

## ME-Keypad 4x4 Board

ME-Keypad 4x4 Board используется для загрузки числовых значения в микроконтроллер. Она состоит из 16 кнопок расположенных в четыре линии и четыре столбца. Подключение к системам разработки происходит спомощью разъема IDC-10F.

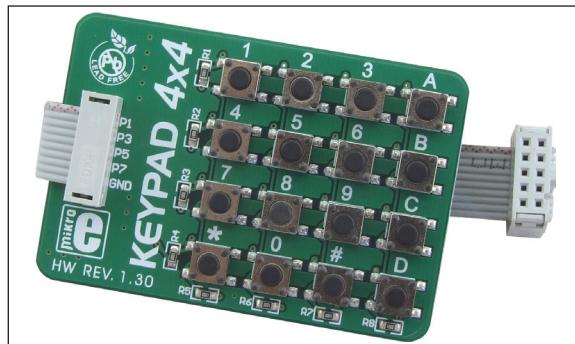


Рисунок 1: ME-Keypad 4x4 Board

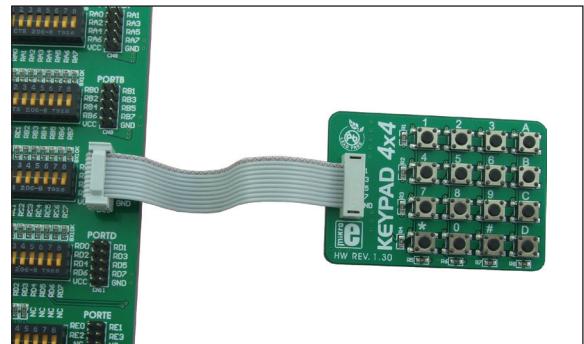


Рисунок 2: ME-Keypad 4x4 Board подключен к системе разработки

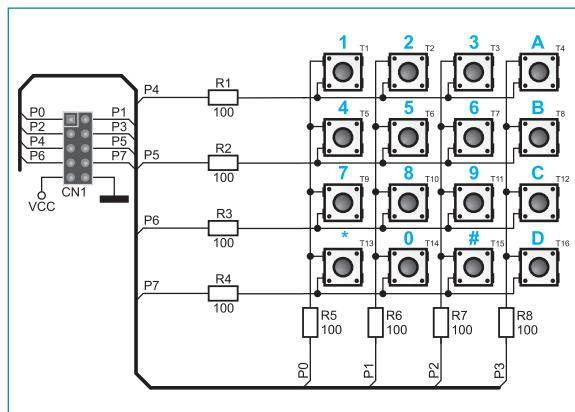


Рисунок 3: ME-Keypad 4x4 Board схемотехническое подключение

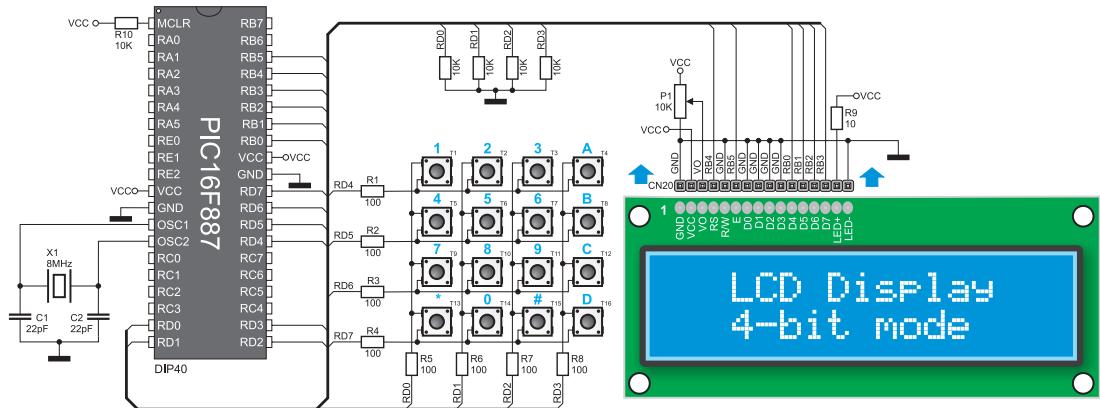


Рисунок 4: Схема подключения клавиатуры, ЖК-дисплея и микроконтроллера

### Пример 1: Программа написана в mikroC PRO for PIC

```

unsigned short kp, cnt, oldstate = 0;
char txt[6]; // Uncomment this block for keypad4x4

char keypadPort at PORTD; // Keypad module connections
                           // End Keypad module connections

sbit LCD_RS at RB4_bit; // LCD module connections
sbit LCD_EN at RB5_bit;
sbit LCD_D4 at RB0_bit;
sbit LCD_D5 at RB1_bit;
sbit LCD_D6 at RB2_bit;
sbit LCD_D7 at RB3_bit;

sbit LCD_RS_Direction at TRISB4_bit;
sbit LCD_EN_Direction at TRISB5_bit;
sbit LCD_D4_Direction at TRISB0_bit;
sbit LCD_D5_Direction at TRISB1_bit;
sbit LCD_D6_Direction at TRISB2_bit;
sbit LCD_D7_Direction at TRISB3_bit;
                           // End LCD module connections

void main() {
    cnt = 0; // Reset counter
    Keypad_Init(); // Initialize Keypad
    ANSEL = 0; // Configure AN pins as digital I/O
    ANSELH = 0;
    Lcd_Init(); // Initialize LCD
    Lcd_Cmd(_LCD_CLEAR); // Clear display
    Lcd_Cmd(_LCD_CURSOR_OFF); // Cursor off
    Lcd_Out(1, 1, "1");
    Lcd_Out(1, 1, "Key :");
    Lcd_Out(2, 1, "Times:");
    do { // Wait for key to be pressed and released
        kp = Keypad_Key_Click(); // Store key code in kp variable
        while (!kp); // Prepare value for output, transform key to its ASCII value
        switch (kp) {
            //case 10: kp = 42; break; // '*' // Uncomment this block for keypad4x3
            //case 11: kp = 48; break; // '0'
            //case 12: kp = 35; break; // '#'
            //default: kp += 48;
    }
}

// Keypad module connections
// End Keypad module connections

// LCD module connections
// End LCD module connections

// Pressed key differs from previous
// Pressed key is same as previous
// Print key ASCII value on LCD
// If counter varialble overflow
// Transform counter value to string
// Display counter value on LCD
}

```

---

## Пример 2: Программа написана в mikroBasic PRO for PIC

```
program Keypad_Test
dim kp, cnt, oldstate as byte
txt as char[7]

' Keypad module connections
dim keypadPort as byte at PORTD
' End Keypad module connections

' Lcd module connections
dim LCD_RS as sbit at RB4_bit
LCD_EN as sbit at RB5_bit
LCD_D4 as sbit at RB0_bit
LCD_D5 as sbit at RB1_bit
LCD_D6 as sbit at RB2_bit
LCD_D7 as sbit at RB3_bit

LCD_RS_Direction as sbit at TRISB4_bit
LCD_EN_Direction as sbit at TRISB5_bit
LCD_D4_Direction as sbit at TRISB0_bit
LCD_D5_Direction as sbit at TRISB1_bit
LCD_D6_Direction as sbit at TRISB2_bit
LCD_D7_Direction as sbit at TRISB3_bit
' End Lcd module connections

main:
oldstate = 0
cnt = 0
Keypad_Init()
ANSEL = 0
ANSELH = 0
Lcd_Init()
Lcd_Cmd(_LCD_CLEAR)
Lcd_Cmd(_LCD_CURSOR_OFF)
Lcd_Out(1, 1, "Key :")
Lcd_Out(2, 1, "Times:")

while TRUE

kp = 0           ' Reset key code variable

' Wait for key to be pressed and released
while ( kp = 0 )
  kp = Keypad_Key_Click()      ' Store key code in kp variable
wend

' Prepare value for output, transform key to it's ASCII value
select case kp
  'case 10: kp = 42    ' Uncomment this block for keypad4x3
  'case 11: kp = 48    ' "0"
  'case 12: kp = 35    ' "#"
  default: kp += 48

case 1          ' Uncomment this block for keypad4x4
case 2
case 3
case 4
case 5
case 6
case 7
case 8
case 9
case 10
case 11
case 12
case 13
case 14
case 15
case 16
end select

if (kp > oldstate) then   ' Pressed key differs from previous
  cnt = 1
  oldstate = kp
else
  Inc(cnt)                  ' Pressed key is same as previous
end if

Lcd_Chр(1, 10, kp)        ' Print key ASCII value on LCD

if (cnt = 255) then        ' If counter variable overflow
  cnt = 0
  Lcd_Out(2, 10, " ")
end if

WordToStr(cnt, txt)        ' Transform counter value to string
Lcd_Out(2, 10, txt)
wend
end.
```

### Пример 3: Программа написана в mikroPascal PRO for PIC

```
program Keypad_Test;

var kp, cnt, oldstate : byte;
    txt : array[6] of byte;

// Keypad module connections
var keypadPort : byte at PORTD;
// End Keypad module connections

// Lcd module connections
var LCD_RS : sbit at RB4_bit;
LCD_EN : sbit at RB5_bit;
LCD_D4 : sbit at RB0_bit;
LCD_D5 : sbit at RB1_bit;
LCD_D6 : sbit at RB2_bit;
LCD_D7 : sbit at RB3_bit;

var LCD_RS_Direction : sbit at TRISB4_bit;
LCD_EN_Direction : sbit at TRISB5_bit;
LCD_D4_Direction : sbit at TRISB0_bit;
LCD_D5_Direction : sbit at TRISB1_bit;
LCD_D6_Direction : sbit at TRISB2_bit;
LCD_D7_Direction : sbit at TRISB3_bit;
// End Lcd module connections

begin
oldstate := 0;
cnt := 0; // Reset counter
Keypad_Init(); // Initialize Keypad
ANSEL := 0; // Configure AN pins as digital I/O
ANSELH := 0;
Lcd_Init(); // Initialize Lcd
Lcd_Cmd(_LCD_CLEAR); // Clear display
Lcd_Cmd(_LCD_CURSOR_OFF); // Cursor off
Lcd_Out(1, 1, 'Key :'); // Write message text on Lcd
Lcd_Out(2, 1, 'Times');

while TRUE do
begin
kp := 0; // Reset key code variable

// Wait for key to be pressed and released
while ( kp = 0 ) do
    kp := Keypad_Key_Click(); // Store key code in kp variable
// Prepare value for output, transform key to it's ASCII value
case kp of
    //case 10: kp = 42; // **
    //case 11: kp = 48; // '0'
    //case 12: kp = 35; // '#'
    //default: kp += 48;
```

```
1: kp := 49; // 1 // Uncomment this block for keypad4x4
2: kp := 50; // 2
3: kp := 51; // 3
4: kp := 65; // A
5: kp := 52; // 4
6: kp := 53; // 5
7: kp := 54; // 6
8: kp := 66; // B
9: kp := 55; // 7
10: kp := 56; // 8
11: kp := 57; // 9
12: kp := 67; // C
13: kp := 42; // *
14: kp := 48; // 0
15: kp := 35; // #
16: kp := 68; // D

end;

if (kp <> oldstate) then // Pressed key differs from previous
begin
    cnt := 1;
    oldstate := kp;
end
else // Pressed key is same as previous
inc(cnt);

Lcd_Ch(1, 10, kp); // Print key ASCII value on Lcd

if (cnt = 255) then // If counter variable overflow
begin
    cnt := 0;
    Lcd_Out(2, 10, ' ');
end;

WordToStr(cnt, txt); // Transform counter value to string

Lcd_Out(2, 10, txt); // Display counter value on Lcd
end;
end.
```