

Ülesanne 1

Iga grupp saab ühe 8 muutujaga digitaal-loogilise funktsiooni ja peab selle realiseerima mikrokontrolleril. Väärtused saadakse lülititelt, mis tuleb lugeda massiivi. Massiiv tuleb anda void tüüpi funktsioonile, milles lahendatakse digitaal-loogika ja kuvatakse LCD-ekraanile kas "True" või "False".

Alustuseks tuleb alla tõmmata projekt:

<http://edu.pld.ttu.ee/~trkal/mikrokontrollerid/Exercise3.zip>

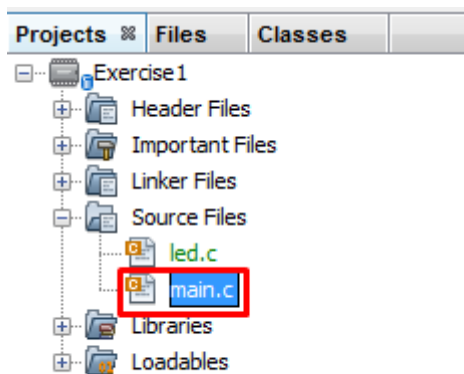
Teiseks, eralda(*extract*) allatõmmatud arhiiv *Desktop*'le.

Kolmandaks, ava terminaalet MPLAB, kasutades käsku *cad* ja *mplab*.

Mplab's projekti avamiseks kasutage *Open Project*:

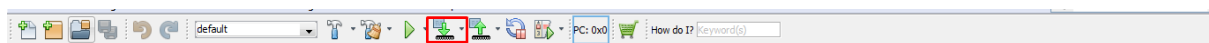


Otsige paremalt aknast *Source Files* alt *main.c* ja tehke topeltklõps selle peal:



Kogu kood kirjutage *main* funktsiooni, lõpmatusse tsüklisse. Ärge unustage komponente initsialiseerida.

Väga tähtis, et kood tööle saada on jätta meelde, et muuta tohib AINULT *main.c* faili ja ainult *main* funktsiooni. Vajadusel võib juurde teha enda funktsioone või lisada vajalikke teeke.



Koodi laadimiseks plaadile kasutage nuppu *Make and Program Device Main Project*.

Funktsioonide näiteid:

Et initsialiseerida 8't lülitit, tuleb kasutada `SWT_Init()` ;

Teistel komponentidel on teised nimed funktsioonile, kõik lühendid leiab *Header files* alt.

Et lugeda lüliteid, saab kasutada kas `SWT_GetValue(int x)`, kus x näitab soovitava lüliti numbrit, või `SWT_GetGroupValue()`, see funktsioon annab kõigi lülite väärtused.

LED'ide väärtustamiseks saab kasutada `LED_SetValue(int x, int y)`, kus x näitab soovitava *LED*'i väärtust ja y annab oleku(1 või 0 ehk põleb või ei põle). Et väärtustada kõik *LED*'d korraga, näiteks lülite väärtuste järgi, siis saab kasutada `LED_SetGroupValue(int x)`, kus siis x näitab ära kõikide *LED*'de oleku.

Nupu oleku saab kätte funktsiooniga `BTN_GetValue(char c)`, kus c on soovitud nupu täht (näiteks keskmine nupp(*center*) puhul on c väärtus `'c'`, vasaku(*left*) nupu puhul aga `'l'`. Funktsioon tagastab väärtuse 1, kui nupp on alla vajutatud ja väärtuse 0, kui nupp on üleval.

LCD-ekraanile kirjutamiseks on funktsioon `LCD_WriteStringAtPos(char* str, int x, int y)`, mis võtab 3 parameetrit. Esimeseks on string, mis kirjutatakse ekraanile(saab olla ainult *string*'i või *char* kujul), teiseks on arv, mis saab olla 0 või 1. 0 kirjutab ülemisele reale ja 1 alumisele ja kolmandaks on arv, mis näitab kui kaugel vasakust äärest hakatakse ekraanile kirjutama. Ekraanile mahub ühte ritta 16 sümbolit.

Rohkem infot erinevate funktsioonide kohta saab projektis sisalduvatest teekidest, aga mitte ühelgi juhul ei tohi neid muuta.

Digitaal-loogika funktsioonid

| Ülesanne 1: | Testitavad kombinatsioonid ja vastused | |
|---|--|----------|
| | 10011000 | 11001011 |
| 1. $(a \& b \& c) (d \& e \& f) !(g \& h)$ | 0 | 1 |
| 2. $!(!a \& !b \& !c) (d \& e \& f) !(g \& h)$ | 1 | 1 |
| 3. $(a b c) \& (d e f) \& !(g h)$ | 1 | 1 |
| 4. $!(!a \& !d \& !g) !b (c \& e) !(f \& h)$ | 1 | 1 |
| 5. $!(!a \& !(b c d e f g h)) \& !b$ | 0 | 0 |
| 6. $!(!a) b (d \& e) !(f \& g \& h) 0$ | 1 | 1 |
| 7. $!(1 \& a) b !c (d \& e \& !(f \& !g \& h))$ | 1 | 1 |
| 8. $!(!(!a \& !b) !(!c \& !d \& !e) !(f \& !g \& !h))$ | 1 | 1 |
| 9. $!a \wedge !b \wedge !c \wedge !d \wedge !e \wedge !f \wedge !g \wedge !h$ | 1 | 1 |
| 0. $(a \& b) \wedge (c \& d) \wedge (!e \& !f) \wedge !(g \& h)$ | 0 | 0 |

Ülesanne 2

Iga grupp saab 2 8 muutujaga digitaal-loogilist funktsiooni, millest esimesest saadakse konstante muutuja ja teist saab reaalajas muuta peale esimese väärtustamist. Kui esimene on väärtustatud kuvatakse LCD ekraanil "True" või "False" ning lülititega saab teisele funktsioonile väärtusi andes muuta lõpp väärtust.

Teine funktsioon: $!(!(a \& b) \wedge !(a \& !b)) \& (c | d) | !e | f | !g | h) \wedge !(y \& !h)$

Ülesande lahti seletus ja kitsendused

1. Lülitite pealt väärtustatakse esimene funktsioon

1.1. Nupu vajutusest arvutatakse esimese funktsiooni väärtus "y"

2. Reaalajas lülitite pealt väärtustada teine funktsioon ja kuvada selle vastus ekraanile

Peab olema realiseeritud vähemalt 2 abi funktsiooni

1. void tüüpi kuhu antakse massiiv ja see väärtustatakse

2. int tüüpi kuhu antakse massiiv, see väärtustatakse ja tagastatakse