

Sveučilište Josipa Jurja Strossmayera u Osijeku
Fakultet elektrotehnike, računarstva i informacijskih
tehnologija

Kolegij: Linux u ugradbenim sustavima

Laboratorijska vježba 2:
Nunchuk - I²C uređaj

1 Uvod

Cilj ove vježbe je prepoznati I²C uređaj u operacijskom sustavu *Linux* te izraditi osnovni *kernel* modul koji će se nadograđivati u narednim laboratorijskim vježbama. Potrebno je izraditi stablo uređaja (engl. *device tree*). Također, potrebno je izraditi kernel modul s osnovnim inačicama `probe()` i `remove()` funkcija.

2 Povezivanje *Nunchuk*-a

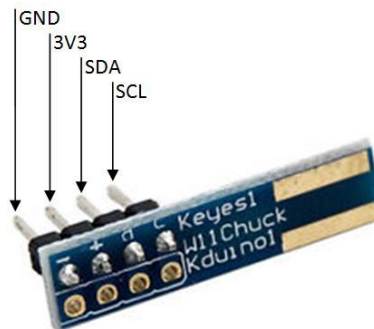
Uzmite *Nunchuk* uređaj kojeg ste dobili od asistenta.

Pomoću konektora ćemo spojiti *Nunchuk* uređaj na J8 pinove *Raspberry Pi* ploče.

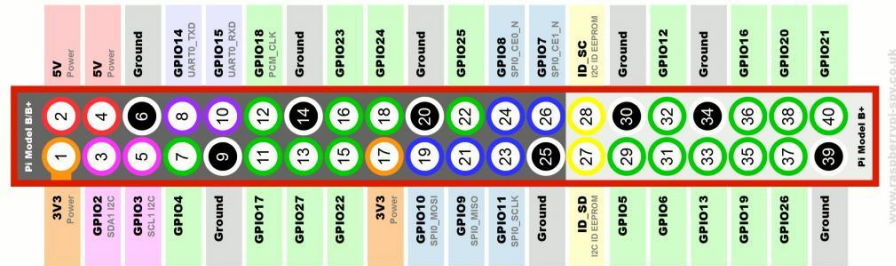
Dokumente s korisnim informacijama o *Nunchuk* uređaju možete preuzeti s adresa:

<https://bootlin.com/labs/doc/nunchuk.pdf>

<https://www.robotshop.com/media/files/PDF/inex-zx-nunchuck-datasheet.pdf>



Slika 1: Raspored pinova Nunchuk konektora



Slika 2: Raspored pinova J8 konektora na Raspberry Pi 3 ploči

Sada povežite *Nunchuk* konektor i Raspberry Pi 3 ploču na sljedeći način:

- GND pin na J8 pin 9 (GND),
- 3V3 pin na J8 pin 1 (3V3),
- SCL pin na J8 pin 5 (GPIO3/SCL1 I2C),
- SDA pin na J8 pin 3 (GPIO2/SDA1 I2C).

3 Izrada posebnog stabla uređaja (engl. *device tree*)

Da bismo *Linux* jezgri omogućili da rukuje novim uređajem, moramo dodati opis tog uređaja u stablo uređaja ploče (engl. *board device tree*). Kako je stablo uređaja za *Raspberry Pi 3* ploču već uključeno u *Linux* jezgru, mi nećemo raditi izmjene direktno u datoteci koja se već koristi za ovu ploču nego ćemo napraviti posebno stablo uređaja za našu ploču s našim uređajem.

Pozicionirajte se u Linux izvorni kod koji bi trebao biti na putanji `/home/rtrk/buildroot/output/build/linux-custom`. Tu se nalazi Linux izvorni kod pomoću kojega smo stvorili sliku sustava (engl. *image*) za našu *Raspberry Pi* ploču koristeći Buildroot u laboratorijskoj vježbi 2. Napravite kopiju datoteke `bcm2710-rpi-3-b.dts` koja se nalazi u poddirektoriju `arch/arm/boot/dts` i nazovite ju `bcm2710-rpi-3-b-custom.dts`. Napravite u istom direktoriju i izmjene u `Makefile` datoteci i omogućite da se i nova `dts` datoteka kompilira.

Unutar nove datoteke, prvo trebamo omogućiti `i2c1` sabirnicu. Zatim,

deklarirajte **Nunchuk** uređaj kao čvor nasljednik **i2c1**. Odaberite **nintendo,nunchuk** za **compatible** postavku. I²C adresu **Nunchuk** uređaja možete provjeriti u **Nunchuk** dokumentu.

Nakon potrebnih izmjena **i2c1** čvor bi trebao izgledati otprilike ovako:

```
&i2c1 {
    pinctrl-names = "default ";
    pinctrl-0 = <&i2c1_pins >;
    clock-frequency = <100000>;
    status="okay ";

    nunchuk: nunchuk@52 {
        compatible ="nintendo ,nunchuk ";
        reg = <0x52>;
        status = "okay ";
    };
};
```

Nakon potrebnih promjena u datoteci stabla uređaja, pozicionirajte se u korijensku putanju Linux izvornog koda (**/home/rtrk/buildroot/output/build/linux-custom**).

Postavite varijable okruženja potrebne za kompiliranje Linux jezgre:

```
export ARCH=arm
export CROSS_COMPILE=arm-linux-gnueabi-
```

Nakon toga, pokrenite kompiliranje datoteka stabla uređaja koristeći naredbu:

```
make dtbs
```

Kopirajte novu **dtb** datoteku u direktorij **tftp server-a** (**/var/lib/tftpboot**) i napravite potrebne izmjene u **Uboot** konfiguraciji da bi *Raspberry Pi* ploča prilikom pokretanja učitala novu datoteku.

Kroz direktorij **/proc/device-tree** možemo provjeriti sve postavke stabla uređaja koje je naš sustav učitao.

Na primjer, možemo provjeriti prisustvo **nunchuk** čvora u stablu uređaja:

```
find /proc/device-tree/ -name *nunchuk*
/proc/device-tree/soc/i2c@7e804000/nunchuk@52
```

Neke od postavki uređaja možemo pročitati pomoću:

```
cat /proc/device-tree/soc/i2c@7e804000/nunchuk@52/status
```

4 Izrada osnovnog upravljačkog programa za Nunchuk

Sada možete krenuti s izradom upravljačkog programa za *Nunchuk*. Unutar direktorija `/home/rtrk/embedded_linux/LV3/solutions` stvorite direktorij imena `nunchuk`. Unutar tog direktorija stvorite dvije datoteke: `nunchuk.c` i `Makefile`.

Struktura `Makefile` datoteke je standardna. Bitan parametar je `KDIR` koji označava putanju do izvornog koda Linux jezgre.

`Makefile` bi trebao izgledati ovako:

```
ifneq ($(KERNELRELEASE),)
obj-m := nunchuk.o
else
KDIR := /opt/faks/buildroot/output/build/linux-custom
all:
$(MAKE) -C $(KDIR) M=$$PWD
endif
```