



راه اندازی

**Spi flash**

در

بسکام

یونس عباس زاده

Electrolab.ir & electrolab.org

## بسمه تعالی

در بسیاری از پروژه ها کمبود حافظه میکرو کنترلر باعث میشود که از حافظه جانبی برای ذخیره اطلاعات استفاده کنیم در اولین مرحله چیزی که به فکر میرسد استفاده از مموری کارت هست اما این برای پروژه هایی هست که شما به کاربر اجازه استفاده از مموری کارت را نمیدهید چرا که شما از این ویژگی میکرو کنترلر برای ذخیره اطلاعات کمپانی یا مشخصات دستگاه یا تنظیمات پر حجم استفاده میکنید مثلا در اسانسور طراح از یک مموری کارت برای ذخیره اهنگ و تصاویر روی نمایشگر استفاده میکنه ودر اسانسور نیازی به نصب مموری کارت از طرف کاربر برای پخش موسیقی ویا گرفتن عکس و... نیست فقط نصاب یا شرکت اسانسوری از این مموری کارت استفاده میکند . اما اگر شما یک دستگاهی مثل یک پخش کننده موسیقی میسازید که شامل یه نمایشگر هست باید محلی برای ذخیره اطلاعاتی از قبیل عکس فولدر عکس پشت زمینه نمایشگر و ... پیدا کنید در این موقع شما استفاده از مموری کارت رو به کاربر محول کردید پس نمیتونید از مموری کارت برای ذخیره اطلاعات استفاده کنید راه حل این است که از ایسی های فلش استفاده کنید این ایسی ها ظرفیتی در حدود 8 مگا بایت دارند این ایسی ها یا استفاده از ارتباط spi با میکرو کنترلر ارتباط برقرار میکنند و عملیات نوشتن و خواندن در این ایسی صورت میگیرد

در این مقاله نحوه اتصال و خواندن و نوشتن اطلاعات در این ایسی ها آموزش داده میشود

عباسزاده

در شروع کار این نکته را متذکر میشوم که تمامی حافظه های فلش در دستورات بیس آورده شده در زیر باهم برابرند

Command Name	1-Byte Command Code		Bytes		
			Address	Dummy	Data
WRITE ENABLE	0000 0110	06h	0	0	0
WRITE DISABLE	0000 0100	04h	0	0	0
READ IDENTIFICATION	1001 1111	9Fh	0	0	1 to 20
READ STATUS REGISTER	0000 0101	05h	0	0	1 to ∞
READ DATA BYTES	0000 0011	03h	3	0	1 to ∞
READ DATA BYTES at HIGHER SPEED	0000 1011	0Bh	3	1	1 to ∞
PAGE WRITE	0000 1010	0Ah	3	0	1 to 256
PAGE PROGRAM	0000 0010	02h	3	0	1 to 256
PAGE ERASE	1101 1011	DBh	3	0	0
SECTOR ERASE	1101 1000	D8h	3	0	0
DEEP POWER-DOWN	1011 1001	B9h	0	0	0
RELEASE from DEEP POWER-DOWN	1010 1011	ABh	0	0	0

دستورات بالا در ساده ترین ایسی فلش وجود دارد و شرکتهای مختلف امکانات مختلف و دستورات مختلفی میذارن مثلا امکان قفل فلش و پاک کردن کل حافظه فلش و...

ایسی فلشی که قراره در این مقاله باهاش کار کنیم W25Q32FV میباشد که دارای 32 مگابایت یا 4 مگابایت حافظه قابل نوشتن هست.

این ایسی ها با پروتکل spi کار میکنند تنظیمات مربوط به spi در بسکام به صورت زیر باید تنظیم بشه

Config Spi = Hard , Interrupt = Off , Data Order = Msb , Master = Yes , Polarity = High , Phase = 1 , Clockrate = 4 , Noss = 1

در ضمن پایه مربوط به انتخاب ایسی در باس spi به صورت زیر تعریف میشه و در هنگام نوشتن و خواندن اطلاعات با reset کردن این پایه ایسی انتخاب میشود به عبارتی این پایه به enable ایسی وصل میشود

Config Pinb.2 = Output

Ss Alias Portb.2

استفاده از این ایسی بسیار آسان هست شما قبل از هر اقدامی مثل نوشتن و خواندن ویپاک کردن باید دستور write enable روی ایسی اجرا کنید برای این کار با فراخوانی زیر برنامه زیر این دستور به ایسی ابلاغ میشود:

Call W\_enable

برای اینکه یک برنامه کامل برای هر ایسی از کارخانه های مختلف طراحی کنید باید id ایسی را که به میکرو وصل کردید را بخوانید و متناسب با آن دستورات مختص به خودش را استفاده کنید . که با دستور زیر id ایسی را بخوانید

Call Get\_id

بعد از اجرای این دستور اطلاعات مربوط در سه بایت به نامهای زیر ذخیره میشه

D\_bytes(1) : manufacturer id

D\_bytes(2) : memory type

D\_bytes(3) :memory capacity

بایت اول مربوط به id کارخانه سازنده هست که با توجه به دیتاشیت در صفحه 25 باید مقدار EF هگز یا 239 دسیمال در این متغیر ذخیره شود همچنین مقادیر بایتهای بعدی باید طبق جدول به صورت زیر باشد

D\_bytes(2) :&H16

D\_bytes(3) :&H40

که به صورت زیر مقادیر این بایتها خوانده و به پورت سریال پرینت میشود

Call W\_enable

"Print "read id

Call Get\_id

Print

(Print "manufacturer id = " ; D\_bytes(1

Print

(Print "memory type = " ; D\_bytes(2

Print

(Print "memory capacity = " ; D\_bytes(3

Print

موقع کار با این ایسی ممکن هست ایسی مشغول به انجام کاری باشد و شما دستوری میفرستید با این امید که پاسخی از ایسی دریافت کنید در این حالت ایسی هیچ نوع عملکردی از خود نشان نمیدهد چرا که مشغول به انجام کاری میباشد خب برای اینکه متوجه شویم که ایسی در حالت مشغول هست یا نه باید بیت شماره 0 از

رجیستر را status register بررسی کنیم که ایا 0 هست یا 1 اگر 0 باشد ایسی آزاد هست اما اگر 1 باشد ایسی مشغول (busy)

هست پس باید بعد از هر عملی (نوشتن خواندن و...) باید منتظر 0 شدن این بیت باشیم اما نحوه خواندن این رجیستر؟ ایسی W25Q32FV دارای 3 تا STATUS REGISTER هست که ما فقط با یکی کار داریم برای این کار با دستور زیر استاتوس رجیستر اول خوانده میشود و مقدار آن باز در بایت (1) D\_bytes ذخیره میشود:

Call W\_enable

Call R\_register

که به صورت زیر بیت شماره 0 را از بایت (1) D\_bytes میخوانیم:

If D\_bytes(1).0 = 0 Then

"Print "not busy

Print

Else

"Print "busy

Waitms 1000

Print

end IF

## نوشتن در حافظه فلش:

در ابتدا توضیح دادیم که قبل از هر دستوری دستور Call W\_enable باید اجرا شود. در مورد نوشتن در حافظه فلش نیز باید این دستورات اجرا یا فراخوانی بشه

برای نوشتن باید آدرس محل شروع نوشتن مشخص بشه با در این ایسی 24 بیت برای آدرس اختصاص داده شده که میتواند 16 مگابایت را آدرس دهی کرد ولی چون ظرفیت ما 4 مگابایت هست ما مشکلی نداریم ما سه بایت برای ذخیره آدرس در نظر میگیریم در ضمن برای اجرای هر عملی یا دستوری باید کد یا آدرس مربوط بفرستیم سپس در سه بایت آدرس مکانی رو که میخواهیم داده در آن نوشته شود را میفرستیم سپس در بایت بعدی خود داده فرستاده میشود به مثال زیر توجه کنید:

B\_bytes(1) = 2

آدرس مربوط به دستور نوشتن طبق جدول صفحه قبل

B\_bytes(2) = 0

B\_bytes(3) = 2

B\_bytes(4) = 0

در سه بایت 2 و 3 و 4 ادرس محل نوشتن رو دادیم که مقدار 512 میباشد

B\_bytes(5) = 5

حالا مقدار خود داده که در اینجا 5 میباشد رو قرار میدهیم

دقت کنید که قبل از هر نوشتن در یک ادرس باید ادرس محل رو در بایتهای بالا قرار بدید و مقدار داده رو هم باید مقداردهی شود سپس توسط دستور زیر عمل نوشتن در حافظه انجام میشود:

W\_data

### نکته بسیار مهم:

اگر داده ای در یک ادرس نوشته شود امکان رونوشت داده (یعنی نوشتن داده ای دیگر در همان ادرس) میسر نیست و باید سکتوری که بایت مورد نظر در آن هست رو کلا پاک کنید و دوباره عملیات نوشتن رو انجام بدید

### خواندن اطلاعات از حافظه:

برای خواندن اطلاعات از حافظه باید ادرس دستور رو که عدد 3 میباشد رو اول بفرستید سپس ادرس رو بفرستید برای این کار ابتدا مقادیر را در بایت های زیر قرار میدهیم :

B\_bytes(1) = 3

درس مربوط به دستور خواندن

B\_bytes(2) = 0

B\_bytes(3) = 2

در سه بایت 2 و 3 و 4 ادرس دیتای رو که میخوایم بخوانیم رو میفرستیم که در اینجا 512 میباشد B\_bytes(4) = 0  
حال دستور زیر را مینوسیم:

R\_data

در پروتوس شبیه ساز این ایسی وجود نداشته و از مشابه این ایسی استفاده شده که امکانات کمتری دارد مثلا قابلیت پاک کردن سکتور وجود ندارد و همچنین ادرس ایدی متفاوتی دارد

به طور کلی برنامه اصلی به صورت زیر خواهد بود و شما تغییرات خود را در این قسمت اعمال خواهید نمود :

```

Do
Call W_enable
Print "read id"
Call Get_id
Print
Print "manufacturer id = " ; D_bytes(1)
Print
Print "memory type      = " ; D_bytes(2)
Print
Print "memory capacity = " ; D_bytes(3)
Print
Waitms 2000
Call W_enable
Call R_register
Print "read register"
Print "register1 = " ; D_bytes(1)
If D_bytes(1).0 = 0 Then
Print "not busy"
Print
Else
Print "busy"
Waitms 1000
Print
End If
E_bytes(1) = 2
E_bytes(2) = 0
E_bytes(3) = 2
E_bytes(4) = 0
E_bytes(5) = 5

Call W_data
Print "WRITE VALUE 5 IN ADDRESS 512"
E_bytes(1) = 3
E_bytes(2) = 0
E_bytes(3) = 2
E_bytes(4) = 0
Call R_data
Print "READ VALUE 5 IN ADDRESS 512"
Loop
End

```

ElectronicsLabs.ir

## با ارزی موفقیت شما عزیزان



### یونس عباس زاده

برای پاسخ به سوالات شما به انجمن

[www.electrolab.org](http://www.electrolab.org)

و یا به وبسایت ما

[www.electrolab.ir](http://www.electrolab.ir)

مراجعه نمایید