Documentatie

Proiectare cu microprocesoare

NUME STUDENT: Ungureanu Iulia Iarina

GRUPA: 30234

Contents

[Documentatie 1](#_Toc124454377)

[Proiectare cu microprocesoare 1](#_Toc124454378)

[1. Introducere 2](#_Toc124454379)

[2. Studiu bibliografic 2](#_Toc124454380)

[3. Soluție propusa si implementare 3](#_Toc124454381)

[4. Testare 4](#_Toc124454382)

[5. Concluzie 5](#_Toc124454383)

1. Introducere

Scopul acestei lucrări este realizarea unui dispozitiv automatizat care sa acționeze la producerea unui zgomot. Dispozitivul consta intr-un ansamblu format dintr-o cutie in care sunt plasate un carusel si o matrice luminoasa. La producerea zgomotului capacul cutiei se deschide, caruselul este ridicat si începe sa se rotească, matricea este de asemenea ridicata si afișează un mesaj luminos. Pentru realizarea acțiunilor de mai sus am folosit o plăcută arduino MEGA 2560, 3 motoare, un buzer, o matrice de 64 de led-uri si un senzor de sunet.

Am ales acest proiect deoarece este ușor de extins si folosește componente variate precum :motoare, led-uri, senzori si buzere. Realizarea de dispozitive cu comanda sonora își găsește aplicabilitate in numeroase domenii din viată reala. Spre exemplu ele pot ajuta la sporirea calității vieții persoanelor cu dizabilități, la automatizarea unor procese legate de conducerea autovehiculelor sau aeronavelor, la dotarea caselor inteligente etc.

Dispozitivul realizat nu este replica unuia existent, ci îmbina in unul singur mai multe concepte diferite, precum aparate cu senzori de sunet, cutii inteligente, decorațiuni luminoase, diferite mecanisme mobil.

1. Studiu bibliografic

* Implementarea deschiderii cutiei:

Solutie 1: [automatic closing and opening box using arduino and ultrasonic sensor | smart box - YouTube](https://www.youtube.com/watch?v=bbn1_nCdKKo) – utilizarea unui servo motor care sa tragă de capac din exteriorul cutiei;

Solutie 2: [Servo opening box lid arduino - YouTube](https://www.youtube.com/watch?v=D8KiOvg8rpM) – utilizarea unui servo motor care sa împingă din interiorul cutiei capacul in sus;

|  |  |  |
| --- | --- | --- |
|  | Solutia1 | Solutia2 |
| Dificultate de implementare | Usor | Mai complex |
| resurse | Un servo motor | |
| Îndeplinește nevoile utilizatorului | Capacul poate sa fie prea lat sau prea greu pentru a fi ridicat complet de servo | Capacul poate fi prea greu pentru a fi ridicat. (se pot utiliza doua servo motoare) |
| Consum de putere | scăzut | |

Am ales sa implementez soluția 2 deoarece cutia de lemn are capacul prea lat pentru a fi ridicat cu soluția 1.

* Îndeplinirea condiției de deschidere:

Solutie1: [Turn on and Off a Lamp Clapping Twice, Using Arduino : 4 Steps - Instructables](https://www.instructables.com/Turn-on-and-Off-a-Lamp-Clapping-Twice-Using-Arduin/) – cu sensor de sunet;

Solutie2: [automatic closing and opening box using arduino and ultrasonic sensor | smart box - YouTube](https://www.youtube.com/watch?v=bbn1_nCdKKo) – cu senzor de distanta;

Solutie3: [DON'T TOUCH BOX !!! - Useless box with arduino - DIY - YouTube](https://www.youtube.com/watch?v=5BRDxsO_TuQ&t=770s) – cu un switch;

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Solutia1 | Solutia2 | | Solutia3 |
| Dificultate de implementare | greu | mediu | | usor |
| resurse | Sound senzor | ultrasonic sensor arduino | | switch |
| Îndeplinește nevoile utilizatorului | da | da | | da |
| Consum de putere |  | |  | 9.1W |

Am ales sa implementez deschiderea automata a cutiei cu senzorul de sunet deoarece pare foarte interesant.

1. Soluție propusa si implementare

Cutiuța muzicala este formata din: o cutie de lemn de 15x20 cm, un carusel de 11 cm , o matrice de 64 de led-uri 8x8 RGB, 4 servo motoare, un senzor de sunet, doua butoane, un buzzer si o placa arduino MEGA 2560.

Am ales un servo SG90 (180 de grade) cu piese metalice pentru a ridica capacul de lemn al cutie la un unghi de 90 de grade si servo-uri de plastic pentru celelalte sarcini. La început am dorit ca matricea de leduri sa fie atașata de capac, dar cum capacul era deja foarte greu servo-ul nu a mai reușit sa îl ridice total, lăsând cutia doar pe jumătate deschisa . Pentru a rezolva aceasta problema am ales sa mai adaug un motor servo pentru ridicarea matricei din cutie la un unghi de 90 de grade. Matricea a trebuit sa fie atașata de servo cu coltul din stanga-sus si primul ei pixel a fost declarat in matrice „Adafruit\_NeoMatrix” ca *NEO\_MATRIX\_BOTTOM* si *NEO\_MATRIX\_RIGHT.*

Pentru rotirea caruselul, acesta a fost lipit pe un motor, iar pentru ridicarea sa din poziția culcat am utilizat un servo care se v-a rotii la 90 de grade. Dimensiunile caruselului erau la limita pentru cutia aleasa, lățimea era exact cat adâncimea cutiei, iar înălțimea era puțin mai mica de cat lățimea cutiei. Pentru a o ridica aveam nevoie de spațiu liber si pentru celelalte doua motoare, deci am poziționat caruselul in diagonala cutiei.

Senzorul de sunet a fost montat in partea din stânga a cutiei îndreptat spre utilizator pentru a putea înregistra eficient bătăile din palme. In partea dreapta a fost montat buzzer-ul.

Butonul roșu este folosit pentru a închide cutia in orice moment. Pinul acestuia are setat o întrerupere cu funcția „RedButoonPin”, care atribuie variabilei stopBox valoarea „1”.

Pentru a executa si jocul de lumini si muzica de fundal am intercalat codurile : ledurile își vor schimba culoarea la intervalul notei muzicale respective. Parcurgem fiecare nota din cântec si in acelasi timp mutam pixeli din matrice cu o poziție la stânga de jos in sus. Primul set de culori din matrice este roșu si verde, după ce acesta se termina trecem la un alt set de culori (verde deschis si magenta). Daca variabila *stopBox* este egala cu „1” asta înseamnă ca butonul roșu a fost apăsat si ieșim din ciclu. La final avem un delay egal cu durata notei curente.

In loop-ul principal citim valoarea de pe senzorul de sunet, daca aceasta este mai mare ca 100 înseamnă ca deschidem cutia cu funcția *OpenBox*, care ridica capacul si matricea. După apelarea si terminarea funcției start aceasta închide cutia cu funcția *CloseBox*. Funcțiile *up* si *down* se ocupa cu ridicarea si coborârea caruselului, acestea sunt apelate la începutul si finalul funcției Start. Tot in aceasta funcție atașam servo-ul pentru rotire si adăugam algoritmul de generare al notelor muzicale si al matricei de leduri. După coborârea caruselului afișam mesajul „Craciun fericit!” cu ajutorul funcției *printTextMatrix* , închidem cutia si atribuim variabilei stopBox valoarea „0”.

Funcția *printTextMatrix* primește ca argument culoarea mesajului de afișat setata cu „setTextColor”. Intr-un loop setam toate led-urile matricei ca stinse, punem cursorul pe rândul x coloana 0, printam mesajul cu *print* si îl afișam cu *show*. Variabila x trebuie sa descrească pentru a afișa textul de la stânga la dreapta, iar delay-ul de la final reprezintă viteza cu care este afișat.

Diagram

Description automatically generated

1. Testare

După montarea matricei de leduri am realizat ca consumul de energie a crescut si toate servo-urile atașate la plăcuta arduino au început sa tremure (“flicker”) la intervale diferite. Am încercat sa setez matricea la o luminozitate mai joasa (.setBrightness(10)), dar nu a rezolvat problema. După o analiza amănunțită a codului am realizat ca matricea mea nu avea setate opțiunile corecte pentru modelul respectiv: NEO\_KHZ800 800 KHz (majoritatea produselor NeoPixel w/WS2812 LEDs). Am înlocuit codul cu „NEO\_KHZ400” pentru produsele FLORA pixels (driver WS2811).

Testare pentru senzorul de sunet setat la sensibilitate maxima:

|  |  |
| --- | --- |
| Valoare de tip analogic citita de senzorul de sunet | Sunete detectate |
| 0-10 | Șoptit, pocnit din degete, bătut din palme, vorbit, ciocănit |
| 10-50 | pocnit din degete, vorbit , bătut din palme, ciocănit |
| 50-100 | Vorbit tare, bătut din palme , ciocănit |
| 100 -150 | Vorbit tare, bătut din palme , ciocănit |
| 150 - 200 | bătut din palme , ciocănit / doar in fata senzorului la o distanta de jumătate de metru |
| >200 | Zgomote înalte la 1-30 centimetri in fata senzorului |

300 nu poate fi atins

In urma testelor putem alege valori de la 50 – 100 daca dorim ca cutia sa fie mai sensibila si sa fie deschisa de la o distanta mai mare. Eu am ales intervalul 100-150 deoarece, in urma testelor, pare sa includă cel mai bine aria sonora a bătutului din palme. Următoarele valori au fost înregistrate in urma mai multor teste de bătut din palme la o distanta de 20-30 de cm in fata senzorului de sunet:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test 1 | Test 2 | Test 3 | Test 4 | Test 5 | Test 6 |
| 20:34:29.104 -> 134  20:34:29.104 -> 52  20:34:29.151 -> 103  20:34:29.151 -> 111 | 20:35:07.214 -> 15  20:35:07.214 -> 126  20:35:07.214 -> 149 | 20:35:53.102 -> 31  20:35:53.102 -> 118  20:35:53.102 -> 97  20:35:53.148 -> 11 | 20:37:32.509 -> 93  20:37:32.509 -> 68  20:37:32.555 -> 145 | 20:38:08.658 -> 50  20:38:08.658 -> 104 | 20:36:12.727 -> 228  20:36:12.727 -> 14  20:36:12.727 -> 27  20:36:12.727 -> 101  20:36:12.773 -> 102 |

5. Concluzie

In concluzie, proiectul a fost terminat, cutia reacționează bine la bătăi din palma si evita cu succes alte zgomote exterioare. Proiectul a fost mai greu de implementat din cauza numărului mare de piese mobile care au trebuit sa fie montat intr-un spațiu redus. Motor-ul servo metalic reușește cu ușurința sa ridice capacul de lemn al cutiei. Matricea de led-uri oferă culori variate si intense, cu o gama mare de opțiuni in utilizare.

Ca modificări ulterioare pot fi adăugate diferite moduri de rulare. Acestea se pot schimba cu ajutorul unei variabile care sa conțină numărul modului actual si care poate fi incrementata de pe butonul galben. In plus trebuie introdus in funcția start o structura de tip „if(mode == thisMode)” pentru fiecare mod implementat. Toate aceste lucruri au fost adăugate deja, pentru alte adaptări utilizatorul trebuie doar sa implementeze alte moduri de rulare. Fiecare mod poate sa conțină diferite melodii, sau pot sa afișeze mesaje noi pe matricea de leduri, pot fi implementate diferite jocuri de culori custom sau imagini de pixeli.