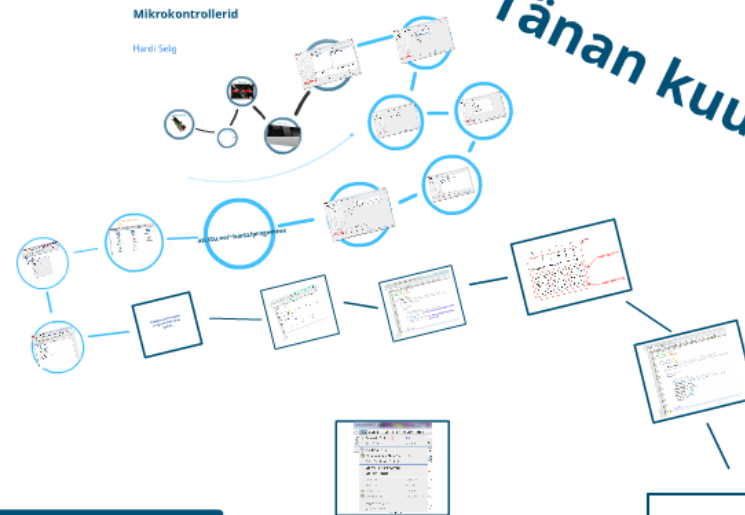


Täna kuulamast :)



Jooksvad tuled

Eeldus

Mõista programmeerimiskeeles C-minikontrolleri konstruktsiooni (struktuuri, tingimusi). Osata kasutada viiteid.

Ülesanne

Programmeeri programmi, mis paneb ühel ajahetkel põlema ühe LEDi. Järgmisel hetkel teine kõrvaleoleva LEDi jne. Tulemuseks peab tekkinud efekt nagu LEDid liiguvad vasakult paremale ning tagasi. LEDide hääletamise vaheline viide vali ise.

Lahenduse kirjeldus

<http://preziki.sajinas.ee/wp-content/uploads/2012/11/Running-leds.pdf>

Soovitus

Proovi välja mõelda natuke lahendusi!

Näidisülesande modifikatsioon

Eeldus

Saada aru tarkvara peatükis kirjelatud programmi näidiskoodist. Mõista sisendite ning väljundite kasutamist ning väljundite realiseerivate funktsioonide töö põhimõtteid.

Ülesanne

Muuta näidisprogrammi koodi nii, et 7-segmendi indikaatori esimene segment (segm_char0) näitaks DIP lüliti summat. Teine segment peab olema olekus NULL ehk ei põle. Ning kolmas segment peab näitama mitmendat korda summeerimist teostatuse (punkt kaasa arvatud). LEDidele väljastada DIP lüliti summa jagamisel tulev jääk. Summa ning jäägi väärtus väljastatakse alles pärast surunupu BTN vajutamist.

Lahenduse kirjeldus

DIP lüliti seatakse soovitud olekusse. Seadit väljastatakse BTN surunupu. Selle tagajärjel väljastatakse 7-segmendile DIP lüliti summa ning summeerimise kordade arv ja LEDidele summa jääk jagamisel 4-ga. Järgmine summa saamiseks korratakse tegevust.

Märkus

DIP lüliti suvalised olekud olekus ON & OFF. ON tähendab, et lüliti väärtus on 1 ning OFF, et lüliti väärtus on 0. Järelikult vastab iga DIP lüliti väärtus olekus ON väärtusele 1.

Munakell

Ülesanne

Programmeeri programmi, mis loendab vastavalt etteantud kahendväärtusele väärtusi alla, kuni jõuab nulli. Hetkelist väärtust näidatakse 7-segmendi indikaatoritel. Loenduri algväärtus antakse ette DIP lülitelt. Kui kell jõuab nulli hakkavad vilkuma LED tuled. Loendur hakkab loendama alles pärast surunupu BTN vajutamist.

Lahenduse kirjeldus

Kasutaja annab DIP lülitele näiteks väärtuse 00001111 ning seejärel vajutab surunupu BTN. 7-segmendi indikaatorile ilmub väärtus 15 ning seejärel hakatakse antud väärtust 1 võrra vähendada, kuni tulemuseks on null. Järgnevalt hakkavad vilkuma LED tuled. Korduste arv pole määratud. Kasutaja poolt antav minimaalne väärtus on 0 ning maksimaalne 255 (11111111). Programm peab olema suuteline võtma vastu kõiki arve antud vahemikust ning väljastama neid 7-segmendi indikaatorile.

Märkus

Väärtuste väljastamisel 7-segmendi indikaatorile lahtuda mõistlikust loetavusest. Liiga kiiresti muutuvaid numbreid on raske lugeda!

LEDi juhtimine

Ülesanne

Koosta programm, mis väljastab igale LEDile 2-sisendiga loogikaventiili väärtuse lähtuvalt DIP lülite olekust. Loogikaventiilid mida simuleerida vali ise (AND, NAND, OR, NOR, XOR, XNOR). Simuleeri nelja loogikaventiili.

Lahenduse kirjeldus

DIP1 ning DIP2 on määratud simuleerima loogikafunktsiooni 2-AND. Järelikult lülitub LED DS1 sisse ainult juhul, kui mõlemad DIP lülid on olekus ON.

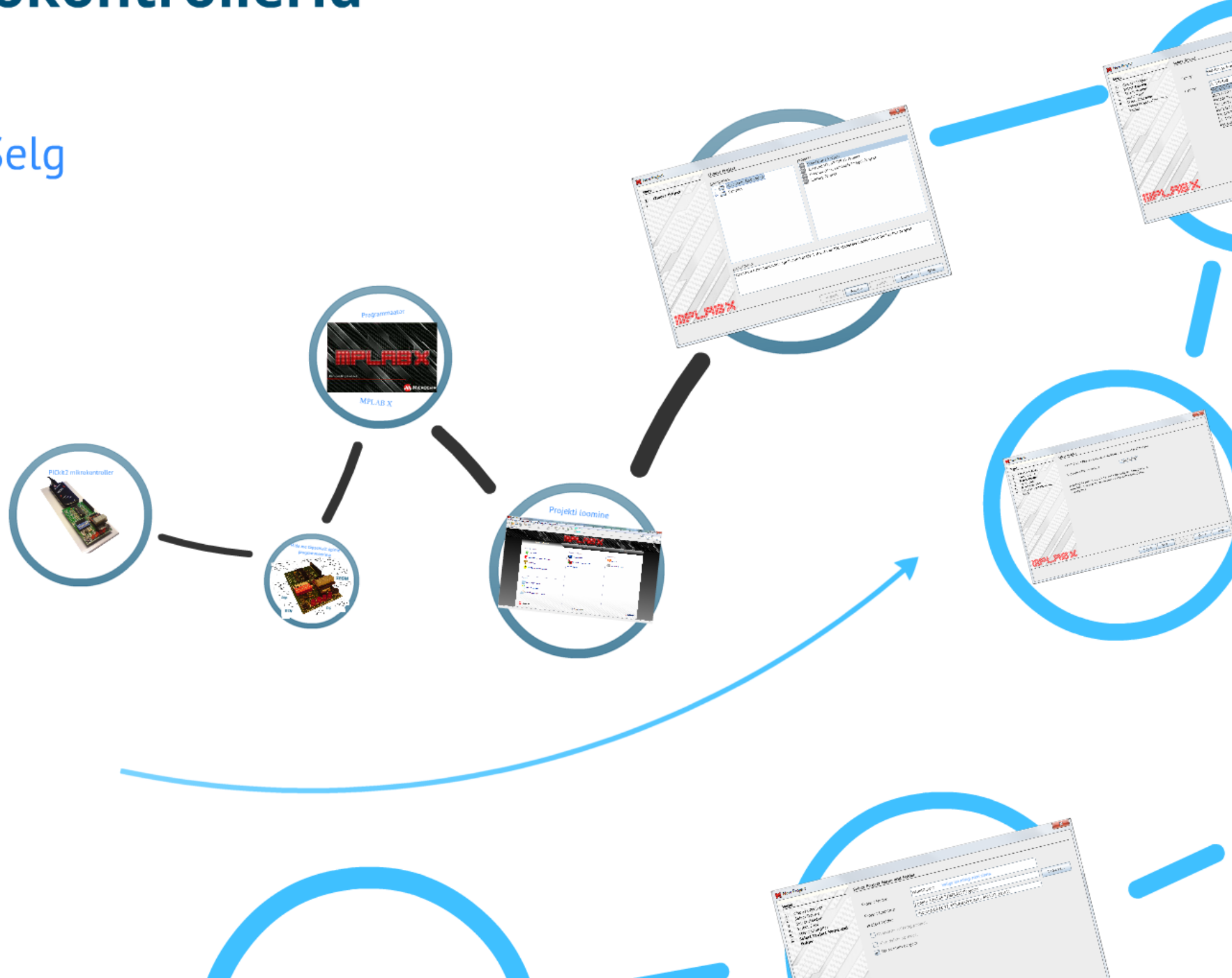
Lisamaterjal

2-sisendiga loogikaelementide tõeväärtustabel.

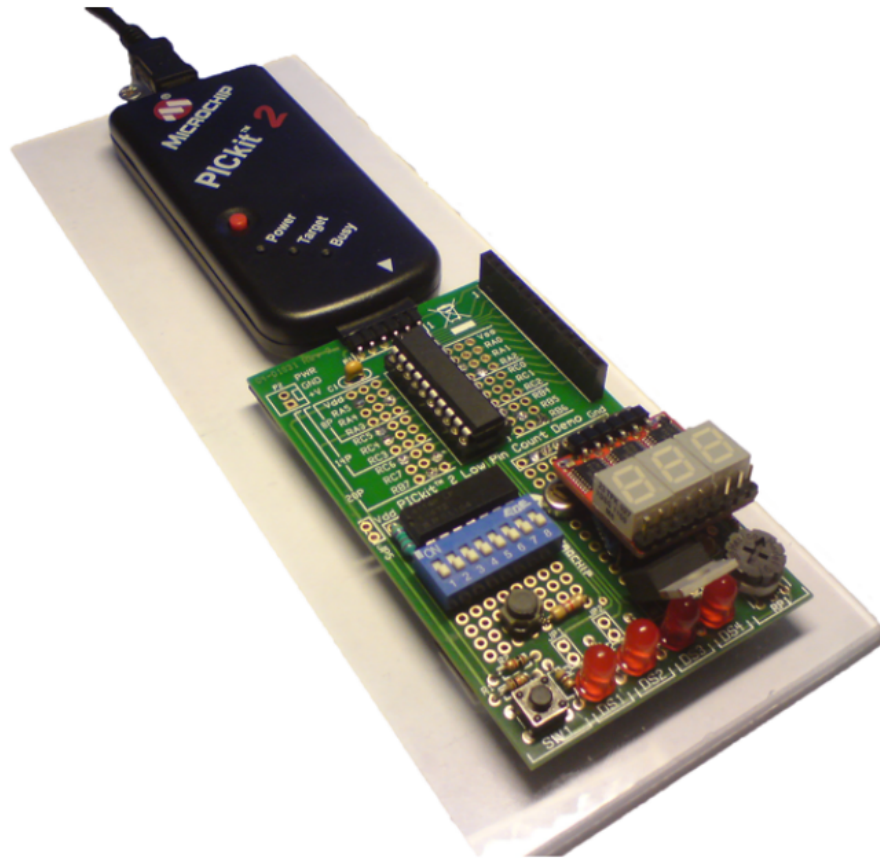
A	B	AND	NAND	OR	NOR	XOR	XNOR
0	0	0	1	0	1	0	1
0	1	0	1	1	0	1	0
1	0	0	1	1	0	1	0
1	1	1	0	1	0	0	1

Mikrokontrollerid

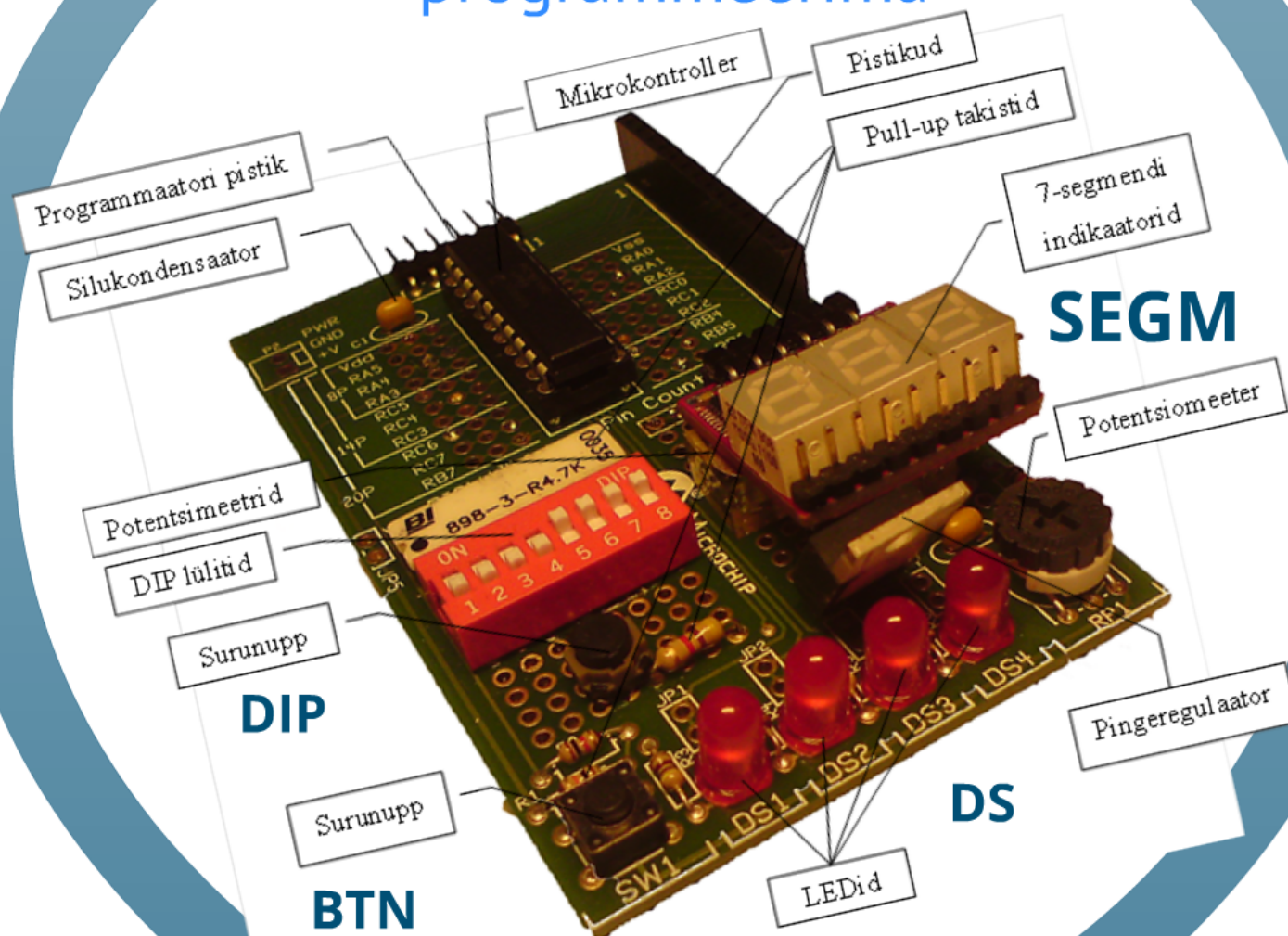
Hardi Selg



PICkit2 mikrokontroller



Mida me täpsemalt õpime programmeerima

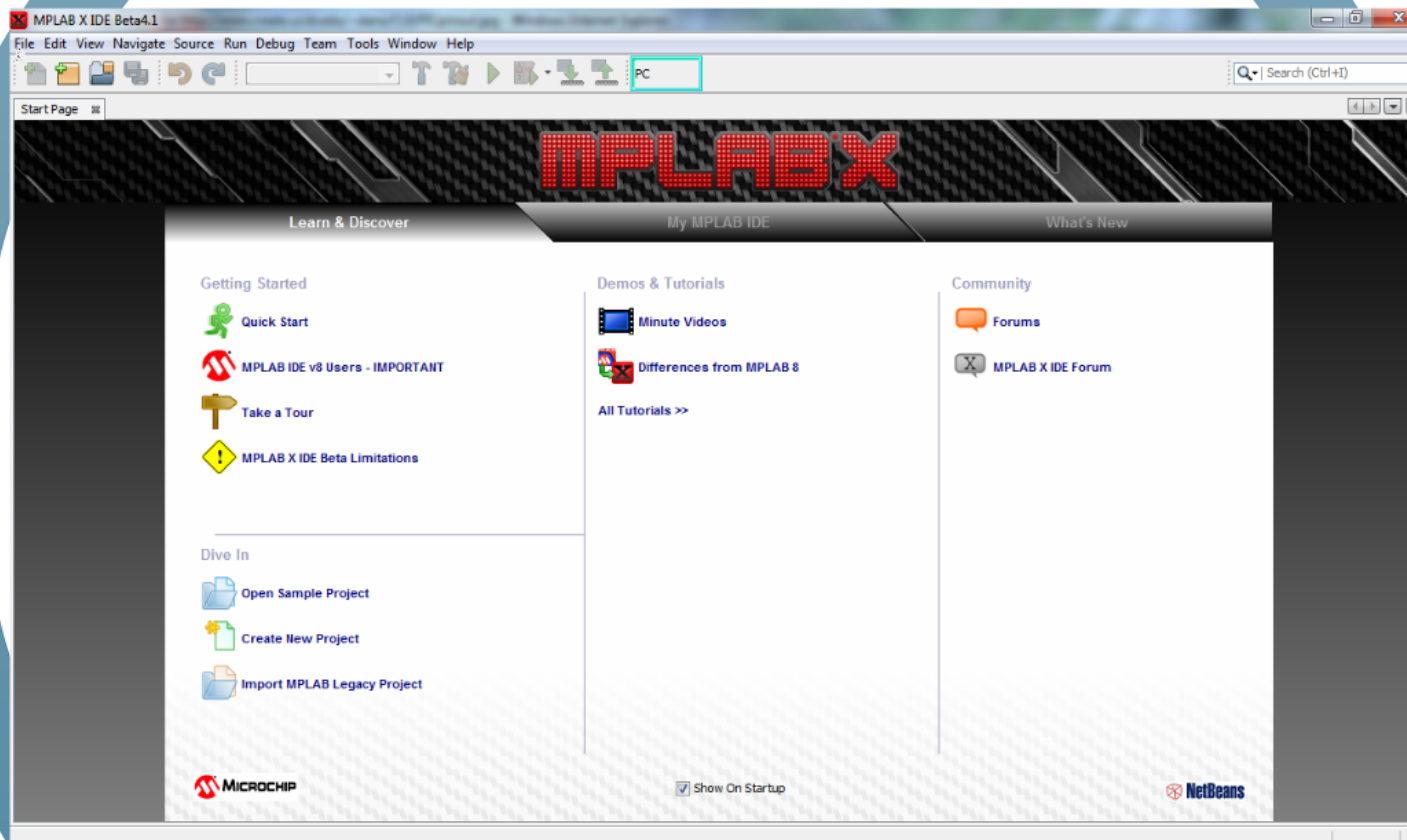


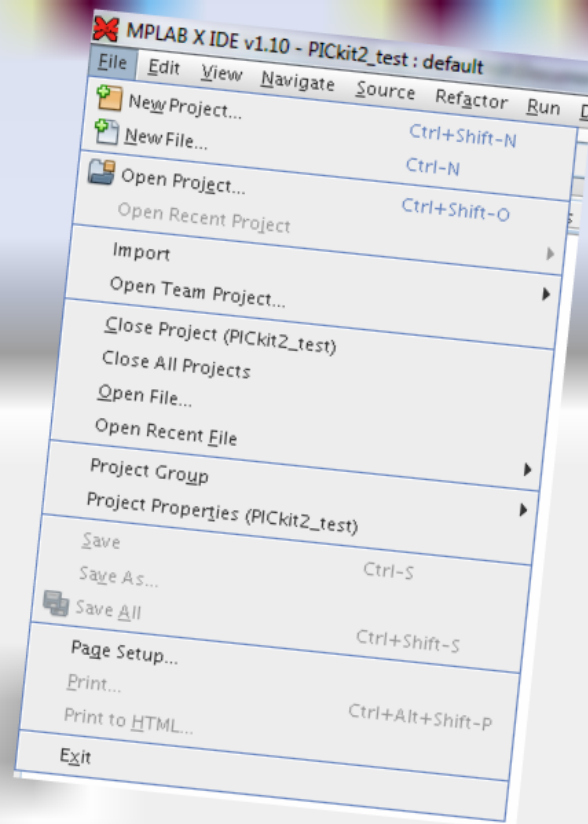
Programmaator

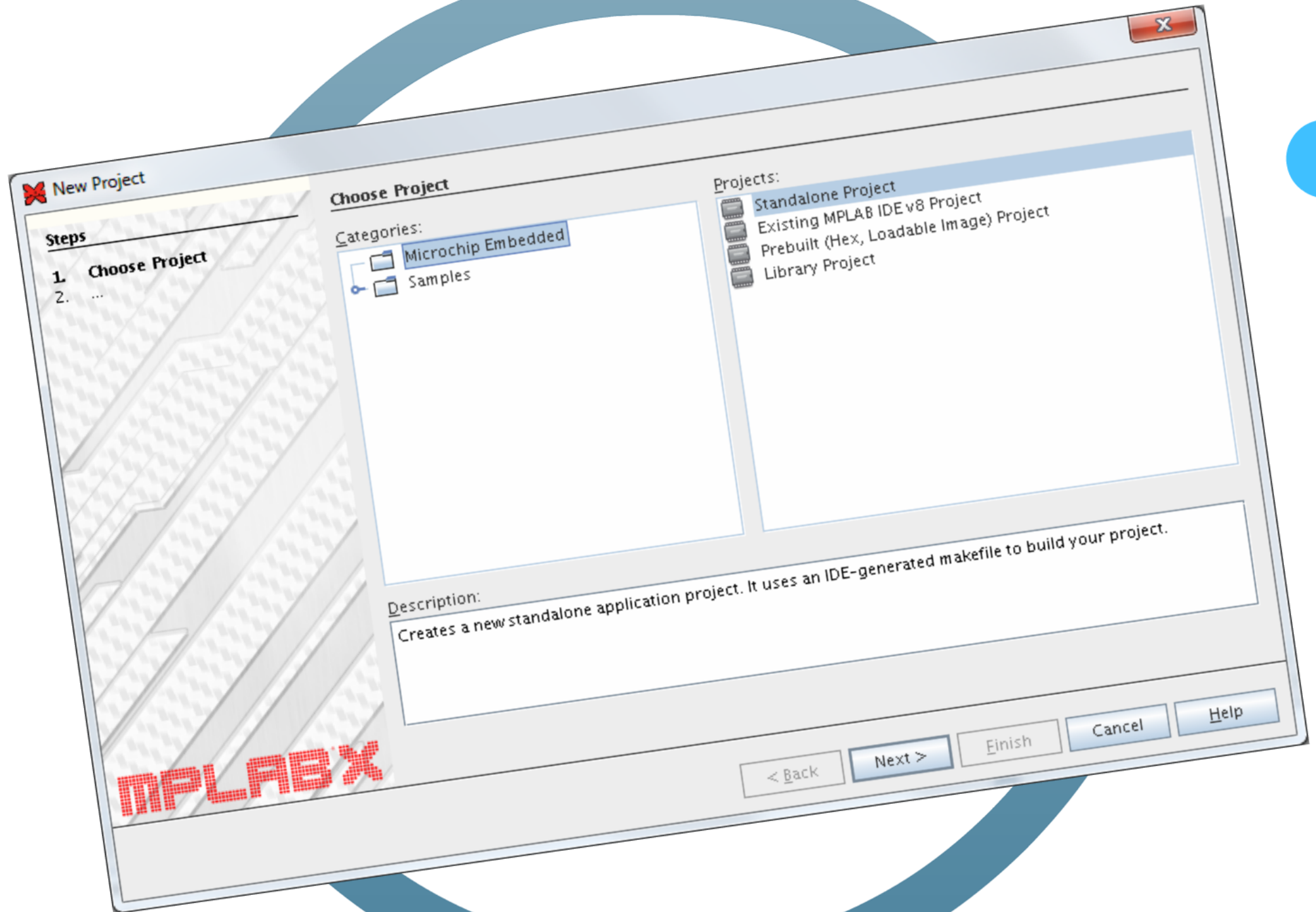


MPLAB X

Projekti loomine









Mid-Range 8-bit MCUs (PIC12/16/M ▼

PIC16F690 ▼

PIC16F690 ▲

Mid-Range 8-bit MCUs (PIC12/16/M

PIC16F690

PIC16F690

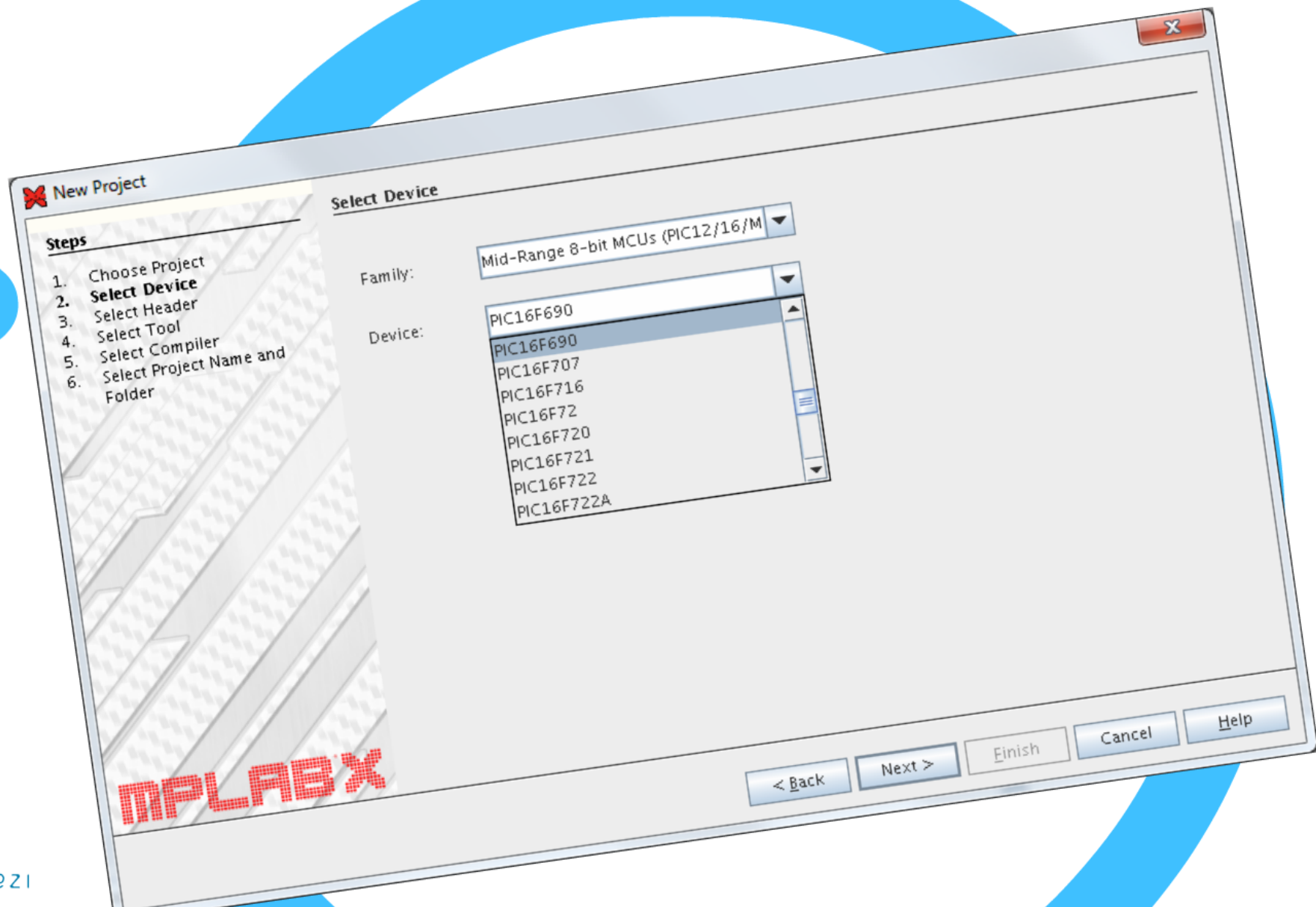
PIC16F707

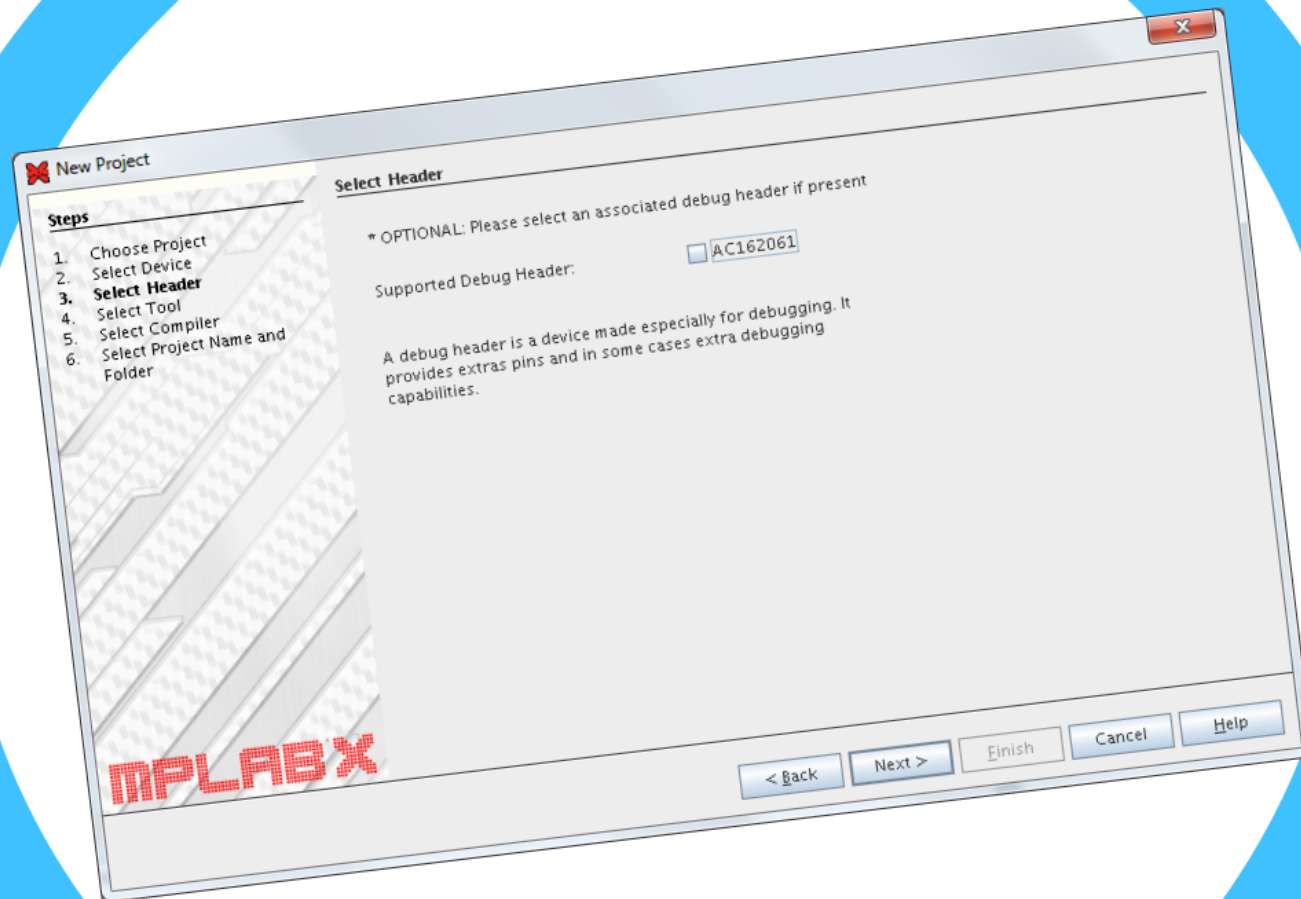
PIC16F716

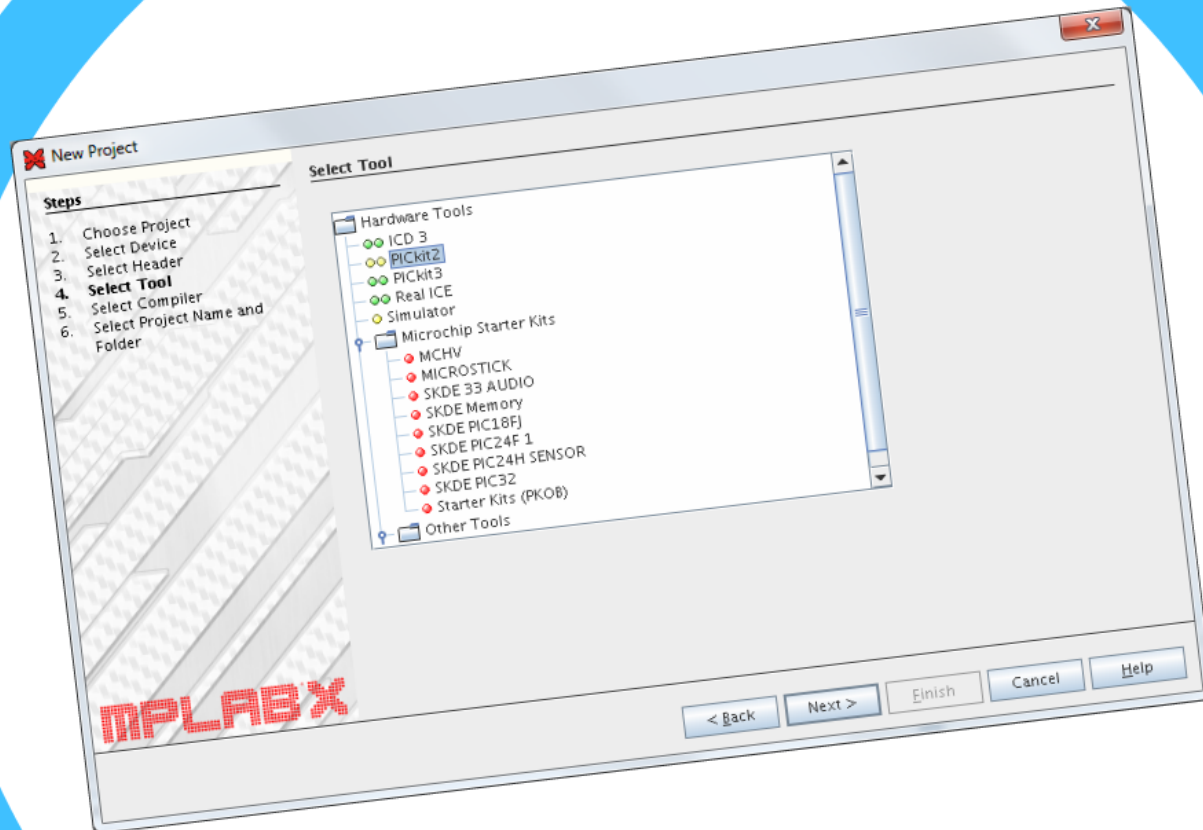
PIC16F72

PIC16F720

PIC16F721







Select Tool

Hardware Tools

  ICD 3

  PICKit2

  PICKit3

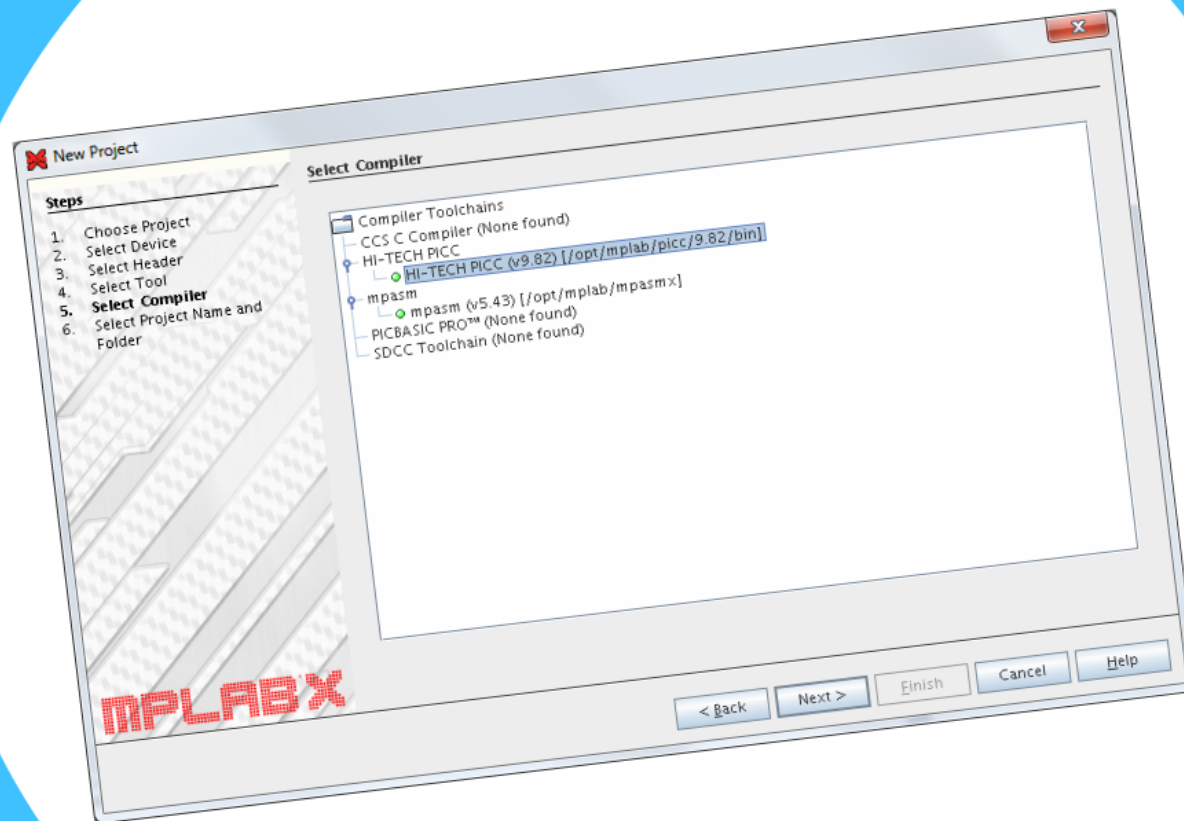
  Real ICE

 Simulator

Microchip Starter Kits

 MCHV

...STICK



Select Compiler

Compiler Toolchains

CCS C Compiler (None found)

HI-TECH PICC

● HI-TECH PICC (v9.82) [/opt/mplab/picc/9.82/bin]

mpasm

● mpasm (v5.43) [/opt/mplab/mpasmx]

PICBASIC PRO™ (None found)

SDCC Toolchain (None found)



and Folder

PICKit2_test valige ise mingi nimi sinna

/home/t061697/MPLABXProjects

/home/t061697/MPLABXProjects/PICKit2_test.X

Brow

existing project.

sources.

project



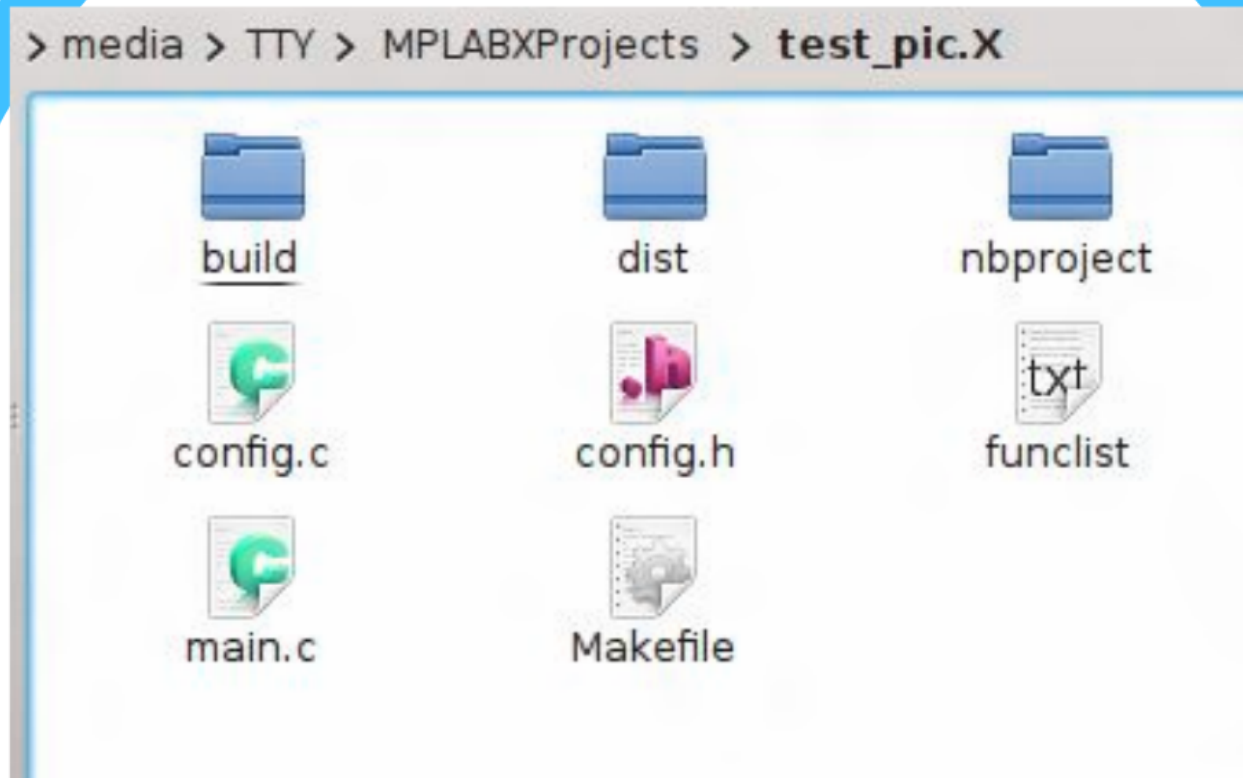
ati.ttu.ee/~hartz/progemine

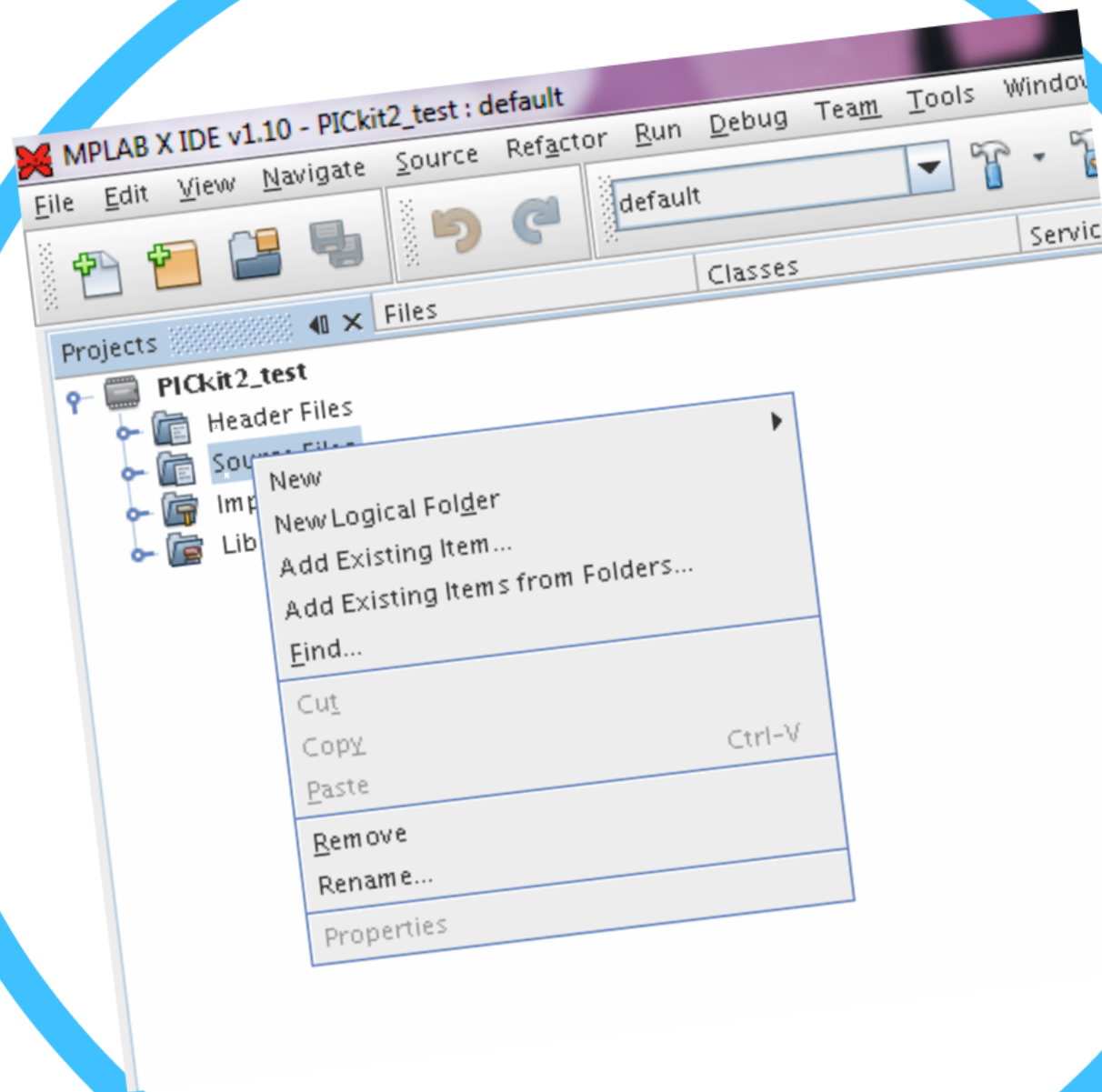


siin on teie matrikli number



t11----



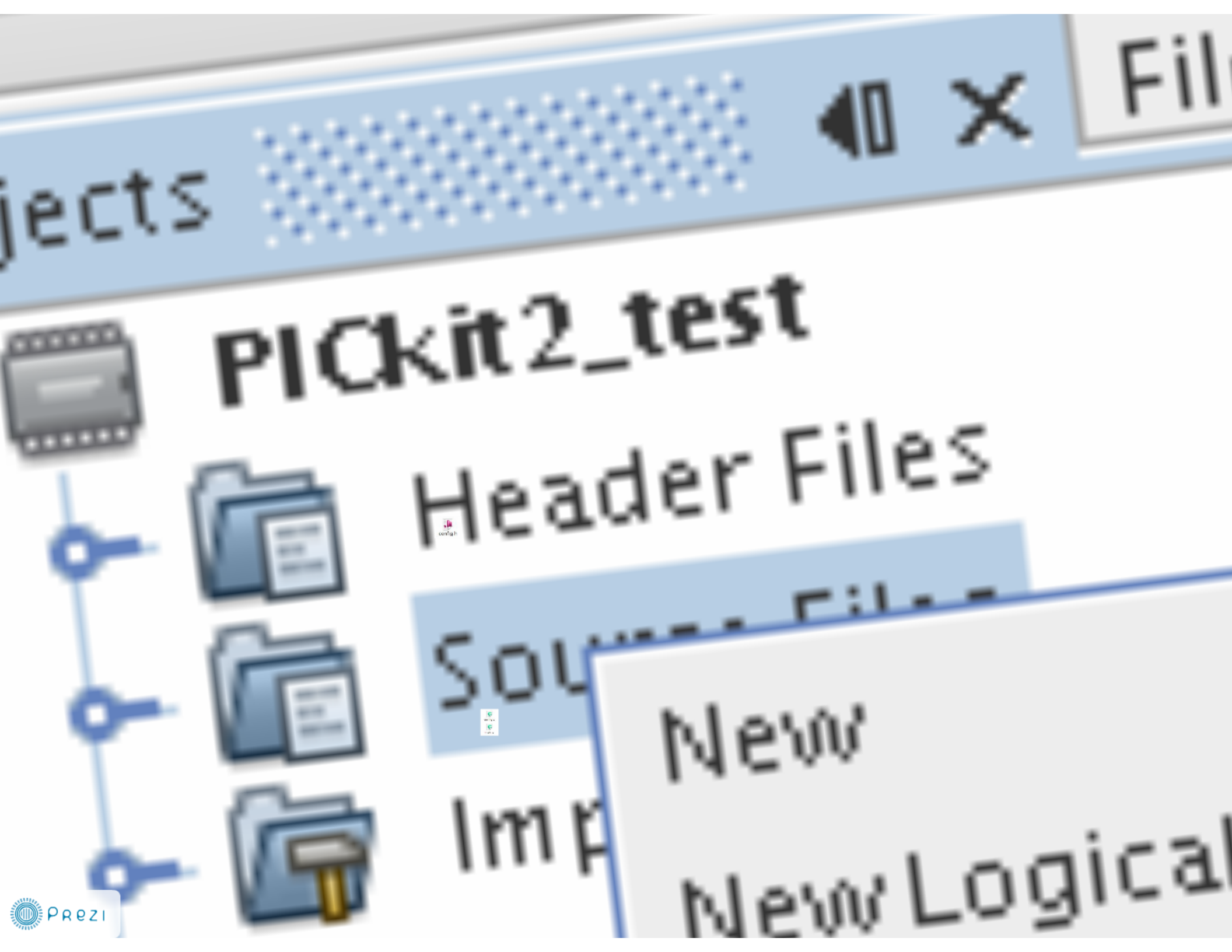




config.c



main.c



jects

PICKit2_test

Header Files

Source Files

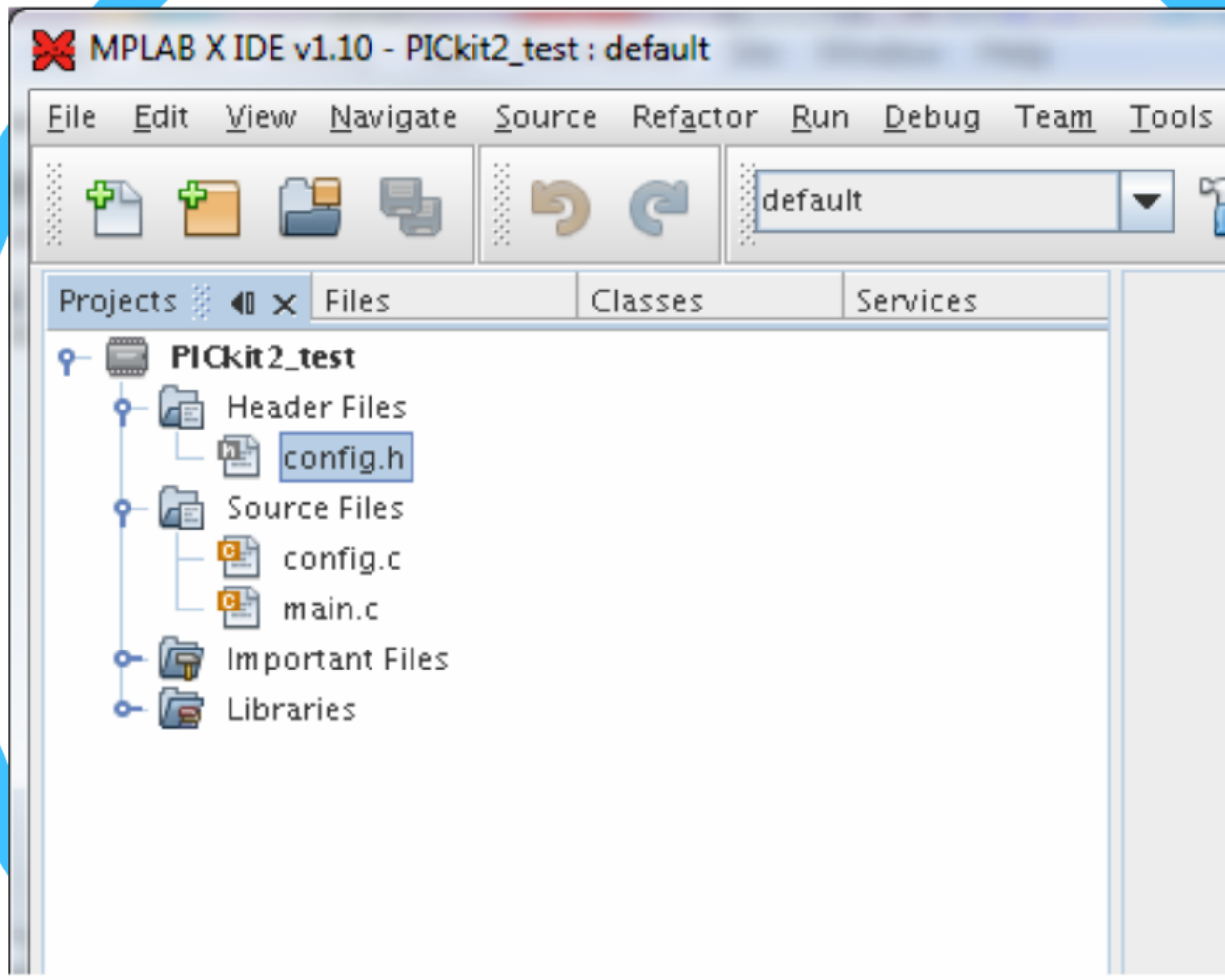
Imp

New

New Logical



config.h



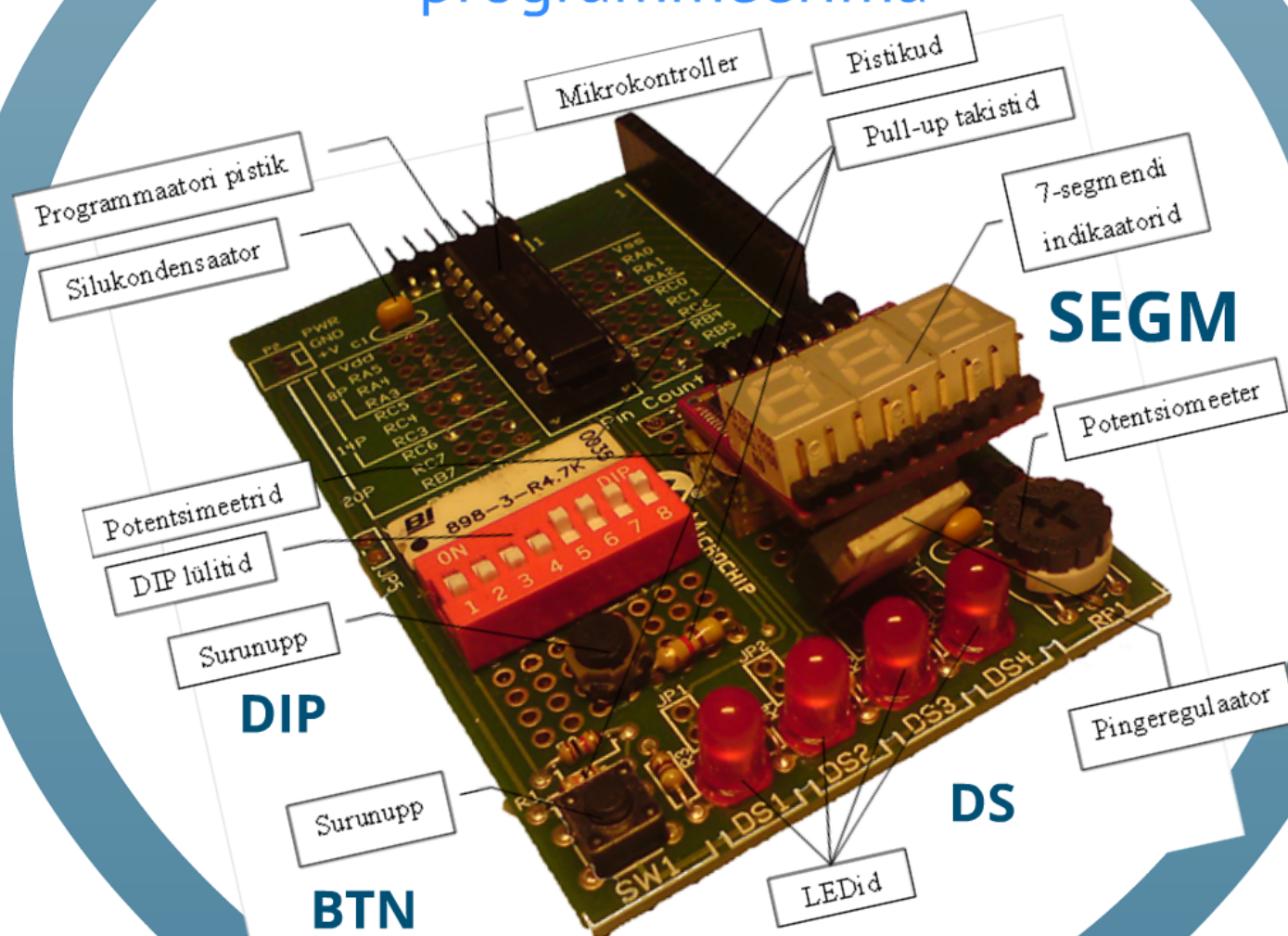
**Lõpuks saame asuda
programmeerimise
kallale**

The image shows a screenshot of a PIC16F690 IDE. The main window displays a C program in a file named `main.c`. The code is as follows:

```
1 //Libraries (mandatory)
2 #include <pic.h>
3 #include "config.h"
4
5 //Main program
6 int main(void) {
7     //Has to be the first function call after main()
8     configuration(); //Includes PIC16F690 basic configuration
9     //Your code goes here...
10
11     //Loop forever
12     while(1)
13     {
14         //To something...
15         delay_100ms; //Creates a delay of 100 ms
16         delay_500ms; //Creates a delay of 500 ms
17
18     }
19     return 0;
20 }
21
22
```

The IDE interface includes a toolbar with various icons for file operations, a status bar at the bottom showing `PC: 0x0` and `z dc c : W:0x0 : bank 0`, and a file explorer on the left showing the `main.c` file.

Mida me täpsemalt õpime programmeerima



```
1 //Libraries (mandatory)
2 #include <pic.h>
3 #include "config.h"
4
5 //Main program
6 int main(void) {
7     //Has to be the first function call after main()
8     configuration(); //Includes PIC16F690 basic configuration
9     //Your code goes here...
10
11     //Loop forever
12     while(1)
13     {
14         //To something...
15         delay_100ms; //Creates a delay of 100 ms
16
17         if (BTN == ON){           kui BTN on alla vajutatud DS1 põleb
18             DS1 = ON;
19         } else if (BTN == OFF){   kui BTN ei ole alla vajutatud,
20             DS1 = OFF;           DS1 ei põle
21         }
22     }
23     return 0;
24 }
25
```

or Run Debug Team Tools Window Help

 Run Main Project 2. F6

Test Project Alt-F6

 Build Main Project

 Clean and Build Main Project 1.

Batch Build Main Project...

Set Project Configuration ▶

Set Main Project ▶

Run File Shift-F6

Test File Ctrl-F6

 Check File Alt-F9

 Validate File Alt+Shift-F9

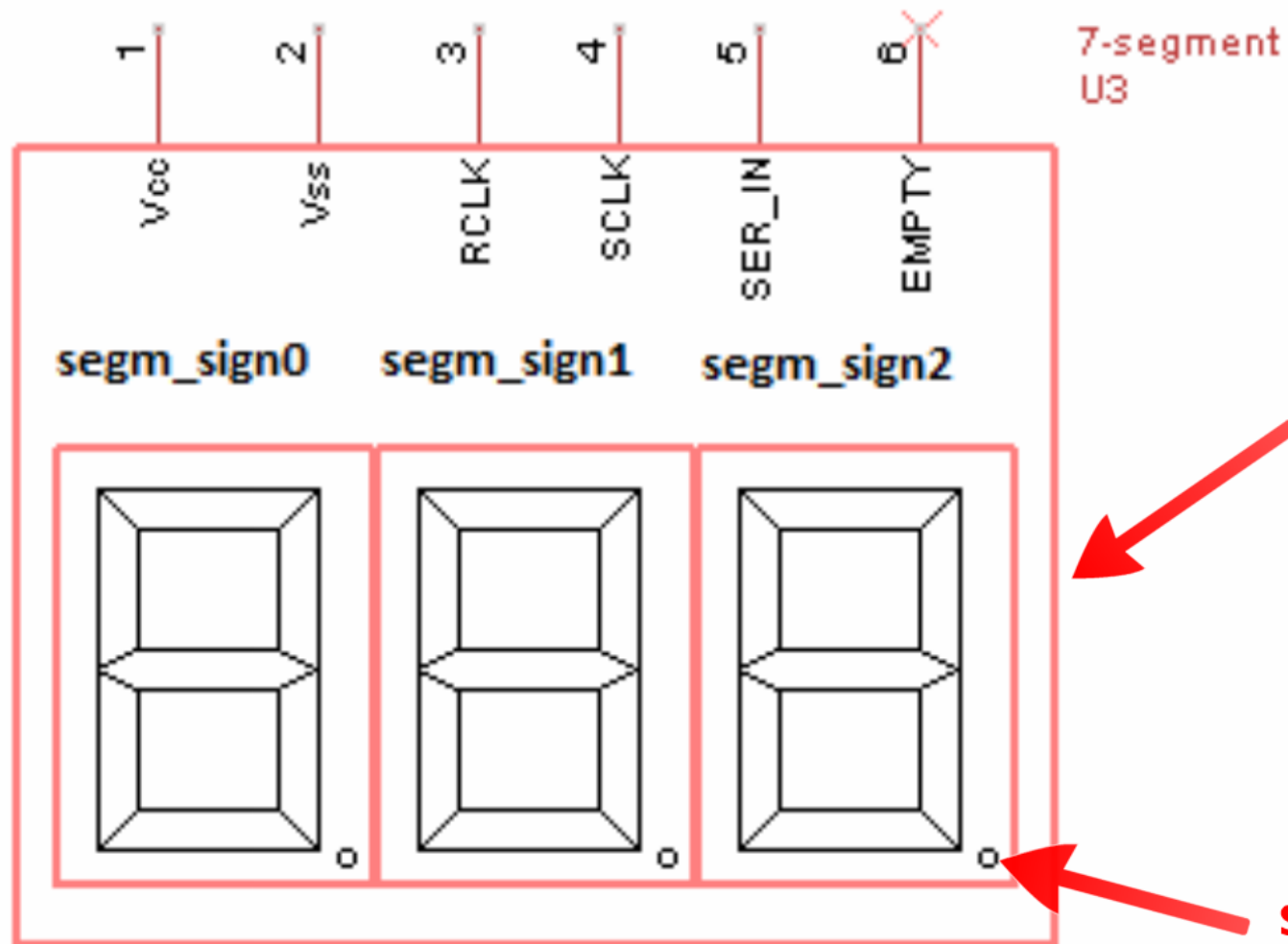
Repeat Build/Run

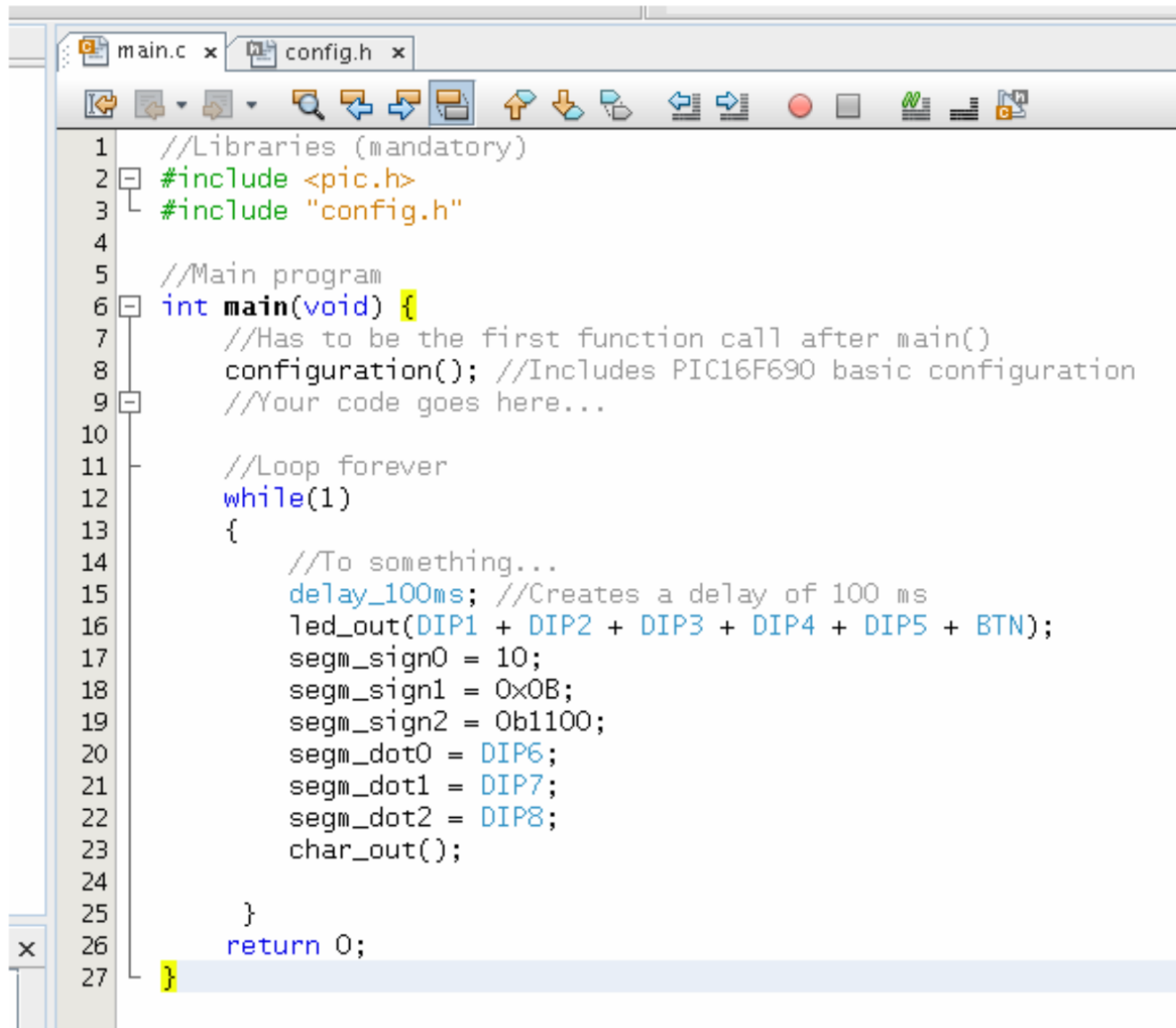
Stop Build/Run

```
1 //Libraries (mandatory)
2 #include <pic.h>
3 #include "config.h"
4
5 //Main program
6 int main(void) {
7     //Has to be the first function call after main()
8     configuration(); //Includes PIC16F690 basic configuration
9     //Your code goes here...
10
11     //Loop forever
12     while(1)
13     {
14         //To something...
15         delay_100ms; //Creates a delay of 100 ms
16
17         if (BTN == ON){
18             DS1 = ON;
19         } else if (BTN == OFF){
20             DS1 = OFF;
21         }
22     }
23     return 0;
24 }
25
```

kui BTN on alla vajutatud DS1 põleb

kui BTN ei ole alla vajutatud, DS1 ei põle





The image shows a code editor window with two tabs: 'main.c' and 'config.h'. The 'main.c' tab is active, displaying a C program. The code includes mandatory libraries, a main function, and a while loop for a forever loop. The loop contains several function calls and assignments for LED and segment control. The code is as follows:

```
1 //Libraries (mandatory)
2 #include <pic.h>
3 #include "config.h"
4
5 //Main program
6 int main(void) {
7     //Has to be the first function call after main()
8     configuration(); //Includes PIC16F690 basic configuration
9     //Your code goes here...
10
11     //Loop forever
12     while(1)
13     {
14         //To something...
15         delay_100ms; //Creates a delay of 100 ms
16         led_out(DIP1 + DIP2 + DIP3 + DIP4 + DIP5 + BTN);
17         segm_sign0 = 10;
18         segm_sign1 = 0x0B;
19         segm_sign2 = 0b1100;
20         segm_dot0 = DIP6;
21         segm_dot1 = DIP7;
22         segm_dot2 = DIP8;
23         char_out();
24
25     }
26     return 0;
27 }
```

or Run Debug Team Tools Window Help

 Run Main Project 2. F6

Test Project Alt-F6

 Build Main Project

 Clean and Build Main Project 1.

Batch Build Main Project...

Set Project Configuration ▶

Set Main Project ▶

Run File Shift-F6

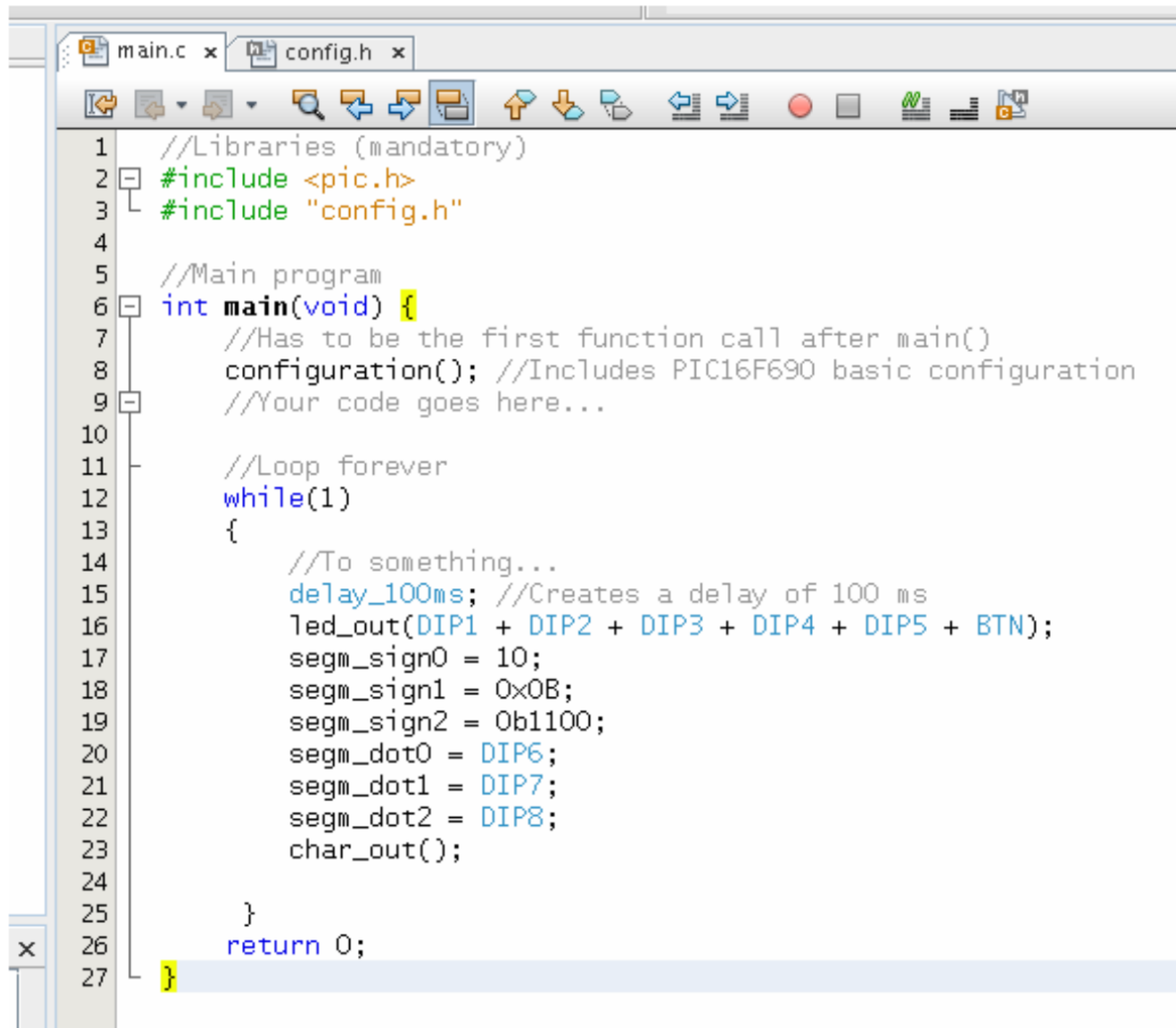
Test File Ctrl-F6

 Check File Alt-F9

 Validate File Alt+Shift-F9

Repeat Build/Run

Stop Build/Run



The image shows a code editor window with two tabs: 'main.c' and 'config.h'. The 'main.c' tab is active, displaying a C program. The code includes mandatory libraries, defines the main program, and contains a while loop for a forever loop. The loop performs a delay and sets various output pins and segments for a 7-segment display.

```
1 //Libraries (mandatory)
2 #include <pic.h>
3 #include "config.h"
4
5 //Main program
6 int main(void) {
7     //Has to be the first function call after main()
8     configuration(); //Includes PIC16F690 basic configuration
9     //Your code goes here...
10
11     //Loop forever
12     while(1)
13     {
14         //To something...
15         delay_100ms; //Creates a delay of 100 ms
16         led_out(DIP1 + DIP2 + DIP3 + DIP4 + DIP5 + BTN);
17         segm_sign0 = 10;
18         segm_sign1 = 0x0B;
19         segm_sign2 = 0b1100;
20         segm_dot0 = DIP6;
21         segm_dot1 = DIP7;
22         segm_dot2 = DIP8;
23         char_out();
24
25     }
26     return 0;
27 }
```

Näidisülesande modifikatsioon

Eeldus

Saada aru tarkvara peatükis kirjeldatud programmi näidiskoodist. Mõista sisendite ning väljundite kasutamist ning väljundeid realiseerivate funktsioonide tööpõhimõtteid.

Ülesanne

Muuta näidisprogrammi koodi nii, et 7-segmendi indikaatori esimene segment (segm_char0) näitaks DIP lülitite summat. Teine segment peab olema olekus NULL ehk ei põle. Ning kolmas segment peab näitama mitmendat korda summeerimist teostatakse (punkt kaasa arvatud). LEDidele väljastada DIP lülititelt saadud summa 4ga jagamisel tulev jääk. Summa ning jäägi vastus väljastatakse alles pärast surunupu BTN vajutamist.

Lahenduse kirjeldus

DIP lülitid seatakse soovitud olekusse. Seejärel vajutatakse BTN surunuppu. Selle tagajärjel väljastatakse 7-segmendile DIP lülitite summa ning summeerimise kordade arv ja LEDidele summa jääk jagamisel 4.ga. Järgmise summa saamiseks korratakse eelnevat.

Märkus

DIP lülitid saavad omada kahte olekus ON & OFF. ON tähendab, et lüliti väärtuseks on 1 ning OFF, et lüliti väärtuseks on 0. Järelikult vastab iga DIP lüliti väärtus olekus ON väärtusele 1.

Jooksvad tuled

Eeldus

Mõista programmeerimiskeeles C enimkasutatavaid konstruktsioone (tsükel, tingimus). Osata kasutada viiteid.

Ülesanne

Programmeeri programm, mis paneks ühel ajahetkel põlema ühe LEDi. Järgmisel hetkel tema kõrvaloleva LEDi jne. Tulemuseks peab tekkima efekt nagu LEDid liiguksid vasakult paremale ning tagasi. LEDide lülitamise vaheline viide vali ise.

Lahenduse kirjeldus

<http://petski.tainas.ee/wp-content/uploads/2012/11/Running-leds.gif>

Soovitus

Proovi välja mõelda nutikas lahendus!

LEDi juhtimine

Ülesanne

Koosta programm, mis väljastab igale LEDile 2-sisendiga loogikaventiili väärtuse lähtuvalt DIP lülitite olekust. Loogikaventiilid mida simuleerida vali ise (AND, NAND, OR, NOR, XOR, XNOR). Simuleeri nelja loogikaventiili.

Lahenduse kirjeldus

DIP1 ning DIP2 on määratud simuleerima loogikafunktsiooni 2-AND. Järelikult lülitub LED DS1 sisse ainult juhul, kui mõlemad DIP lülitid on olekus ON.

Lisamaterjal

2-sisendiga loogikaelementide tõeväärtustabel.

A	B	AND	NAND	OR	NOR	XOR	XNOR
0	0	0	1	0	1	0	1
0	1	0	1	1	0	1	0
1	0	0	1	1	0	1	0
1	1	1	0	1	0	0	1

Munakell

Ülesanne

Programmeeri programm, mis loendab vastavalt etteantud kahendväärtusele väärtusi alla, kuni jõuab nulli. Hetkelist väärtust näidatakse 7-segmendi indikaatoritel. Loenduri algväärtus antakse ette DIP lülitelt. Kui kell jõuab nulli hakkavad vilkuma LED tuled. Loendur hakkab loendama alles pärast surunupu BTN vajutamist.

Lahenduse kirjeldus

Kasutaja annab DIP lülititele näiteks väärtuse 00001111 ning seejärel vajutab surunuppu BTN. 7-segmendi indikaatorile ilmub väärtus 15 ning seejärel hakatakse antud väärtust 1 võrra vähendada, kuni tulemuseks on null. Järgnevalt hakkavad vilkuma LED tuled. Korduste arv pole määratud. Kasutaja poolt antav minimaalne väärtus on 0 ning maksimaalne 255 (1111111). Programm peab olema suuteline võtma vastu kõiki arve antud vahemikust ning väljastama neid 7-segmendi indikaatorile.

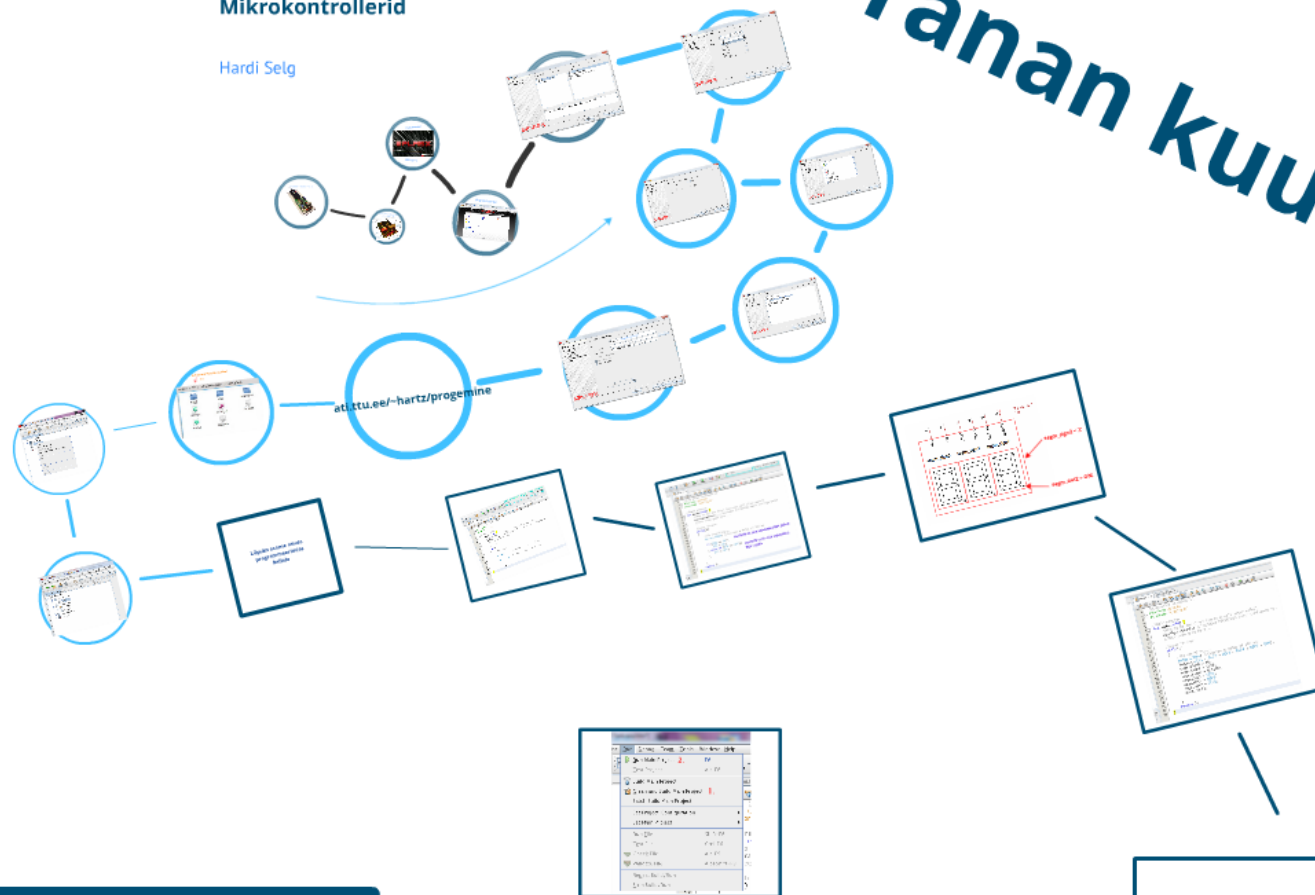
Märkus

Väärtuste väljastamisel 7-segmendi indikaatorile lähtuda mõistlikust loetavusest. Liiga kiiresti muutuvaid numbreid on raske lugeda!

Täna kuulamast :)

Mikrokontrollerid

Hardi Selg



Näidisülesande modifikatsioon

Eeldus

Saada aru tarkvara peatükis kirjeldatud programmi näidiskoodist. Mõista sisendite väljundeid realiseerivate funktsioonide tööpõhimõtteid.

Jooksivad tuled