



# ***Ενσωματωμένα Συστήματα***

(6<sup>ο</sup> εξάμηνο)

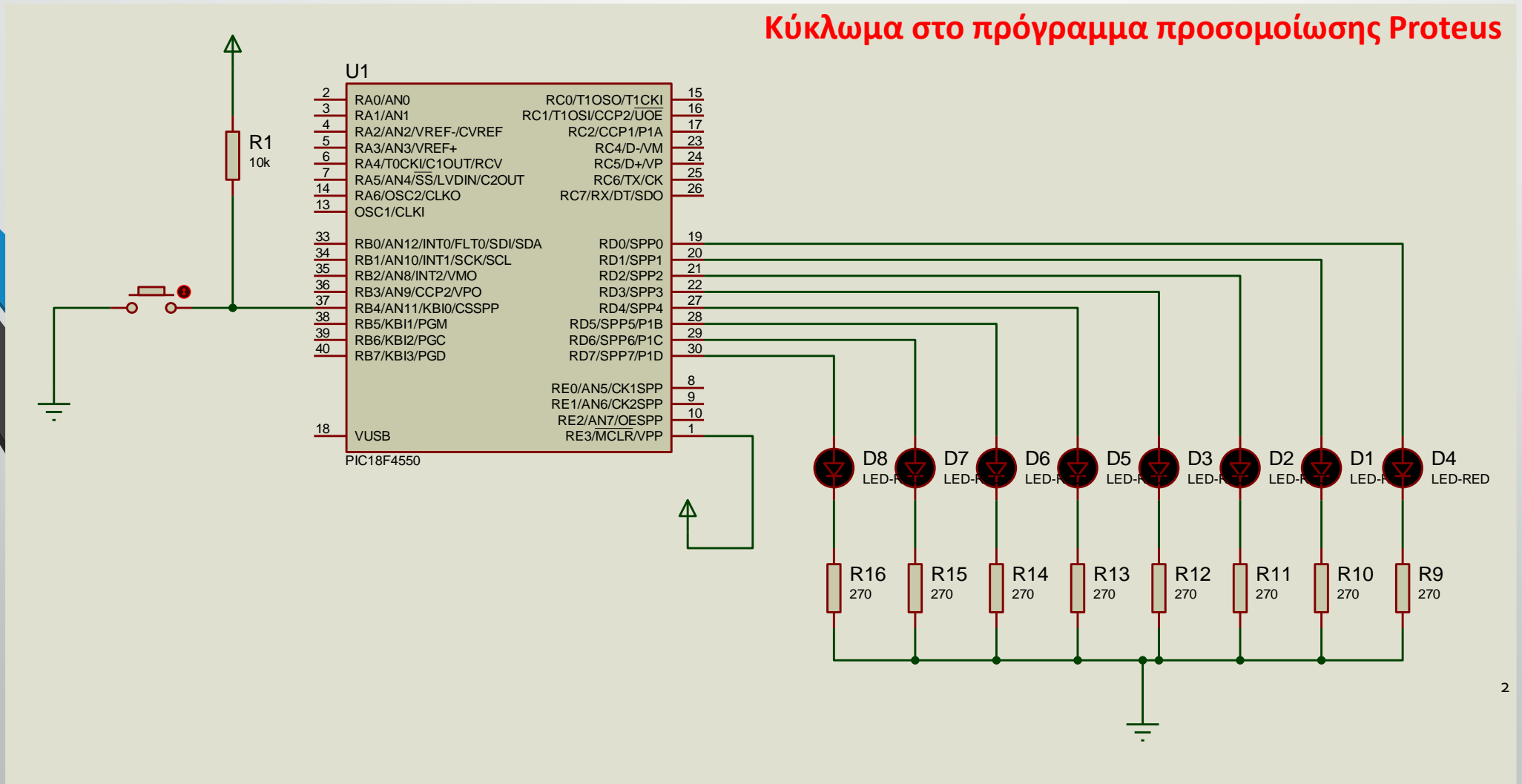
**05-Έλεγχος push-button**

**Διδάσκουσα:** Παπαδοπούλου Μαρία  
Επίκουρη Καθηγήτρια

Θεσσαλονίκη 2025

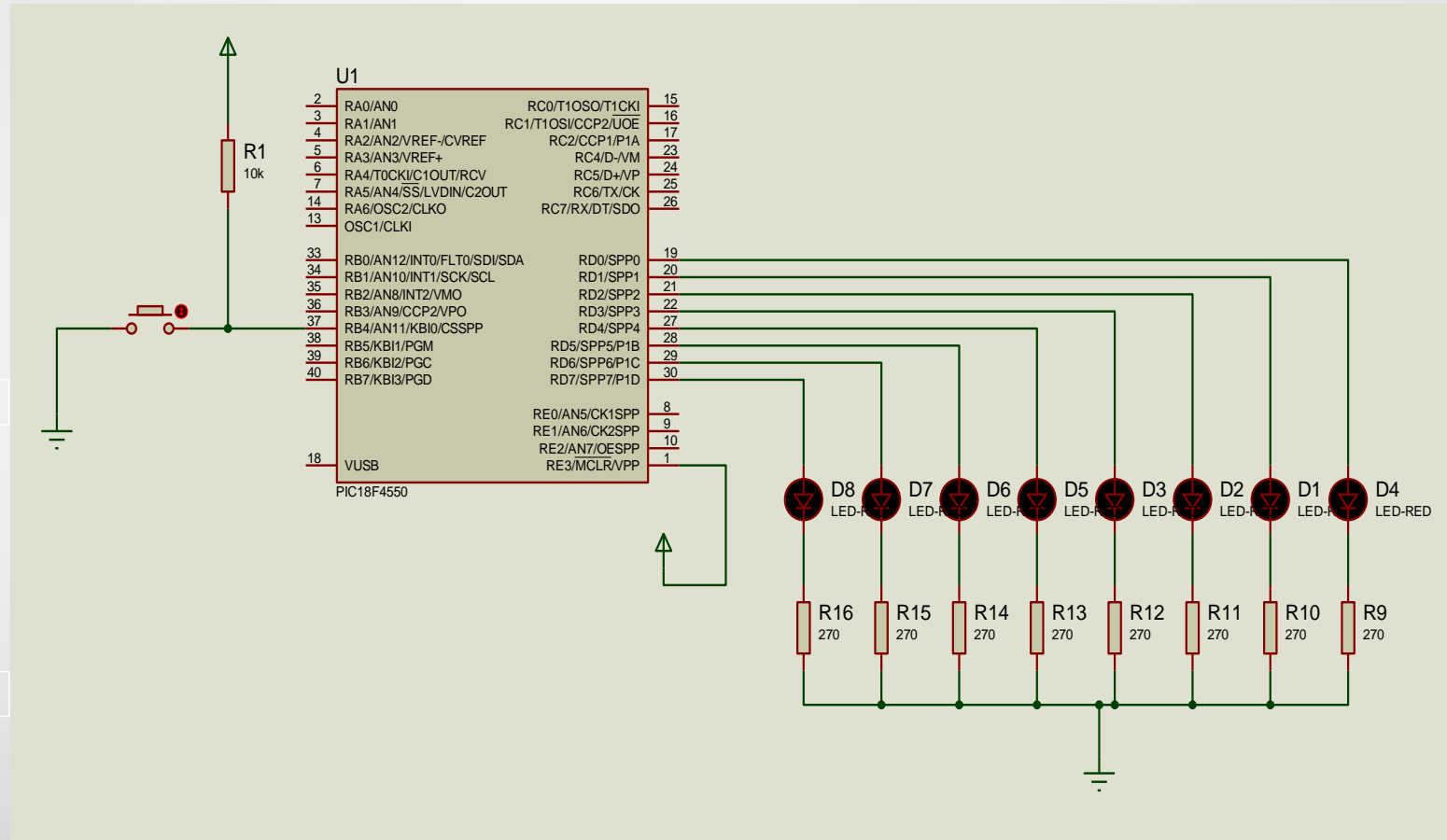
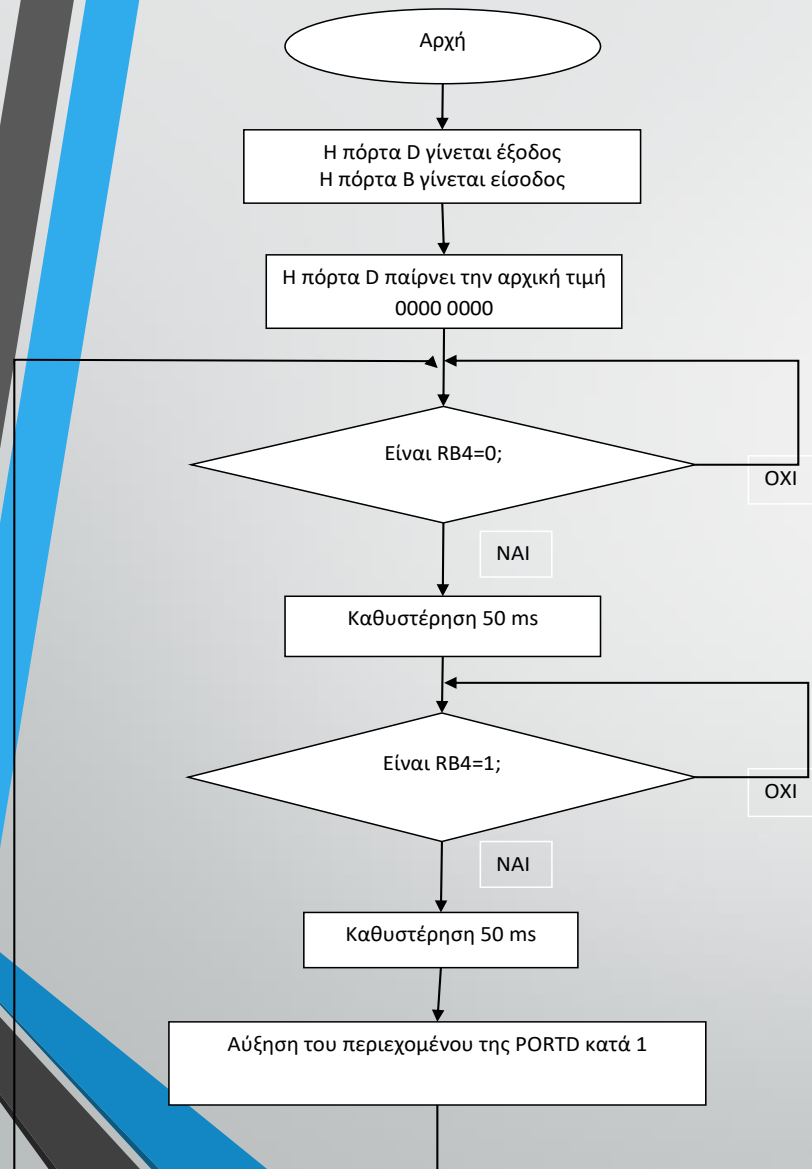
## Άσκηση 2α. Ανάγνωση πατήματος μπουτόν

Να γραφεί πρόγραμμα με το οποίο κάθε φορά που πατάμε ένα μπουτόν, το οποίο είναι συνδεδεμένο στον ακροδέκτη RB4 της πόρτας B, το περιεχόμενο της πόρτας D αυξάνεται κατά 1. Το αποτέλεσμα της αύξησης να φαίνεται σε 8 LED, τα οποία είναι συνδεδεμένα στην πόρτα D του μικροελεγκτή.



## Άσκηση 2α. Ανάγνωση πατήματος μπουτόν

### Διάγραμμα ροής



## Άσκηση 2α. Ανάγνωση πατήματος μπουτόν

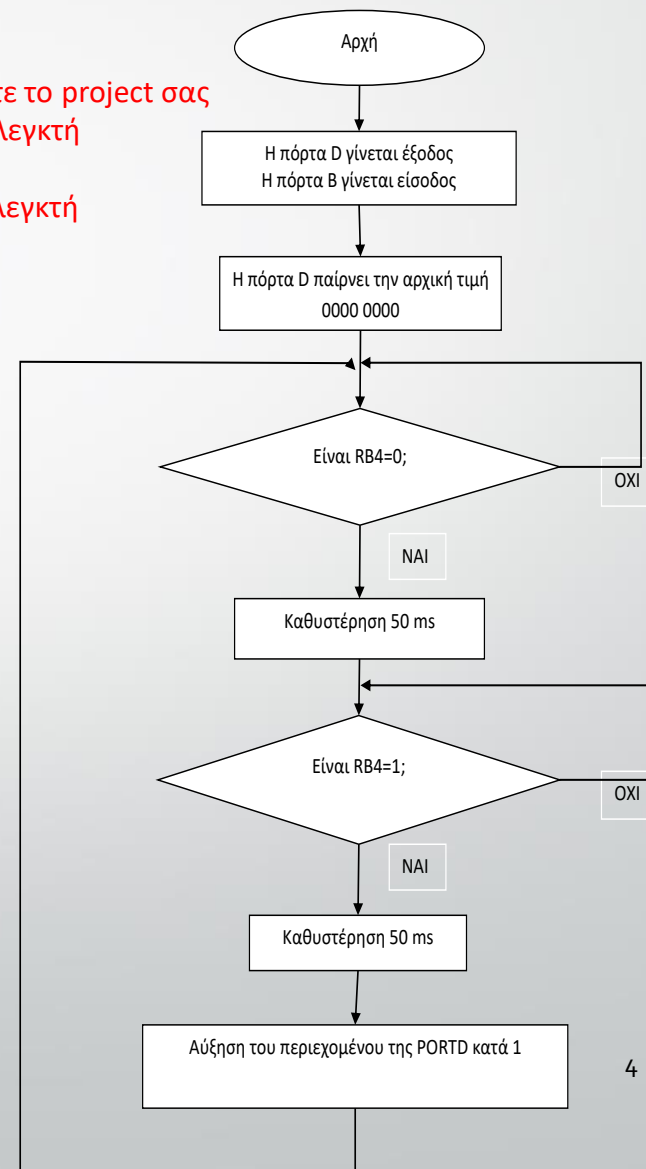
### Πρόγραμμα σε γλώσσα C

**Προσοχή!** Στον φάκελο στον οποίο θα αναπτύξετε το project θα πρέπει να τοποθετήσετε τα αρχεία `main.h` και `18f4550.h` και `add.txt`.

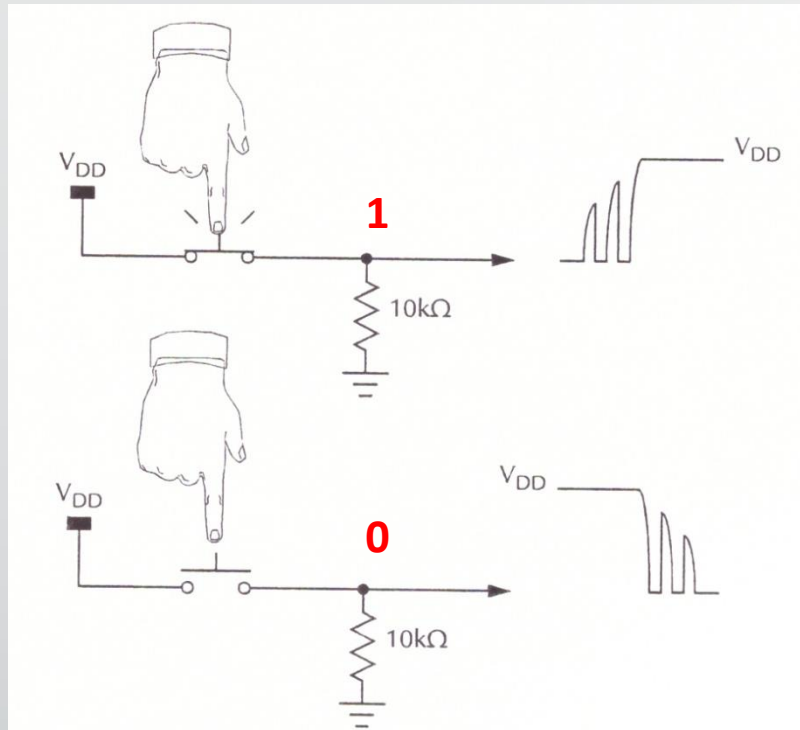
```
#include<main.h>          //Το αρχείο <main.h> περιέχει αρχικές ρυθμίσεις
                           //Πρέπει να τοποθετηθεί οπουδήποτε στον ίδιο φάκελο στον οποίο θα αναπτύξετε το project σας
#define PORTD=0xF83        //F83 είναι η θέση τη καταχωρητή δεδομένων της πόρτας D στη μνήμη του μικροελεγκτή
                           // Δίνουμε στη διεύθυνση 0xF83 το όνομα PORTD
#define PORTB=0xF81        //F81 είναι η θέση τη καταχωρητή δεδομένων της πόρτας D στη μνήμη του μικροελεγκτή
                           // Δίνουμε στη διεύθυνση 0xF81 το όνομα PORTB
// *****Από εδώ αρχίζει το κύριο πρόγραμμα*****
void main()
{
    //άνοιγμα αγκύλης της συνάρτησης main
    set_tris_d(0x00);        //Η θύρα D γίνεται έξοδος (καταχωρητής κατεύθυνσης=0000 0000)
    set_tris_b(0xFF);        //Η θύρα B γίνεται είσοδος (καταχωρητής κατεύθυνσης=1111 1111)
    PORTD=0x00;              //Στην πόρτα D δίνεται η αρχική τιμή 00000000
    // Με την παρακάτω δομή while(TRUE){ } εκτελείται αενάως (για πάντα) το σύνολο των
    // εντολών που είναι μέσα στα άγκιστρα. Η δεσμευμένη λέξη TRUE στη γλώσσα C αντιστοιχεί
    // στην αληθή συνθήκη. Αντί για TRUE θα μπορούσαμε για παράδειγμα να βάλουμε π.χ. 5>1,
    // δηλαδή μια συνθήκη που ισχύει πάντα.
    while(TRUE) {            //Βρόχος που δεν τελειώνει ποτέ (συνθήκη πάντα αληθής)
        while(input(PIN_B4) == 1) { // Αναμονή έως ότου πατηθεί το μπουτόν
            // Στην κατάσταση αναμονής δεν εκτελείται κάποια εντολή
            // Όταν πατηθεί το μπουτόν γίνεται έξοδος από τον βρόχο
            // και εκτελείται η επόμενη εντολή

            delay_ms(50);        // Καθυστερήση 50 ms για αποφυγή φαινόμενου αναπηδήσεων
            while(input(PIN_B4) == 0) { //Αναμονή έως ότου αφεθεί το μπουτόν
                //Στην κατάσταση αναμονής δεν εκτελείται κάποια εντολή
                //Όταν αφεθεί το μπουτόν γίνεται έξοδος από τον βρόχο
                //και εκτελείται η επόμενη εντολή

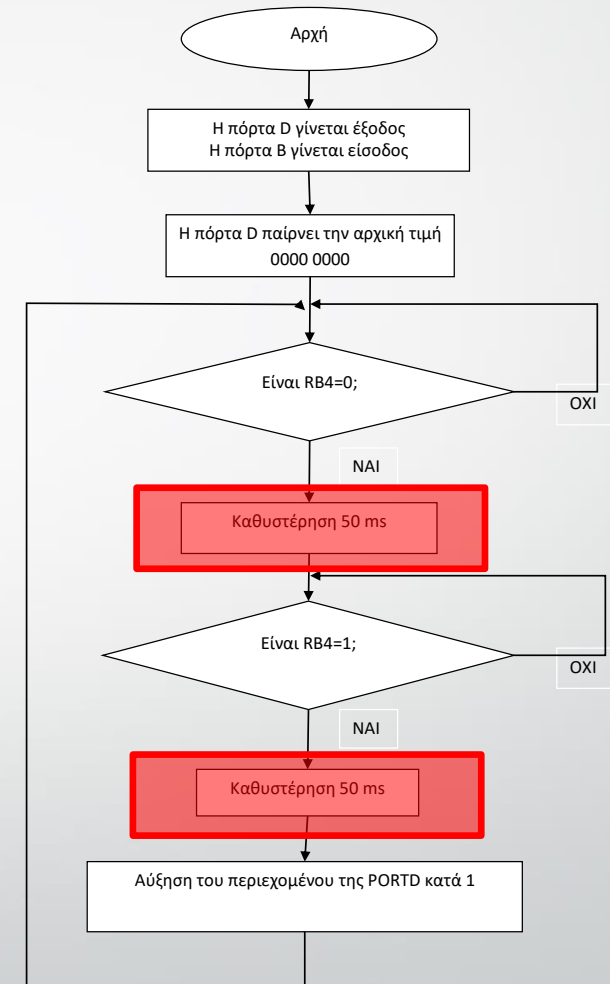
                delay_ms(50);        // Καθυστερήση 50 ms για αποφυγή φαινόμενου αναπηδήσεων
                PORTD=PORTD+1;        //Αύξηση του περιεχομένου της πόρτας D κατά 1
            }
        }
    }
    // κλείσιμο main
}
```



# Φαινόμενο αναπηδήσεων



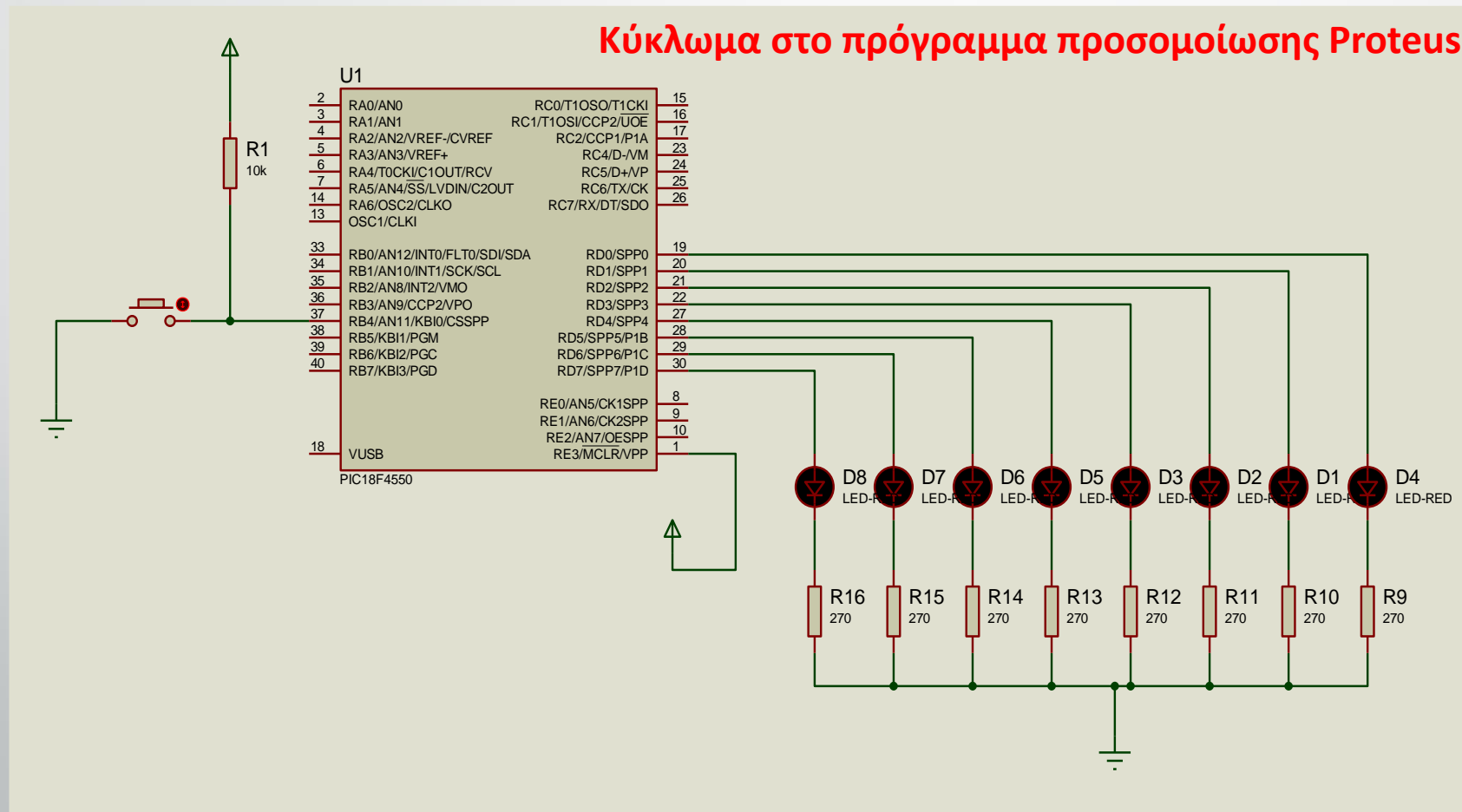
Για να μη ληφθεί ένα πάτημα του μπουτόν σαν περισσότερα, μετά τον έλεγχο του μπουτόν γράφουμε μια **συνάρτηση χρονοκαθυστερήσης**



## Άσκηση 2b. Ανάγνωση πατήματος μπουτόν

Να γραφεί πρόγραμμα με το οποίο για όσο διάστημα είναι πατημένο το μπουτόν που είναι συνδεδεμένο στον ακροδέκτη RB4 εμφανίζεται στην πόρτα D η τιμή 01111110.

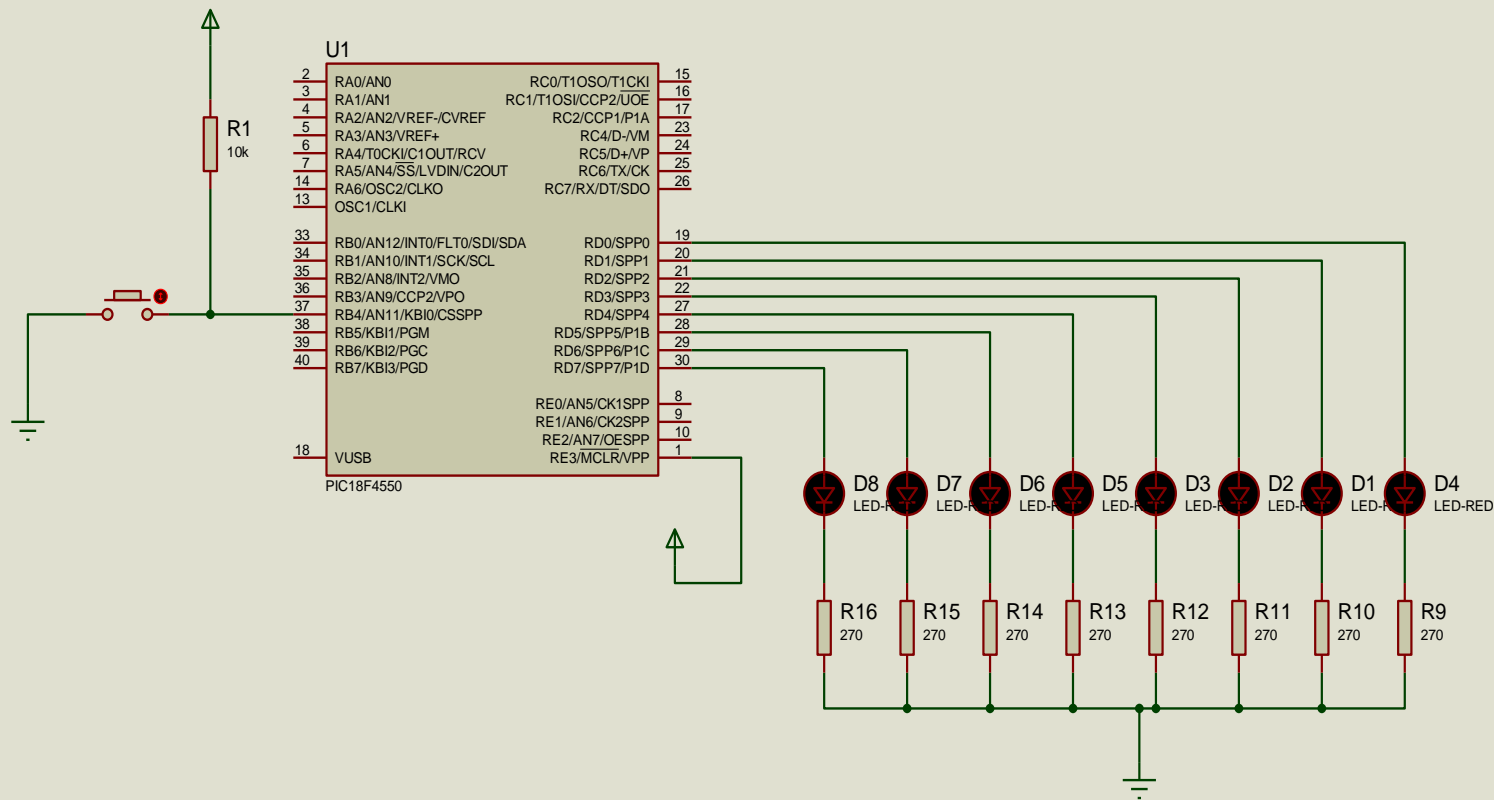
Για όσο διάστημα δεν είναι πατημένο το μπουτόν που είναι συνδεδεμένο στον ακροδέκτη RB4 εμφανίζεται στην πόρτα D η τιμή 10000001.



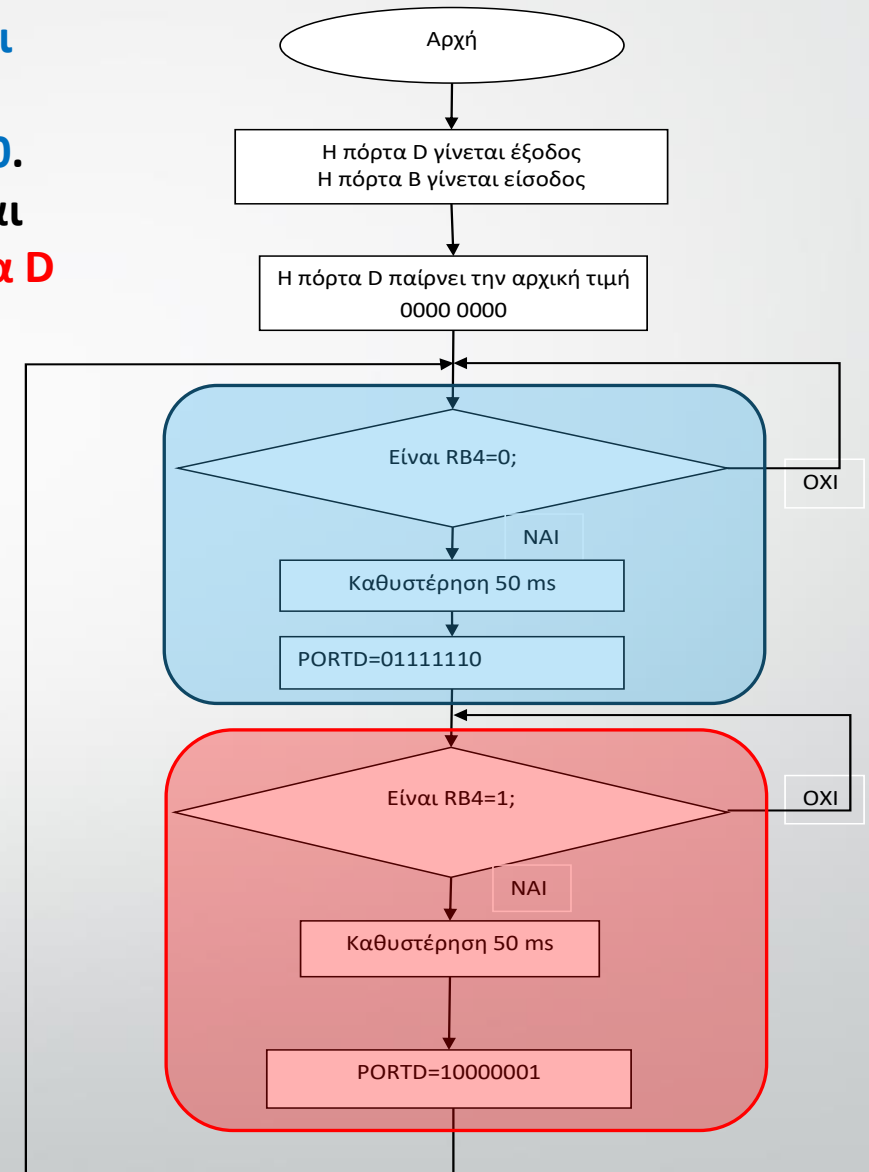
## Άσκηση 2b. Ανάγνωση πατήματος μπουτόν

Να γραφεί πρόγραμμα με το οποίο **για όσο διάστημα είναι πατημένο το μπουτόν** που είναι συνδεδεμένο στον ακροδέκτη RB4 **εμφανίζεται στην πόρτα D η τιμή 01111110**.

**Για όσο διάστημα δεν είναι πατημένο το μπουτόν** που είναι συνδεδεμένο στον ακροδέκτη RB4 **εμφανίζεται στην πόρτα D η τιμή 10000001**.



## Διάγραμμα ροής



## Άσκηση 2b. Ανάγνωση πατήματος μπουτόν

### Πρόγραμμα σε γλώσσα C

**Προσοχή!** Στον φάκελο στον οποίο θα αναπτύξετε το Project θα πρέπει να τοποθετήσετε τα αρχεία main.h και 18f4550.h και add.txt.

```
#include<main.h>          //Το αρχείο <main.h> περιέχει αρχικές ρυθμίσεις
                           //Πρέπει να τοποθετηθεί οπωσδήποτε στον ίδιο φάκελο στον οποίο θα αναπτύξετε το project σας

#define PORTD=0xF83        //F83 είναι η θέση του καταχωρητή δεδομένων της πόρτας D στη μνήμη του μικροελεγκτή
                           // Δίνουμε στη διεύθυνση 0xF83 το όνομα PORTD

#define PORTB=0xF81        //F81 είναι η θέση του καταχωρητή δεδομένων της πόρτας D στη μνήμη του μικροελεγκτή
                           // Δίνουμε στη διεύθυνση 0xF81 το όνομα PORTB

// *****Από εδώ αρχίζει το κύριο πρόγραμμα*****

void main()
{
    //άνοιγμα main
    set_tris_d(0x00);       //Η θύρα D γίνεται έξοδος (καταχωρητής κατεύθυνσης=0000 0000)
    set_tris_b(0xFF);       //Η θύρα B γίνεται είσοδος (καταχωρητής κατεύθυνσης=1111 1111)
    PORTD=0x00;             //Στην πόρτα D δίνεται η αρχική τιμή 00000000

    // Με την παρακάτω δομή while(TRUE){ } εκτελείται αενάως (για πάντα) το σύνολο των
    // εντολών που είναι μέσα στις αγκύλες. Η δεσμευμένη λέξη TRUE στη γλώσσα C αντιστοιχεί
    // στην αληθή συνθήκη. Αντί για TRUE θα μπορούσαμε για παράδειγμα να βάλουμε π.χ. 5>1,
    // δηλαδή μια συνθήκη που ισχύει πάντα.

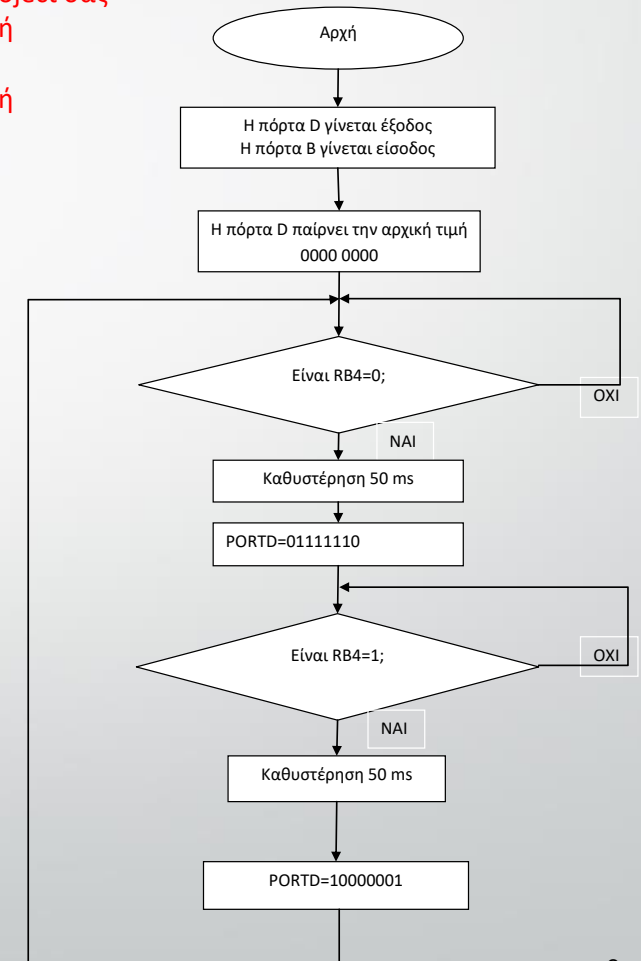
    while(TRUE) {           //Βρόχος που δεν τελειώνει ποτέ (συνθήκη πάντα αληθής)
        while(input(PIN_B4) == 1) { // Αναμονή έως ότου πατηθεί το μπουτόν
            // Στην κατάσταση αναμονής δεν εκτελείται κάποια εντολή
            // Όταν πατηθεί το μπουτόν γίνεται έξοδος από το βρόχο
            // και εκτελείται η επόμενη εντολή

            delay_ms(50);      // Καθυστερήση 50 ms για αποφυγή φαινομένου αναπηδήσεων
            PORTD=0b01111110;  // Αποδίδεται μια τιμή στην πόρτα D
            while(input(PIN_B4) == 0) { //Αναμονή έως ότου αφεθεί το μπουτόν
                //Στην κατάσταση αναμονής δεν εκτελείται καμιά εντολή
                //Όταν αφεθεί το μπουτόν γίνεται έξοδος από τον βρόχο
                //και εκτελείται η επόμενη εντολή

                delay_ms(50);   // Καθυστερήση 50 ms για αποφυγή φαινομένου αναπηδήσεων
                PORTD=0b10000001; //Αποδίδεται μια τιμή στην πόρτα D
            }
        }
    }

    //κλείσιμο while(TRUE)
}

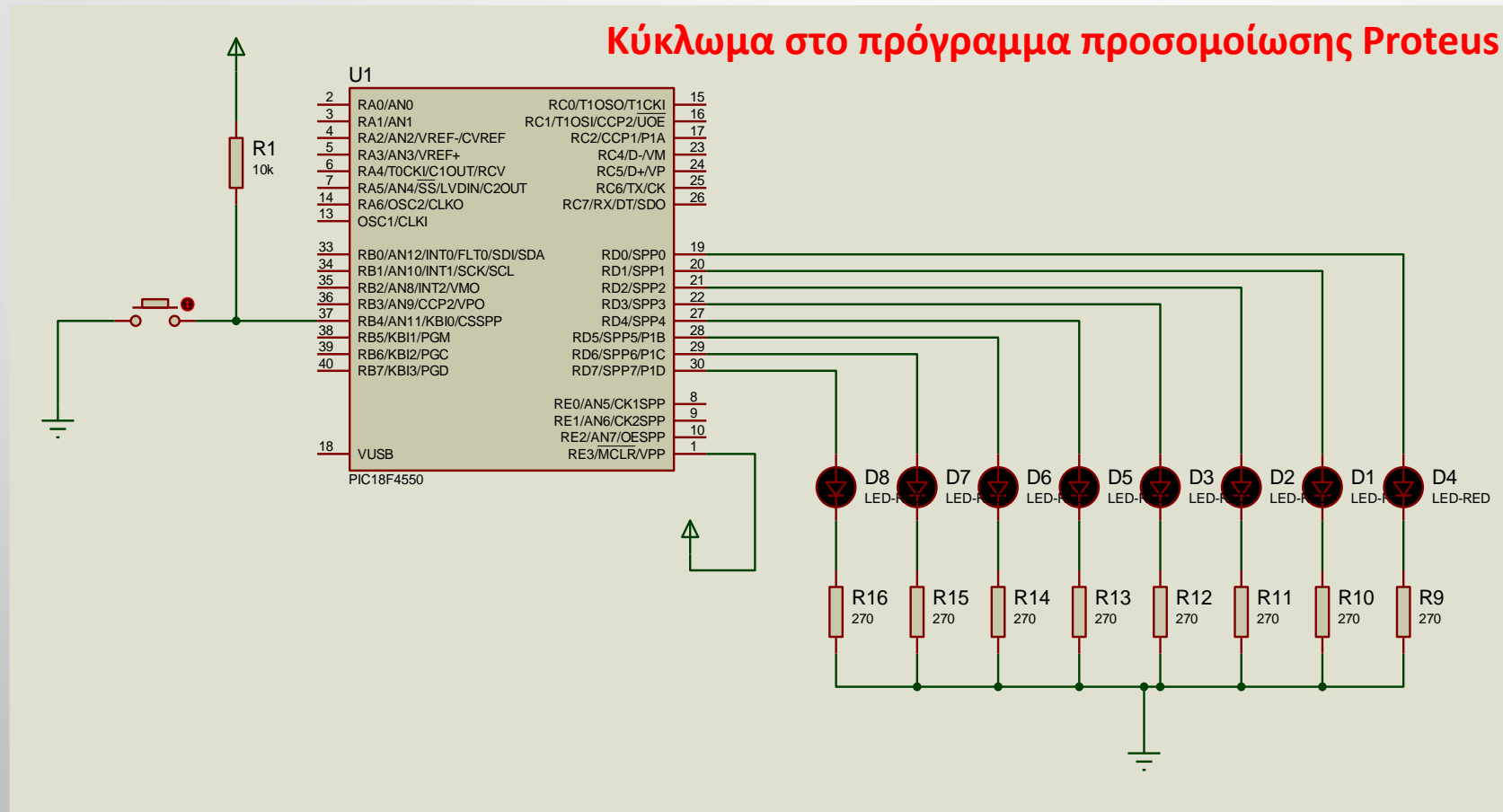
// κλείσιμο main
```





## Άσκηση 2c. Ανάγνωση πατήματος μπουτόν

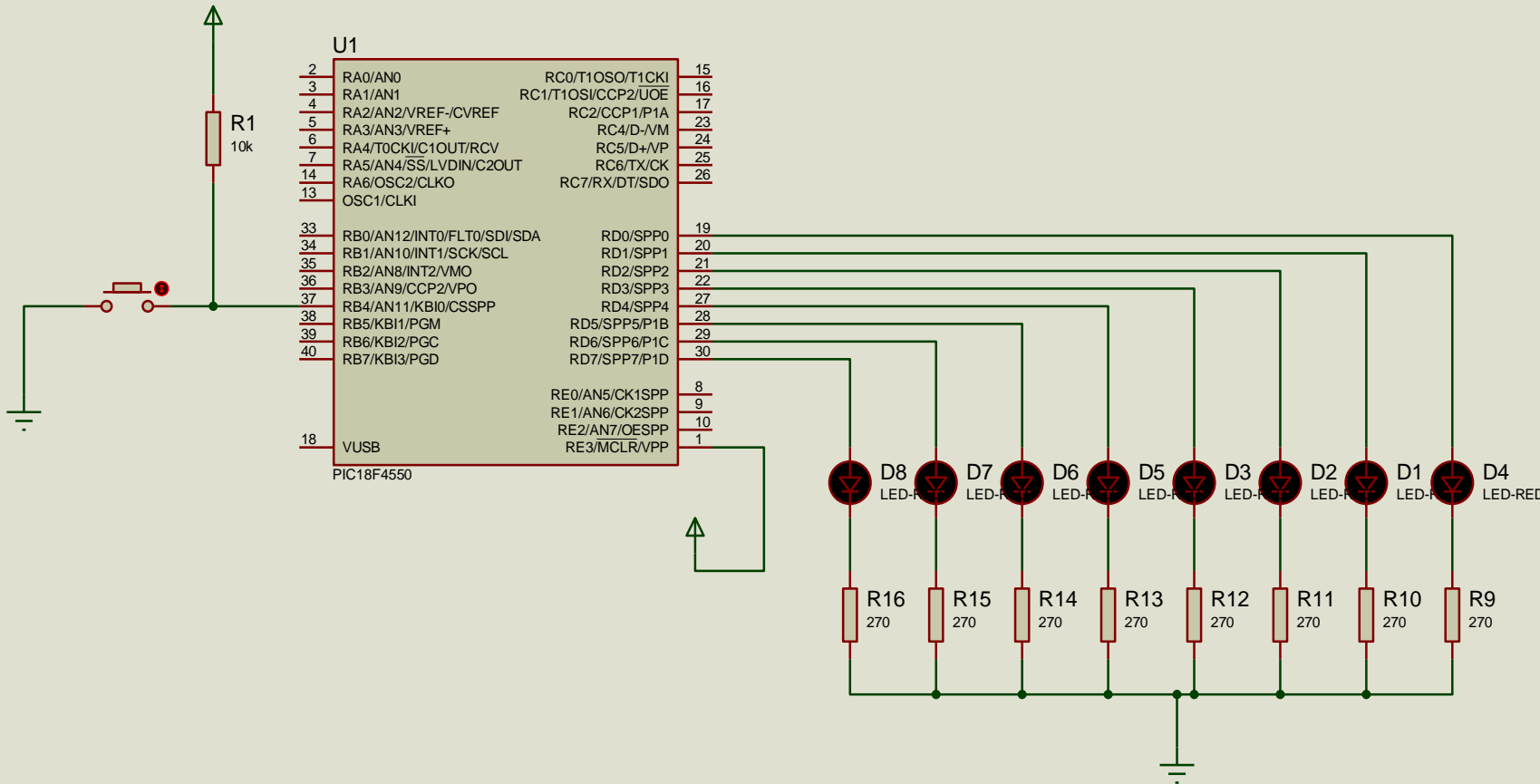
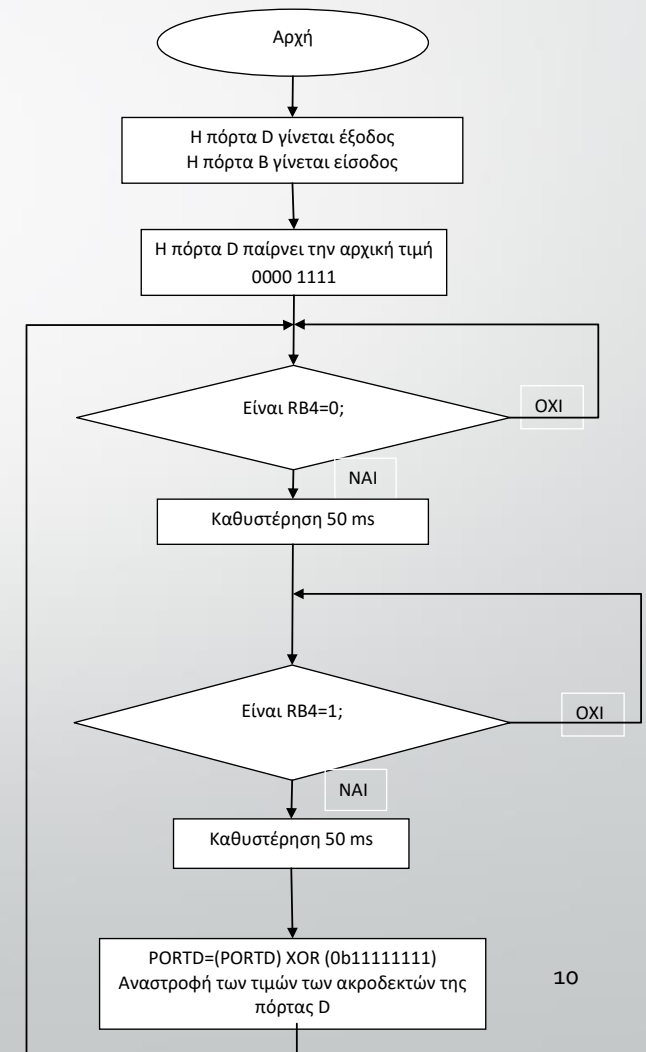
Να γραφεί πρόγραμμα με το οποίο κάθε φορά που πατιέται και αφήνεται το μπουτόν αλλάζει η κατάσταση του κάθε ακροδέκτη της πόρτας D, δηλαδή τα 1 γίνονται 0 και τα 0 γίνονται 1. Στην πόρτα D να δοθεί η αρχική τιμή 00001111.



## Άσκηση 2c. Ανάγνωση πατήματος μπουτόν

Να γραφεί πρόγραμμα με το οποίο κάθε φορά που πατιέται και αφήνεται το μπουτόν αλλάζει η κατάσταση του κάθε ακροδέκτη της πόρτας D, δηλαδή τα 1 γίνονται 0 και τα 0 γίνονται 1. Στην πόρτα D να δοθεί η αρχική τιμή 00001111.

### Διάγραμμα ροής



## Άσκηση 2c. Ανάγνωση πατήματος μπουτόν

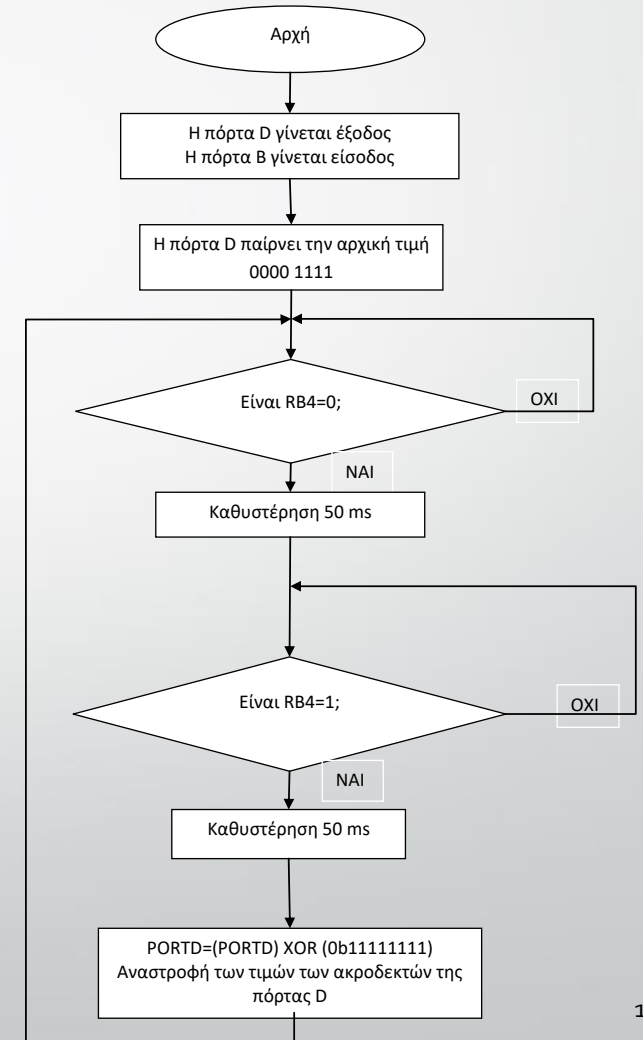
### Πρόγραμμα σε γλώσσα C

**Προσοχή!** Στον φάκελο στον οποίο θα αναπτύξετε το project θα πρέπει να τοποθετήσετε τα αρχεία `main.h` και `18f4550.h` και `add.txt`.

```
#include<main.h>          //Το αρχείο <main.h> περιέχει αρχικές ρυθμίσεις
                          //Πρέπει να τοποθετηθεί οπωσδήποτε στον ίδιο φάκελο στον οποίο θα
                          //αναπτύξετε το project σας
#byte PORTD=0xF83         //F83 είναι η θέση του καταχωρητή δεδομένων της πόρτας D
                          // στη μνήμη του μικροελεγκτή
                          // Δίνουμε στη διεύθυνση 0xF83 το όνομα PORTD
#byte PORTB=0xF81         //F81 είναι η θέση του καταχωρητή δεδομένων της πόρτας D
                          // στη μνήμη του μικροελεγκτή
                          // Δίνουμε στη διεύθυνση 0xF81 το όνομα PORTB
// *****Από εδώ αρχίζει το κύριο πρόγραμμα*****
void main()
{
    //άνοιγμα αγκύλης της συνάρτησης main
    set_tris_d(0x00);      //Η θύρα D γίνεται έξοδος (καταχωρητής κατεύθυνσης=0000 0000)
    set_tris_b(0xFF);      //Η θύρα B γίνεται είσοδος (καταχωρητής κατεύθυνσης=1111 1111)
    PORTD=0x0F;           //Στην πόρτα D δίνεται η αρχική τιμή 00001111
    // Με την παρακάτω δομή while(TRUE){ } εκτελείται αενάως (για πάντα) το σύνολο των
    // εντολών που είναι μέσα στις αγκύλες. Η δεσμευμένη λέξη TRUE στη γλώσσα C αντιστοιχεί
    // στην αληθή συνθήκη. Αντί για TRUE θα μπορούσαμε για παράδειγμα να βάλουμε π.χ. 5>1,
    // δηλαδή μια συνθήκη που ισχύει πάντα.
    while(TRUE) {          //Βρόχος που δεν τελειώνει ποτέ (συνθήκη πάντα αληθής)
        while(input(PIN_B4) == 1) { // Αναμονή έως ότου πατηθεί το μπουτόν
            // Στην κατάσταση αναμονής δεν εκτελείται κάποια εντολή
            // Όταν πατηθεί το μπουτόν γίνεται έξοδος από τον βρόχο
            // και εκτελείται η επόμενη εντολή

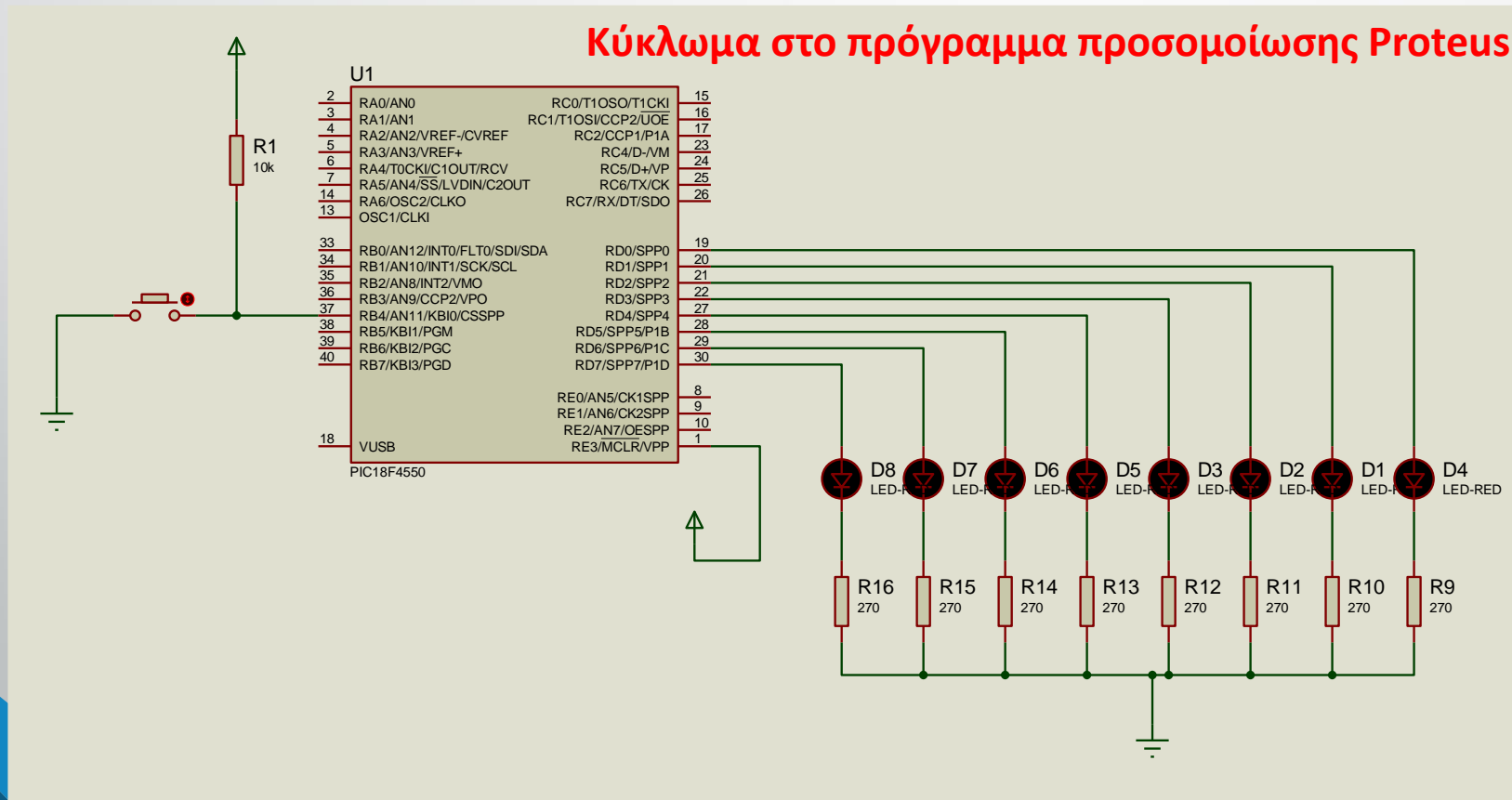
            delay_ms(50);      // Καθυστερήση 50 ms για αποφυγή φαινομένου αναπηδήσεων
            while(input(PIN_B4) == 0) { //Αναμονή έως ότου αφεθεί το μπουτόν
                //Στην κατάσταση αναμονής δεν εκτελείται κάποια εντολή
                //Όταν αφεθεί το μπουτόν γίνεται έξοδος από τον βρόχο
                //και εκτελείται η επόμενη εντολή

                delay_ms(50);    // Καθυστερήση 50 ms για αποφυγή φαινομένου αναπηδήσεων
                PORTD=PORTD^0b11111111; // Αναστροφή των τιμών των ακροδεκτών της πόρτας D
            }
            //κλείσιμο while(TRUE)
        }
        // κλείσιμο main
    }
}
```



## Άσκηση 2d. Ανάγνωση πατήματος μπουτόν με τη χρήση συνάρτησης πατήματος μπουτόν

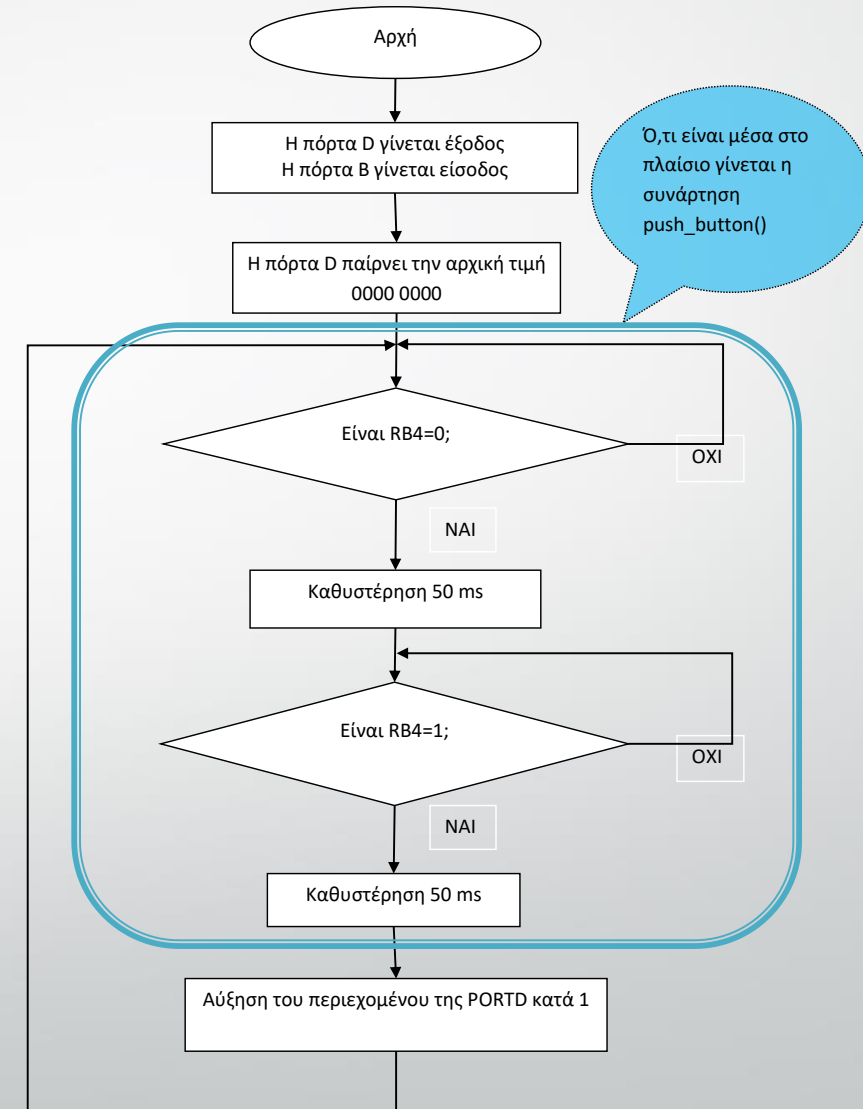
Να γραφεί πρόγραμμα με το οποίο κάθε φορά που πατάμε ένα μπουτόν, το οποίο είναι συνδεδεμένο στον ακροδέκτη RB4 της πόρτας B, το περιεχόμενο της πόρτας D αυξάνεται κατά 1. Το αποτέλεσμα της αύξησης να φαίνεται σε 8 LED τα οποία είναι συνδεδεμένα στην πόρτα D του μικροελεγκτή. Να χρησιμοποιηθεί συνάρτηση με την οποία ελέγχεται ότι πατήθηκε το μπουτόν. Στη συνάρτηση να δοθεί το όνομα `push_button`.



## Άσκηση 2d. Ανάγνωση πατήματος μπουτόν με την χρήση συνάρτησης πατήματος μπουτόν

- Να γραφεί πρόγραμμα με το οποίο κάθε φορά που πατάμε ένα μπουτόν το οποίο είναι συνδεδεμένο στον ακροδέκτη RB4 της πόρτας B το περιεχόμενο της πόρτας D αυξάνεται κατά 1.
- Το αποτέλεσμα της αύξησης να φαίνεται σε 8 LED τα οποία είναι συνδεδεμένα στην πόρτα D του μικροελεγκτή.
- Να χρησιμοποιηθεί συνάρτηση με την οποία ελέγχεται ότι πατήθηκε το μπουτόν. Στη συνάρτηση να δοθεί το όνομα `push_button`.

### Διάγραμμα ροής



## Άσκηση 2d. Ανάγνωση πατήματος μπουτόν με τη χρήση συνάρτησης πατήματος μπουτόν

### Πρόγραμμα σε γλώσσα C

**Προσοχή!** Στον φάκελο στον οποίο θα αναπτύξετε το project θα πρέπει να τοποθετήσετε τα αρχεία main.h και 18f4550.h και add.txt.

```
#include<main.h>          //Το αρχείο <main.h> περιέχει αρχικές ρυθμίσεις
                          //Πρέπει να τοποθετηθεί οπωσδήποτε στον ίδιο φάκελο στον οποίο θα
                          //αναπτύξετε το project σας

#byte PORTD=0xF83         //F83 είναι η θέση του καταχωρητή δεδομένων της πόρτας D στη μνήμη του μικροελεγκτή
                          // Δίνουμε στη διεύθυνση 0xF83 το όνομα PORTD

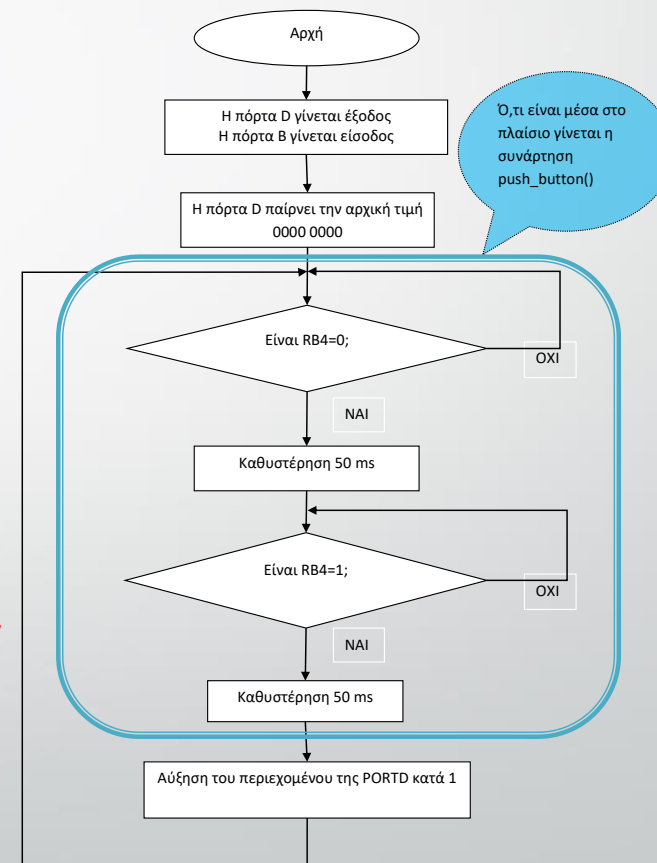
#byte PORTB=0xF81         //F81 είναι η θέση του καταχωρητή δεδομένων της πόρτας D στη μνήμη του μικροελεγκτή
                          // Δίνουμε στην διεύθυνση 0xF81 το όνομα PORTB

void push_button(void)    //Δήλωση της συνάρτησης πατήματος μπουτόν. Προσοχή, πάνω από τη main()
                          //Με αυτήν τη συνάρτηση εκτελούνται όλες οι εντολές με τις οποίες
                          //ο μικροελεγκτής αναγνωρίζει το πλήρες πάτημα (πατώ + αφήνω) ενός μπουτόν

// *****Από εδώ αρχίζει το κύριο πρόγραμμα*****

void main()
{
    // Άνοιγμα main
    set_tris_d(0x00);      //Η θύρα D γίνεται έξοδος (καταχωρητής κατεύθυνσης=0000 0000)
    set_tris_b(0xFF);      //Η θύρα B γίνεται είσοδος (καταχωρητής κατεύθυνσης=1111 1111)
    PORTD=0x00;           //Στην πόρτα D δίνεται η αρχική τιμή 00000000
                          // Με την παρακάτω δομή while (TRUE){ } εκτελείται αενάως (για πάντα) το σύνολο των
                          // εντολών που είναι μέσα στις αγκύλες. Η δεσμευμένη λέξη TRUE στη γλώσσα C αντιστοιχεί
                          // στην αληθή συνθήκη. Αντί για TRUE θα μπορούσαμε για παράδειγμα να βάλουμε π.χ. 5>1,
                          // δηλαδή μια συνθήκη που ισχύει πάντα.

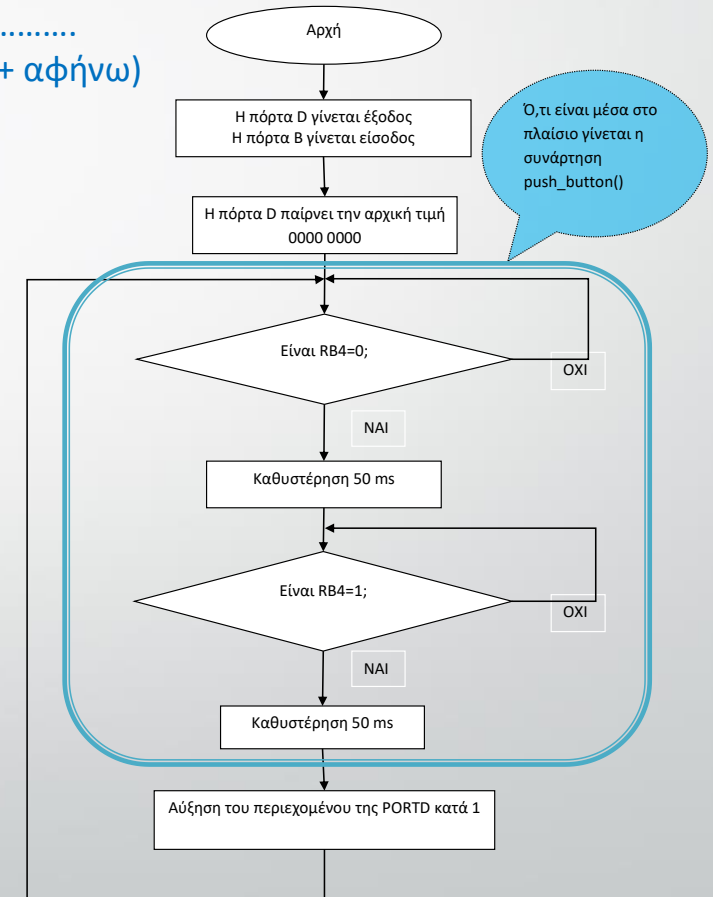
    while(TRUE) { //Βρόχος που δεν τελειώνει ποτέ (συνθήκη πάντα αληθής)
        push_button()    //Γίνεται κλήση της συνάρτησης πατήματος μπουτόν
        PORTD=PORTD+1;    //Αύξηση του περιεχομένου της πόρτας D κατά 1
    }                    // Κλείσιμο while(TRUE)
}                        // Κλείσιμο main
```



## Άσκηση 2d. Ανάγνωση πατήματος μπουτόν με τη χρήση συνάρτησης πατήματος μπουτόν

```
//..... Ορισμός της συνάρτησης push_button.....  
//..... Ο ορισμός της συνάρτησης γράφεται μετά το κύριο πρόγραμμα.....  
// Με τη συνάρτηση αυτή ο μικροελεγκτής περιμένει ένα πλήρες πάτημα του μπουτόν (πατώ + αφήνω)  
// και στη συνέχεια πάει στην επόμενη εντολή
```

```
void push_button(void) { // Ανοίγει το άγκιστρο της συνάρτησης push_button()  
    while(input(PIN_B4) == 1) { // Αναμονή έως ότου πατηθεί το μπουτόν  
    }                          // Στην κατάσταση αναμονής δεν εκτελείται κάποια εντολή  
                              // Όταν πατηθεί το μπουτόν γίνεται έξοδος από τον βρόχο  
                              // και εκτελείται η επόμενη εντολή  
  
    delay_ms(50);             // Καθυστερήση 50 ms για αποφυγή φαινόμενου αναπηδήσεων  
    while(input(PIN_B4) == 0) { // Αναμονή έως ότου αφεθεί το μπουτόν  
    }                          // Στην κατάσταση αναμονής δεν εκτελείται κάποια εντολή  
                              // Όταν αφεθεί το μπουτόν γίνεται έξοδος από τον βρόχο  
                              // και εκτελείται η επόμενη εντολή  
  
    delay_ms(50);             // Καθυστερήση 50 ms για αποφυγή φαινόμενου αναπηδήσεων  
    }                          // Κλείνει το άγκιστρο της συνάρτησης push_button()
```





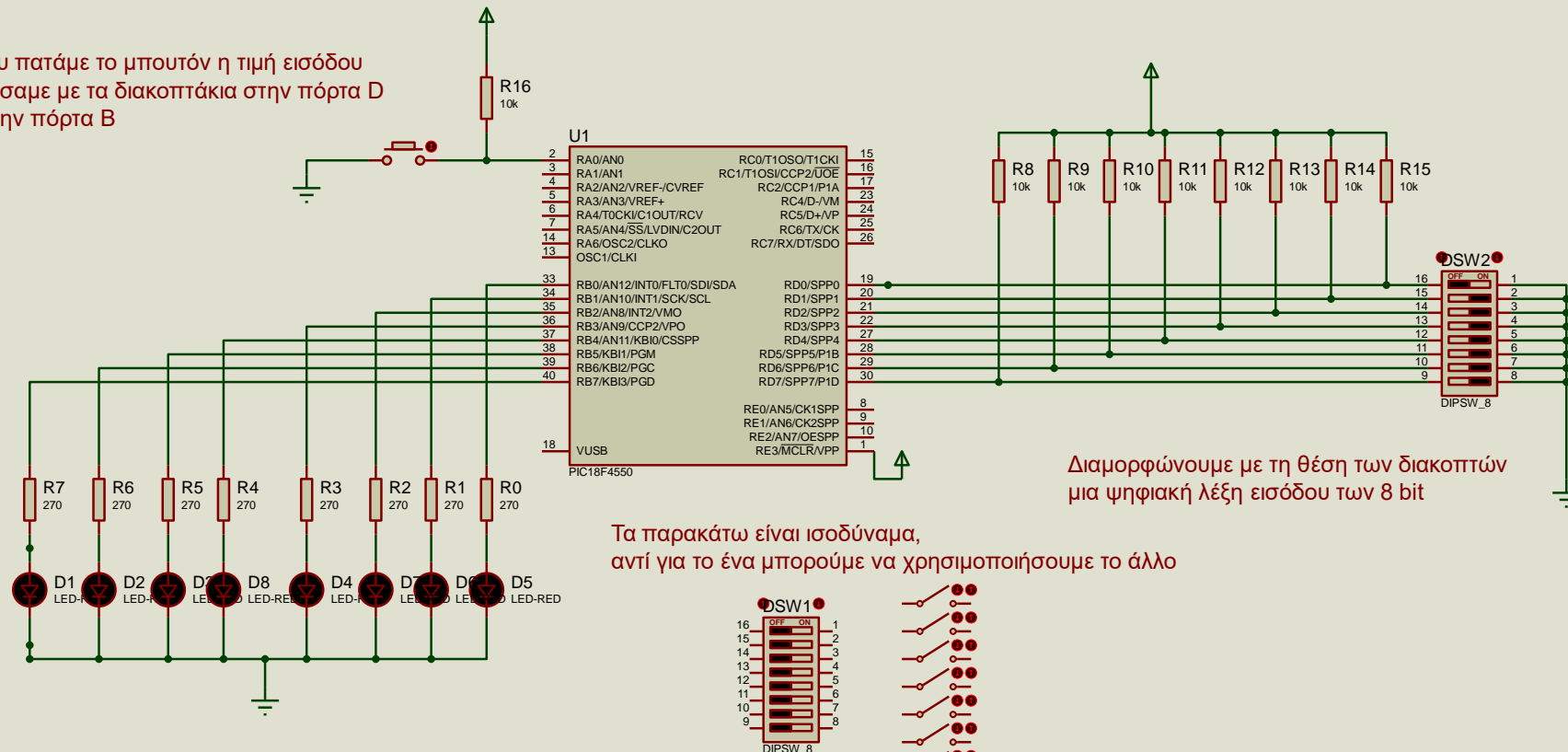
## Άσκηση 2ε. Μεταφορά του περιεχομένου της πόρτας D στην πόρτα B με το πάτημα μπουτόν που συνδέεται στον ακροδέκτη A0 της PORTA

```
while(TRUE){ //Βρόχος που δεν τελειώνει ποτέ (συνθήκη πάντα αληθής)
    push_button(); // Καλείται η συνάρτηση push_button με την οποία αναμένεται
                  // να εκτελεστεί ένα πλήρες πάτημα του μπουτόν

    PORTB=PORTD;

} //Κλείσιμο της αγκύλης του while(TRUE)
```

Κάθε φορά που πατάμε το μπουτόν η τιμή εισόδου που διαμορφώσαμε με τα διακοπτάκια στην πόρτα D μεταφέρεται στην πόρτα B



Το μαύρο είναι η θέση του διακόπτη