

به نام خدا

# راه اندازی ماژول nrf24l01p + هدر فایل (c)

سید حسن مرتضوی

mortazavi91@gmail.com

بهار ۹۲

با سلام

در این مقاله قصد دارم یک هدر فایل برای راه اندازی ماژول nRF24L01P که یکی از بهترین ماژول های مخابراتی با قابلیت اطمینان بالا در محیط های صنعتی میباشد(تست شده)، معرفی کنم.

## قسمت سخت افزاری:

قبل از هر چیز باید توجه کنید که برای تغذیه ماژول از ولتاژ بین ۱.۹ تا ۳.۶ ولت استفاده کنید. می توانید برای تغذیه میکرو از تغذیه ماژول استفاده کنید. ولی اگر از ۵ ولت استفاده میکنید، باید تغذیه ماژول را بین ۲.۲ تا ۳.۶ بگذارید. (درضمن حتما بین ولتاژ مثبت و زمین تغذیه از خازن استفاده کنید)

خوب پایه های ماژول را به این صورت به میکرو وصل کنید(روی میکروهای mega32 و mega16 و مشابهات)(میکرو از طریق spi با ماژول ارتباط برقرار میکند):

IRQ : PORTB.2

CE : PORTB.3

CSN : PORTB.4

MOSI : PORTB.5

MISO : PORTB.6

SCK=CLK : PORTB.7

## قسمت نرم افزاری:

خوب حالا برسیم به قسمت نرم افزاری که با C نوشته شده (این هدر نیاز به وقفه خارجی int2 دارد):

لینک دانلود فایل هدر nrf24l01p.h :

<http://mortazavi91.persianguig.com/Codes/nrf24l01p.h>

قبل از هرچیز فایل هدر را ضمیمه برنامه می کنیم:

```
#include "nrf24l01p.h"
```

خوب حالا در ابتدای تابع main() ، تابع nrf\_config(mode) را فراخوانی کنید، که در صورتی که میخواهید از ماژول به عنوان فرستنده استفاده کنید بجای مود عدد 0 و در صورتی که به عنوان گیرنده استفاده میکنید عدد 1 را به عنوان پارامتر به تابع رد کنید. به این صورت:

```
transmitter : nrf_config(0);
```

```
receiver : nrf_config(1);
```

خوب الان ماژول پیکر بندی شده حالا بریم برای اینکه اطلاعات ارسال یا دریافت کنیم. قبل از این می خوام در رابطه با متغیر nrf\_state صحبت کنم که با خواندن اون در هرجایی از برنامه وضعیت ماژول را مشخص میکند. به این صورت:

- اگر مقدار آن برابر 0 بود ماژول در حالت بی کار است.
- اگر مقدار آن برابر 1 بود یعنی چیزی دریافت شده است. (در مود گیرنده)
- اگر مقدار آن برابر 2 بود یعنی اطلاعات (payload) به درستی ارسال شده است. (successfully sent)

- اگر مقدار آن برابر ۳ بود یعنی اطلاعات (payload) به درستی ارسال شده و در جواب از سوی گیرنده چیزی دریافت شده است. (successfully sent with ACK payload)
- اگر مقدار آن برابر ۴ بود یعنی اطلاعات (payload) به درستی ارسال نشده است. (unsuccessfully sent)

برای ارسال اطلاعات ابتدا تعداد بایتی که می خواهید ارسال کنید را در خانه صفر آرایه payload[] قرار دهید (payload[0]). سپس بایت ها را پشت سرهم در آرایه قرار دهید (حداکثر تا ۳۲ بایت). به عنوان مثال اگر بخواهیم سه بایت 0xAA, 0xBB, 0xCC را ارسال کنیم آرایه payload[] را به این نحو پر میکنیم:

```
payload[0]=3;
```

```
payload[1]=0xAA;
```

```
payload[2]=0xBB;
```

```
payload[3]=0xCC;
```

حالا برای ارسال تابع sent() را فراخوانی میکنیم. به همین راحتی .

برای اطمینان و قرار دادن انحصار متقابل من از سمافور sent\_actived داخل تابع sent() استفاده کردم.

زمانی که وضعیت nrf\_state مخالف صفر شد باید nrf\_state را صفر و sent\_actived رو یک کنید. به این صورت که به طور دائم شرط زیر چک شود:

```

if(nrf_state!=0)
{
    sent_actived=1; //faal sazi ersal mojadad
    nrf_state=0;
}

```

حالا بریم سراغ گیرنده : دیگه باید خودتون بدونید چیکار باید کرد.

اول `nrf_config(1)` و بعد از اون منتظر می مونیم تا `nrf_state` برابر ۱ شود در این صورت مقدار `payload[0]` برابر تعداد بایت های دریافتی و مابقی بایتها اطلاعات دریافتی می باشد. بعد از خوندن اطلاعات حتما دوباره `nrf_state` رو صفر کنید.

به عنوان مثال می تونید به صورت زیر عمل کنید که به طور دائم شرط زیر چک شود:

```

if(nrf_state==1)
{
    for (i=1;i<=payload[0];i++)
    {
        harjake mikhaid berizid=payload[i]
    }
    nrf_state=0;
}

```

تمام. البته این شبه کدهایی که نوشتم فقط به عنوان مثال بود و با توضیحاتی که دادم می تونید  
به هر نحوی که دوست دارید عمل کنید.

به امید موفقیت ... مرتضوی

mortazavi91@gmail.com