

Miniprojekt: Hissen

2017-03-09

Dylan Saleh

Johan Kämpe

Mjukvaruutvecklare inbyggda system

Programmering av inbyggda system

Syfte

Syftet med projektet är att skapa kod som simulerar en hiss, med dess nödvändiga funktioner.

Funktioner i koden ska byggas så att de är generella, utan att veta hur hisssystemets sensorer fungerar. Dessa funktioner ska sedan kunna modifieras efter sensorerna utförande.

Genomförande

Projektet började med att skissa och anteckna nödvändiga funktioner och sensorer som hissen behöver. En verklig hiss undersöktes också.

Projektets valdes att utvecklas med Arduinos utvecklingsmiljö, snarare än ren C-kod.

Texteditorn Atom användes för att skriva källkoden.

Git användes för versionshantering, Google Drive användes för molnlagring, rapportskrivning och backup av filer.

Fritzing användes för att skapa kopplingsschema.

Använd hårdvara

Utvecklingskortet Arduino UNO användes som plattform i hissprojektet. Till kortet anslöts sex tryckknappar och sex lysdioder (LED).

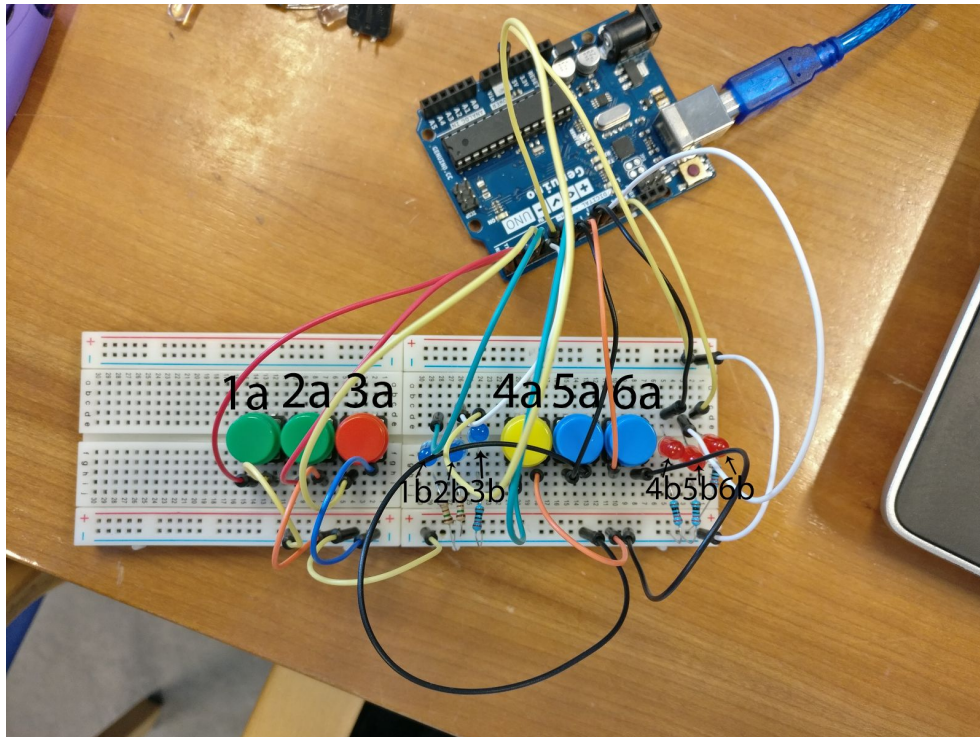
Knapparna simulerar de tryckknappar i ett hisssystem som kallar hissens korg till en våning, samt de knappar som finns inuti hisskorgen för att ta denna till en specifik våning i byggnaden.

Lysdioderna används för att indikera hisskorgens våningsposition, och för indikering till tryckknapparna som kallar på hisskorgen till respektive våning.

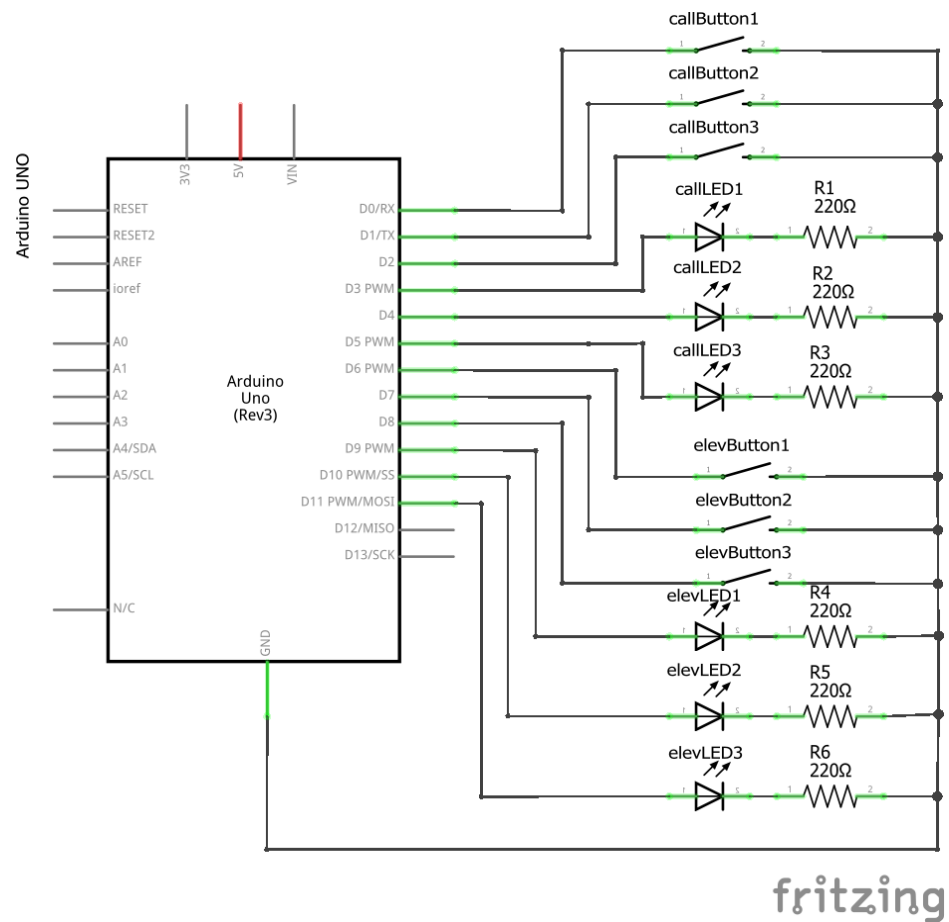
Till lysdioderna kopplas även resistorer, för att limitera strömstyrkan.

Övriga hissfunktioner, som nödstopp, dörröppning, dörrlås och nödsignal simuleras i programkoden med variabler eller utskrifter till Arduino IDEts *Serial Monitor*.

Fotografi på uppkoppling



Kopplingsschema



fritzing

Beskrivning av knappar och lysdioder i fotografi ovan.

Kod i fotografi	Namn	Förklaring
1a	callButton1	Tryckknapp för att kalla hisskorg till våning 1.
2a	callButton2	Tryckknapp för att kalla hisskorg till våning 2.
3a	callButton3	Tryckknapp för att kalla hisskorg till våning 3.
1b	callLED1	Lysdiod-Indikering för callButton1, att hisskorgen är på väg till våning 1.
2b	callLED2	Lysdiod-Indikering för callButton1, att hisskorgen är på väg till våning 2.
3b	callLED3	Lysdiod-Indikering för callButton1, att hisskorgen är på väg till våning 3.
4a	elevButton1	Tryckknappar inuti hisskorg, för att transportera hissen till våning 1
5a	elevButton2	Tryckknappar inuti hisskorg, för att transportera hissen till våning 2
6a	elevButton3	Tryckknappar inuti hisskorg, för att transportera hissen till våning 3
4b	elevLED1	LED-indikering för att indikera att hisskorgens är vid våning 1
5b	elevLED2	LED-indikering för att indikera att hisskorgens är vid våning 2
6b	elevLED3	LED-indikering för att indikera att hisskorgens är vid våning 3

Projektets programkod

Globala variabler

int elevatorCurrentFloor: Används som ett substitut till sensorer, anger hisskorgens nuvarande våningsposition.

int emergencyStopEngaged: Substitut för inslaget nödstopp, sätts till 1 eller 0.

int emergencySignalEngaged: Substitut för inslagen nödsignal, sätts till 1 eller 0.

int motorRunning: Simulera motorkörning för att flytta hisskorg.

int doorLocked[NUM_FLOORS]: Array som håller dörrlås-status, 1 för låst, 0 för olåst.

NUM_FLOORS beskrivs under *Macron* nedan.

int carQueue[NUM_FLOORS]: Kö för att flytta hisskorgen mellan våningar.

unsigned long motorTimer: Timer för att simulera motor-körningstid / hisskorgens förflyttning mellan våningar.

unsigned long doorTimer: Timer för hur länge hissdörrar ska vara öppna innan de stängs och låses.

Funktioner

Argumentet ***int floor*** är en våning i hissens byggnad.

void setup(): Standardfunktion i Arduino-miljön som körs en gång, innan programmets loop.

void loop(): Standardfunktion i Arduino-miljön. Loopas kontinuerligt.

int elevatorFloorPosition(): Ger i returvärde hisskorgens nuvarande våningsposition.

void moveElevatorToFloor(int floor): Transporterar hisskorgen till en våning.

int getButtons(void): Läser av knapparna och ger i returvärde korresponderande våning att flytta hisskorgen till, om en knapp har tryckts in.

int isEmergencyStopEngaged(void): Anger om nödstoppet är intryckt.

int isEmergencySignalEngaged(void): Anger om nödsignal är intryckt.

void sendEmergencySignal(void): Skickar nödsignal, simuleras i *Serial Monitor*.

void setIndicatorLED(int floor): Tänder korresponderande LED för hisskorgens våningsläge, släcker de andra.

void setCallButtonLEDOn(int floor): Tänder en lysdiod för korresponderande *callButton*.

void setCallButtonLEDOff(int floor): Släcker en lysdiod för korresponderande *callButton*.

void addToCarQueue(int floor): Läger en våning till hisskorgens kö.

void removeFromCarQueue(int floor): Tar bort en våning från hisskorgens kö.

void handleCarQueue(void): Behandlar hisskorgens kö, flyttar hisskorgen till nästa våning i kön.

int isDoorLocked(int floor): Anger om dörrlås på våningen är aktiverat eller inte.

int doorsLocked(void): Anger om alla dörrar är låsta eller inte.

void controlDoor(int floor, int openClose): Öppnar eller stänger dörren för en våning.

Macron

enum används för att sätta definitioner av Arduino UNO-pins

Namn	Pin/nummer
pinCallButton1	0
pinCallButton2	1
pinCallButton3	2
pinCallLED1	3
pinCallLED2	4
pinCallLED3	5
pinElevButton1	6
pinElevButton2	7
pinElevButton3	8
pinElevLED1	9
pinElevLED2	10
pinElevLED3	11

callButton: Tryckknappar för att kalla hisskorg till korresponderande våning.

callLED: LED-indikering för *callButtons*, att hisskorgen är på väg

elevButton: Tryckknappar inuti hisskorgen, för att transportera hissen till en annan våning

elevLED: LED-indikering för hisskorgens våningsposition.

0/1-Macron

Används för att lättare kunna läsa och skriva koden.

0	1
FALSE	TRUE
CLOSE	OPEN
NO	YES

Andra macron

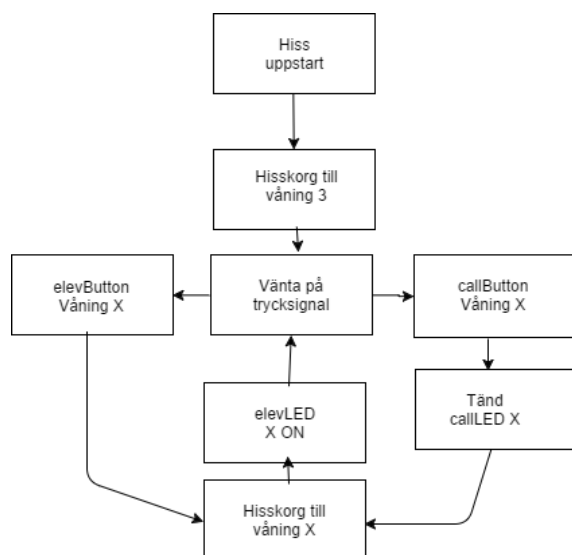
NUM_FLOORS: Anger antalet våningar för hisssystemet

CAR_MOVEMENT_TIME: Tid för förflyttning av hisskorg mellan våningar.

DOOR_OPEN_TIME: Tid för dörrstängning, efter att de öppnats.

EMERGENCY_SIGNAL_FREQUENCY: Utskriftsfrekvens av nödsignal till Serial Monitor.

ELEVATOR_START_FLOOR: Hisskorgens våning vid programstart.

Programkoden som flödesschema FIXA FIXA FIXA

Utskrifter till Serial Monitor

Hissens nödsignal och dörrfunktionalitet simulerades genom utskrift till Arduino IDEs Serial Monitor. För att få utskrift från nödsignalen sätts den globala variabeln *emergencySignalEngaged* till 1.

Nedan visas skärmdumpar från Serial Monitor.

Nödsignal	Dörrar (och kö-info)
<pre> ----- EMERGENCY SIGNAL: BEEP BEEP! Elevator car is at floor: 2 ----- EMERGENCY SIGNAL: BEEP BEEP! Elevator car is at floor: 2 ----- Adding floor to queue: 3 Going to floor: 3 ----- EMERGENCY SIGNAL: BEEP BEEP! Elevator car is at floor: 3 ----- EMERGENCY SIGNAL: BEEP BEEP! Elevator car is at floor: 3 ----- Adding floor to queue: 1 Adding floor to queue: 2 Going to floor: 1 ----- EMERGENCY SIGNAL: BEEP BEEP! Elevator car is at floor: 1 ----- Going to floor: 2 ----- EMERGENCY SIGNAL: BEEP BEEP! Elevator car is at floor: 2 ----- EMERGENCY SIGNAL: BEEP BEEP! Elevator car is at floor: 2 ----- </pre>	<pre> Opening elevator door at floor 2 Closing elevator door at floor 2 Adding floor to queue: 1 Going to floor: 1 Opening elevator door at floor 1 Adding floor to queue: 2 Closing elevator door at floor 1 Going to floor: 2 Opening elevator door at floor 2 Closing elevator door at floor 2 Adding floor to queue: 3 Going to floor: 3 Opening elevator door at floor 3 Adding floor to queue: 2 Adding floor to queue: 1 Adding floor to queue: 3 Closing elevator door at floor 3 Going to floor: 2 Opening elevator door at floor 2 Closing elevator door at floor 2 Going to floor: 1 Opening elevator door at floor 1 Closing elevator door at floor 1 Going to floor: 3 Opening elevator door at floor 3 Closing elevator door at floor 3 </pre>

Diskussion och slutsats

asas