

دانشگاهصنعتیاصفهان

دانشکدهبرقوکامپیوتر

**دستورکار آزمایشگاه طراحی مدارهای واسط**

**آزمایش یازدهم**

تهیه کننده: مهران صفایانی

پاییز95

**آشنایی با ارتباط SPI**

* **اهداف:**
* **آشنایی با پروتکل ارتباطی SPI ونحوه کار با آن**
* **بررسی رجیستر ها و توابع مربوط به SPI**
* **مقدمه:**

در این آزمایش با استاندارد ارتباطی SPI اشنا می شوید سپس با استفاده از این ارتباط اطلاعات یک مبدل قیاسی به رقمی در خارج میکروکنترلر را دریافت می کنید.

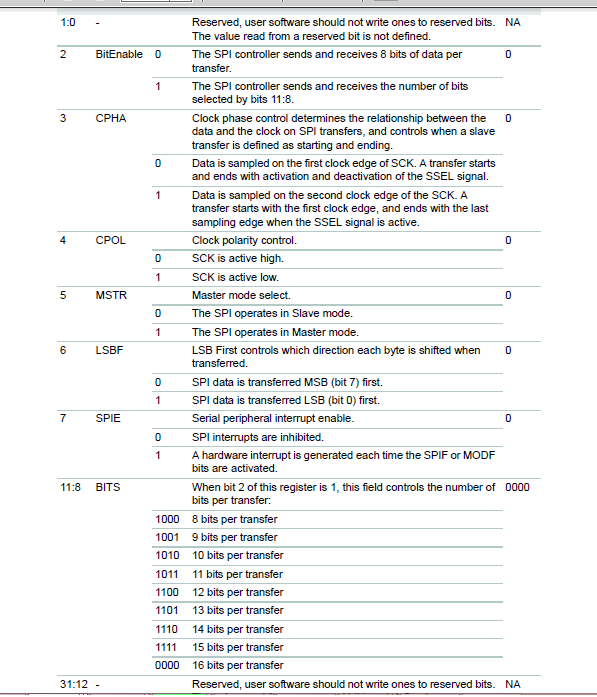
* **بررسی بلوک SPI :**

از این پروتکل جهت ارتباط به صورت سریال سنکرون با ادوات جانبی استفاده می شود . جهت استفاده از این پروتکل یکی از وسایل ارتباطی به صورت MASTER و طرف دیگر ارتباط به صورت SLAVE می باشد . وظیفه MASTER کنترل ارتباط ما بین ادوات ارتباطی مختلف می باشد. جهت ارتباط و ارسال اطلاعات از MASTER به سمت SLAVE از ۴ پایه MISO ،MOSI ،SCK و SS استفاده می شود. وظیفه SCK یا کلاک سنکرون سازی بین MASTER وSLAVE است.MOSI سیم ارسال اطلاعات از MASTER به سمت SLAVE است و MISO به طور عکس اطلاعات را از SLAVE به سمت MASTER می فرستد.از پایه ss زمانی استفاده می شود که چند SLAVE داریم و می خواهیم با آنها ارتباط برقرار کنیم . در SLAVE ها این پایه به صورت ورودی است و زمانی امکان برقرای ارتباط می دهد که مقدار آن صفر باشد و اگر یک باشد SLAVE مد نظرمان غیرفعال می شود.

* **بررسی رجیسترهای SPI و نحوه کار با آن :**

**رجیستر SPCR :**رجیستر کنترل SPI است. بیت دوم این رجیستر جهت ارسال بیش از ۸ بیت این بیت را فعال می کنیم. در صورتی که بخواهیم به صورت ۸ بیت انتقال دهیم نیاز به فعال کردن بیت دوم یا Bit Enabe نیست. بیتهای شماره ۳و۴ به نام های CPHA و CPOL جهت تعیین فاز و پلاریته ی اطلاعات مورد استفاده قرار می گیرد. بیت شماره ۵ یاMSTR جهت تعیین MASTER یا SLAVE است و در صورت یک دادن به این بیت میکرو در مد MASTER قرار می گیرد.بیت شماره ۶ یا LSBF در صورت یک بودن ارسال اطلاعات از بیت کم ارزش صورت می گیرد در صورت صفر بودن از بیت پر ارزش ارسال می شود. بیت شماره ۷ یا SPIE در صورت یک بودن وقفه برای ما فعال می کند. بیتهای 8 تا ۱۱ نیز جهت مشخص کردن تعداد بیت ارسالی در صورت یک بودن بیت دوم این رجیستر است. ساده ترین تنظیم برای این رجیستر بدین صورت است که SPI را درمد MASTER یا SLAVE قرار دهیم و ارسال داده از بیت کم ارزش LSB وارسال به صورت ۸ بیتی صورت گیرد.به عنوان مثال :

LPC\_SPI-> SPCR = (1<<5) | (1<<6);// (Master)(LSBF)



**رجیستر SPSR :** رجیستر وضعیت SPI است که با استفاده از آن می توان به کامل شدن عملیات ارسال و دریافت پی برد.یکی از بیتهای مهم برای ما بیت شماره ۷ یا SPIF است و زمانی که ارسال کامل شد این بیت یک می شود و در صورت فعال بودن بیت SPIE در رجیستر کنترل وقفه رخ می دهد.به عنوان مثال با دستور زیر منتظر کامل شدن ارسال می شویم.

while( !(LPC\_SPI->SPSR & 0x80));//wait for transfer to be completed

**رجیستر SPDR :** رجیستر داده است و داده های ارسالی و دریافتی در داخل این رجیستر قرار می گیرد از دوقسمت ۸ بیتی کم ارزش و پرارزش تشکیل شده است و زمانی که از حالت ۸ بیتی استفاده کنیم تنها قسمت کم ارزش مورد استفاده قرار می گیرد. دستور آن نیز به صورت LPC\_SPI->SPDR است.

**رجیستر SPCCR :** رجیستر تنظیم کلاک واحد SPI است و مقدارPCLK بر مقدار این رجیستر تقسیم می شود.

**رجیستر SPINT :** رجیستر پرچم وقفه است.

## مراحل پیکربندی SPI :

تنظیم پایه ها مربوط به SPI با استفاده از رجیستر PINSELL انجام می گیرد به طوریکه پایه P0.15 برای SCK ، پایه های P0.17 وP0.18 به ترتیب MISO وMOSI قرار گیرد.

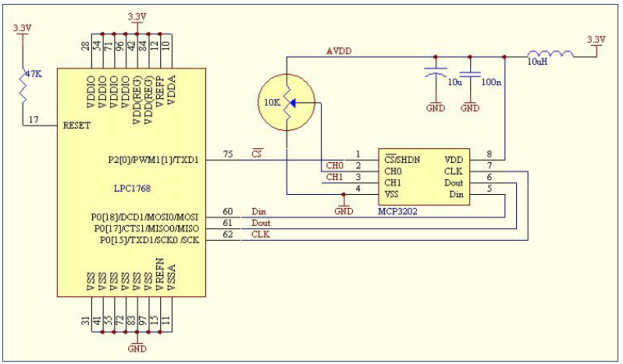
LPC\_PINCON->PINSEL1 |= (3<<2) | (3<<4);

LPC\_PINCON->PINSEL0 |= (3<<30);

* **راه اندازی مبدل آنالوگ به دیجیتال خارجی با پروتکل SPI :**

بر روی بورد آموزشی یک عدد آی سی مبدل آنالوگ به دیجیتال به شماره MCP3202 از شرکت میکروچیپ قرار داده شده است. این آی سی با استفاده از پروتکل ارتباطی SPI به میکرو متصل می‌شود.ارتباط سریال این آی سی دوطرف است و اطلاعات دیجیتال را در قالب ۴ کلمه ۸ بیتی می‌توان خواند.

**برنامه نمونه:** برنامه ای بنوسید که به وسیله آی سی MCP3202 ولتاژ ورودی آی سی به صورت ۱۲ بیتی خوانده شده و نتیجه توسط ۱۲عدد LED نمایش داده شود.(مطابق شکل زیر)



#include <lpc17xx.h>

int main(void){

int data = 0;

LPC\_SC -> PCONP |= (1 << 8); //Enable SPI power/clock control bit

LPC\_PINCON -> PINSEL0 |= 0xC0000000; //Enable P0.15 for SCK

LPC\_PINCON -> PINSEL1 |= 0x0000003C; //Enable P0.17 for MISO and P0.18 for MOSI

LPC\_SPI -> SPCCR = 2; //Divide Pclock/2 for SPI

LPC\_SPI -> SPCR = 0x20; //SPI send & receive 8 bit per transfer and SPI work as Master

LPC\_GPIO0 -> FIODIR0 = 0x01; //Set P0.0 output for CS

LPC\_GPIO2 -> FIODIRL = 0XFFFF; //Set P2.0 - P2.11 output for LEDs

while(1){

LPC\_GPIO0 -> FIOPIN0 = 0x00; //Enable CS

LPC\_SPI -> SPDR = 0x01;

while(!(LPC\_SPI -> SPSR & 0x80)); //Wait for send data

LPC\_SPI -> SPDR = 0xA0;

while(!(LPC\_SPI -> SPSR & 0x80)); //Wait for send data

data = LPC\_SPI -> SPDR;

LPC\_SPI -> SPDR = 0x00;

while(!(LPC\_SPI -> SPSR & 0x80)); //Wait for send data

data = data << 8;

data |= LPC\_SPI -> SPDR;

LPC\_GPIO0 -> FIOPIN0 = 0x01;//Disable CS

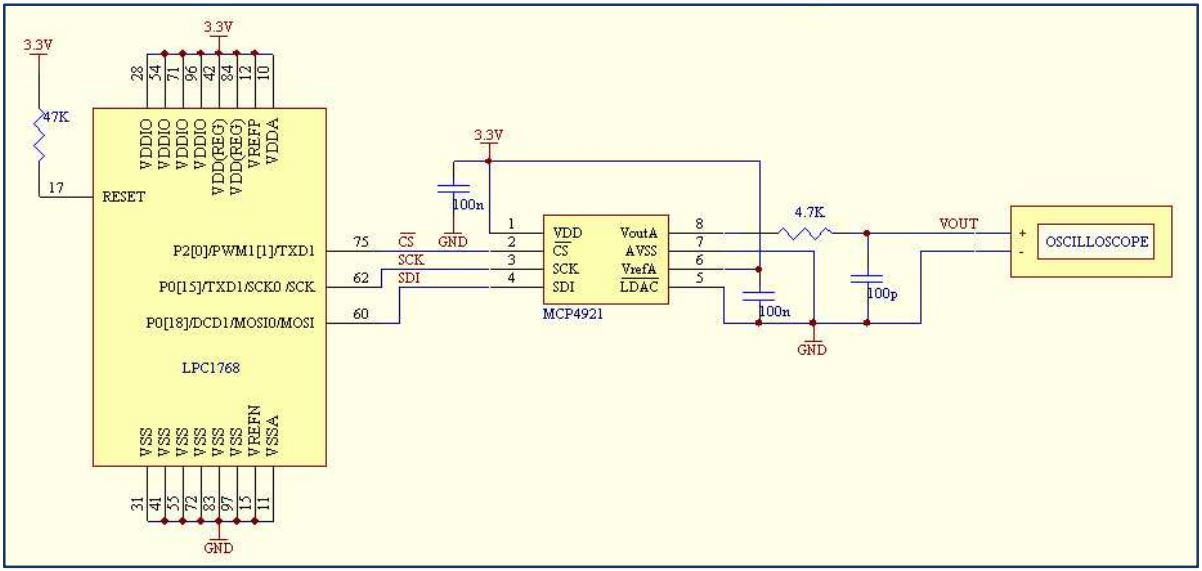
LPC\_GPIO2 -> FIOPINL = data;

data--;

data--;

}

}

* **دستورکار:**
  1. ****با استفاده از واسط SPI مبدل رقمی به قیاسی MCP4921 که بر روی برد تعبیه شده است را راه اندازی نمایید ویک شکل موج سینوسی تولید نمایید و نتیجه را بر روی اسیلوسکوپ مشاهده نمایید.