

Partie 1 Decouverte du produit et de la problematique technique

Ce qu'il se passe si la lumiere est allumee a 100% avec assez de lumiere naturelle
Actuellement les lumieres artificielles restent allumees a pleine puissance meme quand il y a suffisamment de lumiere naturelle provenant des fenetres.

Cela entraine un gaspillage d'energie inutile car les occupants n'ont pas besoin de la lumiere artificielle pour avoir un bon confort visuel.

Valeurs minimale et maximale de luminosite a mesurer

D'apres le cahier des charges :

Valeur minimale : 0 lux obscurite totale

Valeur maximale : au moins 300 lux niveau d'eclairage requis au niveau des tables

La resolution est de 8 bits minimum soit 256 valeurs differentes entre 0 et la valeur max.

Partie 2 Conception

Choix du capteur de luminosite

Le capteur retenu est le capteur de luminosite a photodiode Grove.

Justification :

Il delivre un signal analogique lineaire entre 0 et Vcc compatible avec analogRead() de l'Arduino.

Il est compatible avec une alimentation 3.3 a 5 Vcc soit directement avec la carte Arduino.

Son temps de reponse de 20 a 30 ms est suffisant pour une regulation d'eclairage.

Capteurs ecartés :

Capteur a photoresistance Grove : necessite un pont diviseur de tension supplementaire plus complexe a mettre en oeuvre.

Capteur BH1750 ADA4681 : interface I2C non disponible sur TinkerCAD pour la simulation ecarte pour cette raison.

Integration au schema structurel

Le capteur de luminosite Grove est connecte a l'entree analogique A0 de l'Arduino.

Broche OUT du capteur = A0 de l'Arduino

Broche VCC du capteur = 5V de l'Arduino

Broche GND du capteur = GND de l'Arduino

La LED reste connectee sur la broche D3 (PWM) comme dans le programme existant.

Partie 3 Simulation

Protocole de simulation

Pour faire varier le parametre de luminosite :

1. Lancer la simulation
2. Activer la detection de presence capteur PIR a HIGH
3. Diminuer la valeur de luminosite du phototransistor verifier que la LED s'allume
4. Augmenter la valeur de luminosite verifier que la LED s'eteint
5. Supprimer la presence verifier que la LED reste eteinte dans tous les cas

Interpretation des resultats de simulation

Luminosite insuffisante et presence detectee = LED allumee. Conforme.

Luminosite suffisante et presence detectee = LED eteinte. Conforme.

Pas de presence = LED eteinte quelle que soit la luminosite. Conforme.

Conclusion :

La simulation valide le schema structurel et le programme. Le systeme se comporte conformement au cahier des charges dans tous les cas testes.

Partie 4 Experimentation

Cablage du sous-systeme

Cablage realise avec la carte Arduino le shield Grove et les modules Grove :

Capteur PIR Grove = broche D4

LED Grove = broche D3

Capteur de luminosite Grove broche A0

Alimentation 5V et GND via le shield Grove

Protocole experimental

Pour faire evoluer la luminosite mesuree :

Couvrir le capteur avec la main pour simuler une faible luminosite.

Eclairer le capteur avec une lampe pour simuler une forte luminosite.

Utiliser le luxmetre pour mesurer la vraie valeur en lux.

Pour valider la comparaison luminosite / seuil :

1. Lancer le programme sur l'Arduino.
2. Activer la presence en passant devant le capteur PIR.
3. Faire varier la lumiere et observer le comportement de la LED.
4. Relever les valeurs du luxmetre et les comparer au seuil programme.

Tests de la maquette

Presence detectee luminosite faible = LED allumee. Conforme.

Presence detectee luminosite suffisante = LED eteinte. Conforme.

Pas de presence luminosite faible = LED eteinte. Conforme.

Pas de presence luminosite suffisante = LED eteinte. Conforme.

Conclusion generale

Le systeme fonctionne correctement et repond au cahier des charges.

La LED s'allume uniquement quand une presence est detectee ET que la luminosite est inferieure au seuil de 300 lux.

Quand il y a assez de lumiere naturelle la LED reste eteinte ce qui evite le gaspillage d'energie.

La solution proposee repond bien a la problematique du sujet.