

Μικροεπεξεργαστές και Περιφερειακά

8^ο εξάμηνο

Τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών
Πολυτεχνική Σχολή
Αριστοτέλειο Πανεπιστήμιο Θεσσαλονίκης

Ομάδα 25

Πιπινιάς Ιωάννης¹ και Φανάκης Απόστολος²

¹7965, pipinias@ece.auth.gr

²8261, apostolof@ece.auth.gr

13 Ιουνίου 2021

1 Εισαγωγή

Το παρόν έγγραφο είναι η αναφορά του δεύτερου εργαστηρίου στο μάθημα Μικροεπεξεργαστές και Περιφερειακά. Στο δεύτερο εργαστήριο μας δόθηκε κώδικας που συνδυάζει τον simulator του εργαλείου Keil με το εργαλείο redBlocks στο οποίο έγινε προσομοίωση των φυσικών στοιχείων (οθόνη, leds, διακόπτες κλπ.). Ο κώδικας αφορούσε τη λειτουργία ενός αυτόματου πωλητή (vending machine). Μας ζητήθηκε να τροποποιήσουμε κώδικα ώστε να προσθέσουμε δύο λειτουργικότητες, την προσθήκη ενός κουμπιού με το πάτημα του οποίου εκκινείται η λειτουργία του πωλητή και τον υπολογισμό των ρέστων που θα πρέπει να πάρει ο χρήστης στο τέλος της αγοράς. Με τον συνδυασμό των εργαλείων είναι δυνατή η προσομοίωση και ο έλεγχος του project συνολικά.

Ο κώδικας είναι διαθέσιμος στα συνημμένα καθώς και στο Gitlab repository [εδώ](#).

2 Υλοποίηση

Αρχικά τροποποιήθηκε το project στο εργαλείο redBlocks, όπου με χρήση του γραφικού περιβάλλοντος έγινε προσθήκη ενός νέου κουμπιού START. Για την επίτευξη αυτού δημιουργήθηκε αρχικά ένα νέο digital input interface και έπειτα έγινε χρήση του interface σε ένα νέο view component.

Τα μέρη του κώδικα που τροποποιήθηκαν είναι:

- το header file `LowLevelPlatform.h`
- το header file `PlatformCallbacks.h`
- το source file `Application.cpp`
- το source file `main.cpp`
- το source file `VendingMode.cpp`

Αρχικά, στο header `LowLevelPlatform.h` προστέθηκε ένα νέο type definition για το κουμπί `StartButton`. Έπειτα, στο ίδιο αρχείο, το `StartButton` προστέθηκε στη λίστα των digital inputs ενώ έγινε ρύθμιση για trigger με θετική αιχμή.

Στο αρχείο `PlatformCallbacks.h` προστέθηκε ο ορισμός μία `extern "C"` συνάρτησης με όνομα `onStartPressed`. Ο ορισμός αυτό χρησιμοποιήθηκε ως callback για το κουμπί `StartButton`.

Στο αρχείο `Application.cpp` προστέθηκε η γραμμή `Platform::StartButton::enableCallback()` με σκοπό την ενεργοποίηση του callback που ορίσαμε στο αρχείο `PlatformCallbacks.h`.

Η υλοποίηση της συνάρτησης `onStartPressed` έγινε στο αρχείο `main.cpp`. Η callback αυτή είναι πολύ απλή και αλλάζει μία global boolean μεταβλητή με όνομα `should_continue` σε true.

Άλλες τροποποιήσεις που έγιναν στο αρχείο είναι η προσθήκη ενός βρόχου μέσα στη main ο οποίος καλεί τη `__WFI` (wait for interrupt). Ο βρόχος έχει τοποθετηθεί πριν το run του application και γίνεται έξοδος μόνο όταν η μεταβλητή `should_continue` γίνει true. Έτσι το πρόγραμμα ξεκινά μόνο αφού πατηθεί το κουμπί start. Τέλος, στο αρχείο αυτό προστέθηκε επίσης η συνάρτηση `getChange` η οποία δέχεται ως ορίσματα τα χρήματα που έχει δώσει ο χρήστης και την τιμή του αντικειμένου που επέλεξε, ώστε να υπολογίσει τα ρέστα τα οποία τυπώνει στο debug log.

Στο source file `VendingMode.cpp` έγινε χρήση της `getChange` στα κατάλληλα σημεία ώστε να γίνει ο υπολογισμός των ρεστών. Τα σημεία αυτά, είναι η συνάρτηση `onCoinsInserted` και συγκεκριμένα στην περίπτωση που ο χρήστης έχει ήδη επιλέξει ποτό και η συνάρτηση `onProductButtonPressedEvent` όταν ο χρήστης έχει ήδη εισάγει χρήματα.

3 Προβλήματα που αντιμετωπίστηκαν

Κύριο πρόβλημα που αντιμετωπίσαμε ήταν ότι δεν ήμασταν σίγουροι που πρέπει να καλέσουμε τις συναρτήσεις μας, καθώς μας ζητήθηκε να μην επέμβουμε στον κώδικα c++ (είχαμε κάνει μια αρχική υλοποίηση σε c++, αλλά την αλλάξαμε). Για το λόγο αυτό βάλαμε την `onStartPressed` μέσα στη main πριν την εντολή `app.run()` όπου ξεκινάει και η λειτουργία του μηχανήματος. Την `getChange` όμως την καλούμε μέσα στο `VendingMode.cpp` πριν γίνει η κλήση της συνάρτησης `startReleaseWorkflow()`, δηλαδή πριν το μηχάνημα ελευθερώσει το αντίστοιχο προϊόν. Δε βρήκαμε πως να την καλούμε αλλιώς, ώστε να μην επέμβουμε καθόλου στον κώδικα c++ και να παίρνουμε και τις τιμές για κόστος προϊόντος και χρήματα που έχουν μπει.

Ένα μικρό πρόβλημα που είχαμε στην αρχή, ήταν ότι δεν είχαμε προσθέσει τον κατάλληλο κώδικα στο αρχείο `Application.cpp` με αποτέλεσμα να μη λειτουργεί το κουμπί μας.

Αρχικά είχαμε κάνει υλοποίηση με τροποποίηση του κώδικα c++ με αλλαγές στο `VendingMode`, κατάλληλα `VendingScreen`, όμως όπως μας απαντήσατε και στο e-mail δεν ήταν αυτό που θέλατε, οπότε σπεύσαμε την παραμονή του εργαστηρίου να υλοποιήσουμε την άσκηση εκ νέου.

4 Δοκιμές (testing)

Για τον έλεγχο της υλοποίησης έγινε δοκιμή του κώδικα χρησιμοποιώντας τον συνδυασμό των δύο εργαλείων προσομοίωσης και ακολουθώντας τις διάφορες πιθανές ροές χρήσης του μηχανήματος. Για παράδειγμα, πάτημα οποιουδήποτε άλλου κουμπιού πέραν του START στο ξεκίνημα - το μηχάνημα δεν κάνει τίποτα. Επιλογή προϊόντος και στη συνέχεια εισαγωγή χρημάτων ή και το ανάποδο - τυπώνονται στα ρέστα στο debug log και το μηχάνημα ελευθερώνει το προϊόν.