

## Bonus θέματα στο μάθημα "Εργαστήριο Μικροϋπολογιστών"

Εθνικό Μετσόβιο Πολυτεχνείο - Microlab

Νοέμβριος 2024

Υπεύθυνος : Γεώργιος Αλεξανδρής

Επικοινωνία : galexandris@microlab.ntua.gr

---

## AVR-OS

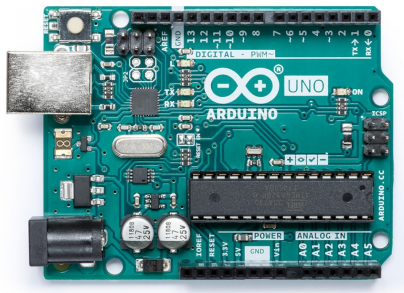
### Exploring the AVR MCU Capabilities in an OS-like Environment

Από την αρχή της μαζικής παραγωγής των υπολογιστικών συστημάτων είχε γίνει κατανοητό ότι η παραγωγή μίας επιπλέον διεπαφής ανάμεσα στην γλώσσα μηχανής (bare metal) και τον χρήστη είναι απαραίτητη, διευκολύνοντας από τις καθημερινές δραστηριότητες των χρηστών, όπως η συγγραφή κειμένων και οι υπολογισμοί, έως και τις απαιτήσεις υψηλών επιπέδων, όπως προσομοιώσεις και μεταγλωττίσεις προγραμμάτων. Αυτή η επιπλέον διεπαφή είναι γνωστή ως το "Λειτουργικό Σύστημα", μία σύνθεση αλγορίθμων και configuration files, επιτρέποντας μία τυποποιημένη εμπειρία χρήστη για κάθε μικροεπεξεργαστή/μικροελεγκτή. Σε αυτή την άσκηση θα κλιθείτε να κατασκευάσετε μία τέτοια διεπαφή για τον μικροελεγκτή του εργαστηρίου, κάνοντας χρήση πολλών υλοποιήσεων που είδατε στις εργαστηριακές ασκήσεις.

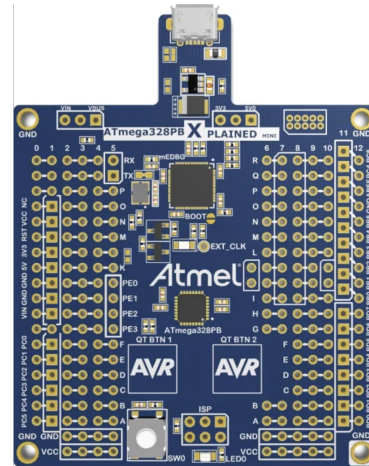
### AVR

Μία από τις πιο διαδεδομένες και πιο παλιές αρχιτεκτονικές μικροελεγκτών που υπάρχουν. Δημιουργήθηκε το 1996 από την Atmel και από το 2016 συντηρείται από την Microchip. Έχει παράξει πολλούς μικροελεγκτές κυρίως 16 και 32 bit. Από τις σημαντικότερες εφαρμογές του είναι η χρήση του στην πλακέτα του Arduino UNO μέχρι και το Rev.3 (το Rev.4 έκανε μετάβαση σε ARM μικροελεγκτή). Έχει χρησιμοποιηθεί σε πάρα πολλές συσκευές κυρίως σαν περιφερειακός μικροελεγκτής μέχρι και σήμερα (η χρήση του δηλαδή είναι για handling side task σε μεγαλύτερα συστήματα). Η αρχιτεκτονική του είναι τύπου RISC. Στις εργαστηριακές ασκήσεις έχετε έρθει αρκετά σε επαφή τόσο με την αρχιτεκτονική αυτή καθώς και με τις δυνατότητες των μικροελεγκτών. Ειδικότερα σε όλες τις εργαστηριακές ασκήσεις έγινε χρήση του μικροελεγκτή **ATmega328PB**.

Ο προγραμματισμός αυτών των μικροελεγκτών μπορεί να γίνει με πολλούς τρόπους εξαρτώντας από την εκάστοτε πλακέτα ανάπτυξης. Στα πλαίσια αυτής της άσκησης θα γίνει χρήση του **MPLAB X IDE** και της πλακέτας **ntuAboard** όπως ακριβώς και στις υπόλοιπες εργαστηριακές ασκήσεις μέχρι τώρα.



(a) Arduino UNO r3



(b) ATmega328PB Xplained

Figure 1: Παραδείγματα AVR based development boards

## Περιφερειακά

Κατά την διάρκεια των εργαστηριακών σας ασκήσεων χρησιμοποιήσατε

- Διάφορα περιφερειακά (Keypad, LED οθόνη, αισθητήρες, ESP-WiFi extender)
- Τον PCA9555 IO Expander
- Πρωτόκολλα για συνδεσμολογίες των παραπάνω (I2C, TWI, UART)

Στα πλαίσια αυτής της άσκησης, δοσμένες συναρτήσεις για τα παραπάνω θα πρέπει να υλοποιηθούν με κατάλληλο τρόπο, ώστε μέσα από αυτές να προκύψουν κατάλληλα "systemcalls", δηλαδή κατηγοριοποιημένες εντολές που πάντα θα καλούν τις συγκεκριμένες συναρτήσεις. Ουσιαστικά, βασισμένοι στις υλοποιήσεις που ήδη έχετε πραγματοποιήσει, θα κατασκευαστούν συναρτήσεις περιτύλιξής τους (wrapper functions), καλώντας τις με συγκεκριμένες παραμέτρους. Η οργάνωση των εντολών καθώς και το πως θα "εξυπηρετώνται" από το πυρήνα του συστήματος αφήνετε σαν επιλογή στον φοιτητή.

## Προδιαγραφές του OS

Για την διεπαφή του χρήστη με το λειτουργικό σύστημα θα γίνει η χρήση τερματικού σε περιβάλλον εντολών (CLI- Command Line Interface). Για την οπτικοποίηση των ενεργειών του λειτουργικού συστήματος θα γίνει χρήση του πρωτοκόλλου **VGA** σε οθόνη. Για την είσοδο των εντολών θα πρέπει να γίνει η χρήση του πρωτοκόλλου **USB** μέσω συνδεδεμένης συσκευής πληκτρολογίου. Οι υποστηριζόμενες εντολές του λειτουργικού σας συστήματος θα πρέπει να είναι οι εξής:

- **setuser <username>**: Αλλαγή του ονόματος χρήστη. Προσοχή: Το όνομα χρήστη πρέπει να παραμένει γραμμένο στην πλακέτα ακόμα και μετά το reset.
- **setpass <password>**: Αλλαγή του κωδικού πρόσβασης. Προσοχή: Ο κωδικός πρόσβασης πρέπει να παραμένει γραμμένος στην πλακέτα ακόμα και μετά το reset.
- **lcd <text>**: Εμφάνιση κειμένου στην οθόνη LCD.
- **gpiow <PORTx> <0xHH>**: Εγγραφή τιμής σε GPIO port.

- **gpior <PORTx>**: Ανάγνωση τιμής από GPIO port. Προσοχή: Η τιμή πρέπει να εμφανίζεται στο τερματικό.
- **temperature** : Ανάγνωση της θερμοκρασίας από τον αισθητήρα.
- **potentiometer** : Ανάγνωση της τιμής του ποτενσιόμετρου.

Για την υλοποίηση αυτών των μεθοδολογιών υπάρχουν οι εξής παραδοχές:

- Προφανώς και ο AVR δεν είναι σχεδιασμένος για την υποστήριξη ενός ολοκληρωμένου OS, πολλές από τις υλοποιήσεις είναι λογικό να μην μπορούν να υλοποιηθούν μαζικά. Γι αυτό τον λόγο στις προδιαγραφές του OS οι λειτουργίες που είναι με πορτοκαλί είναι προαιρετικές, αρκεί να υλοποιηθεί τουλάχιστον μία από αυτές, αιτιολογώντας επαρκώς γιατί δεν υλοποιήθηκαν οι υπόλοιπες.
- Στα πλαίσια των ασκήσεων εξετάστηκαν και άλλες υλοποιήσεις (WiFi, PWM) οι οποίες δεν αναφέρονται στις προδιαγραφές, εννοείται πως είστε ελεύθεροι να τις υλοποιήσετε, αν το επιλέξετε, με τα κατάλληλα high level commands του OS.
- Στις υλοποιήσεις σας θα πρέπει να προσέξετε τις ανάγκες ως προς την μνήμη.

## Παραδοτέα

Τα παραδοτέα αυτής της άσκησης θα είναι μία παρουσίαση που θα περιλαμβάνει:

- Περιγραφή των προδιαγραφών τις πλακέτας μαζί με τα περιφερειακά που θα χρησιμοποιηθούν.
- Περιγραφή των υλοποιήσεων των περιφερειακών και των συναρτήσεων του λειτουργικού συστήματος.
- Επίδειξη (demo) των δυνατοτήτων του λειτουργικού συστήματος μέσω του τερματικού.
- Ανάλυση αιτιολόγηση των εμποδίων που παρουσιάστηκαν κατά την υλοποίηση.
- Θετικά και αρνητικά της υλοποίησης λειτουργικού συστήματος σε τέτοιου είδους μικροελεγκτές.

## Βιβλιογραφία-Βοηθητικό υλικό

- [USB Protocol](#)
- [USB Keyboards](#)
- [A quick Introduction to VGA](#)(Προφανώς υπάρχουν και άλλα tutorial)
- [VGRA Connector](#)
- [Using the AVR EEPROM](#) (Username, Password storage)
- Όλες οι εργαστηριακές ασκήσεις σας μαζί με τα αντίστοιχα documentation.