Arduino Forum

Using Arduino => Networking, Protocols, and Devices => Topic started by: silly_cone on Dec 29, 2016, 06:14 pm

Title: **ESP8266 version 12E with NRF24L01 - SOLVED!**
Post by: **silly_cone** on **Dec 29, 2016, 06:14 pm**

This is something that's been giving me difficulty for months now. I almost abandoned the project, but last night decided to give it another go, and thanks to the dutiful efforts and brilliance of others, I have found the solution!

The problem was that I wanted to connect an ESP 12-E to an NRF24L01 module to integrate it into an RF24 network. This would allow the 12-E to act as a bridge between my house RF24 network and the internet. But it seemed that no matter how I connected the NRF to the 12-E, they just wouldn't communicate.

But then I came across these two pages (below), where people had figured it out! It requires the use of the hardware SPI pins on the 12-E, which also require a pull-down resistor from pin D8 on the 12-E to activate HSPI mode.

Pin connections are as follows:

NRF ESP-12E
CSN ---------- D8
MOSI ---------- D7
MISO ---------- D6
SCK ---------- D5
CE ---------- D4

Also remember, D8 must be tied to ground via a 3-5 KOhm. The page I referenced recommends a 4.75 kOhm resisitor, but I used a 3.3 kOhm successfully. Do not connect directly to ground, use a resistor!

Then in your code, initialize the radio normally, using "RF24 wirelessSPI(2, 15);".

While I was testing it last night, I had difficulty getting it to transmit properly at first. I had it set up to send an incrementing integer every 500 mS, but it wasn't being properly received by the other node. I increased the interval to once every second, then it began to work perfectly and ran for about 10 hours before I unhooked it. I don't know if it was coincidental that it started working once I changed the interval or not.

Also, I have not tested it full RF24 network functionality yet.

websites referenced:
https://github.com/TMRh20/RF24/issues/125 (https://github.com/TMRh20/RF24/issues/125)
http://d.av.id.au/blog/esp8266-hardware-spi-hspi-general-info-and-pinout/ (http://d.av.id.au/blog/esp8266-hardware-spi-hspi-general-info-and-pinout/)

Title: **Re: ESP8266 version 12E with NRF24L01 - SOLVED!**
Post by: **silly_cone** on **Jan 03, 2017, 08:26 pm**

Update:

RF24 Network implements successfully on the ESP8266. I now have a functioning system that retrieves Date and Time information from NIST through wi-fi and broadcasts that out to the RF24

network. Just remember to add a few delay()'s throughout the code to keep the ESP running stably.

Title: **Re: ESP8266 version 12E with NRF24L01 - SOLVED!**
Post by: **issac_itd** on **Nov 29, 2017, 02:43 am**

Thank u very much!

Just what i was looking for.

My ESP-12 is now working with NRF24L01 module!

Title: **Re: ESP8266 version 12E with NRF24L01 - SOLVED!**
Post by: **derver** on **Mar 07, 2018, 10:37 am**

erorr
ets Jan 8 2013,rst cause:2, boot mode:(3,7)

load 0x4010f000, len 1384, room 16
tail 8
chksum 0x2d
csum 0x2d
v0c897c37
~ld

help me

Title: **Re: ESP8266 version 12E with NRF24L01 - SOLVED!**
Post by: **derver** on **Mar 07, 2018, 10:45 am**

turns out this error is encountered?

Soft WDT reset

ctx: cont
sp: 3ffef820 end: 3ffefb60 offset: 01b0

>>>stack>>>
3ffef9d0: 401063bc 3ffee8f0 3ffee910 401064b5
3ffef9e0: 00000010 3ffee8f0 3ffee910 402029f8
3ffef9f0: 00000017 3ffee8f0 3ffee8f0 40202a40
3ffefa00: 00000017 00000000 3ffee8f0 40202a71
3ffefa10: 00000010 00000000 3ffee8f0 40202eeb
3ffefa20: 00002cee 000000cc 3ffee8f0 40202fe9
3ffefa30: cccc3ce3 000000cc 00000000 00000000
3ffefa40: 3ffee768 00000001 00000000 40202489
3ffefa50: 00000003 00000000 3ffee8f0 00000000
3ffefa60: 00000001 00000000 3ffee768 40202503
3ffefa70: 00050001 feefeffe feefeffe feefeffe
3ffefa80: 00000000 3ffee910 3ffee8f0 401064b5
3ffefa90: 401063bc 3ffee910 3ffee8f0 00000004
3ffefaa0: 00000038 3ffefb30 3ffee768 402027f7
3ffefab0: 3ffefb38 3ffee8f0 3ffee910 40202a00
3ffefac0: 401063bc 3ffee768 3ffefb30 00000018
3ffefad0: 00000004 3ffefb30 3ffee768 4020286a
3ffefae0: 00000000 3ffefb38 00000038 3ffeeb2c

3ffefaf0: 00000001 3ffee768 3ffeeb08 402039b1
3ffefb00: 3ffe88e0 3ffeeb2c 40203ba4 3ffeeb2c
3ffefb10: 00000001 3ffee768 3ffeeb08 402029b3
3ffefb20: 3fffdad0 3ffee768 3ffeeb25 40201fa0
3ffefb30: 00010000 00000072 00000000 40203bc5
3ffefb40: 3fffdad0 00000000 3ffeeb25 40203bf0
3ffefb50: feefeffe feefeffe 3ffeeb40 40100108
<<<stack<<<
!ç
JçJ!!çHç

Title: **Re: ESP8266 version 12E with NRF24L01 - SOLVED!**
Post by: **jandrioli** on **Mar 14, 2018, 04:17 pm**

Quote from: derver on Mar 07, 2018, 10:45 am

> turns out this error is encountered?
>
> Soft WDT reset
>
> ctx: cont
> sp: 3ffef820 end: 3ffefb60 offset: 01b0
>
> >>>stack>>>
> 3ffef9d0: 401063bc 3ffee8f0 3ffee910 401064b5
> <<<stack<<<
> !ç
> JçJ!!çHç

Your issues have possibly 2 causes:

1 - bad code, like heap/stack collision, or some other bad referencing type of code bug

2 - bad sketch upload configuration; make sure the settings selected are correct for your device. For example the Flash (SPIFFS) size, speed, mode (DIO/QIO), all of these play crucial role.

Good luck.

Title: **Re: ESP8266 version 12E with NRF24L01 - SOLVED!**
Post by: **srinivasa_rao_pedada** on **May 22, 2018, 06:08 am**

hello silly_cone can you please post your sample code which was successfully working with nrf24l01, which would help me a lot in modifying mine.

Title: **Re: ESP8266 version 12E with NRF24L01 - SOLVED!**
Post by: **Robin2** on **May 22, 2018, 09:16 am**

This Simple nRF24L01+ Tutorial (http://forum.arduino.cc/index.php?topic=421081) may help. I have not tried it with an ESP8266 but as the TMRh20 RF24 library seems to work the examples should.

Wireless problems can be very difficult to debug so get the wireless part working on its own before you start adding any other features.

...R

Title: **Re: ESP8266 version 12E with NRF24L01 - SOLVED!**
Post by: **srinivasa_rao_pedada** on **May 22, 2018, 11:29 am**

Hello Robin thanks for the reply. I don't have much knowledge in programming and sorry if there are

any technical or grammatical errors.

I have successfully programmed and compiled NRF24L01 modules with arduino uno, but i need both LORA and RF communication because my project involves around 12 number of master slave communication. So i want to compile the same program with NODEMCU and NRF24L01. So i made the pin configuration like this

NRF ESP-12E
CSN ---------- D8
MOSI ---------- D7
MISO ---------- D6
SCK ---------- D5
CE ---------- D4

and D8 pin grounded through 4.7K ohm resistor.

my code which was successfully working arduino is:

Title: **Re: ESP8266 version 12E with NRF24L01 - SOLVED!**
Post by: **srinivasa_rao_pedada** on **May 22, 2018, 11:42 am**

code attached below. When I upload the code it gives me the following error in serial monitor.

Soft WDT reset

ctx: cont
sp: 3ffef820 end: 3ffefb60 offset: 01b0

>>>stack>>>
3ffef9d0: 401063bc 3ffee8f0 3ffee910 401064b5
3ffef9e0: 00000010 3ffee8f0 3ffee910 402029f8
3ffef9f0: 00000017 3ffee8f0 3ffee8f0 40202a40
3ffefa00: 00000017 00000000 3ffee8f0 40202a71
3ffefa10: 00000010 00000000 3ffee8f0 40202eeb
3ffefa20: 00002cee 000000cc 3ffee8f0 40202fe9
3ffefa30: cccc3ce3 000000cc 00000000 00000000
3ffefa40: 3ffee768 00000001 00000000 40202489
3ffefa50: 00000003 00000000 3ffee8f0 00000000
3ffefa60: 00000001 00000000 3ffee768 40202503
3ffefa70: 00050001 feefeffe feefeffe feefeffe
3ffefa80: 00000000 3ffee910 3ffee8f0 401064b5
3ffefa90: 401063bc 3ffee910 3ffee8f0 00000004
3ffefaa0: 00000038 3ffefb30 3ffee768 402027f7
3ffefab0: 3ffefb38 3ffee8f0 3ffee910 40202a00
3ffefac0: 401063bc 3ffee768 3ffefb30 00000018
3ffefad0: 00000004 3ffefb30 3ffee768 4020286a
3ffefae0: 00000000 3ffefb38 00000038 3ffeeb2c
3ffefaf0: 00000001 3ffee768 3ffeeb08 402039b1
3ffefb00: 3ffe88e0 3ffeeb2c 40203ba4 3ffeeb2c
3ffefb10: 00000001 3ffee768 3ffeeb08 402029b3
3ffefb20: 3fffdad0 3ffee768 3ffeeb25 40201fa0
3ffefb30: 00010000 00000072 00000000 40203bc5
3ffefb40: 3fffdad0 00000000 3ffeeb25 40203bf0
3ffefb50: feefeffe feefeffe 3ffeeb40 40100108

<<<stack<<<

I tested the NODEMCU module with wifisacn and ap examples and found working fine. Please suggest me hardware or software modifications if any to get it done.

Title: **Re: ESP8266 version 12E with NRF24L01 - SOLVED!**
Post by: **srinivasa_rao_pedada** on **May 24, 2018, 07:54 am**

Hello everyone, sorry if there is anything wrong in my question. Basically I need an SPI communication example which involves NODEMCU and NRF24L01, wherein it searches radio in all the six pipes (tmrh20 library) and displays received information on the serial monitor. Thanks in advance

Title: **Re: ESP8266 version 12E with NRF24L01 - SOLVED!**
Post by: **Robin2** on **May 24, 2018, 10:27 am**

I have no experience of using an nRF24 with an ESP8266 so I can't help with the error messages.

Quote

> Basically I need an SPI communication example which involves NODEMCU and NRF24L01, wherein it searches radio in all the six pipes (tmrh20 library) and displays received information on the serial monitor.

It will be much easier to help if you describe the project you want to implement so we understand the context for the solution you have in mind.

For example I have never found any need to use more than 1 pipe (even with lots of nRF24 slaves) so I can't imagine why you need 6.

...R

Title: **Re: ESP8266 version 12E with NRF24L01 - SOLVED!**
Post by: **srinivasa_rao_pedada** on **May 24, 2018, 11:01 am**

I have 12 slaves placed at a stretch of 1 km distance. I have allotted 1 pipe for two slaves to avoid packet loss and ease of identification. 12 slaves are used to control some 50 induction motors like switching on and getting on feedback and local alarm in case of fail to start etc.. All these I/O s are handled by arduino uno and NRF24L01 module combination. Whereas feedback part for remote viewing of alarm and running status will be uploaded to thingspeak channel using NODEMCU and NRF24L01 modules.

Title: **Re: ESP8266 version 12E with NRF24L01 - SOLVED!**
Post by: **Robin2** on **May 24, 2018, 11:30 am**

Interesting. A bit more detail will be useful.

What Arduino is at each slave location?

How do the distant slaves communicate with the master unit? Is each of the 12 slave communicating directly with the master? Or is (say) slave A communicating with slave B and is slave B passing on the data to slave C so that it gets to the master as a series of "hops"?

If the slaves communicate directly with the master I don't see any advantage in using multiple pipes on the master. As it only has one radio receiver if two or more slaves transmit at the same time all the messages will be garbled.

How often do messages need to be exchanged between the master and the slaves?

Is there two-way communication between the master and the slaves.

...R

Title: **Re: ESP8266 version 12E with NRF24L01 - SOLVED!**
Post by: **srinivasa_rao_pedada** on **May 24, 2018, 11:56 am**

1: Arduino uno is used throughout.
2: For certain commands like On command and alarm feedback , power failure will be passed to distant slaves by nearby slaves.

3. All of them may communicate at the same time not often but it depends alarm trigger.

4. All modules have two way communication.

Title: **Re: ESP8266 version 12E with NRF24L01 - SOLVED!**
Post by: **Robin2** on **May 24, 2018, 12:25 pm**

Thanks.

It sounds as if most of the time all the slaves communicate directly with the master but occasionally some of them communicate via other slaves. How is that managed?

Quote

> For certain commands like On command and alarm feedback , power failure

Is "power failure" a message just like ON or ALARM or is it the reason why the commands are not sent directly to the slave?

What do you use the ESP8266 for?
Which node is it connected to?

...R

Title: **Re: ESP8266 version 12E with NRF24L01 - SOLVED!**
Post by: **srinivasa_rao_pedada** on **May 24, 2018, 04:22 pm**

1. Occasional communication will be triggered by received integer. It will be compared to a predefined value and then communication starts which are managed by different pipe allocation.

2. Power failure is just a message like ON /ALARM.

3. This project is being developed for an aquaculture farmer who are facing problems with frequent power cuts during rainy season during night time. It is a difficult situation for them switching on the motors many times which are placed at 1km stretch walking on road full of mud.

4. ESP 8266 is used only for running status of all motors and power failure and alarms etc. data will be uploaded to a thingspeak channel which can be monitored remotely.

Title: **Re: ESP8266 version 12E with NRF24L01 - SOLVED!**
Post by: **Robin2** on **May 24, 2018, 04:52 pm**

Quote from: srinivasa_rao_pedada on May 24, 2018, 04:22 pm

> 1. Occasional communication will be triggered by received integer. It will be compared to a predefined value and then communication

> starts which are managed by different pipe allocation.

Perhaps I did not explain my question clearly. What I want to know is how the decision is sometimes made for the farthest Arduino to talk directly to the master and at other times the decision is made to talk to the master through other nodes.

Quote

> 4. ESP 8266 is used only for running status of all motors and power failure and alarms etc. data will be uploaded to a thingspeak channel which can be monitored remotely.

Does that mean the ESP8266 is just being used as a WiFi module connected to the master Uno?

Do all the slave Arduinos have the same program? If so please post the program and the program that is running on the master.

...R

Title: **Re: ESP8266 version 12E with NRF24L01 - SOLVED!**
Post by: **swilli38** on **Jun 23, 2018, 11:43 pm**

I recognise this was a while ago but I'm having the same trouble to make an NRF24L01 receive data to an ESP8266 (NODEMCU development board). I've had data TX and RX perfectly with NRF using Arduino Nano and Pro Minis.

However, as soon as I replace with an ESP8266 it seems to receive different data size (I sent 7 but claims it received 16) and the data reads zero. The RX program on the ESP8266 loops through OK.

I've tried adding a pull down resister to pin D8 but no change.

Would it be possible for you to advise further? perhaps share your code and PIN out for the ESP8266 to the NRF. Many thanks.

Title: **Re: ESP8266 version 12E with NRF24L01 - SOLVED!**
Post by: **silly_cone** on **Jul 03, 2018, 10:05 pm**

I hadn't seen these new posts until just now. Its been a long time since I've thought about this, but as luck has it I'll be working on this same kind of problem tomorrow, so I should have some insight to share as I re-familiarize myself with it.

Title: **Re: ESP8266 version 12E with NRF24L01 - SOLVED!**
Post by: **Robin2** on **Jul 03, 2018, 10:52 pm**

Quote from: swilli38 on Jun 23, 2018, 11:43 pm

> However, as soon as I replace with an ESP8266 it seems to receive different data size (I sent 7 but claims it received 16) and the data reads zero. The RX program on the ESP8266 loops through OK.

I seem to have missed this Reply until now.

I wonder is the difference due to different sizes for data elements. For example a Nano treats the smallest variable size as 8 bits (a byte or char), but maybe an ESP8266 uses 16 bits for the smallest variable size. However an nRF24 sends all data as bytes.

...R

Title: **Re: ESP8266 version 12E with NRF24L01 - SOLVED!**
Post by: **silly_cone** on **Jul 06, 2018, 08:18 pm**

There shouldn't be any issues with variable or byte size; esp will use byte or uint8_t variables just fine and treats them as 1 byte.

I think you may need to share some code swilli38.

These are the NodeMCU units I'm using:

When I use two nodes of these boards, each with an NRF24L01 with a 4.7K pull-down on D8 I get rock-solid communication between the two when using the code below. And when I have the receiver print the size of the packet using the sizeof() command, its spot-on.

Receiver Code:
Code: [Select]

```
#include <SPI.h>
#include <nRF24L01.h>
#include <RF24.h>

const int pinCE = 2;
const int pinCSN = 15;
byte gotByte = 0;

RF24 wirelessSPI(pinCE, pinCSN);
const uint64_t pAddress = 0xB00B1E5000LL;

void setup()
{
  Serial.begin(9600);
  wirelessSPI.begin();
  wirelessSPI.setAutoAck(1);
  wirelessSPI.enableAckPayload();
  wirelessSPI.setRetries(5,15);
  wirelessSPI.openReadingPipe(1,pAddress);
  wirelessSPI.startListening();
  wirelessSPI.printDetails();
}

void loop()
{
    while(wirelessSPI.available()){
     wirelessSPI.read( &gotByte, sizeof(gotByte) );
     Serial.print("Recieved packet number: ");
     Serial.println(gotByte);
     Serial.print("Size: ");
     Serial.println(sizeof(gotByte));
    }

  delay(100);
}
```

Transmitter Code:
Code: [Select]

```
#include <SPI.h>
#include <nRF24L01.h>
#include <RF24.h>

const int pinCE = 2;
const int pinCSN = 15;
byte counter = 1;
bool done = false;
RF24 wirelessSPI(pinCE, pinCSN);
const uint64_t pAddress = 0xB00B1E5000LL;

void setup()
{
  Serial.begin(9600);
  wirelessSPI.begin();
  wirelessSPI.setAutoAck(1);
  wirelessSPI.enableAckPayload();
  wirelessSPI.setRetries(5,15);
  wirelessSPI.openWritingPipe(pAddress);
  wirelessSPI.stopListening();
  wirelessSPI.printDetails();
}


void loop()
{
   if (!wirelessSPI.write( &counter, 1 )){
       Serial.println("packet delivery failed");
   }
   else {
       counter++;
   }
    delay(750);
}
```

Title: **Re: ESP8266 version 12E with NRF24L01 - SOLVED!**
Post by: **giovi61** on **Feb 25, 2019, 05:59 pm**

Hi, I see that many days have passed since you did your last surgery and I hope you can answer me. I tried to implement the code you wrote, but I could not run esp-12 (NodeMcu v3) + NRF24L01. The NRF module works normally with arduino Uno and Nano, but not with the Esp-12. The result is a continuous reset of the module:

STATUS = 0x0e RX_DR=0 TX_DS=0 MAX_RT=0 RX_P_NO=7 TX_FULL=0
RX_ADDR_P0-1 = 0xb00b1e5000 0xc2c2c2c2c2
RX_ADDR_P2-5 = 0xc3 0xc4 0xc5 0xc6
TX_ADDR = 0xb00b1e5000
RX_PW_P0-6 = 0x20 0x00 0x00 0x00 0x00 0x00
EN_AA = 0x3f
EN_RXADDR = 0x03
RF_CH = 0x4c
RF_SETUP = 0x07
CONFIG = 0x0e
DYNPD/FEATURE = 0x03 0x06

Exception (28):
epc1=0x40206d8a epc2=0x00000000 epc3=0x00000000 excvaddr=0x00008d58 depc=0x00000000

>>>stack>>>

ctx: cont
sp: 3ffffc90 end: 3fffffc0 offset: 01a0
3ffffe30: 3ffef0f0 00008d5a 3ffffe90 40205539
3ffffe40: 3ffe8020 00000006 00000000 40238c58
3ffffe50: 3ffe8020 3ffe8020 3ffef0f0 40209948

```
3ffffe60:  3ffe8020 3ffe8020 3ffef0f0 40209acc
3ffffe70:  00000004 ffffffff ffffffff 40238bdb
3ffffe80:  40238bda 3ffe8020 3ffef0f0 40209cf0
3ffffe90:  00000000 ffffffff 00000000 00000000
3ffffea0:  00000004 0000000d 00302073 401007b8
3ffffeb0:  00363430 3fff7820 00000008 00000002
3ffffec0:  3fffff40 3fffff30 00000000 00000008
3ffffed0:  00ffff50 3fffff40 00000004 00000002
3ffffee0:  00000002 00000000 00000000 3fffff60
3ffffef0:  3fffff70 3fffff60 00000008 0000001e
3fffff00:  00000009 00000001 3ffe8020 40209f92
3fffff10:  3fffff50 3fffff40 00000004 3ffee334
3fffff20:  3fffdad0 40238bcc 3ffe8020 40209f92
3fffff30:  3fffff70 3fffff60 00000004 401001cd
3fffff40:  0000001d 00000000 3ffee2a8 40206d20
3fffff50:  3fffff70 3fffff60 00000004 40202a14
3fffff60:  00000006 00008d5a 00000001 00000005
3fffff70:  5ffffe00 00000000 3ffee2a8 4020317c
3fffff80:  3fffdad0 00000000 3ffee2a8 402032c1
3fffff90:  3fffdad0 00000000 3ffee2a8 402024b2
3fffffa0:  feefeffe feefeffe 3ffee304 402036e0
3fffffb0:  feefeffe feefeffe 3ffe84f8 40100a0d
<<<stack<<<

 ets Jan  8 2013,rst cause:2, boot mode:(3,6)

load 0x4010f000, len 1384, room 16
tail 8
chksum 0x2d
csum 0x2d
v21db8fc9
```

Someone over silly_cone managed to make the 2 modules work together. If yes, please share your experience, problems and solutions.
Thank you