**Ενσωματωμένα Συστήματα – Εργασία 2023-2024**

ΣΤΑΔΙΟ Γ:

**Δήλωση ομάδας:**

Βελαώρα Μαρία, 5ο Έτος

Σενή Εβελίνα, 5ο Έτος

**Θεωρητική προσέγγιση:**

Για την ανάπτυξη του ενσωματωμένου συστήματος μας (smart lighting), χρησιμοποιήσαμε το online simulator Tinkercard, ένα περιβάλλον σχεδίασης και προσομοίωσης ενσωματωμένων συστημάτων.

Στο σύστημα που αναπτύξαμε, έχουμε χρησιμοποιήσει τα εξής στοιχεία:

* Arduino
* Αισθητήρα κίνησης
* Αισθητήρα φωτός
* Relay SPTD
* Τροφοδοσία
* Λαμπτήρα



Το στοιχείο Relay SPDT το χρησιμοποιήσαμε ώστε να εξασφαλίσουμε τον αυτοματισμό των φώτων, σύμφωνα με τις συνθήκες που έχουμε ορίσει στο σύστημά μας. Για παράδειγμα, ανίχνευση κίνησης και φωτεινότητας.

Γενικά το στοιχείο αυτό γνωρίζουμε ότι λειτουργεί ως ένας ηλεκτρομηχανικός διακόπτης και στο σύστημα μας παίρνει την θέση της συσκευής ελέγχου.

Στο ενσωματωμένο σύστημα που αναπτύξαμε ο αισθητήρας κίνησης λειτουργεί κανονικά, εντός των τιμών που του έχουμε ορίσει.

Συγκεκριμένα λαμβάνει ως μέγιστη δυνατή τιμή το 150, δηλαδή

για Distance <= 150 φως ΟΝ, ενώ

για Distance > 150 φως OFF.

Σε αυτό το σύστημα, ο αισθητήρας φωτός λειτουργεί και όταν σέρνουμε την μπάρα από το πλήρες φως σε καθόλου φως. Τότε ο λαμπτήρας ανάβει. Το μόνο πρόβλημα που αντιμετωπίζουμε στον αισθητήρα αυτόν είναι ότι όταν η μπάρα είναι στο καθόλου φως, τότε ο λαμπτήρας αναβοσβήνει και δεν μένει σταθερά αναμμένος, πρόβλημα που οφείλεται στον κώδικα που έχουμε συντάξει και δεν βρήκαμε τρόπο λύσης.

Σημειώνεται ότι για να λειτουργήσει το σχέδιο που παρουσιάζεται και παρακάτω χρειάστηκε να συντάξουμε και τον κώδικα σε γλώσσα C++ που αναλύει τις λειτουργίες των επιμέρους στοιχείων.

Ο κώδικας αυτός έχει την εξής **λειτουργική δομή**:

* **Αρχικοποιούνται αισθητήρες και έξοδοι**: Ορίζει τα pins για τον αισθητήρα κίνησης και το relay. Επίσης, ορίζει τον terminal1 ως αναλογικό εισαγωγικό ακροδέκτη για τον αισθητήρα φωτός.
* **Ελέγχεται η φωτεινότητα**: Διαβάζει την τιμή της φωτεινότητας από τον αισθητήρα φωτός και ενεργοποιεί/απενεργοποιεί το Relay ανάλογα.
* **Ελέγχεται η απόσταση**: Παίρνει το σήμα από τον αισθητήρα κίνησης και υπολογίζει την απόσταση με βάση τον χρόνο που επιστρέφει το σήμα.

**Ο Κώδικας**:

#define trig 3

#define echo 2

#define relay 8

int terminal1=A5;

int terminal2\_value;

void setup() {

pinMode(terminal1, INPUT);

pinMode(trig, OUTPUT);

pinMode(echo, INPUT);

pinMode(relay, OUTPUT);

}

void loop() {

long duration, distance;

terminal2\_value=analogRead(terminal1);

if (terminal2\_value>512)

{

digitalWrite(relay, LOW);

}

if(terminal2\_value<512)

{

digitalWrite(relay, HIGH);

}

digitalWrite(trig, LOW);

delay(20);

digitalWrite(trig, HIGH);

delay(20);

digitalWrite(trig, LOW);

duration = pulseIn(echo, HIGH);

distance = duration\*0.0343/2;

if (distance <= 150)

{

digitalWrite(relay, HIGH);

delay(2000);

}

if (distance >= 151)

{

digitalWrite(relay, LOW);

delay(2000);

}

}

**Εκτίμηση (πειραματικών μετρήσεων):**

Τι περιμένουμε μετά την υλοποίηση του ενσωματωμένου συστήματος που υλοποιήσαμε;

Διεξάγουμε μια σειρά δοκιμών για τον έλεγχο της αξιοπιστίας και ακρίβειας των συσκευών μας.

Αναμένουμε ότι ο αισθητήρας κίνησης, θα ανιχνεύει και θα μας δίνει ακριβείς μετρήσεις για την καθορισμένη απόσταση των 150 cm. Όσον αφορά τα διαφορετικά επίπεδα φωτεινότητας, εκτιμάμε ότι ο αισθητήρας φωτός θα ρυθμίζει το φως του λαμπτήρα, ο οποίος θα ανάβει όταν υπάρχει στο περιβάλλον χαμηλή φωτεινότητα ή και καθόλου φως, και θα σβήνει σε συνθήκες υψηλής φωτεινότητας από το περιβάλλον.

Ακόμη εξετάζεται η απάντηση του συστήματος σε συνθήκες που ποικίλλουν, από διακυμάνσεις στην τροφοδοσία, στις πιθανές επιδράσεις από άλλες συσκευές.

Οι μετρήσεις αυτές βοηθούν να κατανοήσουμε την αξιοπιστία και την αποδοτικότητα του συστήματος σε όσο πιο πραγματικές συνθήκες λειτουργίας γίνεται.

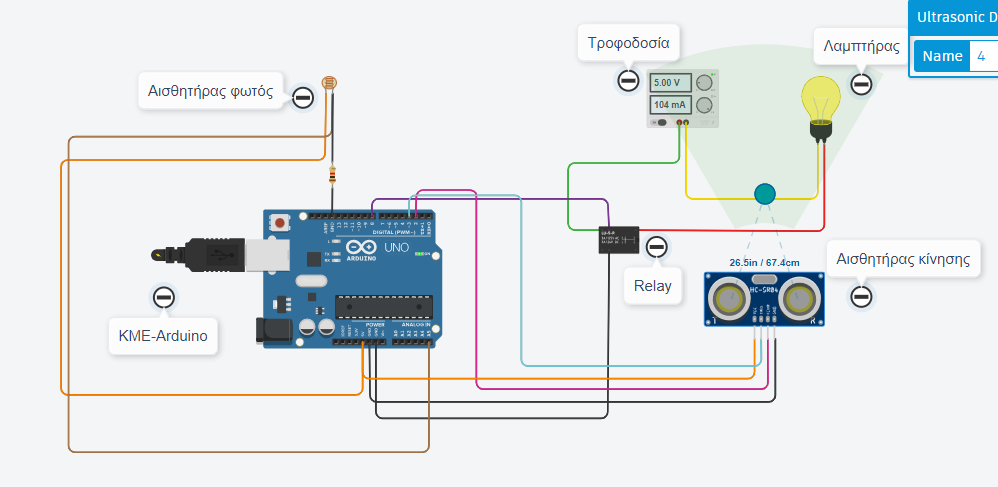
**Διαδικασίες ελέγχου ορθής λειτουργίας:**

* Το συνολικό σύστημα έχει ως εξής:

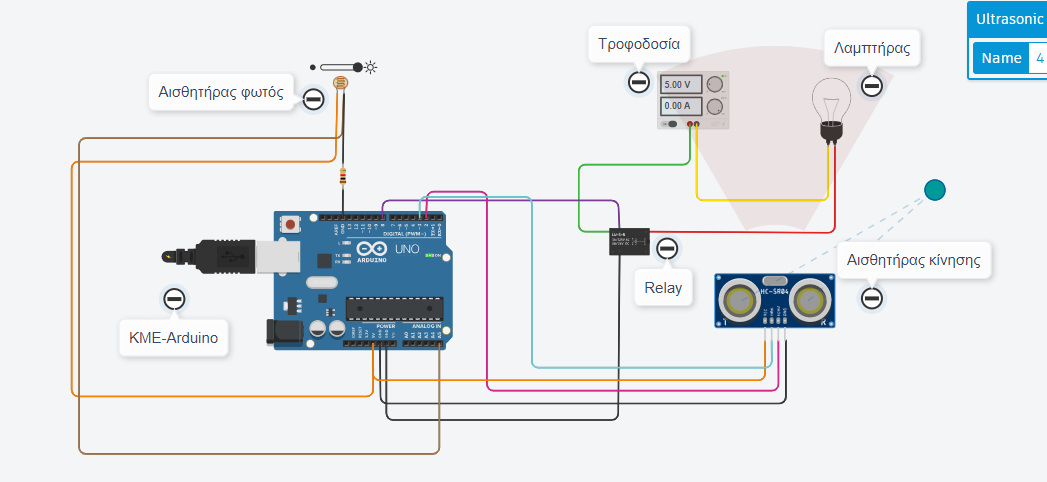
Εικόνα που περιέχει κείμενο, διάγραμμα, στιγμιότυπο οθόνης, Σχέδιο

Περιγραφή που δημιουργήθηκε αυτόματα

* Παρουσίαση του αισθητήρα κίνησης ON:



* Παρουσίαση του αισθητήρα κίνησης OFF:



* Παρουσίαση του αισθητήρα φωτός ON:

Εικόνα που περιέχει κείμενο, διάγραμμα, στιγμιότυπο οθόνης, Σχέδιο

Περιγραφή που δημιουργήθηκε αυτόματα

* Παρουσίαση του αισθητήρα φωτός OFF:

Εικόνα που περιέχει κείμενο, διάγραμμα, στιγμιότυπο οθόνης, Σχέδιο

Περιγραφή που δημιουργήθηκε αυτόματα

* Το συνολικό σύστημα, ως κύκλωμα:

Εικόνα που περιέχει διάγραμμα, Σχέδιο, γραμμή, τεχνικό σχέδιο

Περιγραφή που δημιουργήθηκε αυτόματα

Μπορείτε κι εσείς να τρέξετε το εν λόγω πρόγραμμα και να δείτε τον τρόπο λειτουργίας του ακολουθώντας τον παρακάτω σύνδεσμο:

<https://www.tinkercad.com/things/fJI6TYndAay-smart-lighting>

**Προτάσεις / Βελτιστοποιήσεις:**

* Μια καλή προσθήκη, είχαμε την ιδέα όμως δεν καταφέραμε την υλοποίηση, θα ήταν μια σχετική **εφαρμογή κινητού**. Προσεγγίσαμε θεωρητικά το ζήτημα εμπλέκοντας την έννοια του IoT, για το πόση διευκόλυνση θα παρείχε μια τέτοια δυνατότητα. Ο χρήστης θα μπορεί όπου κι αν βρίσκεται να έχει έλεγχο του φωτισμού στο σπίτι του, τόσο για εξοικονόμηση (για παράδειγμα φεύγεις από το σπίτι και δεν θυμάσαι αν φρόντισες να κλείσεις τα φώτα), όσο και για προγραμματισμό για άλλους προσωπικούς λόγους (για παράδειγμα, έχεις κατοικίδιο και θες να έχεις ανοιχτό κάποιο συγκεκριμένο φως).
* Το παραπάνω θα μπορούσε να συνδυαστεί και με την λειτουργία του **χρονοδιακόπτη**, είτε φυσικά στο σπίτι, σαν οντότητα εντός του smart home συστήματος, είτε σαν επιλογή εντός της εφαρμογής που προαναφέραμε. Σαν έννοια αυτό που προσφέρει είναι, ευκολία στην καθημερινότητα, αφού ορίζεις για παράδειγμα, πότε θες να ανάψουν τα φώτα και σιγά-σιγά να ξυπνήσεις ή να κλείσουν σε όλα τα δωμάτια για συγκεκριμένες ώρες κ.ο.κ.
* Μια άλλη καλή επιλογή, που επίσης προσπαθήσαμε να συμπεριλάβουμε ανεπιτυχώς, είναι η λειτουργία **RGB**. Να μπορούμε δηλαδή να επιλέξουμε για τον φωτισμό μας οποιοδήποτε χρωματισμό από μια γκάμα συνδυασμών των βασικών χρωμάτων : κόκκινο, πράσινο, μπλε. Αυτό θα δώσει μεταφορικά και κυριολεκτικά χρώμα στην καθημερινότητα μας και θα δώσει λίγο περισσότερο «ενδιαφέρον» στο σύστημα μας, τραβώντας και την προσοχή του καταναλωτή σε μια ανταγωνιστική αγορά.
* Απαραίτητος ίσως και να είναι και ένας **ανεμιστήρας**. Αυτό, ξεπερνώντας το «αισθητικό» κομμάτι θα βοηθούσε πρακτικά στην συντήρηση του συστήματος μας και θα μας προφύλασσε από την υπερθέρμανση του και τις πιθανές φθορές.

**Παρουσίαση σχετικού εργαλείου:**

Το Tinkercard παρέχει ένα εικονικό περιβάλλον το οποίο μας επιτρέπει να αναπτύξουμε και να προσομοιώσουμε το κύκλωμά μας. Ο προσομοιωτής αυτός βοηθάει αρκετά στο να εντοπίσουμε και να διορθώσουμε τυχόν λάθη κατά των σχεδιασμό του ενσωματωμένου μας. Το πρόγραμμα αυτό επιτρέπει, επίσης, την ανάπτυξη και την δοκιμή του κώδικα για την λειτουργία του Arduino και καθορίζει έτσι διάφορες παραμέτρους και επιμέρους λειτουργίες που θα πρέπει να επιτελεί το σύστημα μας. Μετά από την σχεδίαση του συστήματος στην πλατφόρμα αυτή μπορούμε να προχωρήσουμε και στην φυσική του υλοποίησή.

**Βιβλιογραφία**

* Tinkercard

<https://www.commonsense.org/education/reviews/tinkercad>

* What is SPDT Switch : Working & Its Applications

<https://www.elprocus.com/spdt-switch/>

* UNO R3

<https://docs.arduino.cc/hardware/uno-rev3>

* Smart switches or smart bulbs? How to choose the right smart lighting or your home, Jennifer Pattison

<https://www.theverge.com/23156554/smart-bulbs-switch-lighting-guide-how-to>