

Απλοποίηση και αυτοματοποίηση του προγραμματισμού του Arduino με τη βοήθεια ενός φύλλου εργασίας που ονομάζεται WIDE (Worksheet Integrated Development Environment).

Το WIDE μια νέα προσέγγιση προγραμματισμού για το Arduino, που χρησιμοποιεί πίνακες μέσα σε ένα περιβάλλον υπολογιστικών φύλλων εργασίας (LibreOffice). Η μέθοδος αυτή παράγει αυτόματα πάνω από το 80% του κώδικα, απλοποιώντας τον προγραμματισμό. Οι πίνακες είναι εύκολο να συμπληρωθούν, να τροποποιηθούν και να αντιμετωπιστούν προβλήματα. Υπάρχουν ξεχωριστά φύλλα εργασίας για τον ορισμό αναλογικών και ψηφιακών ακροδεκτών, λογικών μεταβλητών, περιγραφή κατάστασης, ροή κατάστασης, εκτύπωση, μενού, επικοινωνία με υπέρυθρες, Bluetooth, επικοινωνία RS485 και αποθήκευση μεταβλητών σε EEPROM. Ο παραγόμενος κώδικας συλλέγεται σε ένα φύλλο εργασίας και στη συνέχεια αντιγράφεται στο Arduino IDE για μεταγλώττιση και αποθήκευση στην ηλεκτρονική κάρτα. Αυτή η μέθοδος έχει εφαρμοστεί με επιτυχία σε διάφορα έργα, επιτρέποντας να δαπανηθεί περισσότερος χρόνος για την επίλυση προβλημάτων παρά για την συγγραφή κώδικα.

Βασικά χαρακτηριστικά του WIDE:

1. Μοντέλο μηχανής πεπερασμένων καταστάσεων (Finite State Model) : Διασπά πολύπλοκες εργασίες σε μικρότερες, διαχειρίσιμες καταστάσεις.
2. Πίνακες λογιστικών φύλλων εργασίας: Εύκολη συμπλήρωση, κατανόηση, τροποποίηση και αντιμετώπιση προβλημάτων σε σύγκριση με την παραδοσιακή συγγραφή κώδικα.
3. Αυτόματη δημιουργία κώδικα: Μειώνει τα σφάλματα και απλοποιεί τη διαδικασία προγραμματισμού.
4. Αρθρωτή δομή: Διαφορετικά φύλλα εργασίας για συγκεκριμένες εργασίες, όπως για διαμορφωση ακροδεκτών, μεταβλητές, καταστάσεις, επικοινωνία κ.λπ.
5. Φιλικό προς το χρήστη περιβάλλον εργασίας: Ενθαρρύνει τη σταδιακή ανάπτυξη και την ευκολότερη αποσφαλμάτωση.

Οφέλη από τη χρήση του WIDE:

1. Απλοποιημένος προγραμματισμός: Ιδανικό για αρχάριους και μη ειδικούς.
2. Μειωμένα σφάλματα: Η αυτόματη παραγωγή κώδικα και η δομημένη προσέγγιση ελαχιστοποιούν τα σφάλματα.
3. Οργανωμένα έργα: Η αρθρωτή δομή προάγει τη σαφήνεια και τη συντηρησιμότητα.
4. Ταχύτερη ανάπτυξη: Λιγότερος χρόνος δαπανάται στο συντακτικό και περισσότερος στη λειτουργικότητα.
5. Βελτιωμένη αποσφαλμάτωση: Η λογική οργάνωση διευκολύνει τον εντοπισμό και τη διόρθωση σφαλμάτων.

Πιθανές εφαρμογές:

1. Εκπαιδευτικό εργαλείο για την εκμάθηση προγραμματισμού Arduino.
2. Έργα χομπίστες για ταχεία κατασκευή πρωτοτύπων και ανάπτυξη.
3. Απλοποίηση του προγραμματισμού μικροελεγκτών για ποικίλα ακροατήρια.

Arduino nano											
Arduino Nano	Nano every		NAME		Type of pin	internal pull up	Type of variable			Create variable of previous value	
pins	microprocessor and motor shield	comments	names in program	array subscript(n)	Input/ output/ PWM	internal pullup	type (boolean, int)	initial value if output or pwm	of the value in the previous cycle	invert the logic of the signal	
0		reserved for USB Rx, or bluetooth	RX		OUTPUT		boolean	0			
1		reserved for USB Tx, or bluetooth	TX		OUTPUT		boolean	0			
2	INT0 timing pin	IR sensor			INPUT	1	boolean	0			
3	PWM A (timer 2 8bit), Motor A speed PWM	PWM value for speed of motor A	SpeedA		PWM		boolean	0			
4		rs485 set to Hi to send, Low to Receive (connect to RE-DE together)			INPUT	1	boolean	0			
5	(timer 0 8 bit, pwm)	Button	Button5		INPUT		boolean	0	1		
6	(timer 0 8 bit, pwm)	Detects obstacle in the doors path	Button6		INPUT		boolean	0	1		

Setup the pins in a table

Describe the operation of the project in states

4	STATE DESCRIPTION		LoopOfOperation	LoopOfChangeState
5	Value of State variable	State name	State description	Set variables or add actions or functions during state, use ";" after each action
6	10	StOpened	Opened (at open position) closing (moving to close position)	ConfigOpened10(); printLCD216(0);
7	20	StClosing	operateMoving();	ConfigRunCCW();
8	30	StClosed	closed (at close position)	ConfigClosed30(); printLCD216(0);
9	40	StOpening	opening (moving to open)	operateMoving(); ConfigRunCW();
10	100	StTripByTimer	program stops needs SoftReset	TriplLed =1; TriplLed =1; ConfigStop(); printLCD216(0);
11	110	StTripByHCurrent	program stops needs SoftReset	TriplLed =1; TriplLed =1; ConfigStop(); printLCD216(0);
12	200	StInteractive	program outputs on screen and waits response by user	ConfigStop(); printLCD216(0); SoftResetB =0;

Describe the flow from state to state

STATE MACHINE					
state number	State name	Trigger (use == for equality check) && for AND and for OR use parenthesis for comparison pairs of and &&	New state number	New State name	Notes
10	StOpened	CombinedControlB	20	StClosing	when mechanically the door fully opens the DoorOpened sensor should be activated
20	StClosing	TimeToDecelerateOn	25	StClosingStop	
30	StClosed	CombinedControlB	40	StOpening	when mechanically the door fully closes the DoorClosed sensor should be activated
40	StOpening	TimeToDecelerateOn	45	StOpeningStop	
20	StClosing	(ControlInputRise Obstacle)	50	StopImmediately	
40	StOpening	(ControlInputRise Obstacle)	50	StopImmediately	

The flow of the program is created automatically

Το WIDE, μια επαναστατική προσέγγιση στον προγραμματισμό του Arduino με βάση τα φύλλα εργασίας, δίνει τη δυνατότητα σε άτομα χωρίς εμπειρία στον προγραμματισμό να εμβαθύνουν στον κόσμο των ηλεκτρονικών και να ζωντανέψουν τις ιδέες τους. Το επιτυγχάνει αυτό αυτοματοποιώντας την πλειονότητα της δημιουργίας κώδικα, επιτρέποντας στους χρήστες να επικεντρωθούν στις βασικές λειτουργίες και τη λογική των έργων τους, αντί να κολλήσουν στις περιπλοκότητες του παραδοσιακού προγραμματισμού.

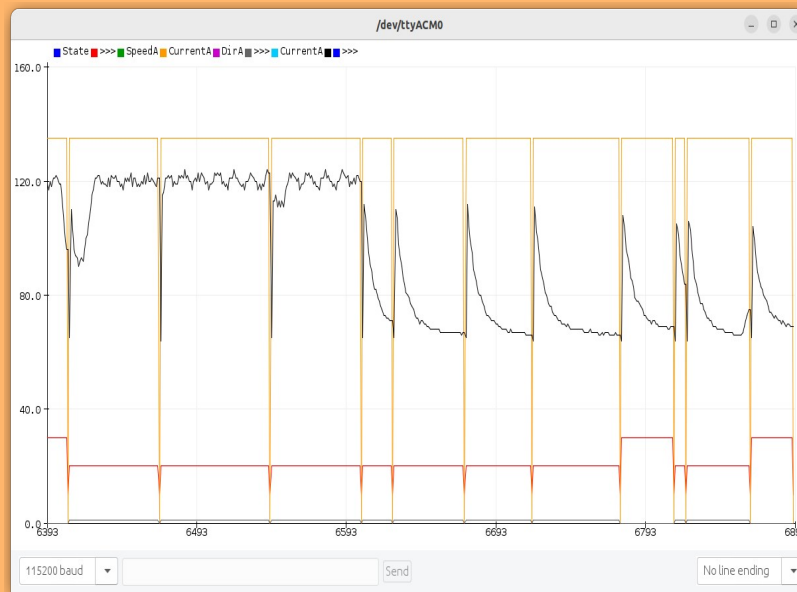
Φανταστείτε να σχεδιάζετε το πρόγραμμα Arduino μέσα από τον οικείο φακό των φύλλων εργασίας, συμπληρώνοντας πίνακες αντί να γράφετε γραμμές κώδικα. Το WIDE προσφέρει ειδικά υπολογιστικά φύλλα εργασίας για διάφορες πτυχές του προγραμματισμού, από τον ορισμό ακροδεκτών και τη δημιουργία κανόνων λογικής μέχρι τη δημιουργία διεπαφών χρήστη. Αυτή η διασθητική διεπαφή προάγει την οργάνωση και προάγει τη βαθύτερη κατανόηση των εσωτερικών λειτουργιών του προγράμματος, ακόμη και για όσους δεν έχουν προηγούμενες γνώσεις προγραμματισμού.

Μόλις ολοκληρώσετε τα φύλλα εργασίας, το WIDE μεταφράζει απρόσκοπτα το σχέδιό σας στον αντίστοιχο κώδικα Arduino, έτοιμο να φορτωθεί στην πλακέτα σας και να δώσει ζωή στο έργο σας. Αυτό όχι μόνο εξοικονομεί πολύτιμο χρόνο και εξαλείφει την πιθανότητα συντακτικών λαθών που συχνά ταλαιπωρούν τον παραδοσιακό προγραμματισμό, αλλά δίνει επίσης τη δυνατότητα στους χρήστες να επαναλαμβάνουν και να πειραματίζονται πιο ελεύθερα.

Τα οφέλη του WIDE εκτείνονται πολύ πέρα από την ευκολία χρήσης. Χειριζόμενο τη δημιουργία κώδικα, απαλλάσσει τους χρήστες από τις πολυπλοκότητες του προγραμματισμού, επιτρέποντάς τους να επικεντρωθούν στην ουσία του έργου τους: στην επίλυση ενός προβλήματος, στη δημιουργία κάτι νέου ή απλώς στην έκφραση της δημιουργικότητάς τους. Αυτή η μετατόπιση της εστίασης ευνοεί την καινοτομία και καθιστά το Arduino πιο προσιτό σε ένα ευρύτερο κοινό, συμπεριλαμβανομένων των μαθητών, των χομπίστες και όλων όσων έχουν μια σπίθα περιέργειας και επιθυμούν να εξερευνήσουν τον κόσμο της ηλεκτρονικής. Έτσι, αν είχατε ποτέ μια ιδέα για ένα ηλεκτρονικό έργο αλλά νιώθατε εκφοβισμένοι από την προοπτική του προγραμματισμού, το WIDE είναι εδώ για να σας δώσει τη δυνατότητα να μετατρέψετε αυτή την ιδέα σε πραγματικότητα.

Προβολή μεταβλητών από το Arduino σε γραφήματα για βαθύτερη κατανόηση

View variables from Arduino in graphs get deeper understanding



Ενότητες κατάρτισης για το WIDE

Ενότητα 1

1. Απλό έργο με χρήση καταστάσεων (παράδειγμα Blink)
2. Μοναδικά αναλογικά σήματα εισόδου
3. Μεταβαλλόμενη ένταση των εξόδων, όπως ο έλεγχος της έντασης μιας λάμπας.
4. Κατάσταση βαθμονόμησης και κατάσταση λειτουργίας.
5. Ανίχνευση ανερχόμενης ακμής κουμπού (προηγούμενες τιμές σημάτων)
6. Ανύψωση των σημάτων εξόδου
7. Επικοινωνία μικροελεγκτή και υπολογιστή (οδηγίες από τον υπολογιστή)
8. Προσθήκη δέκτη IR για να δέχεται οδηγίες από ελεγκτές IR
9. Γραφική παράσταση των μεταβλητών του Arduino
10. Έλεγχος κινητήρα
11. Ένα απλό έργο, ένα έργο όπως ένα παντζούρι πόρτας

Ενότητα 2 Πέρα από την εισαγωγή

- 1) Ένα έργο με περισσότερες από 5 καταστάσεις
- 2) Καταστάσεις προστασίας έκτακτης ανάγκης που ενεργοποιούνται σε οποιαδήποτε κατάσταση για την προστασία του συστήματος
- 3) Επικοινωνία με Bluetooth
- 4) Επικοινωνία με RS485
- 5) Προστασία των μεταβλητών χρήστη στην EEPROM
- 6) Προσθήκη μενού στο πρόγραμμά σας
- 7) Προσθήκη οθόνης I2C LCD
- 8) Κανονική λειτουργία - Λειτουργία έκτακτης ανάγκης

- 9) Εντολές με χρονοδιακόπτες
- 10) Arduino IOT χρησιμοποιώντας το Arduino Cloud
- 11) Σχεδιασμός ταμπλό στο Arduino Cloud

Contact

Μέθοδος προγραμματισμού του Arduino με τη χρήση του προεπεξεργαστή WIDE

Πρόσωπο επικοινωνίας

Νίκος Ιωάννου

99918088

mckluino@gmail.com