



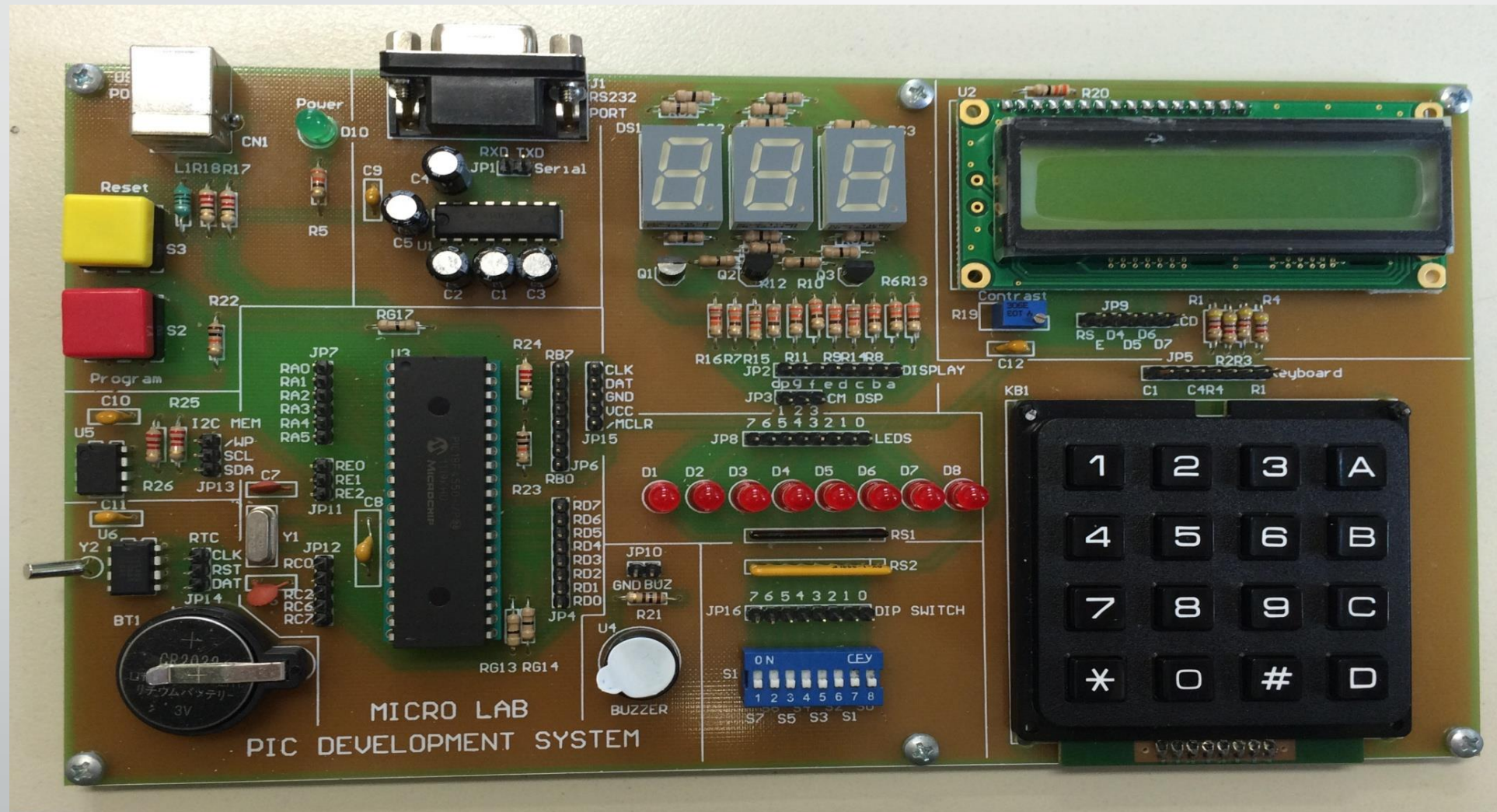
Ενσωματωμένα Συστήματα

(6^ο εξάμηνο)
10-Keypad 4x4

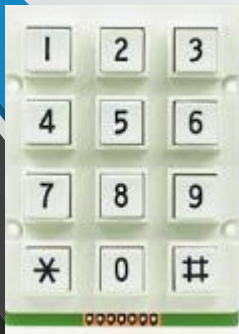
Διδάσκουσα: Παπαδοπούλου Μαρία
Επίκουρη Καθηγήτρια

Θεσσαλονίκη 2025

Πληκτρολόγιο 4x4

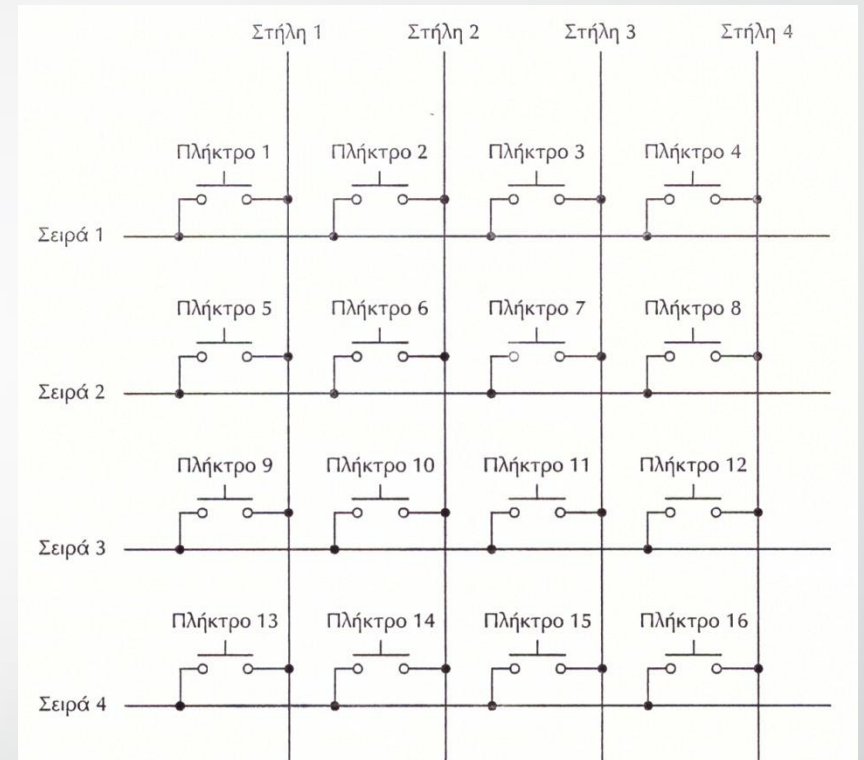
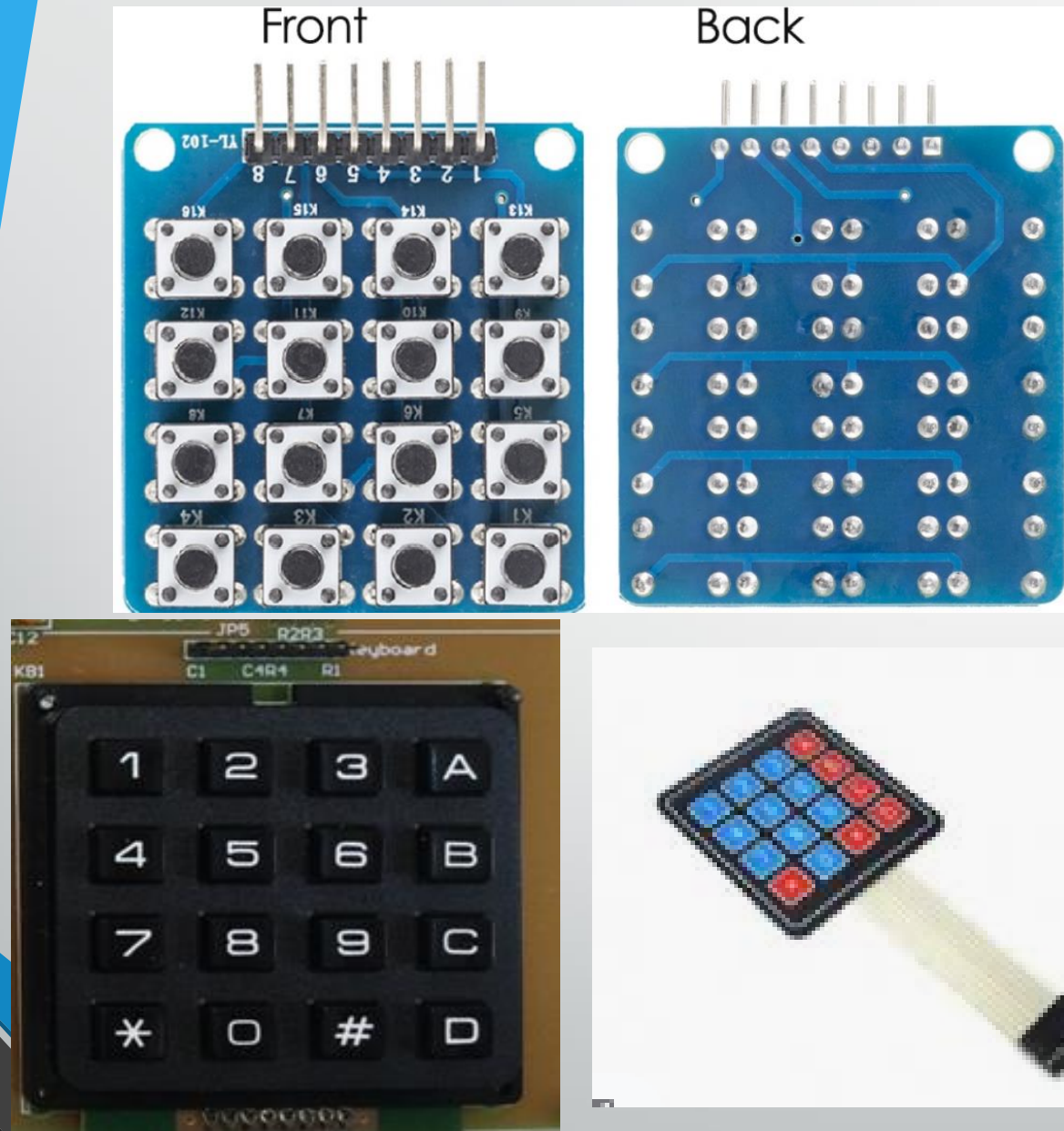


Επικοινωνία μικροελεγκτή με πληκτρολόγιο



Πληκτρολόγια Matrix

Πληκτρολόγιο 4x4

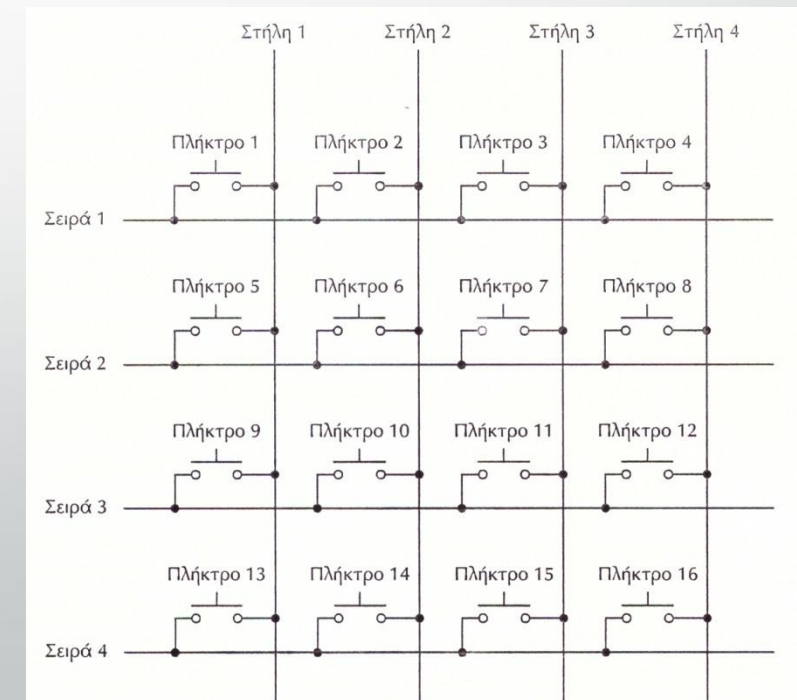


**Υπάρχουν 4 αγωγοί στηλών
και 4 αγωγοί σειρών**

Κάθε μπουτόν συνδέει μια
συγκεκριμένη στήλη με μια
συγκεκριμένη γραμμή

Γιατί δεν χρησιμοποιούμε για κάθε μπουτόν έναν ακροδέκτη του μικροελεγκτή για να ελέγχουμε αν πατιέται το αντίστοιχο μπουτόν;
Δεν θα ήταν ποιο εύκολος ο έλεγχος;

Μέγεθος πληκτρολογίου	Αριθμός απαιτούμενων ακροδεκτών	
	ανεξάρτητο	matrix
3 × 3	9	6
4 × 4	16	8
4 × 6	24	10
5 × 5	25	10
6 × 6	36	12



Τι σημαίνει ανάγνωση του πληκτρολογίου από τον μικροελεγκτή;



- Ανάγνωση του πληκτρολογίου σημαίνει ότι κάθε φορά που πατιέται ένα πλήκτρο ή συνδυασμός πλήκτρων αποδίδεται σε μια μεταβλητή, π.χ. `k`, μια τιμή που υποδεικνύει το ποιο ακριβώς πλήκτρο πατήθηκε.
- Η τιμή αυτής της μεταβλητής μας δείχνει ποιο πλήκτρο ή ποιος συνδυασμός πλήκτρων πατήθηκε

Παράδειγμα:

Όταν πατηθεί το πλήκτρο 0 (μηδέν) η μεταβλητή `k` παίρνει την τιμή `(48)d`

Το ότι η μεταβλητή `k` πήρε την τιμή `(48)d` υποδεικνύει ότι πατήθηκε το πλήκτρο του μηδενός

ΕΡΩΤΗΜΑΤΑ

1. Στο πληκτρολόγιο 4x4 τι τιμή αποδίδεται στη μεταβλητή **k** όταν διαβαστεί ότι ένα πλήκτρο πατήθηκε;
2. Τι τιμή αποδίδεται στη μεταβλητή **k** όταν δεν πατηθεί κάποιο πλήκτρο;

1. Όταν πατηθεί ένα πλήκτρο αποδίδεται στη μεταβλητή **k** ο ASCII κώδικας του πλήκτρου που πατήθηκε.



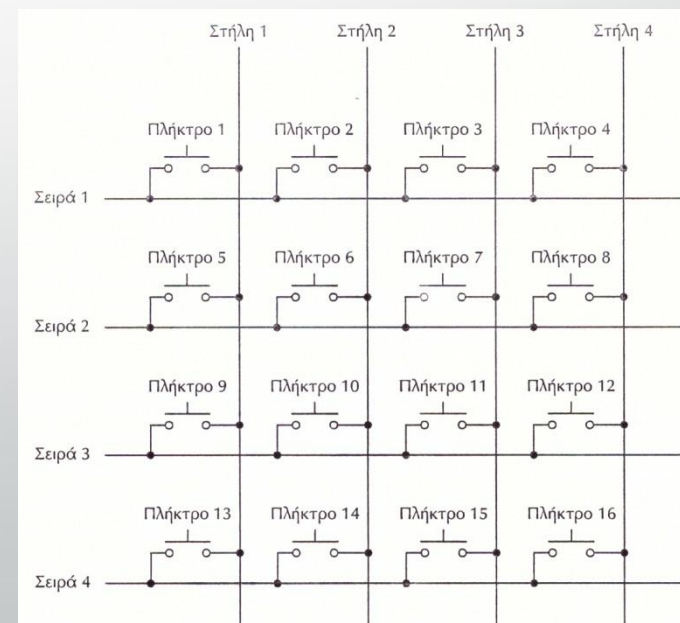
Παράδειγμα

Πατήθηκε το επάνω δεξιό πλήκτρο.

$k = 0x41$ ($0x41$ είναι ο ASCII κώδικας του **A**)

Πατήθηκε το επάνω αριστερό πλήκτρο

$k = 0x31$ ($0x31$ είναι ο ASCII κώδικας του **1**)

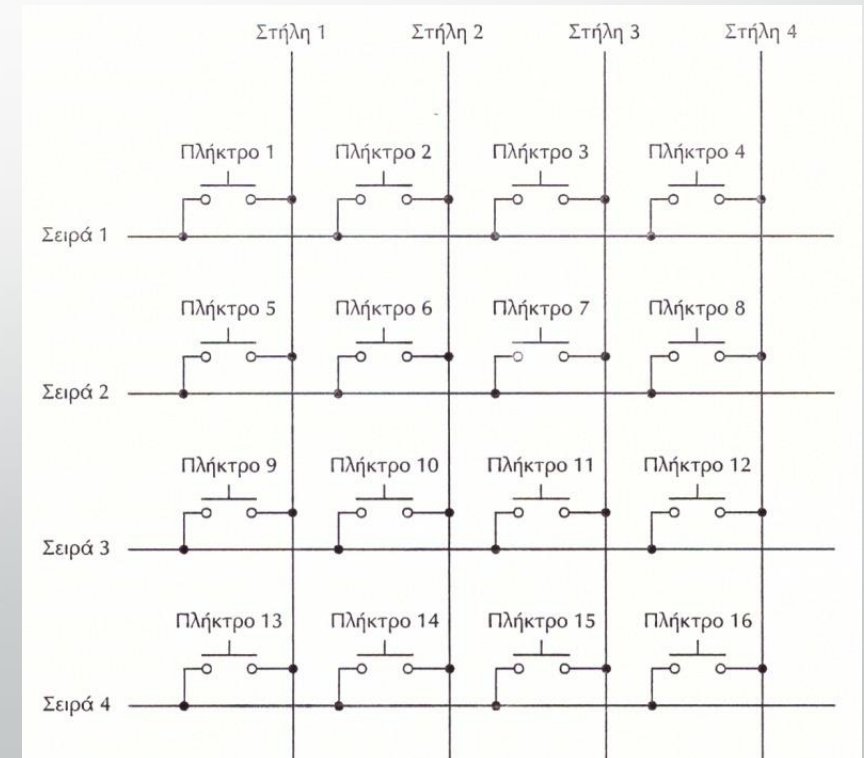


Τι τιμή αποδίδεται στη μεταβλητή `k` όταν δεν πατηθεί κάποιο πλήκτρο;

2. Όταν δεν πατηθεί κανένα πλήκτρο επιλέγουμε να αποδίδεται στη μεταβλητή `k` η τιμή **μηδέν**

Επομένως για να διαβαστεί το πληκτρολόγιο θα πρέπει να κατασκευαστεί μια συνάρτηση, π.χ. με όνομα `k=kbd_getc()` η οποία κάνει τα εξής:

1. Όταν πατηθεί κάποιο πλήκτρο αποδίδει στη μεταβλητή `k` τον ASCII κώδικα του πλήκτρου που πατήθηκε.
2. Όταν δεν πατηθεί κάποιο πλήκτρο αποδίδει στη μεταβλητή `k` την τιμή **0**.



Βασικές αρχές χρήσης της keypad.h

1. Το αρχείο keypad.h το βάζουμε στον ίδιο φάκελο όπου έχουμε το πρόγραμμά μας
2. Την κάνουμε include στην αρχή του προγράμματος μας
`include<keypad.h>`
3. Μέσα στη main καλούμε την `kbd_init()`; // **Συνάρτηση αρχικοποίησης**
4. Ανοίγουμε το αρχείο keypad.h (π.χ. με το notepad) και ελέγχουμε/τροποποιούμε τον τρόπο σύνδεσης των ακροδεκτών του πληκτρολογίου στους ακροδέκτες του μικροελεγκτή.



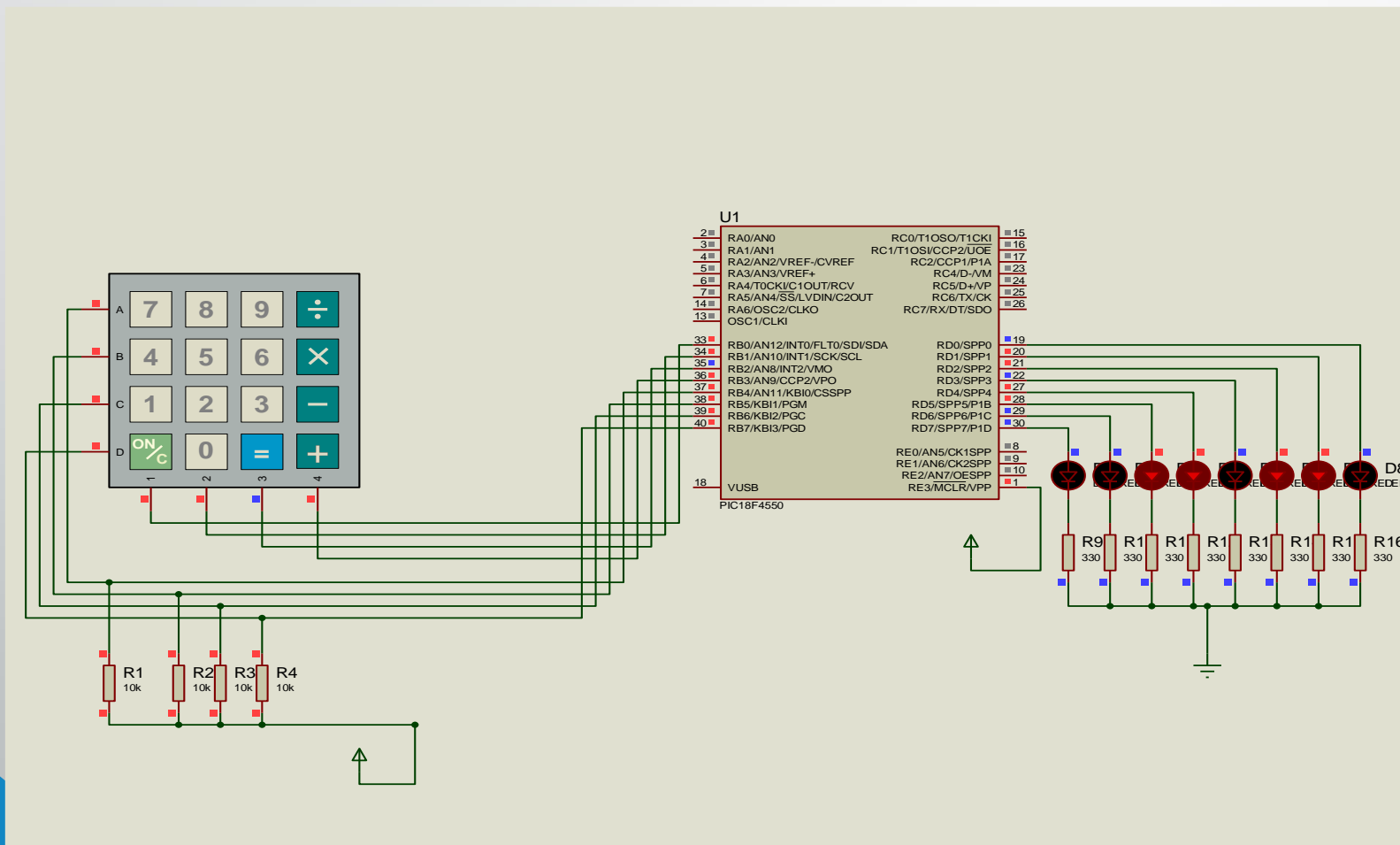
Επίσης τοποθετούμε σε έναν πίνακα τους ASCII κώδικες των συμβόλων του πληκτρολογίου

Συνάρτηση: `k=kbd_getc()`; // Η μεταβλητή `k` είναι `char(8 bit)`

1. Όταν πατηθεί κάποιο πλήκτρο αποδίδεται στη μεταβλητή `k` ο **ASCII κώδικας** του πλήκτρου που πατήθηκε.
2. Όταν δεν πατηθεί κάποιο πλήκτρο αποδίδεται στη μεταβλητή `k` η τιμή **0**.

Άσκηση 9α. Εισαγωγή δεδομένων από το πληκτρολόγιο

Να γραφεί πρόγραμμα με το οποίο κάθε φορά που πατιέται ένα πλήκτρο στο πληκτρολόγιο 4x4, ο ASCII κώδικας του πλήκτρου που πατήθηκε εμφανίζεται στα 8 LED που συνδέονται στην πόρτα D του μικροελεγκτή. Να χρησιμοποιηθεί ο οδηγός (driver) keypad.h



Προσοχή!

Στην πόρτα D εμφανίζεται η τιμή $(0011\ 0110)_b = 0x36$ δηλαδή ο ASCII κώδικας του 6.
Επομένως το πλήκτρο που πατήθηκε είναι το 6

Άσκηση 9α. Εισαγωγή δεδομένων από το πληκτρολόγιο

Εμφάνιση του ASCII κώδικα του πλήκτρου που πατήθηκε στην πόρτα D

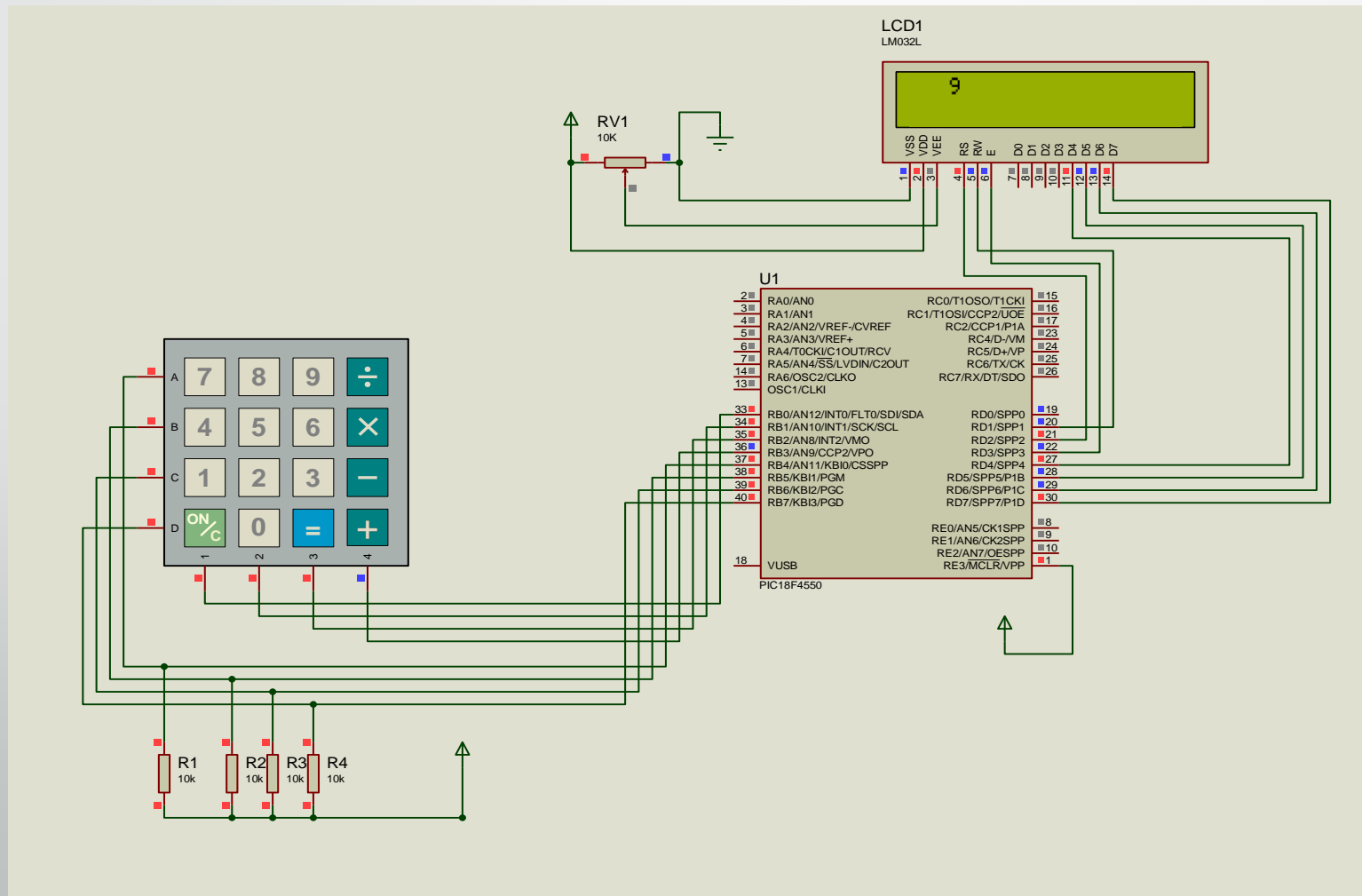
```
#include<main.h>
#include<keypad.h> //για proteus
#define PORTB=0xf81 //υπάρχει παραλλαγή.
#define PORTD=0xf83 // της keypad
//Ορισμοί συναρτήσεων
void init(void);
//Κύριο πρόγραμμα
void main() {
    char k; // Μεταβλητή k για αποθήκευση του ASCII κώδικα του πλήκτρου που πατήθηκε
    init(); // Ρουτίνα αρχικοποίησης {set_tris_d(0x00); PORTD==0}. Τη γράφουμε μετά την main()
    kbd_init(); // Εκτέλεση της ρουτίνας αρχικών ρυθμίσεων του μικροελεγκτή

    while(TRUE) {
        k=kbd_getc(); // Ανάγνωση του πληκτρολογίου. Μεταφέρεται στη μεταβλητή k
                       // ο ASCII κώδικας του πλήκτρου που πατήθηκε.
                       // Αν δεν πατηθεί κανένα πλήκτρο η μεταβλητή k παίρνει την τιμή 0
        if(k!=0) { // Αν πατηθεί κάποιο πλήκτρο, δηλαδή αν k≠0 εκτελούνται όσα είναι
                  // μέσα στις αγκύλες
            PORTD=k; // Μεταφέρεται στην πόρτα D ο ASCII κώδικας του πλήκτρου που
                    // πατήθηκε, δηλαδή η τιμή της μεταβλητής k.

        }
    } // Κλείνει η while
} // Κλείνει η main
```

Άσκηση 9b. Εισαγωγή δεδομένων από το πληκτρολόγιο

Να γραφεί πρόγραμμα με το οποίο ο μικροελεγκτής 18F4550, ο οποίος είναι συνδεδεμένος σε πληκτρολόγιο 4x4 και σε οθόνη LCD 2 γραμμών, κάθε φορά που πατιέται ένα πλήκτρο εμφανίζει το πλήκτρο που πατήθηκες στην 5^η θέση της πρώτης γραμμής της οθόνης LCD.



Άσκηση 9b. Εισαγωγή δεδομένων από το πληκτρολόγιο.

-----Πάνω από την main() { } -----

Πρόγραμμα

```
#include <main.h>
```

```
#include <keypad.h>   // πρόγραμμα οδήγησης (driver) για ανάγνωση από πληκτρολόγιο 4X4  
                      // Προσοχή!!! Η πόρτα B χρησιμοποιείται για το  
                      // πληκτρολόγιο. Να ελεγχθούν οι ορισμοί των συνδέσεων  
                      // στο αρχείο keypad.h. Ανοίξτε το αρχείο keypad.h με το notepad και  
                      // διορθώστε τους ορισμούς των συνδέσεων σε περίπτωση που δεν  
                      // ανταποκρίνονται στο σχέδιο.
```

```
#include <flex_lcd.h>   // πρόγραμμα οδήγησης (driver) για εμφάνιση σε οθόνη LCD δύο γραμμών  
                      // Προσοχή!!! Η πόρτα D χρησιμοποιείται για την οδήγηση της  
                      // οθόνης. Να ελεγχθούν οι ορισμοί των συνδέσεων  
                      // στο αρχείο flex_lcd.h. Ανοίξτε το μ αρχείο flex_lcd.h με το notepad και  
                      // διορθώστε τους ορισμούς των συνδέσεων σε περίπτωση που δεν  
                      // ανταποκρίνονται στο σχέδιο.
```

```
#byte PORTB=0xf81   // θέση του καταχωρητή δεδομένων της πόρτας B  
                    // στην μνήμη δεδομένων του μικρολεγκτή PIC 18F4550
```

```
#byte PORTD=0xf83   // θέση του καταχωρητή δεδομένων  
                    // της πόρτας D
```

```
// Δήλωση συνάρτησης αρχικοποίησης
```

```
void init(void);
```

Άσκηση 9b. Εισαγωγή δεδομένων από το πληκτρολόγιο.

//Κύριο πρόγραμμα

void main()

{

char k; // μεταβλητή char για αποθήκευση του ASCII κώδικα του πληκτρο
// που πατήθηκε

init(); // αρχικοποίηση, η συνάρτηση αρχικοποίησης έχει γραφεί στο τέλος

lcd_init(); //αρχικοποίηση της οθόνης lcd.

kbd_init(); // αρχικοποίηση της συνάρτησης ανάγνωσης από το πληκτρολόγιο

while(TRUE) { // εκτελείται συνεχώς ένας βρόχος

k=kbd_getc(); //σε κάθε εκτέλεση του βρόχου διαβάζεται ο ASCII κώδικας
// του πλήκτρου που πατήθηκε. Αν δεν πατήθηκε κανένα
// πλήκτρο επιστρέφεται στην μεταβλητή k η τιμή 0

if (k!=0) { //αν πατήθηκε κάποιο πλήκτρο(k!=0) εκτελείται αυτό που είναι
// μέσα στις αγκύλες

lcd_gotoxy(4,1); // ετοιμασία της οθόνης για εγγραφή στην
// 5^η θέση της 1^{ης} γραμμής

lcd_putc(k); // αποστέλλεται στην 5^η θέση της 1^{ης} γραμμής
// της οθόνης ο ASCII κώδικας του πλήκτρου
// που πατήθηκε. Εμφανίζεται το σύμβολο που
//αντιστοιχεί στον ASCII κώδικα του πλήκτρου που
//πατήθηκε

}

//κλείνει η αγκύλη του if

} //κλείνει η αγκύλη του while(TRUE)

// κλείνει η αγκύλη της main

}

Άσκηση 9b. Εισαγωγή δεδομένων από το πληκτρολόγιο.

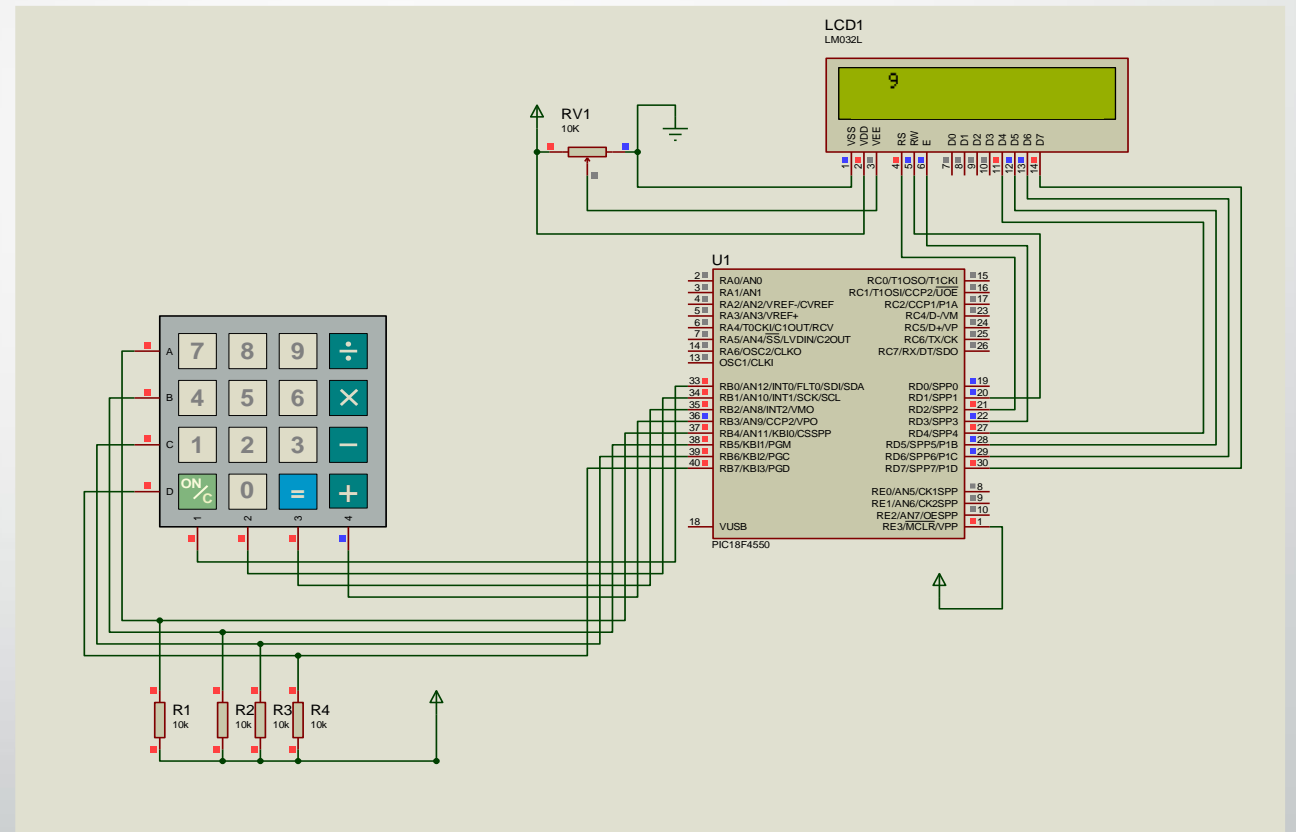
```
----- init() { } -----  
// Κώδικας της συνάρτησης αρχικοποίησης  
void init(void){  
    set_tris_d(0x00); // Η πόρτα D γίνεται έξοδος, χρησιμοποιείται  
                      // για την οδήγηση της οθόνης LCD  
}
```

Ερώτηση:

Πού καθορίζεται η ρύθμιση της πόρτας B, δηλαδή ότι η μισή είναι έξοδος και η μισή είσοδος;

Απάντηση:

Η ρύθμιση αυτή είναι μέσα στον κώδικα της keypad.h και δεν χρειάζεται να την ξαναβάλουμε στην init(){ }



Πώς γνωρίζει η keypad.h τι σύμβολο έχουμε σχεδιάσει πάνω σε κάθε πλήκτρο του πληκτρολογίου 4x4;

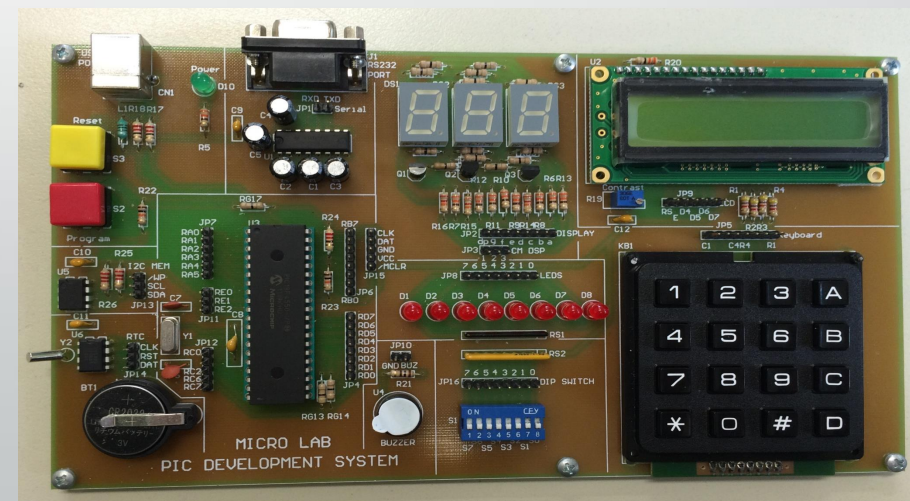
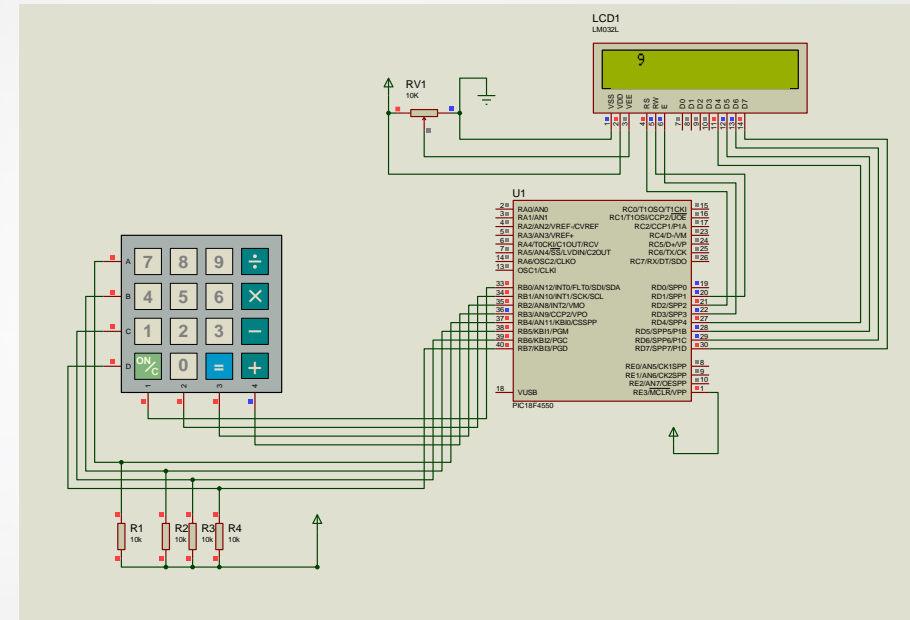
Δεν το γνωρίζει, εμείς το δηλώνουμε μέσα στον κώδικα της keypad.h

//Keypad connection:

```
#define row0 PIN_B4 // Προσοχή!!!  
#define row1 PIN_B5 // Δηλώνουμε τον ακροδέκτη  
#define row2 PIN_B6 // στον οποίο συνδέεται  
#define row3 PIN_B7 // η κάθε γραμμή(row) και  
#define col0 PIN_B0 // η κάθε στήλη(column)  
#define col1 PIN_B1 // του πληκτρολογίου  
#define col2 PIN_B2 //  
#define col3 PIN_B3 //
```

//----- Variable definition-----

```
char const KEYS[4][4] = // Προσοχή!!!  
{'1','2','3','A'}, // Σε αυτόν τον πίνακα γράφουμε  
{'4','5','6','B'}, // τους ASCII κώδικες των χαρακτήρων  
{'7','8','9','C'}, // που σημειώθηκαν πάνω σε κάθε  
{'*','0','#','D'}; // πλήκτρο
```



Σε ποιο σημείο της keypad.h δηλώνεται ποιοι ακροδέκτες της πόρτας B είναι ακροδέκτες εισόδου και ποιοι είναι ακροδέκτες εξόδου;

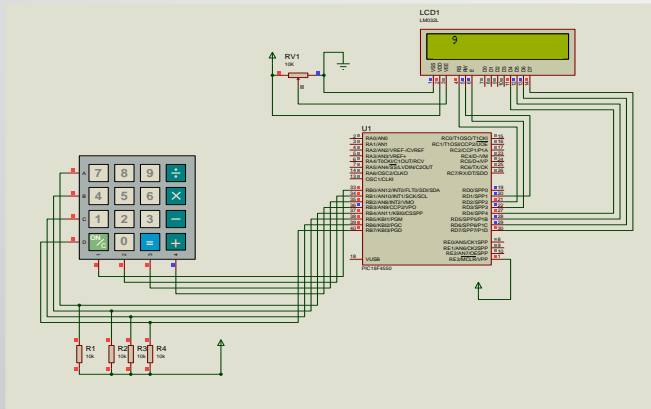
```
void kbd_init(void)
```

```
{
```

```
    set_tris_b(0xF0); // Εδώ !!!!
```

```
    output_b(0x0F);
```

```
}
```



```
//Keypad connection:
```

```
#define row0 PIN_B4 // Προσοχή!!!
```

```
#define row1 PIN_B5 // Δηλώνουμε τον ακροδέκτη
```

```
#define row2 PIN_B6 // στον οποίο συνδέεται
```

```
#define row3 PIN_B7 // η κάθε γραμμή (row) και
```

```
#define col0 PIN_B0 // η κάθε στήλη (column)
```

```
#define col1 PIN_B1 // του πληκτρολογίου
```

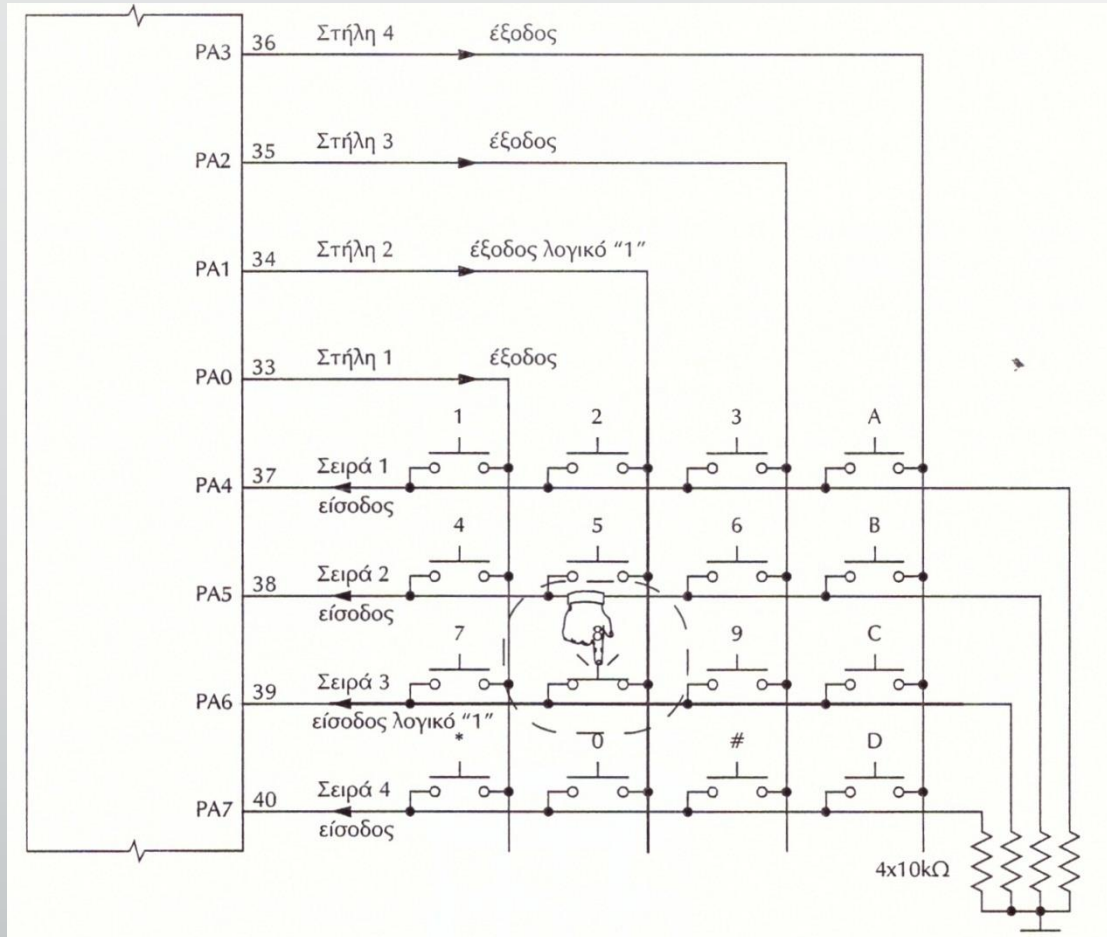
```
#define col2 PIN_B2
```

```
#define col3 PIN_B3
```

```
set_tris_b(0xF0); // 0xF0=0b11110000. Οι ακροδέκτες B4, B5, B6, B7 γίνονται  
// είσοδοι και οι ακροδέκτες B0, B1, B2 και B3 γίνονται  
// έξοδοι
```

Προσοχή!!! Αν χρησιμοποιήσουμε την πόρτα D θα πρέπει να κάνουμε τις σχετικές αλλαγές στην στον πίνακα Keypad connection και να προσαρμόσουμε στην πόρτα D τις εντολές της kbd_init(void).

Πώς λειτουργεί η `k=Keypad.h(1);`



Ακροδέκτης
πληκτρολογίου

C0 C1 C2 C3 R0 R1 R2 R3
↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓

Ακροδέκτης
μικροελεγκτή

A0 A1 A2 A3 A4 A5 A6 A7

Οι ακροδέκτες του πληκτρολογίου

C0 C1 C2 C3 R0 R1 R2 R3

| | | | | | | |



Στήλη 1	Στήλη 2	Στήλη 3	Στήλη 4
1	0	0	0
0	1	0	0
0	0	1	0
0	0	0	1

C0 στο πληκτρολόγιο σημαίνει
Column 0 δηλαδή στήλη 0

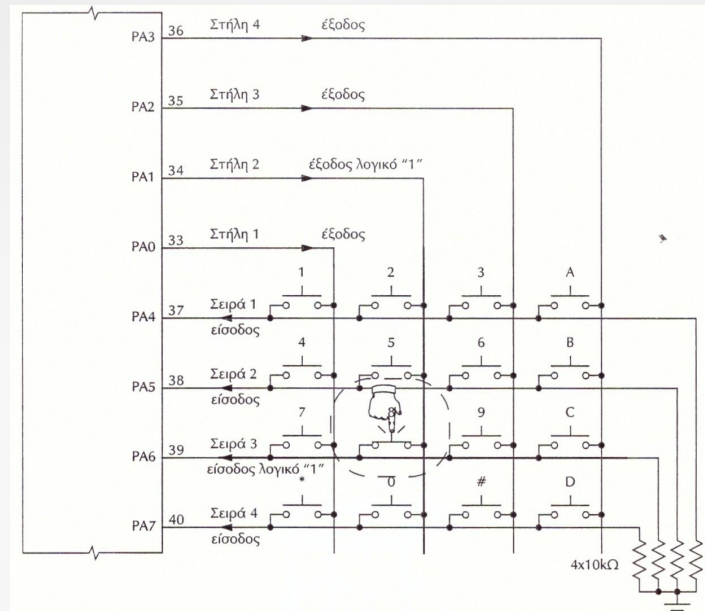
R2 στο πληκτρολόγιο σημαίνει Row 2
δηλαδή γραμμή 2

C0 C1 C2 C3 R0 R1 R2 R3

| | | | | | | |



Πώς λειτουργεί η Keypad.h (2);



Οι στήλες 1, 2, 3, 4 συνδέονται στους ακροδέκτες εξόδου του μικροελεγκτή B0, B1, B2, B3

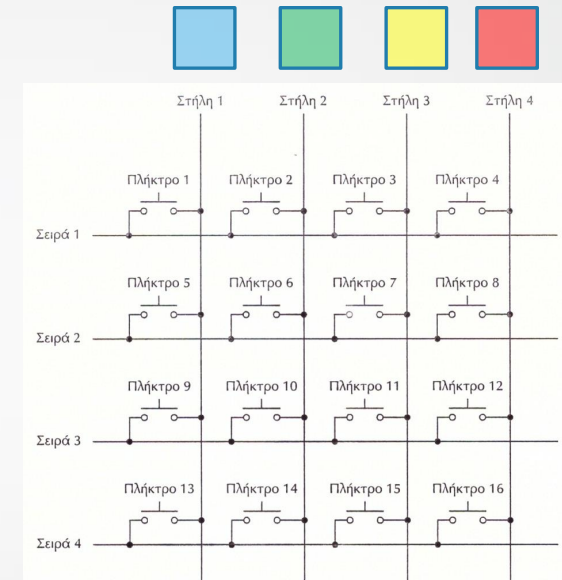
Οι σειρές 1, 2, 3, 4 συνδέονται στους ακροδέκτες εισόδου του μικροελεγκτή B4, B5, B6, B7

Στήλη1 Στήλη2 Στήλη3 Στήλη4 Δίνονται διαδοχικά οι τιμές 1000, 0100, 0010, 0001	Σειρά1 Σειρά2 Σειρά 3 Σειρά 4 Διαβάζονται οι τιμές στις σειρές και ανιχνεύεται το πλήκτρο που πατήθηκε	Πλήκτρο	k= ASCII κώδικας πλήκτρου
1000	0100	Πλήκτρο 5	k='5'
0001	0100	Πλήκτρο B	k='B'
0010	0001	Πλήκτρο #	k='#'

Στέλνουμε διαδοχικά 1 στην πρώτη, δεύτερη, τρίτη, και τέταρτη στήλη και διαβάζουμε σε ποια γραμμή εμφανίζεται. Έτσι ανιχνεύουμε το πλήκτρο που πατήθηκε.

Στήλη1 Στήλη2 Στήλη3 Στήλη4 Δίνονται διαδοχικά οι τιμές 1000, 0100, 0010, 0001	Σειρά1 Σειρά2 Σειρά 3 Σειρά 4 Διαβάζονται οι τιμές στις σειρές και ανιχνεύεται το πλήκτρο που πατήθηκε	Πλήκτ ρο	k= ASCII κώδικας πλήκτρ ου
1000	1000	1	'1'
1000	0100	5	'4'
1000	0010	9	'7'
1000	0001	13	'*'
0100	1000	2	'2'
0100	0100	6	'5'
0100	0010	10	'8'
0100	0001	14	'0'
0010	1000	3	'3'
0010	0100	7	'6'
0010	0010	11	'9'
0010	0001	15	'#'
0001	1000	4	'A'
0001	0100	8	'B'
0001	0010	12	'C'
0001	0001	16	'D'

Όταν δεν πατιέται κανένα πλήκτρο η μεταβλητή k παίρνει την τιμή 0 !!!



`k=kbd_getc();`



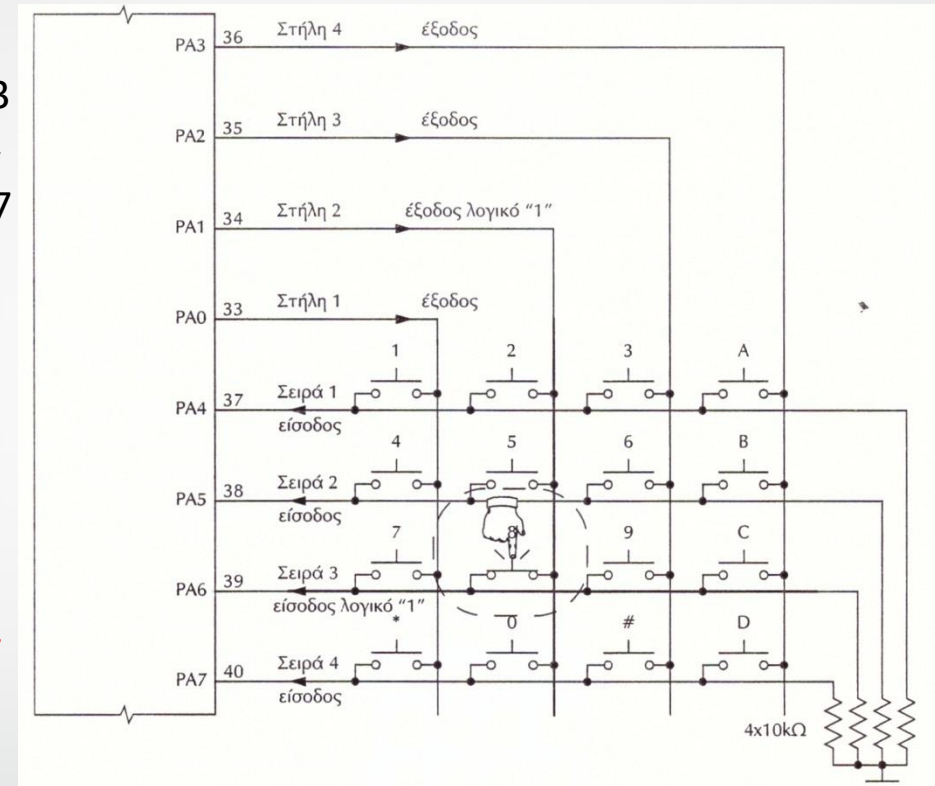
Τρόπος σύνδεσης του πληκτρολογίου 4x4 στο μικροελεγκτή

Ακροδέκτης
πληκτρολογίου C0 C1 C2 C3 R0 R1 R2 R3
↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓
Ακροδέκτης A0 A1 A2 A3 A4 A5 A6 A7
μικροελεγκτή

//Δήλωση μέσα στην keypad.h

```
#define row0 PIN_A4 // Προσοχή!!!  
#define row1 PIN_A5 // Δηλώνουμε τον ακροδέκτη  
#define row2 PIN_A6 // στον οποίο συνδέεται  
#define row3 PIN_A7 // η κάθε γραμμή(row) και  
#define col0 PIN_A0 // η κάθε στήλη(column)  
#define col1 PIN_A1 // του πληκτρολογίου  
#define col2 PIN_A2 //  
#define col3 PIN_A3 //
```

```
void kbd_init(void)  
{  
    set_tris_a(0xF0); //Προσοχή  
    output_a(0x0F);  
}
```



Ακροδέκτες του πληκτρολογίου

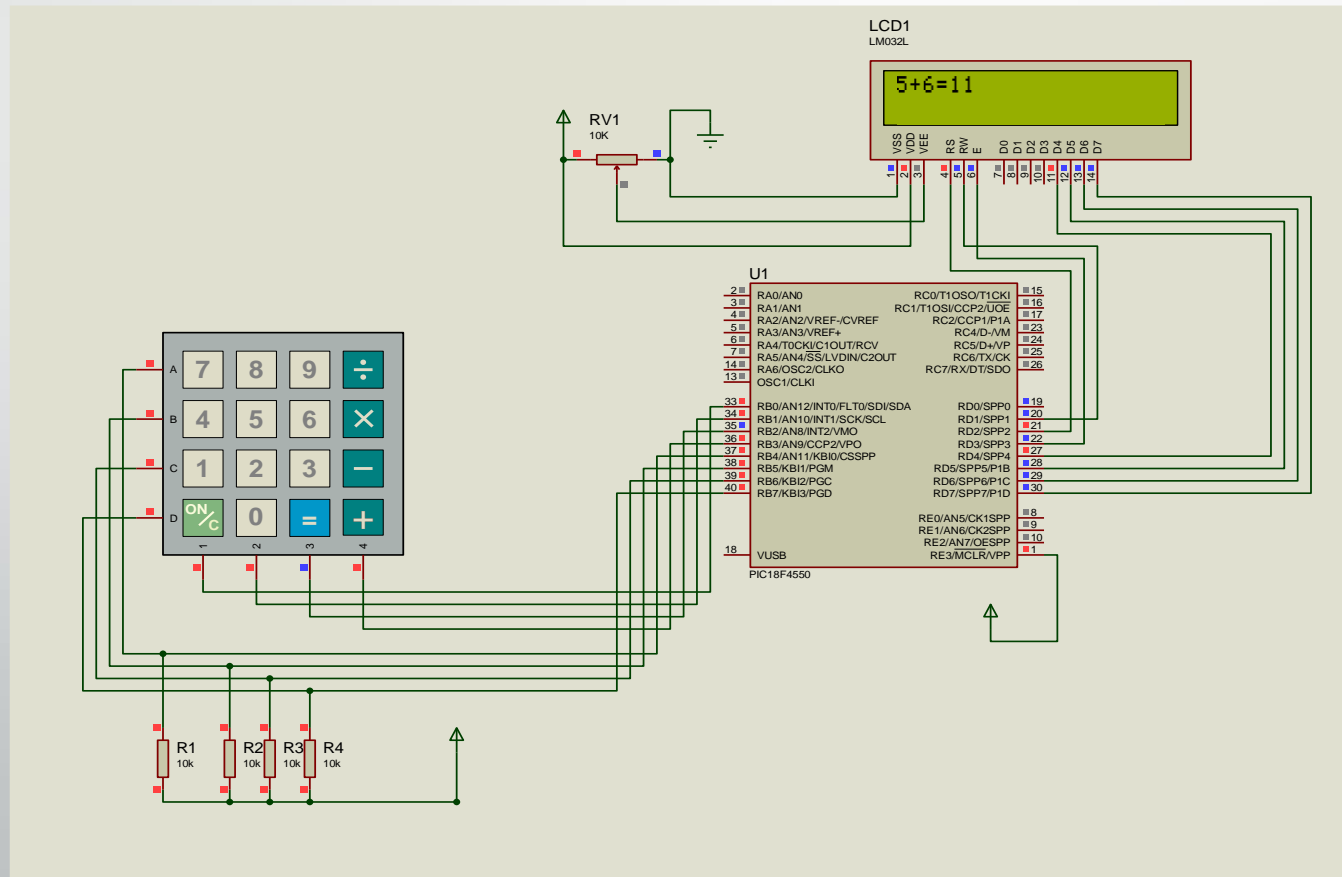
C0	C1	C2	C3	R0	R1	R2	R3

Το C είναι από τη λέξη Column (στήλη)
Το R είναι από τη λέξη Row (γραμμή)

Άσκηση 9c. Εισαγωγή δεδομένων από το πληκτρολόγιο

Να γραφεί πρόγραμμα με το οποίο εισάγονται δύο μονοψήφιοι ακέραιοι αριθμοί από το πληκτρολόγιο και το αποτέλεσμα της άθροισης τους εμφανίζεται στην οθόνη για 2 δευτερόλεπτα. Στη συνέχεια να μπορούν να εισαχθούν δύο νέοι αριθμοί.

Παράδειγμα: Όταν πληκτρολογείται το 5 να εμφανίζεται το 5+, όταν στη συνέχεια πληκτρολογείται το 6 να εμφανίζεται στην οθόνη το $5+6=11$. Μετά από 2 δευτερόλεπτα θα πρέπει να καθαρίζεται η οθόνη και να μπορούν να εισαχθούν δύο νέοι αριθμοί.



Πρόγραμμα άσκησης 9c (Δηλώσεις)

//Πρόγραμμα

#include <main.h>

#include <keypad.h> // πρόγραμμα ανάγνωσης από πληκτρολόγιο 4X4

//Προσοχή!!! Η πόρτα B χρησιμοποιείται για το
//πληκτρολόγιο. Να ελεγχθούν οι ορισμοί των συνδέσεων
// στο αρχείο keypad.h. Ανοίξτε το με το notepad και
//διορθώστε τους ορισμούς των συνδέσεων αν δεν
//ανταποκρίνονται στο σχέδιο.

#include <flex_lcd.h> //πρόγραμμα εμφάνισης σε οθόνη LCD δύο γραμμών

//Προσοχή!!! Η πόρτα D χρησιμοποιείται για την οδήγηση

της

//οθόνης. Να ελεγχθούν οι ορισμοί των συνδέσεων
// στο αρχείο flex_lcd.h. Ανοίξτε το με το notepad και
//διορθώστε τους ορισμούς των συνδέσεων αν δεν
//ανταποκρίνονται στο σχέδιο.

#byte PORTB=0xf81 //θέση του καταχωρητή δεδομένων της πόρτας B

#byte PORTD=0xf83 // θέση του καταχωρητή δεδομένων της πόρτας D

// Δήλωση μεταβλητών

char state=1;

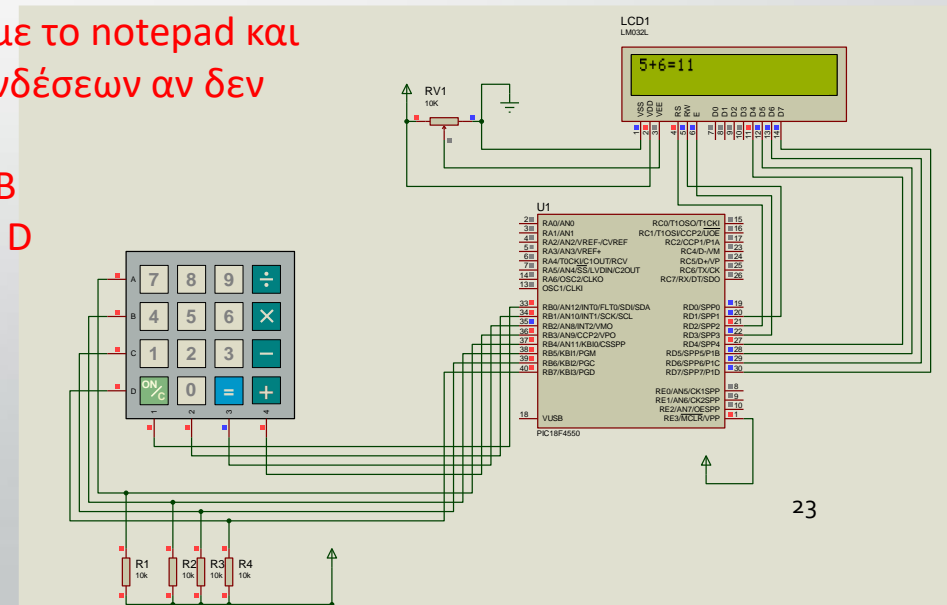
char N1=0;

char N2=0;

char sum=0;

// Δήλωση ρουτινών, συναρτήσεων

void init(void);



Πρόγραμμα άσκησης 9c (Κύριο πρόγραμμα-1)

//Κύριο πρόγραμμα

```
void main()
```

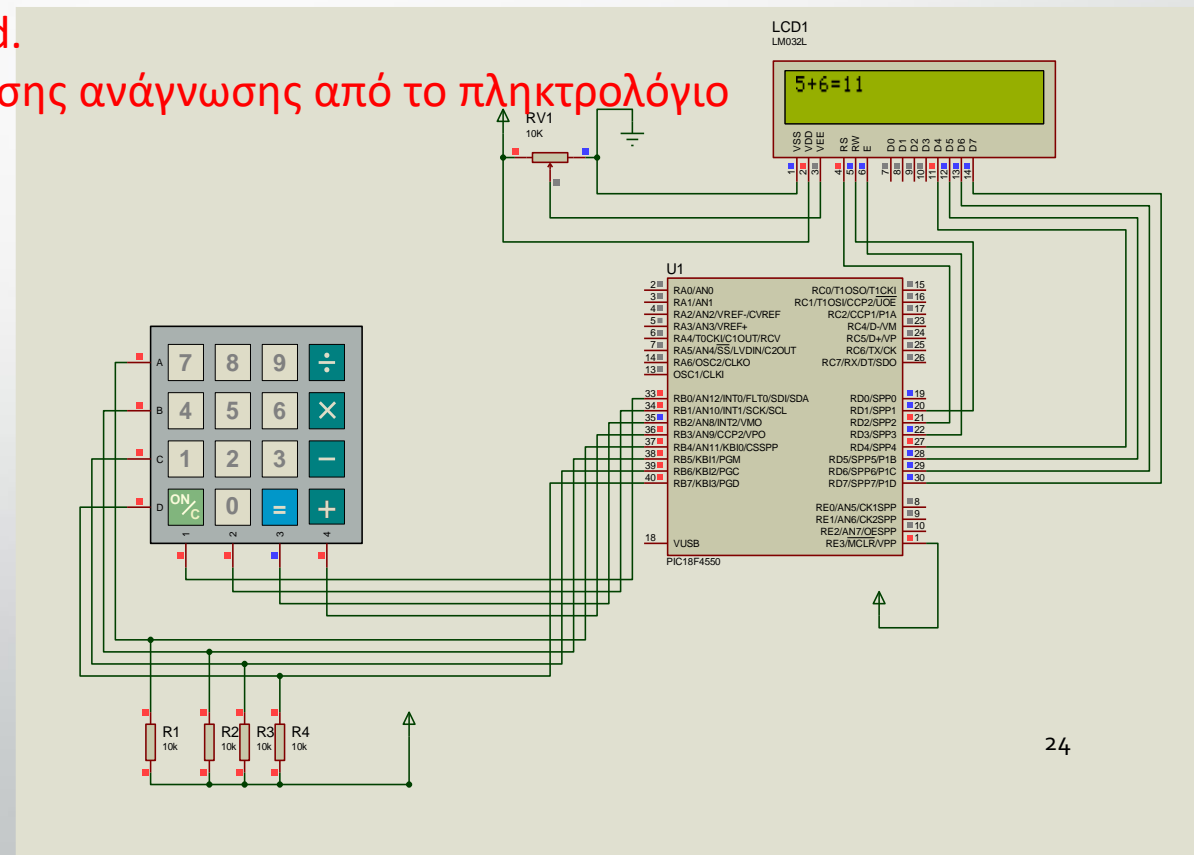
 $\{$

```
char k;    // μεταβλητή char για αποθήκευση του ASCII κώδικα του πληκτρο  
           // που πατήθηκε
```

```
init(); // αρχικοποίηση, η συνάρτηση αρχικοποίησης έχει γραφεί στο τέλος
```

```
lcd_init(); //αρχικοποίηση της οθόνης lcd.
```

```
kbd_init(); // αρχικοποίηση της συνάρτησης ανάγνωσης από το πληκτρολόγιο
```



Πρόγραμμα άσκησης 9c (Κύριο πρόγραμμα-2)

```
while(TRUE) { // εκτελείται συνεχώς ένας βρόχος

    switch(state){
        case 1:
            k=kbd_getc(); //Στη μεταβλητή k μεταφέρεται ο ASCII κώδικας
                        // του πλήκτρου που πατήθηκε
            while(k!=0){ // Όταν πατηθεί κάποιο πλήκτρο(δηλαδή όταν k≠0).
                        // Προσοχή! Το περιεχόμενο μέσα στις αγκύλες του
                        //while εκτελείται μόνο όταν πατηθεί ένα πλήκτρο
                        //δηλαδή μόνο όταν k!=0.
            N1=k&0b00001111; // Μηδενίζονται τα 4 πρώτα bit του ASCII κώδικα του
                        // πλήκτρου που πατήθηκε, και επομένως η
                        // μεταβλητή N1
                        // λαμβάνει την τιμή του πλήκτρου που πατήθηκε.
                        // Για παράδειγμα αν πατηθεί το πλήκτρο 7 το k παίρνει
                        //την τιμή 0x37 και η μεταβλητή N1 την τιμή 0x07.
            printf(lcd_putc,"%1d+",N1); //Εμφάνιση της μεταβλητής N1 και του συμβόλου +
            state=2; //Μετάβαση στην κατάσταση 2 για εμφάνιση του δεύτερου αριθμού
            break; // έξοδος από το while (k!=0)
        } //κλείσιμο της αγκύλης του switch(state)
        break; // Οι υπόλοιπες εντολές του while(TRUE) δεν εκτελούνται και
            //γίνεται μετάβαση στην αρχή του while(TRUE), δηλαδή στο
            // switch(state)
            //αλλά τώρα έγινε state=2. Οπότε θα γίνει μετάβαση στο case 2:
```

Πρόγραμμα άσκησης 9c (Κύριο πρόγραμμα-3)

case 2 :

```
k=kbd_getc();//Στην μεταβλητή k μεταφέρεται ο ASCII κώδικας
// του πλήκτρου που πατήθηκε
while(k!=0) //Όταν πατηθεί κάποιο πλήκτρο(δηλαδή όταν k≠0).
// Προσοχή! Το περιεχόμενο μέσα στις αγκύλες του
//while εκτελείται μόνο όταν πατηθεί ένα πλήκτρο
//δηλαδή μόνο όταν k!=0.
{
N2=k&0b00001111; // Μηδενίζονται τα 4 πρώτα bit του ASCII κώδικα του
// πλήκτρου που πατήθηκε, και επομένως
// η μεταβλητή N2
// λαμβάνει την τιμή του πλήκτρου που πατήθηκε.
// Για παράδειγμα αν πατηθεί το πλήκτρο 7 το k παίρνει
//την τιμή 0x37 και η μεταβλητή N1 την τιμή 0x07.
printf(lcd_putc,"%1d=",N2); //Εμφάνιση της τιμής του δεύτερου πλήκτρου
// που πατήθηκε και του συμβόλου =
sum=N2+N1; // Υπολογισμός του αθροίσματος των δύο
// αριθμών
printf(lcd_putc,"%1d",sum); // εμφάνιση του αθροίσματος των δύο αριθμών

state=1; // η κατάσταση γίνεται 1 ώστε στη συνέχεια να μπορεί να εισαχθεί
// νέος αριθμός.
delay_ms(2000); //Αναμονή για 2 δευτερόλεπτα
printf(lcd_putc, "\f"); //Καθαρισμός της οθόνης
break; // έξοδος από το while(k!=0)
} //κλείνει το while(k!=0)
} //κλείνει το switch(state)
} //κλείνει η αγκύλη του while(TRUE)
} // κλείνει η αγκύλη της main
```

Πρόγραμμα άσκησης 9c (Ρουτίνα αρχικών ρυθμίσεων)

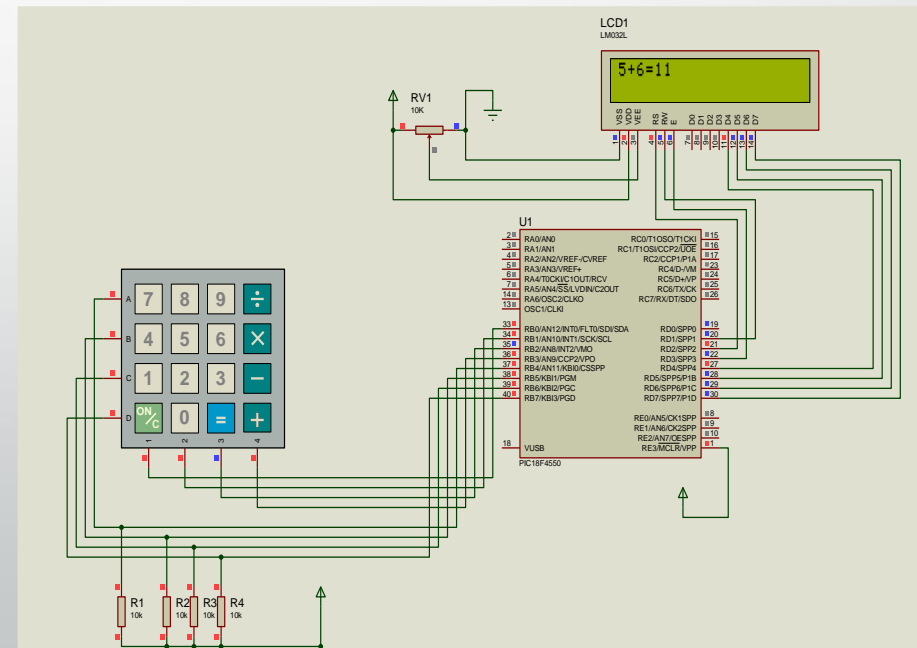
// Κώδικας της συνάρτησης αρχικοποίησης

```
void init(void){  
    set_tris_d(0x00);    // Η πόρτα D γίνεται έξοδος, χρησιμοποιείται για την οδήγηση  
                        // της οθόνης LCD  
}
```

Πού ορίζονται οι ακροδέκτες της πόρτας B σαν ακροδέκτες εισόδου-εξόδου;

Μέσα στην keypad.h Την ανοίγουμε και κάνουμε έλεγχο!!!

```
void kbd_init(void)  
{  
    set_tris_b(0xF0); //Εδώ !!!!  
    output_b(0x0F);  
}
```



Άσκηση 9d. Εισαγωγή δεδομένων από το πληκτρολόγιο

Να γραφεί πρόγραμμα με το οποίο εμφανίζονται στη σειρά οι χαρακτήρες των πλήκτρων που πατιούνται. Όταν πατηθεί το πλήκτρο C να καθαρίζεται η οθόνη και να μπορούν να εμφανιστούν στην οθόνη νέοι χαρακτήρες.

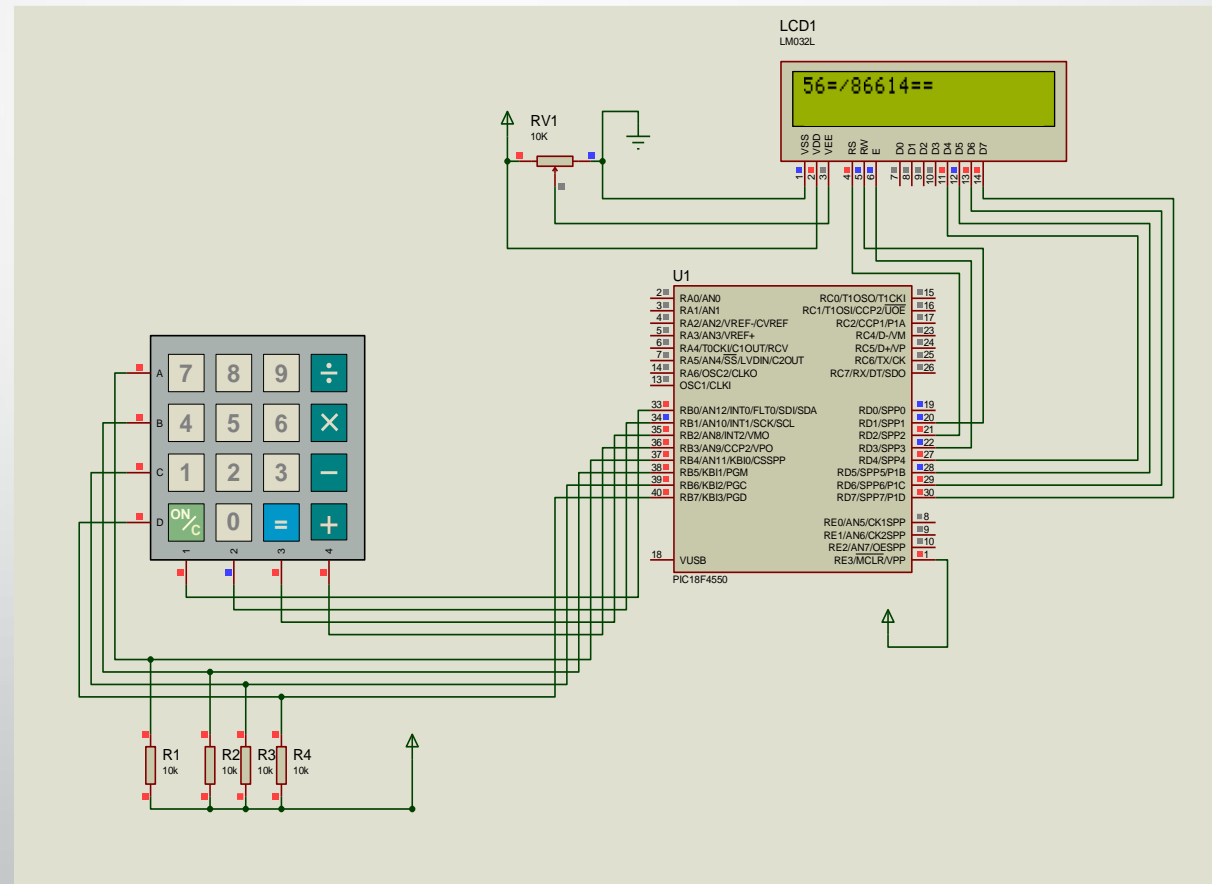
Βοήθημα:

Εντολή για την εμφάνιση στην οθόνη ενός χαρακτήρα του οποίου ο ASCII κώδικας είναι *k*:

```
printf(lcd_putc,"%c", k);
```

Εντολή για τον καθαρισμό της οθόνης:

```
printf(lcd_putc,"\f");
```



Άσκηση 9d. Εισαγωγή δεδομένων από το πληκτρολόγιο (Δηλώσεις)

//Πρόγραμμα

#include <main.h>

#include <keypad.h> // πρόγραμμα ανάγνωσης από πληκτρολόγιο 4X4

//Προσοχή!!! Η πόρτα B χρησιμοποιείται για το
//πληκτρολόγιο. Να ελεγχθούν οι ορισμοί των συνδέσεων
// στο αρχείο keypad.h. Ανοίξτε το με το notepad και
//διορθώστε τους ορισμούς των συνδέσεων αν δεν
//ανταποκρίνονται στο σχέδιο.

#include <flex_lcd.h> //πρόγραμμα εμφάνισης σε οθόνη LCD δύο γραμμών

//Προσοχή!!! Η πόρτα D χρησιμοποιείται για την οδήγηση της
//οθόνης. Να ελεγχθούν οι ορισμοί των συνδέσεων
// στο αρχείο flex_lcd.h. Ανοίξτε το με το notepad και
//διορθώστε τους ορισμούς των συνδέσεων αν δεν
//ανταποκρίνονται στο σχέδιο.

#byte PORTB=0xf81 //θέση του καταχωρητή δεδομένων της πόρτας B

#byte PORTD=0xf83 // θέση του καταχωρητή δεδομένων της πόρτας D

// Δήλωση global μεταβλητών(Σ' αυτό το πρόγραμμα δεν χρειάζονται)

// Δήλωση ρουτινών, συναρτήσεων

void init(void);

//Κύριο πρόγραμμα

.....

Άσκηση 9d. Εισαγωγή δεδομένων από το πληκτρολόγιο (Κύριο πρόγραμμα)

```
//Κύριο πρόγραμμα
void main()
{
while(TRUE) {      // Βρόχος που εκτελείται συνεχώς
    k=kbd_getc();    // μεταφέρεται στην μεταβλητή k ο ASCII κώδικας του
                    // πλήκτρου που πατήθηκε

    if (k!=0 && k!='C') {printf(lcd_putc,"%c",k);} //Εάν πατηθεί ένα πλήκτρο(k!=0) και εάν το
                                                //πλήκτρο που πατήθηκε δεν είναι το C
                                                //να εμφανιστεί στην οθόνη ο χαρακτήρας
                                                //του οποίου ο ASCII κώδικας είναι k.
    else if (k=='C') {printf(lcd_putc,"\f");}    // Εάν πατηθεί το πλήκτρο C να καθαριστεί η οθόνη

} // Κλείσιμο της αγκύλης του while(TRUE)
} // κλείνει η αγκύλη της main

// Κώδικας της συνάρτησης αρχικοποίησης
void init(void){
set_tris_d(0x00); // Η πόρτα D γίνεται έξοδος,
                  // χρησιμοποιείται για την οδήγηση της οθόνης LCD
}
```