6η Εργαστηριακή Αναφορά

Παπαδόπουλος Χαράλαμπος 03120199

Στρίφτης Γεώργιος 03121200

**Άσκηση 1**

Η πρώτη άσκηση αφορά την ανάγνωση και επεξεργασία δεδομένων από ένα πληκτρολόγιο 4x4 που είναι συνδεδεμένο στην πλακέτα ntuAboard\_G1. Ο κώδικας περιλαμβάνει τις συναρτήσεις scan\_row, scan\_keypad, και scan\_keypad\_rising\_edge, οι οποίες διαδοχικά ελέγχουν ποια πλήκτρα είναι πατημένα.

* Η scan\_row ελέγχει μία συγκεκριμένη γραμμή του πληκτρολογίου για πιεσμένους διακόπτες, αναγνωρίζοντας ποια στήλη αντιστοιχεί σε κάθε πλήκτρο.
* Η scan\_keypad καλεί τη scan\_row για κάθε γραμμή, επιτρέποντας έτσι την ανάγνωση του πλήρους πληκτρολογίου.
* Τέλος, η scan\_keypad\_rising\_edge καταγράφει τις αλλαγές στις καταστάσεις των πλήκτρων, αποφεύγοντας σφάλματα που προέρχονται από τον σπινθηρισμό των διακοπτών.

Μέσω της συνάρτησης keypad\_to\_ascii, τα πατημένα πλήκτρα αντιστοιχίζονται σε κωδικούς ASCII, ενώ τα συγκεκριμένα πλήκτρα συνδέονται με LEDs στις θύρες PB0 έως PB3, τα οποία ανάβουν όσο το αντίστοιχο πλήκτρο παραμένει πατημένο.

|  |
| --- |
| **void** **setup**() {  twi\_init();  PCA9555\_0\_write(REG\_CONFIGURATION\_1, **0xF0**);  PCA9555\_0\_write(REG\_CONFIGURATION\_0, **0x00**); //Set EXT\_PORT0 as output  PCA9555\_0\_write(REG\_OUTPUT\_1, **0x0F**); // check last line  DDRB = **0xFF**;  }  **uint16\_t** **scan\_row**(**uint8\_t** i, **uint16\_t** input)  {  **uint16\_t** temp = **0**;  **switch**(i){  **case** **1**:  PCA9555\_0\_write(REG\_OUTPUT\_1, **0**b11111110);  temp = ~(PCA9555\_0\_read(REG\_INPUT\_1));  temp &= **0xF0**;  input = temp >> **4**;  **break**;  **case** **2**:  PCA9555\_0\_write(REG\_OUTPUT\_1, **0**b11111101);  temp = ~(PCA9555\_0\_read(REG\_INPUT\_1));  temp &= **0xF0**;  input |= temp;  **break**;  **case** **3**:  PCA9555\_0\_write(REG\_OUTPUT\_1, **0**b11111011);  temp = ~(PCA9555\_0\_read(REG\_INPUT\_1));  temp &= **0xF0**;  temp = temp << **4**;  input |= temp;  **break**;  **case** **4**:  PCA9555\_0\_write(REG\_OUTPUT\_1, **0**b11110111);  temp = ~(PCA9555\_0\_read(REG\_INPUT\_1));  temp &= **0xF0**;  temp = temp << **8**;  input |= temp;  **break**;  }    **return** input;  }  **uint16\_t** **scan\_keypad**()  {  **uint16\_t** input = **0**;  **for**(**uint16\_t** i = **1**; i < **5**; i++){  input = scan\_row(i, input);  }    **return** input;  }  **static** **uint16\_t** pressed\_keys;  **void** **scan\_keypad\_rising\_edge**()  {  **uint16\_t** pressed\_keys\_tempo;  pressed\_keys\_tempo = scan\_keypad();  \_delay\_ms(**20**);  pressed\_keys\_tempo &= scan\_keypad();  **if**(pressed\_keys\_tempo != pressed\_keys){  pressed\_keys ^= pressed\_keys\_tempo;  pressed\_keys &= pressed\_keys\_tempo;  }  }  **static** **int** ascii[] = {**42**, **48**, **35**, **68**, **55**, **56**, **57**, **67**, **52**, **53**, **54**, **66**, **49**, **50**, **51**, **65**};  **int** **keypad\_to\_ascii**()  {  **uint16\_t** temp = pressed\_keys;  **for**(**int** i = **0**; i < **16**; i++) {  **if**(temp == **1**) **return** ascii[i];  temp = temp >> **1**;  }  **return** **0**;  }  **int** **main**(){  setup();  **int** output;  **while**(**1**){  scan\_keypad\_rising\_edge();  output = keypad\_to\_ascii();  **switch**(output){  **case** **0**:  PORTB = **0x00**;  **break**;  **case** **65**:  PORTB = **0x01**;  **break**;  **case** **56**:  PORTB = **0x02**;  **break**;  **case** **54**:  PORTB = **0x04**;  **break**;  **case** **42**:  PORTB = **0x08**;  **break**;  }  \_delay\_ms(**50**);  }  } |

**Άσκηση 2**

Στη δεύτερη άσκηση, ο κώδικας διαβάζει την κατάσταση του πληκτρολογίου και εμφανίζει τον τελευταίο πατημένο χαρακτήρα σε οθόνη LCD 2x16. Για την υλοποίηση αυτή, η συνάρτηση scan\_keypad\_rising\_edge καλείται συνεχώς για να ανιχνεύσει αν έχει πατηθεί κάποιο πλήκτρο, το οποίο στη συνέχεια μετατρέπεται στον αντίστοιχο ASCII χαρακτήρα και αποστέλλεται στην οθόνη. Αυτό επιτρέπει τη συνεχή απεικόνιση του τελευταίου πατημένου πλήκτρου, με την προϋπόθεση ότι μόνο ένα πλήκτρο είναι πατημένο κάθε φορά.

|  |
| --- |
| **int** **main**(**void**) {  setup();    **while**(**1**) {  scan\_keypad\_rising\_edge();  **if**(pressed\_keys != **0**) {  lcd\_clear\_display();  lcd\_data(keypad\_to\_ascii());  }  \_delay\_ms(**50**);  }  } |

**Άσκηση 3**

Η τρίτη άσκηση περιλαμβάνει την υλοποίηση μιας ηλεκτρονικής κλειδαριάς που βασίζεται σε έναν διψήφιο κωδικό από το πληκτρολόγιο 4x4. Ο κώδικας διαβάζει διαδοχικά δύο αριθμούς και συγκρίνει την είσοδο με έναν προκαθορισμένο κωδικό. Αν ο κωδικός είναι σωστός, τα LEDs PB0 έως PB5 ανάβουν για 3 δευτερόλεπτα, ενώ αν είναι λανθασμένος, τα ίδια LEDs αναβοσβήνουν για 5 δευτερόλεπτα. Μετά την είσοδο του κωδικού, το πρόγραμμα δεν επιτρέπει άλλες εισόδους για 5 δευτερόλεπτα.

|  |
| --- |
| **int** **main**(){  setup();  **int** nums[**2**];    **while**(**1**){  **int** counter = **0**;  **while**(counter < **2**) {  scan\_keypad\_rising\_edge();  **if**(pressed\_keys != **0**) {  nums[counter] = keypad\_to\_ascii();  counter++;  scan\_keypad\_rising\_edge();  **while**(pressed\_keys != **0**){  scan\_keypad\_rising\_edge();  }  }  }    **if**(nums[**0**] == **50** && nums[**1**] == **48**) {  PORTB = **0xFF**;  \_delay\_ms(**5000**);  PORTB = **0x00**;  }  **else**{  **for**(**int** i = **0**; i < **5**; i++) {  PORTB = **0xFF**;  \_delay\_ms(**500**);  PORTB = **0x00**;  \_delay\_ms(**500**);  }  }  }  } |