

Projektovanje GUI upotrebom Qt radnog okvira

11.1. Klase za rad sa vremenom

U ovom poglavlju će biti predstavljene klase `QTime` i `QTimer` koje se koriste u radu u realnom vremenu.

QTime

`QTime` klasa sadrži funkcije za rad sa satom realnog vremena i koristi se za lakše rukovanje objektima `QTimer`, `QTimeEdit` i sl. `QTime` objekat sadrži polja za broj sati, minuta, sekundi i milisekundi počevši od ponoći (00:00:00). Može da pročita sistemsko vreme i meri protekli period vremena. Sem toga, može da upoređuje vreme i inkrementira trenutno vreme za određeni broj sekundi ili milisekundi. `QTime` objekat koristi 24-časovni režim tako da nema ugrađene koncepte: AM (prepodne) i PM (poslepodne).

`QTime` objekat se tipično pravi tako što se konstruktoru direktno prosledi broj sati, minuta, sekundi i milisekundi, ili se iskoristi statička funkcija `currentTime()` koja vraća `QTime` objekat sa trenutnim lokalnim sistemskim vremenom.

```
int sati = 1, minuti = 2, sekunde =3, msekunde = 4;  
QTime vreme (sati, minuti, sekunde, msekunde);
```

```
QTime vreme (QTime::currentTime());
```

Metoda `setHMS()` postavlja vrednosti objekta na one prosleđene preko parametara.

```
QTime vreme.setHMS(sati, minuti, sekunde);
```

Iz QTime objekta se pomoću metoda **hour()**, **minute()**, **second()** i **msec()** mogu izdvojiti vrednosti sati, minuta, sekundi i milisekundi u celobrojnom formatu.

```
int sati = vreme.hour();
int minuti = vreme.minute();
int sekunde = vreme.second();
int msekunde = vreme.msec();
```

Iste vrednosti se mogu vratiti kao string koristeći metodu **toString()**, gde se kao parametar prosleđuje željeni format. Obratiti pažnju na tabelu sa znakovima za formatiranje.

```
QString format = "hh:mm:ss";
QString str = vreme.toString(format);
```

| format | str |
|---------------|--------------|
| hh:mm:ss.zzz | 14:13:09.042 |
| h:m:s a | 2:13:9 pm |
| H:m:s a | 14:13:9 pm |

| | |
|----------|---|
| h | sati bez vodeće nule (0 do 23 ili 1 do 12 ako je uključen AM/PM prikaz) |
| hh | sati sa vodećom nulom (00 to 23 or 01 to 12 ako je uključen AM/PM prikaz) |
| m | minuti bez vodeće nule (0 do 59) |
| mm | minuti sa vodećom nulom (00 do 59) |
| s | sekunde bez vodeće nule (0 do 59) |
| ss | sekunde sa vodećom nulom (0 do 59) |
| z | milisekunde bez vodeće nule (0 to 999) |
| zzz | milisekunde sa vodećom nulom (000 to 999) |
| AP ili A | uključuje AM/PM prikaz, AP će biti "AM" ili "PM". |
| ap ili a | uključuje AM/PM prikaz, AP će biti "am" ili "pm". |

Funkcija *fromString()* radi suprotnu operaciju, uzima kao parametare string i format a kao povratnu vrednost vraća objekat *QTime*.

```
QTime vreme = QTime::fromString("1mm12car00",  
"m'mm'hcarss"); // vreme je 12:01.00
```

Metode **addSecs()** i **addMSecs()** dodaju odgovarajući broj sekundi ili milisekundi na trenutnu vrednost *QTime* objekta. Ukoliko se prosledi broj sa negativanim predznakom, vreme će se umanjiti.

```
QTime n(14, 0, 0); // n == 14:00:00  
QTime t;  
  
t = n.addSecs(70); // t == 14:01:10  
t = n.addSecs(-70); // t == 13:58:50  
t = n.addSecs(10 * 60 * 60 + 5); // t == 00:00:05  
t = n.addSecs(-15 * 60 * 60); // t == 23:00:00
```

Za poređenje objekata *QTime*, preklapljeni su sledeći operatori:

==, !=, <, >, <= i >=.

Manji objekat je onaj koji sadrži ranije vreme.

Metoda **isValid()** vraća true ako je vreme validno, u suprotnom vraća false. Na primer, vreme 23:30:55.746 is je validno, ali 55:12:30 nije.

QTimer

Podešavanje tajmera u Qt je veoma jednostavno. Potrebno je napraviti objekat klase *QTimer*, povezati njegov signal *timeout()* sa odgovarajućim slotom, i startovati ga metodom *start(int brojMilisekundi)* sa parametrom koji određuje periodu javljanja signala *timeout()* u milisekundama. Konkretno u Primeru 2 u nastavku, su korišćena dva tajmera, čije je ponašanje definisano u konstruktoru glavnog prozora:

```
// kreiranje objekta klase QTimer, parent je glavni  
QTimer *myTimer1 = new QTimer(this);  
  
// prozor ui (pokazivač na njega je this)  
QTimer *myTimer2 = new QTimer(this);  
  
// spajanje sa slotovima prethodno definisanim  
// kao public slots u dialog.h
```

```

connect(myTimer1, SIGNAL(timeout()), this,
        SLOT(rtcReadTimeSlot()));
connect(myTimer2, SIGNAL(timeout()), this, SLOT(lcdSlot()));

myTimer1->start(1000); //startovanje tajmera, perioda 1 sekund
myTimer2->start(500); //startovanje tajmera, perioda 0.5 sekundi

```

11.2. QMessageBox

Message box-ovi su vrlo često korišćeni elementi grafičkog interfejsa. Qt klasa `QMessageBox` omogućava jednostavno instanciranje osnovnih tipova ovih *box*-ova, kao što su: informacija (eng. *information*), pitanje (eng. *question*), upozorenje (eng. *warning*), itd. Za pitanje se moraju definisati i tasteri. Neki primeri upotrebe su:

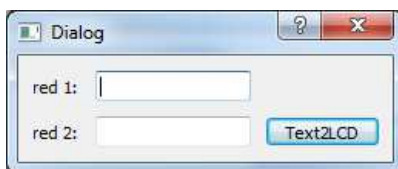
```

QMessageBox::question(this, "Test", "Quit?",
                     QMessageBox::Yes|QMessageBox::No);
QMessageBox::information(this, "Application Name",
                        "An information message." );
QMessageBox::warning(this, "Upozorenje!",
                    "Pogresan format!\nMora da bude hh:mm:ss");

```

Zadatak 1

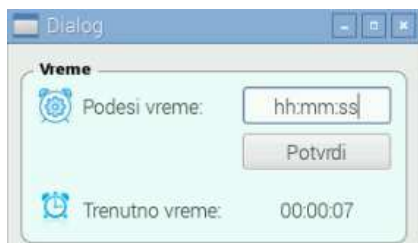
Potrebno je projektovati GUI koji omogućava ispis unetog teksta na LCD-u nakon pritisnutog tastera, u skladu sa slikom 11.1. Koristiti `QLineEdit` i `QPushButton` i ograničiti veličinu unetog teksta na 16 karaktera. Za prihvatanje teksta iz edit polja koristiti `QString`.



Slika 11.1. Izgled GUI u zadatku 1

Zadatak 2

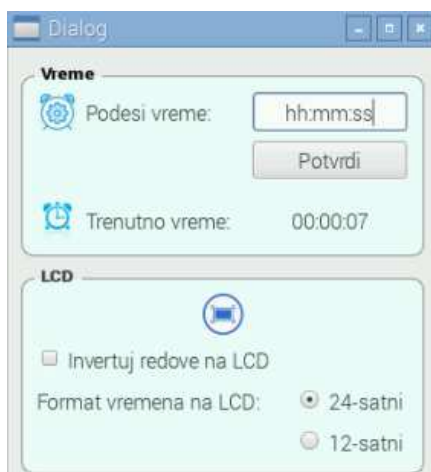
Potrebno je projektovati GUI koji omogućava podešavanje trenutnog vremena RTCC-a i ispis u skladu sa GUI prikazanim na slici 11.2. Koristiti potrebne funkcije iz vežbe 8. U slučaju unosa vremena u pogrešnom formatu obavestiti korisnika o tome odgovarajućim *MessageBox*-om.



Slika 11.2. Izgled GUI u zadatku 2

Zadatak 3

Potrebno je projektovati GUI koji predstavlja proširenje prethodnog zadatka i omogućava ispis vremena na LCD-u uz odabir određenih formata za to u skladu sa slikom 11.3.



Slika 11.3. Izgled GUI zadatka 3

Zadatak 4

Prethodnom zadatku potrebno je dodati funkciju alarma. Korisnik treba da ima mogućnost unosa vremena za alarm u formatu hh:mm:ss, a kada se trenutno vreme poklopi sa unešenim, treba uključiti LED vezanu na RPi koja emulira alarm.