פרוייקטון-משחק ניחושים.

שם המדריך: ברונפמן אירינה

מגישים:
יעקב ליאור פינטו
ניר נחמן גולדנברג

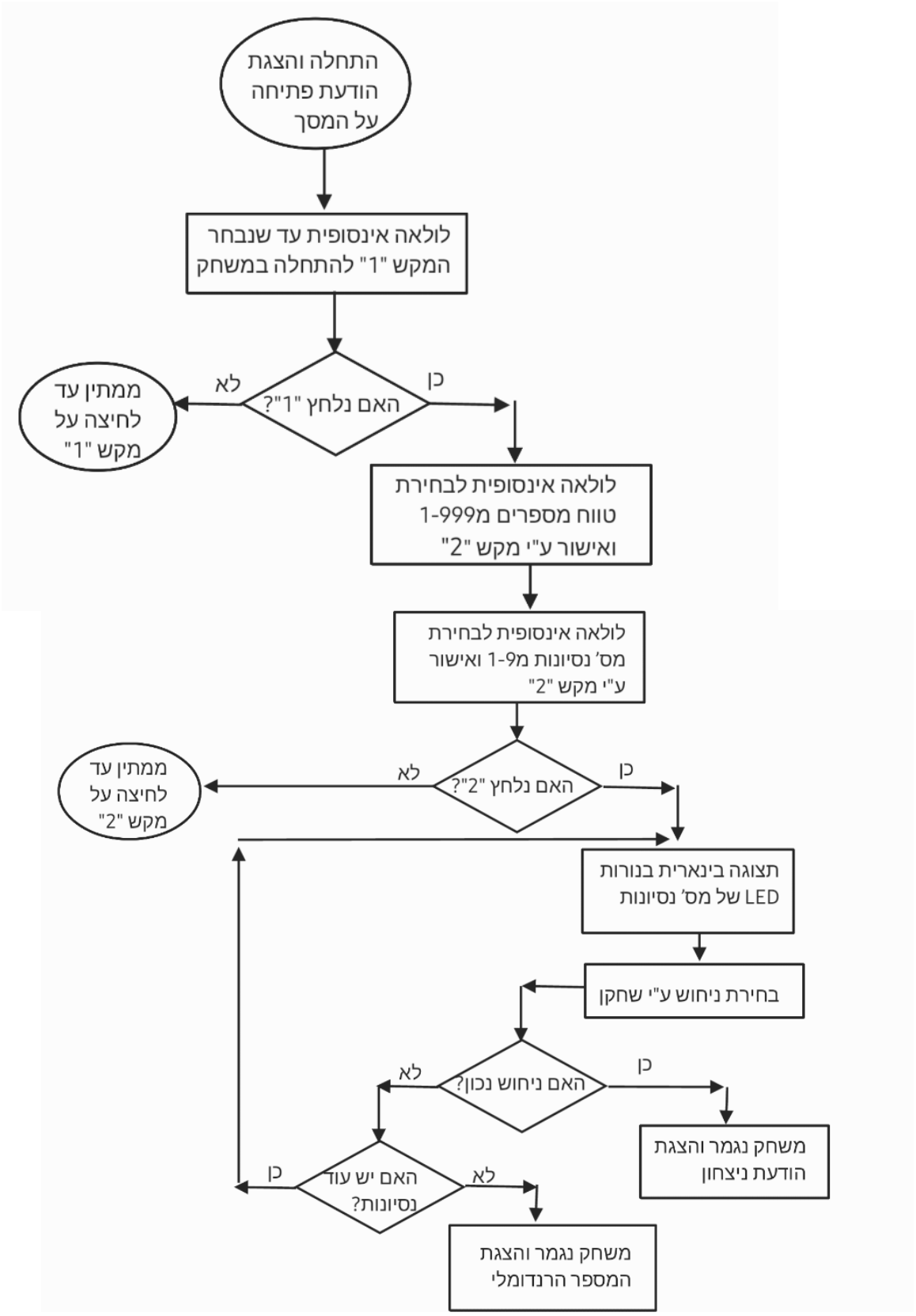
תקציר:

התוכנית מתארת משחק "ניחוש מספר" , הכולל תצוגת LCD ונורותLED . עם התחלת התוכנית, מוצגת הודעת כניסה והוראות ראשוניות. המשתמש לוחץ על כפתורים כדי לבחור טווח מספרים ומספר ניסיונות לניחוש, אשר נקבעים בעזרת מדידות אנלוגיות. מספר רנדומלי נבחר בטווח שנקבע, והמשתמש נדרש לנחש אותו תוך מספר מוגבל של ניסיונות .

בכל ניחוש, התוכנית מציגה את הניחוש הנוכחי, מצב הניחוש (גבוה או נמוך מדי), ומספר הניסיונות שנותרו באמצעות תצוגת LED והודעות ב-בהתאם בצג הLCD.

אם המשתמש מצליח לנחש את המספר, מופיעה הודעת ניצחון, אחרת, מופיעה הודעת הפסד עם המספר הנכון. נורות ה LED-מתעדכנות בהתאם להתקדמות המשתמש במשחק, כולל הבהוב עבור ניסיון ניחוש והצגת מספר הניסיונות שנותרו

תרשים זרימה לתוכנית:

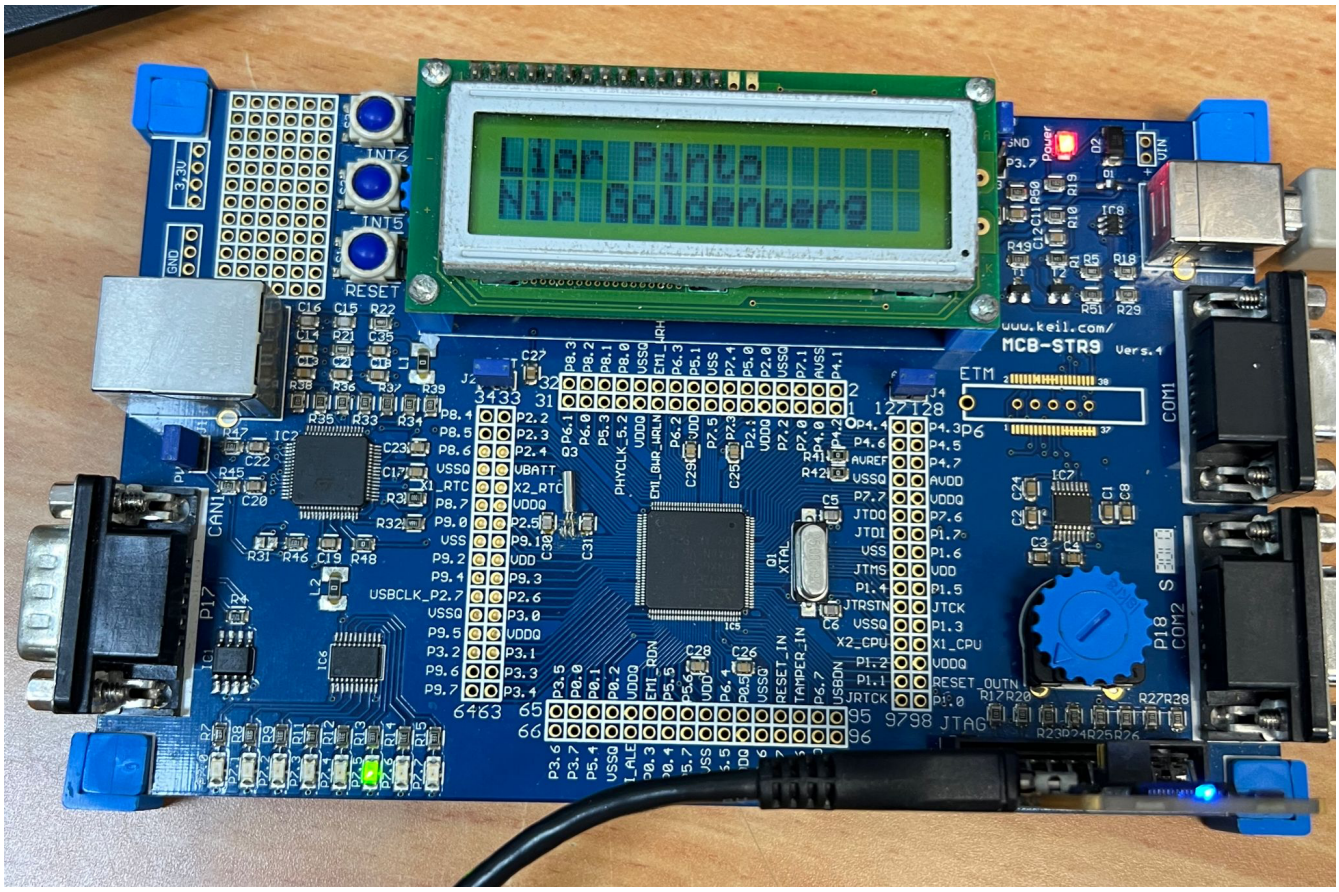


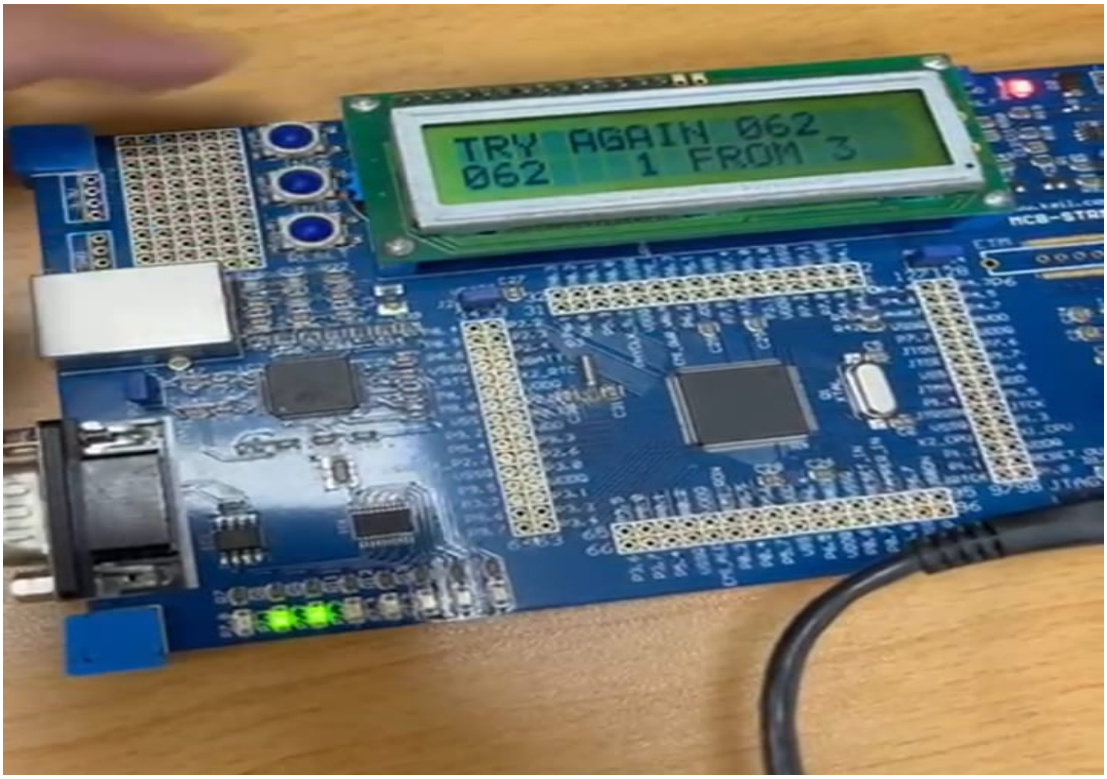
```c
#include <91x_lib.h>

#define   LED       GPIO7->DR[0x3FC]
#define   SW        GPIO3->DR[0x3FC]

int measure(void);// measure p4.0 analog return 10 bit integer
void dly(int i); //delay function depended by i
void printnums(int num, int rplace, int cplace); // print function to print guess/number/range

void dly(int i)
{
int j,k;
for (j=1;j<100;j++)
for(k=0;k<i;k++);
}

void printnums(int num, int rplace, int cplace) //print function
{
set_cursor(rplace,cplace);
lcd_putchar((num/100)+0x30); //print Meot as char
set_cursor(rplace+1,cplace);
lcd_putchar(((num/10)%10)+0x30); //print Asarot as char
set_cursor(rplace+2,cplace);
lcd_putchar((num%10)+0x30); //print Ahadot as char
}

int measure(void)
{
    ADC->CR         &= 0xFF7f;
    ADC->CR         = 0x0002;
    ADC->CR         |= 0x0001;
    dly(10000);
    while(((ADC->CR) & 0x8000)==0); // wait for end of converge
    return ((ADC->DR0) & 0x03FF);
}

int main (void)
{
unsigned int td=5000, number=0, tap=0, tries, guess, range, i=0; //Define Variables
// td - for dly function // number - random number // tap - how many times youve already try
// tries - how many tries youve get // guess - youre guess // range - your (range-1)
// restart - for strat new game with the same settings

Intio();
Int_a2d();
LED=0x0; // turn off LEDs
lcd_init();
lcd_clear();

lcd_print ("WELCOME TO GUESS");
set_cursor(0,1);
lcd_print ("A NUMBER GAME!");
dly(td*20);
LED=0xF; // turn on half LEDs
lcd_clear();

lcd_print ("SET YOUR SETTINGS");
dly(td*20);
lcd_clear();

while((SW&0X40)!=0) //Random number
{
      set_cursor(0,0);
    lcd_print ("PRESS 1 TO START");
    set_cursor(0,1);
    lcd_print ("GUESS 0 -");
    printnums(range, 10, 1);
    range=(measure()*998/1023); // range == measure between 0-999
    dly(td);
}
```

```
72      LED=0xAA; // turn on half (Alternately) LEDs
73      lcd_clear();
74
75      while((SW&0X20)!=0) //Random number
76      {
77          tries=(measure()*8/1023)+1; // range == measure between 1-9
78          set_cursor(0,0);
79          lcd_putchar((tries)+0x30);
80          set_cursor(2,0);
81          lcd_print ("TRIES TO WIN");
82          set_cursor(0,1);
83          lcd_print ("PRESS 2 TO START");
84          number=(number+1)%(range); // 0 <= number <= range
85          dly(100);
86      }
87
88      LED=0xFF; // turn on all LEDs
89
90      lcd_clear();    //clear LCD
91      lcd_print ("RANDOM NUMBER");
92      set_cursor(0,1);
93      lcd_print ("PICKING...");
94
95      for (i = 0; i < 3; i++) //blinking LEDs for users experience
96      {
97          LED = 0x00;
98          dly(td*4);
99          LED = 0xFF;
100         dly(td*4);
101     }
102     dly(td);
103
104     lcd_clear();    //clear LCD
105     lcd_print ("PRESS 2 TO");
106     set_cursor(0,1);
107     lcd_print ("MAKE A GUESS");
```

```
108     dly(td*20);
109
110         while (tap<tries) // game loop
111         {
112         lcd_clear();    //clear LCD
113         lcd_print ("MAKE A GUESS");
114         printnums(guess, 0, 1);
115         guess=(measure()*range/1023); // guess == measure between 1-range
116         LED=(2^((tries+2)-tap)); //shows on LED how much tries lefts
117         dly(td);
118
119         if ((SW&0X20)==0)
120             {
121                 tap++;
122
123                 lcd_clear();
124                 if (guess==number)
125                 {
126                     tap=tries; //finish game
127                 }
128                 else
129                 {
130                 lcd_clear();
131                 if (guess>number)
132                 lcd_print ("TOO HIGH!");
133                 else lcd_print ("TOO LOW!");
134                     dly(td*20);
135                     lcd_clear();
136                     lcd_print ("TRY AGAIN");
137                     printnums(guess, 10, 0);
138                     set_cursor(6,1);
139                     lcd_putchar(tap+0x30); // how many times tried
140                     set_cursor(8,1);
141                     lcd_print ("FROM");
142                     set_cursor(13,1);
143                     lcd_putchar(tries+0x30); // how many times in total
```

```
144                     dly(td*20);
145                 }
146             }
147
148         if (tap>=tries) //when loosing print the right number
149             {
150                 lcd_clear();
151                 if (guess==number)
152                     lcd_print ("YOU WIN! :-)");
153                 else  lcd_print ("GAME OVER :-(");
154
155                 set_cursor(0,1);
156                 lcd_print ("NUMBER IS");
157                 printnums(number, 10, 1);
158                 dly(td*20);
159             }
160         }
161     }
```


Lior Pinto
Nir Goldenberg
MCB-STR9
Vers.1
RESET
INT4
INT5
ETM
COM1
COM2
JTAG


TRY AGAIN 062
062 1 FROM 3

קטע הקוד משורה 1-35

```
#include <91x_lib.h>

#define   LED      GPIO7->DR[0x3FC]
#define   SW       GPIO3->DR[0x3FC]

int measure(void);// measure p4.0 analog return 10 bit
integer
void dly(int i); //delay function depended by i
void printnums(int num, int rplace, int cplace); // print
function to print guess/number/range

void dly(int i)
{
int j,k;
for (j=1;j<100;j++)
for(k=0;k<i;k++);
}

void printnums(int num, int rplace, int cplace) //print function
{
set_cursor(rplace,cplace);
lcd_putchar((num/100)+0x30); //print Meot as char
set_cursor(rplace+1,cplace);
lcd_putchar(((num/10)%10)+0x30); //print Asarot as char
set_cursor(rplace+2,cplace);
lcd_putchar((num%10)+0x30); //print Ahadot as char
}

int measure(void)
{
   ADC->CR       &= 0xFF7f;
   ADC->CR       = 0x0002;
   ADC->CR       |= 0x0001;
   dly(10000);
   while(((ADC->CR) & 0x8000)==0); // wait for end of
converge
   return ((ADC->DR0) & 0x03FF);
}
```

קטע הקוד משורה 37-73

```
int main (void)
{
unsigned int td=5000, number=0, tap=0, tries, guess, range,
i=0; //Define Variables
// td - for dly function // number - random number // tap - how
many times youve already try
// tries - how many tries youve get // guess - youre guess //
range - your (range-1)
// restart - for strat new game with the same settings

Intio();
Int_a2d();
LED=0x0; // turn off LEDs
lcd_init();
lcd_clear();

lcd_print ("WELCOME TO GUESS");
set_cursor(0,1);
lcd_print ("A NUMBER GAME!");
dly(td*20);
LED=0xF; // turn on half LEDs
lcd_clear();

lcd_print ("SET YOUR SETTINGS");
dly(td*20);
lcd_clear();

while((SW&0X40)!=0) //Random number
{
            set_cursor(0,0);
   lcd_print ("PRESS 1 TO START");
   set_cursor(0,1);
   lcd_print ("GUESS 0 -");
   printnums(range, 10, 1);
   range=(measure()*998/1023); // range == measure
between 0-999
   dly(td);
}

LED=0xAA; // turn on half (Alternately) LEDs
lcd_clear();
```

קטע הקוד משורה 75-117

```
while((SW&0X20)!=0) //Random number
{
   tries=(measure()*8/1023)+1; // range == measure
between 1-9
   set_cursor(0,0);
   lcd_putchar((tries)+0x30);
   set_cursor(2,0);
   lcd_print ("TRIES TO WIN");
   set_cursor(0,1);
   lcd_print ("PRESS 2 TO START");
   number=(number+1)%(range); // 0 <= number <= range
   dly(100);
}

LED=0xFF; // turn on all LEDs

lcd_clear();      //clear LCD
lcd_print ("RANDOM NUMBER");
set_cursor(0,1);
lcd_print ("PICKING...");

for (i = 0; i < 3; i++) //blinking LEDs for users experience
{
   LED = 0x00;
   dly(td*4);
   LED = 0xFF;
   dly(td*4);
}
dly(td);

lcd_clear();      //clear LCD
lcd_print ("PRESS 2 TO");
set_cursor(0,1);
lcd_print ("MAKE A GUESS");
dly(td*20);

   while (tap<tries) // game loop
   {
   lcd_clear();  //clear LCD
   lcd_print ("MAKE A GUESS");
   printnums(guess, 0, 1);
   guess=(measure()*range/1023); // guess == measure
between 1-range
   LED=(2^((tries+2)-tap)); //shows on LED how much tries
lefts
   dly(td);
```

קטע הקוד משורה 119-161

```
    if ((SW&0X20)==0)
       {
          tap++;

          lcd_clear();
          if (guess==number)
          {
             tap=tries; //finish game
          }
          else
          {
          lcd_clear();
          if (guess>number)
          lcd_print ("TOO HIGH!");
          else lcd_print ("TOO LOW!");
             dly(td*20);
             lcd_clear();
             lcd_print ("TRY AGAIN");
             printnums(guess, 10, 0);
             set_cursor(6,1);
             lcd_putchar(tap+0x30); // how many times tried
             set_cursor(8,1);
             lcd_print ("FROM");
             set_cursor(13,1);
             lcd_putchar(tries+0x30); // how many times in total
             dly(td*20);
          }
       }

    if (tap>=tries) //when loosing print the right number
      {
          lcd_clear();
          if (guess==number)
             lcd_print ("YOU WIN! :-)");
          else  lcd_print ("GAME OVER :-(");

          set_cursor(0,1);
          lcd_print ("NUMBER IS");
          printnums(number, 10, 1);
          dly(td*20);
       }
    }
}
```