceiveArray(BUFFER_SIZE, buffer);
            ^
Manchester/Manchester.cpp: In member function 'void Manchester::transmit(uint16_t)':
Manchester/Manchester.cpp:131:31: warning: narrowing conversion of '(((int)data) >> 8)' from 'int' to 'uint8_t {aka unsigned char}' inside { } [-Wnarrowing]
uint8_t byteData[2] = {data >> 8, data & 0xFF};
                       ^
Manchester/Manchester.cpp:131:42: warning: narrowing conversion of '(((int)data) & 255)' from 'int' to 'uint8_t {aka unsigned char}' inside { } [-Wnarrowing]
uint8_t byteData[2] = {data >> 8, data & 0xFF};
                       ^
SparkIntervalTimer/SparkIntervalTimer.cpp: In member function 'void IntervalTimer::start_SIT(uint16_t, bool)':
```

SparkIntervalTimer/SparkIntervalTimer.cpp:178:10: warning: statement has no effect [-Wunused-value]
 scale == uSec; // Default to microseconds
          ^

SparkIntervalTimer/SparkIntervalTimer.cpp: In member function 'void IntervalTimer::interrupt_SIT(action)':
SparkIntervalTimer/SparkIntervalTimer.cpp:261:15: warning: variable 'TIMx' set but not used [-Wunused-but-set-variable]
 TIM_TypeDef* TIMx;
              ^

SparkIntervalTimer/SparkIntervalTimer.cpp: In member function 'void IntervalTimer::resetPeriod_SIT(uint16_t, bool)':
SparkIntervalTimer/SparkIntervalTimer.cpp:305:26: warning: unused variable 'timerInitStructure' [-Wunused-variable]
 TIM_TimeBaseInitTypeDef timerInitStructure;
                         ^

SparkIntervalTimer/SparkIntervalTimer.cpp: In member function 'void IntervalTimer::start_SIT(uint16_t, bool)':
SparkIntervalTimer/SparkIntervalTimer.cpp:191:43: warning: 'TIMx' may be used uninitialized in this function [-Wmaybe-uninitialized]
 TIM_ITConfig(TIMx, TIM_IT_Update, ENABLE);
                                           ^

SparkIntervalTimer/SparkIntervalTimer.cpp: In member function 'void IntervalTimer::stop_SIT()':
SparkIntervalTimer/SparkIntervalTimer.cpp:247:18: warning: 'TIMx' may be used uninitialized in this function [-Wmaybe-uninitialized]
 TIM_DeInit(TIMx);
                  ^

SparkIntervalTimer/SparkIntervalTimer.cpp: In member function 'void IntervalTimer::resetPeriod_SIT(uint16_t, bool)':
SparkIntervalTimer/SparkIntervalTimer.cpp:336:23: warning: 'TIMx' may be used uninitialized in this function [-Wmaybe-uninitialized]
 TIMx->PSC = prescaler;
                       ^

.../.../.../build/target/user/platform-6/libuser.a(Manchester.o): In function Manchester':
Manchester/Manchester.cpp:55: undefined reference toSystemCoreClock'
Manchester/Manchester.cpp:55: undefined reference to Wiring_TIM2_Interrupt_Handler'
Manchester/Manchester.cpp:55: undefined reference toWiring_TIM3_Interrupt_Handler'
Manchester/Manchester.cpp:55: undefined reference to Wiring_TIM4_Interrupt_Handler'
../../../build/target/user/platform-6/libuser.a(SparkIntervalTimer.o): In functionWiring_TIM2_Interrupt_Handler_override':
SparkIntervalTimer/SparkIntervalTimer.cpp:37: undefined reference to TIM_GetITStatus'
SparkIntervalTimer/SparkIntervalTimer.cpp:39: undefined reference toTIM_ClearITPendingBit'
.../.../.../build/target/user/platform-6/libuser.a(SparkIntervalTimer.o): In function Wiring_TIM3_Interrupt_Handler_override':
SparkIntervalTimer/SparkIntervalTimer.cpp:47: undefined reference toTIM_GetITStatus'
SparkIntervalTimer/SparkIntervalTimer.cpp:49: undefined reference to

```
TIM_ClearITPendingBit' ../../../build/target/user/platform-
6/libuser.a(SparkIntervalTimer.o): In
functionWiring_TIM4_Interrupt_Handler_override':
SparkIntervalTimer/SparkIntervalTimer.cpp:57: undefined reference to TIM_GetITStatus'
SparkIntervalTimer/SparkIntervalTimer.cpp:59: undefined reference
toTIM_ClearITPendingBit'
.../.../.../build/target/user/platform-6/libuser.a(SparkIntervalTimer.o): In function
IntervalTimer::start_SIT(unsigned short, bool)':
SparkIntervalTimer/SparkIntervalTimer.cpp:163: undefined reference
toRCC_APB1PeriphClockCmd'
SparkIntervalTimer/SparkIntervalTimer.cpp:153: undefined reference to
RCC_APB1PeriphClockCmd' SparkIntervalTimer/SparkIntervalTimer.cpp:158:
undefined reference toRCC_APB1PeriphClockCmd'
SparkIntervalTimer/SparkIntervalTimer.cpp:189: undefined reference to TIM_TimeBaseInit'
SparkIntervalTimer/SparkIntervalTimer.cpp:190: undefined reference toTIM_Cmd'
SparkIntervalTimer/SparkIntervalTimer.cpp:191: undefined reference to TIM_ITConfig'
SparkIntervalTimer/SparkIntervalTimer.cpp:200: undefined reference toNVIC_Init'
.../.../.../build/target/user/platform-6/libuser.a(SparkIntervalTimer.o): In function
IntervalTimer::stop_SIT()': SparkIntervalTimer/SparkIntervalTimer.cpp:240:
undefined reference toTIM_Cmd'
SparkIntervalTimer/SparkIntervalTimer.cpp:244: undefined reference to NVIC_Init'
SparkIntervalTimer/SparkIntervalTimer.cpp:247: undefined reference
toTIM_DeInit'
collect2: error: ld returned 1 exit status
make: *** [1aaa671af00ccc1525b2ee0c970353f68ea01b9103a7086cfca92cba9e0e.elf] Error 1
```

peekay123 🛡
Elite
Jul '15

@smogs , support for the SparkIntervalTimer with Photon is not yet available unless you compile locally. It is coming VERY soon I believe 😄

chuank
Aug '15

Hi @peekay123 ,

I'm compiling locally and attempting to use your Manchester library with my Photon.

My setup is to use Arduino Pro Minis transmitting off 433MHz transmitters using the Radiohead library. As a test, they are currently broadcasting a single uint8_t at each press of a pushbutton. My original Arduino-based receiver, using the same Radiohead library, receives the char just fine, but not the Photon. It seems like your Manchester library port expects uint16_ts, could you point me towards the right direction to tweak the code? Or would you suggest that I use a different library on my Arduino side to send uint16_ts?

p.s. I've pulled the latest SparkIntervalTimer develop branch from your repo, and compile without error.

chuank   Aug '15

Guess I typed too soon…

Using Mchr3k's Arduino Manchester library I was able to get the Pro Minis talking nicely to the Photons. It does appear that the Radiohead library is not compatible with this library, for anyone who might encounter this issue here.

Faajer   Nov '15

Been trying to do communication from my photon to an arduino nano using "RF wireless receiver module & transmitter module board for arduino super regeneration 315/433MHZ DC5V (ASK /OOK)" Failing so far though. As you succeeded, care to share some code snippets?
I'm receiving at the photon and sending from the arduino nano.

---

chuank   Dec '15

Hi @Faajer ,

The first question will be: do you have a Rx & Tx 433MHz module pair?

Did you try using @peekay123 's Manchester library?. The sample code in there worked quite well for me. I compile all of my code locally for other reasons but otherwise they compiled fine for me using the Photon as Rx.

A couple of things to note:

1. the baud rate – usually the slower you set it, the more reliable it is. I've had decent reception with 1200/2400bps
2. voltage you are supplying the Rx and Tx with. Most of the 433MHz Tx's out there can be drive anywhere from 3.3V to 7.5V. Higher voltage provides stronger transmit power
3. are you using a proper antenna design, of right length? this link could help

Also, are you using the Tx on your Photon or Arduino Nano?

If the Tx is on your Photon, and if you're powering your Photon via USB, try hooking up the power to the Tx using the VIN pin (5V/USB) instead of 3v3.

If you're using the Rx on your Photon, I have noticed that powering the Rx at either VIN (5V/USB) or from the 3V3 pin does make a difference to reception quality. Haven't been able to figure out why, though. Try either option, and see if one provides better results. There's so many variants of these cheap ASK/OOK super regenerative Rx's out there so give this a try.

Faajer   Dec '15

Hi @chuank

I do have a Rx & Tx 433MHz module pair.
I'm using the Manchester library from the Particle IDE (builtin Manchester library) for the Photon. And the Manchester direct download for the Arduino

I've tried fooling around with the baud rate setting it at any possible value.

Using the Photon as the receiver. I make use of the 3V3 port but also tried to drain power from the arduino 5v port which is powered from usb.

It just doesn't hit the man.receiveComplete() so there's little to test until I can actually catch the signal.
The 2 devices are next to each other so range shouldn't be an issue.

It's also not clear to me what other port on the photon besides the 3V3 port can be used for power?
I'm sorry if this comes over somewhat noobish but I'm new to these kinds of projects.

Thanks for responding btw.

---

chuank   Dec '15

Hi @Faajer ,

If you're powering via USB on your Photon, use the VIN pin to provide +5V to your Rx. Also try moving to the other DATA pin on the Rx, and check that you've got the right pins declared in your Particle code. Avoid using breadboards if you can, too. While there shouldn't be any major RF interference issues between the 2.4GHz WiFi antenna and the 433's, try orienting them perpendicularly to see if you can pick up any new signals.

For testing, send a regular ping at 1sec intervals on my Tx, and get your Photon to blip the D7 LED whenever it gets a new byte of data. This gives immediate feedback on whether it's simply a poor signal (when you'll get inconsistent Rx blips), or point out something else in your code that's not working.

Also, make sure you're using https://mchr3k.github.io/arduino-libs-manchester/ as the library on the Arduino side.

I did remember having some difficulties getting 433MHz RF to work the first time with the Photon, but it was resolved after checking the points I listed in my previous post.

Hope this helps!

Faajer   Dec '15

@chuank Thanks for the tips. I got it working now (extended the antennas with 17cm wire) and at a friends I can manage to send messages up to 14 meters including 2 walls.
At my own place I barely send up to 7 meters. Guess it also depends on noise.

11-12 meters would be perfect for my setup but I doubt I'll ever get that far in this environment.

---

chuank   Dec '15

That's great to hear @Faajer !

I've had surprisingly good response with good 'ol 17.8cm hookup wire. Also, look online for loaded coil antenna designs where you basically coil up the base of the antenna like a helix using a lamp

cord to help form the helix. This gave me a bit more range, but my deployment area (along a narrow hallway) reflects/absorbs the crap out of my Tx's signal. I'd be lucky to get the kind of range you are getting with consistency. What kind of walls are they?

You can also try using a compact 433mhz chip antenna, but those will be a lot tougher to tune (since you can't shorten the chip itself!).

Since I don't have a spectrum analyzer, I'm finding my RF tests to be a lot like magic voodoo. 🙂 I'm guessing that with the cheap Rx/Tx combos which might not be exactly well-tuned, you will have differing performance/range even with the same wire!

---

brettski   Jan '16

@wgbartley I have read that the receiver you have from SparkFun is noisy, have you experienced this yourself or have you had solid communications with it?

Brett

---

wgbartley 🛡   Jan '16
Elite

I've had good luck with it in the past. However, something happened to mine, and I get nothing but random output from it now. It's probably something I've done to break it over the years (and I've had it for 5+ years at least)!

---

daylight14   Mar '16

Has anyone gotten the Manchester library to work on a Photon? I bought a cheep RX/TX pair for 433MHz and can't get them to work. I've been through most of the suggestions on this thread so far so I'm starting to think I might have a dud RX or TX. I just want to make sure it's possible on a Photon and not just the Core before buying a new pair.

---

daylight14   Mar '16

Never mind, I kinda got it to work. The Rx can pick up the Tx when they are connected to the same photon. But when I put the Tx on a different photon it stops working. So I have no idea what I'm doing wrong, but they aren't duds.

Visit particle.io/mesh to check out our third generation of IoT hardware: the **Argon** (Wi-Fi + Mesh + BLE), the **Boron** (LTE + Mesh + BLE), and the **Xenon** (Mesh + BLE).   ✕

---

peekay123 🛡   Jun '14
Elite

@peanutsguy , I have published the Manchester library to the IDE. It needs the SparkIntervalTimer library to be added as well. 😄

peanutsguy   Jun '14

Great work! Thanks! I'll check it right now 😃

---

krvarma   Aug '14
Level 4 | Dev Guru

@peekay123 , I am having some issues using this library. I have one Core with 433Mhz Transmitter and TX_Basic demo and another with Receiver and RX_Basic demo running. The TX demo Core' LED is blinking, but the RX demo is not working. The below if condition never becomes TRUE.

```
if (man.receiveComplete()){
...
}
```

Can you give some tips to debug and correc this?

---

wyealex   May '15

I'm interested in using the Manchester library to decode the output from an Oregon Scientific temperature sensor. The decoding guidelines here mention that the bitrate is 1024Hz for the sensor I have. Can someone clarify if this is the rate I need to set in my receiver, or whether the MAN_1200 option will suffice since it is above the 1024Hz rate?

---

peekay123 🛡   May '15
Elite

@wyealex , the library was a port I did originating HERE. A quick search on google led me to some Arduino code for your sensor. It may be best to port a library specifically written for the sensor.

> ### Arduino Wireless Weather Station & Web Server
>
> Objective Replace my PC (which has been on 24-7 for 6 years) and weather station software with an Arduino – Lower power consumption, less maintenance and...

> ### Arduino Intercepting Oregon Scientific Weather Station Data
>
> The Experiment: Arduino Intercepting Oregon Scientific 433Mhz Wireless Weather Station Data. Build a 433Mhz wireless weather station receiver around an Arduino UNO micro-controller, to recei…

Find the best Arduino library for your application and I can help port it if needed. 😃

wyealex — May '15

Thanks for pointing me in the right direction peekay. My sensor is certainly on the list (THN132N) so this should be applicable. I am actually prototyping this project on an Arduino Mega before my Spark Photon arrives (hopefully later this month), so I will try to test it out directly before attempting to port it. Thanks for the offer of help with the port - I might take you up on it before too long though this is my first experience with code this dense so it might be a while before I understand it fully!

---

smogs — Jul '15

I would be really keen to use this library but it wont let me compile any of the examples for my photon 🙁

I was hoping you may be able to help @peekay123 given you did the first port… this is way beyond my still new skills
I get the following errors:

```
manchesterrx_array.cpp: In function 'void loop()':
manchesterrx_array.cpp:47:20: warning: operation on 'moo' may be undefined [-Wsequence-point]
if (man.receiveComplete()) {
                   ^
manchesterrx_array.cpp:43:13: warning: unused variable 'receivedSize' [-Wunused-variable]
man.beginReceiveArray(BUFFER_SIZE, buffer);
                   ^
Manchester/Manchester.cpp: In member function 'void Manchester::transmit(uint16_t)':
Manchester/Manchester.cpp:131:31: warning: narrowing conversion of '(((int)data) >> 8)' from 'int' to 'uint8_t {aka unsigned char}' inside { } [-Wnarrowing]
uint8_t byteData[2] = {data >> 8, data & 0xFF};
                   ^
Manchester/Manchester.cpp:131:42: warning: narrowing conversion of '(((int)data) & 255)' from 'int' to 'uint8_t {aka unsigned char}' inside { } [-Wnarrowing]
uint8_t byteData[2] = {data >> 8, data & 0xFF};
                   ^
SparkIntervalTimer/SparkIntervalTimer.cpp: In member function 'void IntervalTimer::start_SIT(uint16_t, bool)':
SparkIntervalTimer/SparkIntervalTimer.cpp:178:10: warning: statement has no effect [-Wunused-value]
scale == uSec; // Default to microseconds
                   ^
SparkIntervalTimer/SparkIntervalTimer.cpp: In member function 'void IntervalTimer::interrupt_SIT(action)':
SparkIntervalTimer/SparkIntervalTimer.cpp:261:15: warning: variable 'TIMx' set but not used [-Wunused-but-set-variable]
TIM_TypeDef* TIMx;
                   ^
SparkIntervalTimer/SparkIntervalTimer.cpp: In member function 'void IntervalTimer::resetPeriod_SIT(uint16_t, bool)':
SparkIntervalTimer/SparkIntervalTimer.cpp:305:26: warning: unused variable 'timerInitStructure' [-Wunused-variable]
```

```
TIM_TimeBaseInitTypeDef timerInitStructure;
^
SparkIntervalTimer/SparkIntervalTimer.cpp: In member function 'void
IntervalTimer::start_SIT(uint16_t, bool)':
SparkIntervalTimer/SparkIntervalTimer.cpp:191:43: warning: 'TIMx' may be used uninitialized in
this function [-Wmaybe-uninitialized]
TIM_ITConfig(TIMx, TIM_IT_Update, ENABLE);
^
SparkIntervalTimer/SparkIntervalTimer.cpp: In member function 'void IntervalTimer::stop_SIT()':
SparkIntervalTimer/SparkIntervalTimer.cpp:247:18: warning: 'TIMx' may be used uninitialized in
this function [-Wmaybe-uninitialized]
TIM_DeInit(TIMx);
^
SparkIntervalTimer/SparkIntervalTimer.cpp: In member function 'void
IntervalTimer::resetPeriod_SIT(uint16_t, bool)':
SparkIntervalTimer/SparkIntervalTimer.cpp:336:23: warning: 'TIMx' may be used uninitialized in
this function [-Wmaybe-uninitialized]
TIMx->PSC = prescaler;
^
.../.../.../build/target/user/platform-6/libuser.a(Manchester.o): In function Manchester':
Manchester/Manchester.cpp:55: undefined reference toSystemCoreClock'
Manchester/Manchester.cpp:55: undefined reference to Wiring_TIM2_Interrupt_Handler'
Manchester/Manchester.cpp:55: undefined reference
toWiring_TIM3_Interrupt_Handler'
Manchester/Manchester.cpp:55: undefined reference to Wiring_TIM4_Interrupt_Handler'
../../../build/target/user/platform-6/libuser.a(SparkIntervalTimer.o): In
functionWiring_TIM2_Interrupt_Handler_override':
SparkIntervalTimer/SparkIntervalTimer.cpp:37: undefined reference to TIM_GetITStatus'
SparkIntervalTimer/SparkIntervalTimer.cpp:39: undefined reference
toTIM_ClearITPendingBit'
.../.../.../build/target/user/platform-6/libuser.a(SparkIntervalTimer.o): In function
Wiring_TIM3_Interrupt_Handler_override':
SparkIntervalTimer/SparkIntervalTimer.cpp:47: undefined reference
toTIM_GetITStatus'
SparkIntervalTimer/SparkIntervalTimer.cpp:49: undefined reference to
TIM_ClearITPendingBit' ../../../build/target/user/platform-
6/libuser.a(SparkIntervalTimer.o): In
functionWiring_TIM4_Interrupt_Handler_override':
SparkIntervalTimer/SparkIntervalTimer.cpp:57: undefined reference to TIM_GetITStatus'
SparkIntervalTimer/SparkIntervalTimer.cpp:59: undefined reference
toTIM_ClearITPendingBit'
.../.../.../build/target/user/platform-6/libuser.a(SparkIntervalTimer.o): In function
IntervalTimer::start_SIT(unsigned short, bool)':
SparkIntervalTimer/SparkIntervalTimer.cpp:163: undefined reference
toRCC_APB1PeriphClockCmd'
SparkIntervalTimer/SparkIntervalTimer.cpp:153: undefined reference to
RCC_APB1PeriphClockCmd' SparkIntervalTimer/SparkIntervalTimer.cpp:158:
undefined reference toRCC_APB1PeriphClockCmd'
```

SparkIntervalTimer/SparkIntervalTimer.cpp:189: undefined reference to `TIM_TimeBaseInit'
SparkIntervalTimer/SparkIntervalTimer.cpp:190: undefined reference to `TIM_Cmd'
SparkIntervalTimer/SparkIntervalTimer.cpp:191: undefined reference to `TIM_ITConfig'
SparkIntervalTimer/SparkIntervalTimer.cpp:200: undefined reference to `NVIC_Init'
.../.../.../build/target/user/platform-6/libuser.a(SparkIntervalTimer.o): In function `IntervalTimer::stop_SIT()': SparkIntervalTimer/SparkIntervalTimer.cpp:240: undefined reference to `TIM_Cmd'
SparkIntervalTimer/SparkIntervalTimer.cpp:244: undefined reference to `NVIC_Init'
SparkIntervalTimer/SparkIntervalTimer.cpp:247: undefined reference to `TIM_DeInit'
collect2: error: ld returned 1 exit status
make: *** [1aaa671af00ccc1525b2ee0c970353f68ea01b9103a7086cfca92cba9e0e.elf] Error 1

---

peekay123 🛡
Elite

Jul '15

@smogs , support for the SparkIntervalTimer with Photon is not yet available unless you compile locally. It is coming VERY soon I believe 😄

---

chuank

Aug '15

Hi @peekay123 ,

I'm compiling locally and attempting to use your Manchester library with my Photon.

My setup is to use Arduino Pro Minis transmitting off 433MHz transmitters using the Radiohead library. As a test, they are currently broadcasting a single uint8_t at each press of a pushbutton. My original Arduino-based receiver, using the same Radiohead library, receives the char just fine, but not the Photon. It seems like your Manchester library port expects uint16_ts, could you point me towards the right direction to tweak the code? Or would you suggest that I use a different library on my Arduino side to send uint16_ts?

p.s. I've pulled the latest SparkIntervalTimer develop branch from your repo, and compile without error.

---

chuank

Aug '15

Guess I typed too soon…

Using Mchr3k's Arduino Manchester library I was able to get the Pro Minis talking nicely to the Photons. It does appear that the Radiohead library is not compatible with this library, for anyone who might encounter this issue here.

Faajer

Nov '15

Been trying to do communication from my photon to an arduino nano using "RF wireless receiver module & transmitter module board for arduino super regeneration 315/433MHZ DC5V (ASK /OOK)"

Failing so far though. As you succeeded, care to share some code snippets?
I'm receiving at the photon and sending from the arduino nano.

---

chuank   Dec '15

Hi @Faajer ,

The first question will be: do you have a Rx & Tx 433MHz module pair?

Did you try using @peekay123 's Manchester library?. The sample code in there worked quite well for me. I compile all of my code locally for other reasons but otherwise they compiled fine for me using the Photon as Rx.

A couple of things to note:

1. the baud rate – usually the slower you set it, the more reliable it is. I've had decent reception with 1200/2400bps
2. voltage you are supplying the Rx and Tx with. Most of the 433MHz Tx's out there can be drive anywhere from 3.3V to 7.5V. Higher voltage provides stronger transmit power
3. are you using a proper antenna design, of right length? this link could help

Also, are you using the Tx on your Photon or Arduino Nano?

If the Tx is on your Photon, and if you're powering your Photon via USB, try hooking up the power to the Tx using the VIN pin (5V/USB) instead of 3v3.

If you're using the Rx on your Photon, I have noticed that powering the Rx at either VIN (5V/USB) or from the 3V3 pin does make a difference to reception quality. Haven't been able to figure out why, though. Try either option, and see if one provides better results. There's so many variants of these cheap ASK/OOK super regenerative Rx's out there so give this a try.

Faajer   Dec '15

Hi @chuank

I do have a Rx & Tx 433MHz module pair.
I'm using the Manchester library from the Particle IDE (builtin Manchester library) for the Photon.
And the Manchester direct download for the Arduino

I've tried fooling around with the baud rate setting it at any possible value.

Using the Photon as the receiver. I make use of the 3V3 port but also tried to drain power from the arduino 5v port which is powered from usb.

It just doesn't hit the man.receiveComplete() so there's little to test until I can actually catch the signal.
The 2 devices are next to each other so range shouldn't be an issue.

It's also not clear to me what other port on the photon besides the 3V3 port can be used for power?
I'm sorry if this comes over somewhat noobish but I'm new to these kinds of projects.

Thanks for responding btw.

chuank　　　　　　　　　　　　　　　　　　　　　　　　**Dec '15**

Hi @Faajer ,

If you're powering via USB on your Photon, use the VIN pin to provide +5V to your Rx. Also try moving to the other DATA pin on the Rx, and check that you've got the right pins declared in your Particle code. Avoid using breadboards if you can, too. While there shouldn't be any major RF interference issues between the 2.4GHz WiFi antenna and the 433's, try orienting them perpendicularly to see if you can pick up any new signals.

For testing, send a regular ping at 1sec intervals on my Tx, and get your Photon to blip the D7 LED whenever it gets a new byte of data. This gives immediate feedback on whether it's simply a poor signal (when you'll get inconsistent Rx blips), or point out something else in your code that's not working.

Also, make sure you're using https://mchr3k.github.io/arduino-libs-manchester/ as the library on the Arduino side.

I did remember having some difficulties getting 433MHz RF to work the first time with the Photon, but it was resolved after checking the points I listed in my previous post.

Hope this helps!

Faajer　　　　　　　　　　　　　　　　　　　　　　　　**Dec '15**

@chuank Thanks for the tips. I got it working now (extended the antennas with 17cm wire) and at a friends I can manage to send messages up to 14 meters including 2 walls.
At my own place I barely send up to 7 meters. Guess it also depends on noise.

11-12 meters would be perfect for my setup but I doubt I'll ever get that far in this environment.

---

chuank　　　　　　　　　　　　　　　　　　　　　　　　**Dec '15**

That's great to hear @Faajer !

I've had surprisingly good response with good 'ol 17.8cm hookup wire. Also, look online for loaded coil antenna designs where you basically coil up the base of the antenna like a helix using a lamp cord to help form the helix. This gave me a bit more range, but my deployment area (along a narrow hallway) reflects/absorbs the crap out of my Tx's signal. I'd be lucky to get the kind of range you are getting with consistency. What kind of walls are they?

You can also try using a compact 433mhz chip antenna, but those will be a lot tougher to tune (since you can't shorten the chip itself!).

Since I don't have a spectrum analyzer, I'm finding my RF tests to be a lot like magic voodoo. 🙂 I'm guessing that with the cheap Rx/Tx combos which might not be exactly well-tuned, you will have differing performance/range even with the same wire!

---

brettski　　　　　　　　　　　　　　　　　　　　　　　**Jan '16**

@wgbartley I have read that the receiver you have from SparkFun is noisy, have you experienced this yourself or have you had solid communications with it?

Brett

---

wgbartley 🛡   **Jan '16**
Elite

I've had good luck with it in the past. However, something happened to mine, and I get nothing but random output from it now. It's probably something I've done to break it over the years (and I've had it for 5+ years at least)!

---

daylight14   **Mar '16**

Has anyone gotten the Manchester library to work on a Photon? I bought a cheep RX/TX pair for 433MHz and can't get them to work. I've been through most of the suggestions on this thread so far so I'm starting to think I might have a dud RX or TX. I just want to make sure it's possible on a Photon and not just the Core before buying a new pair.

---

daylight14   **Mar '16**

Never mind, I kinda got it to work. The Rx can pick up the Tx when they are connected to the same photon. But when I put the Tx on a different photon it stops working. So I have no idea what I'm doing wrong, but they aren't duds.