

VJEŽBA 5: SPI SUČELJE. AKCELEROMETAR LIS3DSH.

I. Cilj vježbe

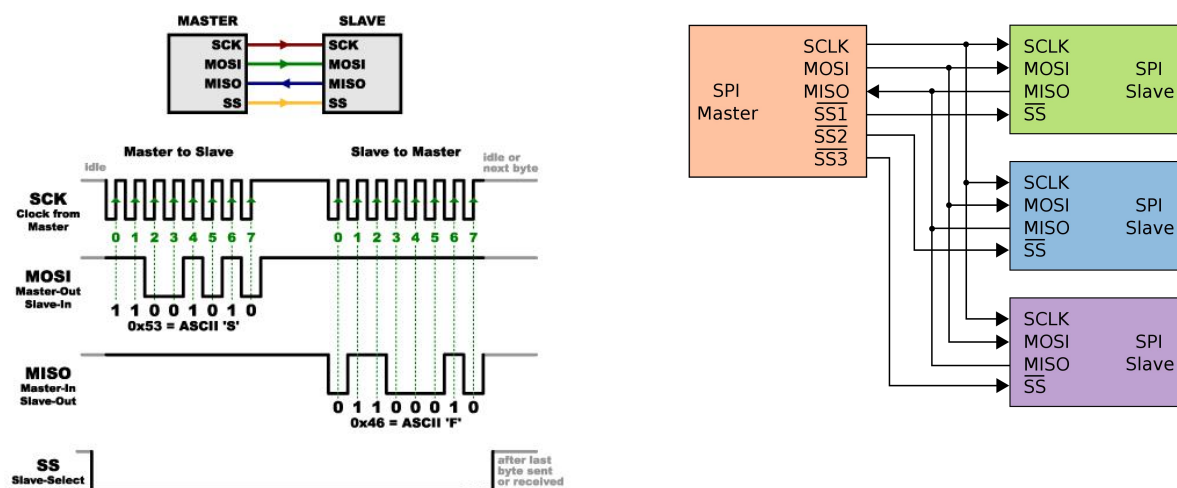
Upoznati se sa SPI sučeljem (engl. Serial Peripheral Interface) na mikroupravljaču STM32F407VG. Korištenje SPI sučelja za komunikaciju s akcelerometrom LIS3DSH. Razumijevanje mjernih signala dobivenih od akcelerometra.

II. Opis vježbe

Na laboratorijskim vježbama koristi se STM32F4 Discovery razvojna ploča kao hardverska komponenta dok se Atollic TrueStudio koristi kao pripadna programska podrška. U vježbi se demonstrira SPI komunikacija između STM32F407VG i akcelerometra LIS3DSH koji je ugrađen na razvojnu ploču. Mjerenja akcelerometra se prikazuju u stvarnom vremenu na računalu te se pomoću dobivenih mjerenja izračunava kut nagiba razvojne ploče s obzirom na horizontalnu ravninu.

II.1. SPI sučelje na mikroupravljaču STM32F407VG

SPI je serijsko komunikacijsko sučelje koje omogućava mikroupravljaču razmjenu podataka s različitim uređajima poput GPS-a, čitača SD kartica, zaslona i sl. Ovo je half/full duplex sinkrona serijska komunikacija što znači da postoje zasebne linije za takt i razmjenu podataka. Izvor takta (linija SCK) je *master* uređaj (najčešće mikroupravljač), a na sabirnicu može biti spojeno više *slave* uređaja. Kada se podaci šalju od *master* prema *slave* uređaju tada se podaci šalju preko linije MOSI (engl. *Master Out / Slave In*). Kada *slave* uređaj treba odgovoriti *master* uređaju, tada *master* uređaj i dalje generira potrebni takt, a *slave* uređaj šalje podatke putem linije MISO (engl. *Master In / Slave Out*). *Master* uređaj može odabrati *slave* uređaj s kojim će komunicirati pomoću linije SS (engl. *Slave Select*). Ako je ovaj signal u visokoj logičkoj razini, tada je *slave* uređaj odspojen od sabirnice. Neposredno prije slanja podataka *slave* uređaju, *master* će povući SS liniju u nisku logičku razinu. Primjer SPI komunikacije dan je na slici 5.1 a) dok je na slici b) prikazana najčešća topologija u slučaju više *slave* uređaja (topologija zvijezde).



Sl. 5.1 SPI komunikacija, a) vremenski dijagram, b) spajanje više *slave* uređaja.

Mikroupravljač STM32F407VG ima tri SPI sučelja: SPI1 na APB2 sabirnici i brzine do 42 Mbit/s te SPI2 i SPI23 na APB1 sabirnici i brzine do 21 Mbit/s. SPI sučelje na mikroupravljaču STM32F407VG je visoko konfigurabilno: simplex, half ili full duplex komunikacija, programirajiva veličina podatkovnog okvira, brzine prijenosa te redoslijeda bitova, automatski izračun CRC-a, DMA prijenos, različiti izvori zahtjeva za prekidom, itd. SPI sučelje spojeno je na odgovarajuće GPIO pinove mikroupravljača. Tako je npr. SPI1 spojen na pinove PA5 (SCK), PA6 (MISO), PA7 (MOSI). Kako bi se GPIO pinovi mogli koristiti za SPI komunikaciju, potrebno ih je podesiti tako da se koristi alternativna funkcija GPIO pina te ih povezati sa SPI periferijom. Stoga, postupak podešavanja SPI sučelja se sastoji od nekoliko koraka:

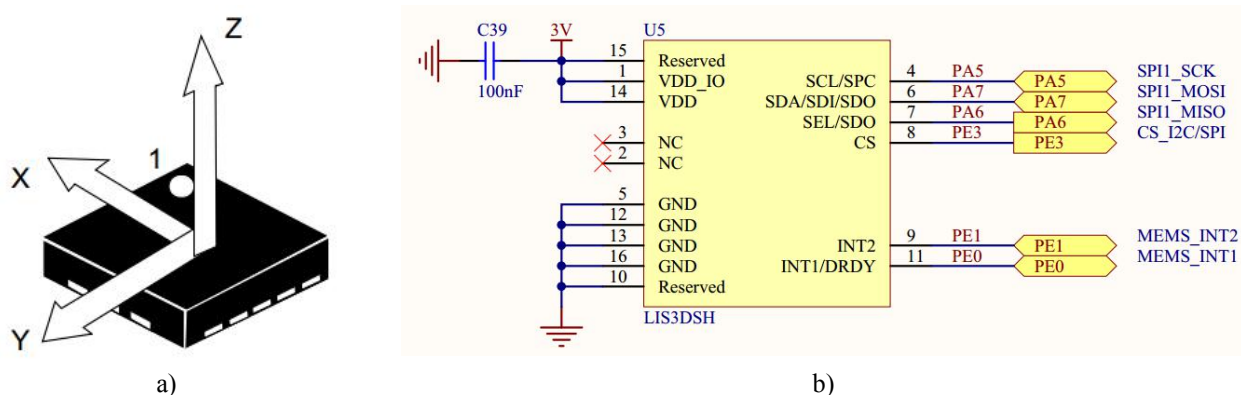
1. Omogućiti takt za SPI sučelje.
2. Konfigurirati GPIO pin tako da radi u alternativnom načinu te ga povezati s SPI periferijom.
3. Podesiti SPI sučelje (način rada, brzina prijenosa, veličina podatkovnog okvira i sl.).

4. Opcionalno omogućiti generiranje zahtjeva za prekidom i podesiti NVIC.
5. Omogućiti rad SPI sučelja.

Podešavanje i rad sa SPI sučeljem podržan je s odgovarajućim funkcijama u Standard Peripheral Library (SPL).

II.2. LIS3DSH

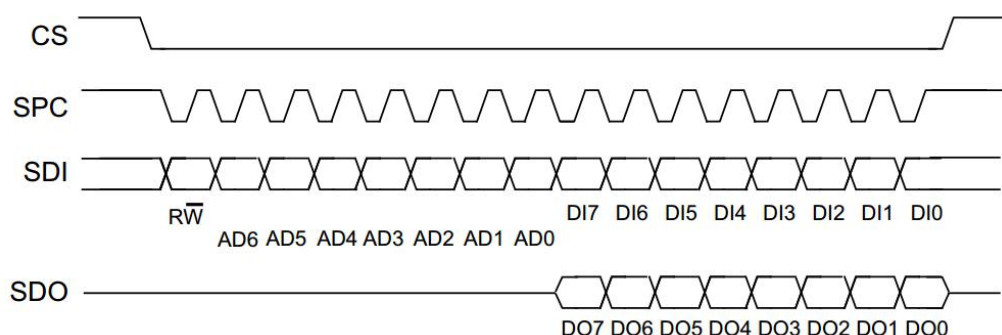
LIS3DSH je 3-osni linearni akcelerometar niske potrošnje i visokih performansi. Ovaj akcelerometar ima mogućnost dinamičkog odabira mjernog raspona ($\pm 2g/\pm 4g/\pm 6g/\pm 8g/\pm 16g$) te mogućnost mjerenja ubrzanja s frekvencijom u rasponu od 3.125 Hz do 1.6 kHz. LIS3DSH ima integrirani FIFO buffer koji omogućuje pohranjivanje mjerenja. Na slici 5.2 a) je prikazan LIS3DSH u LGA-16 kućištu zajedno s pripadajućim koordinatnim sustavom. LIS3DSH može mjeriti ubrzanje u sve tri osi te daje rezultat u obliku 16 bitnog broja za svaku os. Konfiguraciju i dohvaćanje mjernih vrijednosti moguće je obaviti putem SPI ili I2C sučelja. Na slici 5.2 b) prikazan je spoj LIS3DSH i mikroupravljača STM32F407VG na Discovery razvojnoj ploči. Za komunikaciju se koristi SPI1 pri čemu je LIS3DSH *slave*, a STM32F407VG *master* uređaj. MOSI linija se spaja na SDI (engl. *Serial Data In*) pin LIS3DSH, MISO linija se spaja na SDO (engl. *Serial Data Out*) pin LIS3DSH, SCK linija na SPC (engl. *Serial Port Clock*) pin LIS3DSH. Kao signal za odabir *slave* uređaja (SS) koristi se GPIO pin PE3 koji se spaja na CS (engl. *Chip Select*) pin LIS3DSH-



Sl. 5.2 Akcelerometar LIS3DSH a) referentni koordinatni sustav, b) spajanje na mikroupravljač STM32F407VG na Discovery razvojnoj ploči.

Rad s akcelerometrom LIS3DSH podrazumijeva čitanje vrijednosti iz odgovarajućih registara odnosno zapisivanje vrijednosti u odgovarajući registar. Registri LIS3DSH su veličine 8 ili 16 bita te svaki ima odgovarajuću adresu. Detaljne informacije za svaki registar moguće je pronaći u *datasheetu* LIS3DSH, kao primjerice adresu pojedinog registra, početnu vrijednost, mogućnost pisanja i čitanja, značenje pojedinog bita i sl. Jedan od važnih registara je CTRL_REG4 (adresa 0x20h) kojim se podešava frekvencija mjerenja te uopće omogućuje mjerenje ubrzanja u smjeru pojedine osi.

LIS3DSH je *slave* uređaj prilikom SPI komunikacije s mikroupravljačem STM32F407VG. Čitanje vrijednost registra ili zapisivanje vrijednosti u neki registar LIS3DSH odvija se unutar 16 pulseva signala takta SPC (ili višekratnika broja 8 ako se zapisuje ili čita više od jednog bajta). Komunikacija započinje slanjem prvog bajta preko SDI (MOSI) linije. Prvi bit ovog bajta definira radi li se o zapisivanju neke vrijednosti u registar LIS3DSH (vrijednost bita 0) ili se radi o čitanju vrijednosti nekog registra LIS3DSH (vrijednost bita 1). Preostalih 7 bitova predstavlja adresu registra koji se čita ili adresu registra u kojeg se zapisuje vrijednost. U slučaju operacije zapisivanje vrijednosti, STM32F407VG preko SDI (MOSI) linije šalje bajt koji predstavlja željenu vrijednost registra LIS3DSH. U slučaju operacije čitanja, LIS3DSH preko SDO (MISO) linije šalje trenutnu vrijednost registra od interesa. Ovakav princip komunikacije dan je na slici 5.3.



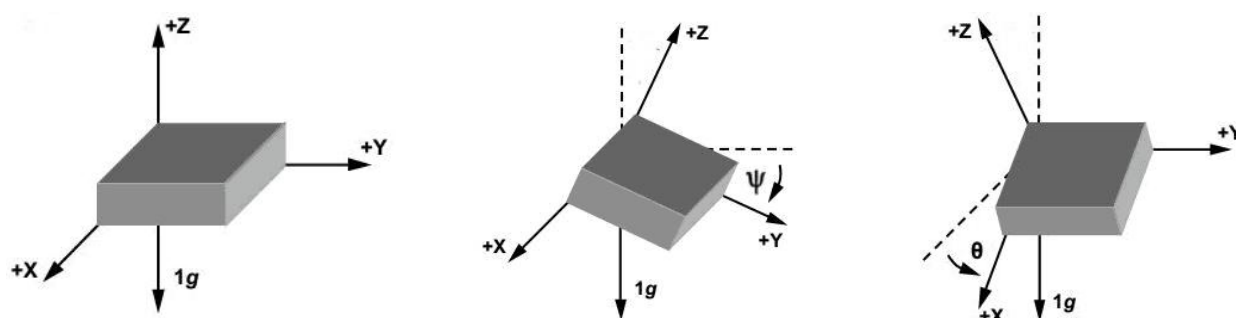
Sl. 5.3 SPI komunikacija s LIS3DSH.

II.3. Mjerenje nagiba

Jedna od vrlo čestih aplikacija akcelometra uključuje mjerenje nagiba (rotacija ekrana na pametnim telefonima i tabletima, sustav za zaštitu automobila i sl.). Ovo podrazumijeva izračunavanje položaja akcelometra s obzirom na gravitaciju os Zemlje odnosno horizontalnu ravninu. Rotacija akcelometra s obzirom na horizontalnu ravninu opisuje se pomoću *roll* i *pitch* kuteva kao što je prikazano na slici 5.4. Kada je Z os akcelometra poravnata s gravitacijskom osi (npr. akcelometar mirno leži na ravnoj podlozi), tada će očitavanja Z komponente akcelometra biti približno jednaka ubrzanju sile teže ($1g$) dok su X i Y komponente akcelometra približno jednake 0 (pod pretpostavkom da ne postoji *offset* mjerenja). Međutim, kada se akcelometar zarotira oko svoje X ili Y koordinatne osi, tada se vektor ubrzanja sile “osjeti” i u X odnosno Y smjeru. Iz izmjerenih vrijednosti ubrzanja u smjeru a_x , a_y i a_z moguće je dobiti kuteve nagiba u radijanima pomoću sljedećih izraza:

$$\text{Roll} = \text{atan2}(a_y, a_z)$$

$$\text{Pitch} = \text{atan2}(-a_x, \sqrt{(a_y)^2 + (a_z)^2})$$

Sl. 5.4 Akcelometar LIS3DSH a) poravnat s horizontalnom ravninom, b) *roll*, c) *pitch*.

III. Priprema za vježbu

1. Nabrojite tri različite primjene akcelometra.
2. Otvorite LIS3DSH DocID022405 Rev 3 *datasheet*. Možete li objasniti koja je razlika između *bypass* načina rada i *stream* načina rada akcelometra LIS3DSH?
3. U kojem registru se trebaju izmijeniti vrijednosti ako se želi promijeniti mjerni opseg akcelometra LIS3DSH?
4. Želite promijeniti samo vrijednost bita na određenom mjestu nekog registra akcelometra LIS3DSH. Kako ćete to učiniti (pretpostavite da su vam na raspolaganju funkcije za pisanje u neki registar odnosno za čitanje nekog registra LIS3DSH)?
5. Zadnja izmjerena vrijednost ubrzanja u smjeru x-osi spremljena je u dva 8 bitna registra LIS3DSH: OUT_X_L i OUT_X_H. Ove dvije vrijednosti predstavljaju “donji” i “gornji” bajt 16 bitne vrijednosti ubrzanja u smjeru X-osi. Na koji način dobiti 16 bitnu vrijednost ubrzanja u smjeru X-osi ako je na raspolaganju funkcija za čitanje nekog registra LIS3DSH?

IV. Rad na vježbi

1. Preuzmite i izvršite import projekta “LV5_accelerometer_template” koji se nalazi na adresi https://gitlab.com/rgrbic/UGRS_LV. Ovaj projekt sadrži funkcije za rad s akcelometrom LIS3DSH, te glavni program u kojem se periodički čitaju vrijednosti ubrzanja sve tri osi akcelometra te se ispisuju na serijski port. Proučite izvorni kod ovog projekta. Razumijete li sve dijelove koda? Razumijete li kod funkcije za inicijalizaciju LIS3DSH, te kod funkcija za razmjenu podataka s LIS3DSH preko SPI sučelja?

2. Spojite STM32F4 Discovery razvojnu ploču s osobnim računalom. Uključite Arduino Serial Plotter. Možete li objasniti mjerenja koja dobivate na ekranu? Pomičite STM32F4 Discovery razvojnu ploču. Koje su minimalne, a koje maksimalne vrijednosti koje mjerite? Dopunite kod iz zadatka 1 tako da se prikazuju mjerenja sve tri osi.
3. Dopunite/izmijenite programski kod tako da povećate mjerni opseg (npr. na $\pm 4g$). Što se događa s mjerenjima koja prikazujete u Arduino Serial Plotteru? Na kraju vratite mjerni opseg na $\pm 2g$.
4. Dopunite/izmijenite programski kod tako da se uključivanjem/isključivanjem pojedinih LED signalizira u kojem je smjeru nagnuta razvojna ploča.
5. Dopunite/izmijenite programski kod tako da na serijski port ispisujete vrijednosti ubrzanja u m/s^2 .
6. Dopunite/izmijenite programski kod tako da na serijski port ispisujete vrijednosti *pitch* i *roll* kuteva u stupnjevima. Dobivate li točne vrijednosti? Dodajte odgovarajući *offset* za svaku os kako biste dobili točniji rezultat.
7. Mjerenja dobivena s akcelerometra mogu sadržavati veliku količinu šuma. Implementirajte filter s pomičnim usrednjavanjem te prikažite mjerenja u Serial Plotteru s i bez primijenjenog filtra. Kako veličina prozora utječe na filtriranje signala? Za pohranjivanje mjerenja koristite kružni spremnik (engl. *circular buffer*).

V. Za one koji žele znati više

Akcelerometar LIS3DSH ima opciju detektiranja slobodnog pada (engl. *freefall*) koja se može primjerice koristiti za zaštitu hard diska u ugrađenom računalnom sustavu. Podesite akcelerometar LIS3DSH tako da preko odgovarajućeg pina signalizira mikroupravljaču slobodni pad Discovery razvojne pločice. Prilikom slobodnog pada uključite sve četiri LED na Discovery razvojnoj ploči.