



Projekts “Dari pats – automašīnas projekts” (4. Līmenis)

Skolnieku daba lapa

Komanda:

Mērķis: attālināti vadīt DIY automašīnu

Ko var izmantot, lai attālināti vadītu un kontrolētu DIY automašīnu? Padomājiet par to un meklējiet informāciju tiešsaistē.

Zemāk uzrakstiet savas atbildes.

Kā darbojas Bluetooth? Meklējiet informāciju tiešsaistē un uzrakstiet atbildes zemāk.

Pārskatiet vairākus scenārijus, kā var izveidot DIY automašīnu stūri attālinātai vadībai, un izvēlieties vienu, ko demonstrēt. Uzzīmējiet savu DIY automašīnu un uzskaitiet nepieciešamos izgatavošanas materiālus.

	Materiālu saraksts:
--	----------------------------

Laiks virknes slēguma veidošanai!

Šajā 4.līmenī DIY automašīna tiks attālināti vadīta ar mobilo tālruni, ar Android sistēmu, izmantojot Bluetooth tehnoloģiju. Pirmkārt, Bluetooth modulis HM-06 jāpievieno Arduino sensora vairogam saskaņā ar 1.

HM-06 modulis	Arduino Sensora plate
RXD	3
TXD	2
GND	G
VCC	V

1. tabula. HM-06 Bluetooth moduļa savienošana ar Arduino sensora plati (pin-out).

Laiks praktiski darboties!

Izveidosim ķēdi, izmantojot savu Arduino plati un atbilstošās elektriskās sastāvdaļas.

Datorā jāinstalē bibliotēka RemoteXY. Lai instalētu bibliotēku, izpildiet tālāk sniegtos norādījumus.

1. Lejupielādējiet RemoteXY bibliotēku no vietnes <https://remotexy.com/en/library/>
2. Nepieciešams izgūt no ZIP arhīva RemoteXY bibliotēku *Arduino* → *libraries* mapē.
3. Palaidiet Arduino IDE programmatūru.

RemoteXY bibliotēka ļauj mobilajā tālrunī izveidot lietotāja interfeisu. Jūs varat izveidot interfeisu, velkot un novietojot dažādu veidu pogas no sadaļas *Elements* uz virtuālo mobilo tālruni vietnē <https://remotexy.com/en/editor/>, kā parādīts 1. attēlā. Katrai pogai ir savs mainīgais, kas nosaukumu var mainīt sadaļā *Properties*. Varat arī mainīt pogas krāsu un pogas tekstu.



Laiks programmēšanai!

Pievienojiet savu Arduino USB.

Pabeidzot lietotāja interfeisa dizainu, noklikšķiniet uz pogas Iegūt avota kodu / Get source code. Pēc tam jūs redzēsiet lietotāja saskarnes avota kodu. Šis kods tiks integrēts jūsu kodā no 1. līmeņa. Ja rakstāt kodu, izmantojot mBlock, lūdzu, nokopējiet risinājumu no Arduino IDE no pievienotā dokumenta. Pirmkārt, jūs varat redzēt lietotāja saskarnes konfigurāciju starp RemoteXY un END RemoteXY / RemoteXY include library un END RemoteXY include bibliotēku un komentārus avota kodā. Kopējiet šo koda daļu savā kodā no 1. līmeņa sākumā (pēc bibliotēku iekļaušanas). SoftwareSerial bibliotēka ir iekļauta divas reizes, tāpēc noņemiet vienu no tām, piemēram:

```
#include <Arduino.h>
#include <Wire.h>
#include <SoftwareSerial.h> Remove the repeating library

////////////////////////////////////
// RemoteXY include library //
////////////////////////////////////

#define REMOTEXY_MODE_SOFTSERIAL
#include <SoftwareSerial.h>

#include <RemoteXY.h>

// RemoteXY connection settings
#define REMOTEXY_SERIAL_RX 2
#define REMOTEXY_SERIAL_TX 3
#define REMOTEXY_SERIAL_SPEED 9600

// RemoteXY configurate
#pragma pack(push, 1)
uint8_t RemoteXY_CONF[] =
{ 255,4,0,0,0,43,0,10,13,0,
  1,0,43,11,12,12,2,31,87,0,
  1,0,43,27,12,12,2,31,83,0,
  1,0,23,27,12,12,2,31,65,0,
  1,0,63,27,12,12,2,31,68,0 };

// this structure defines all the variables and events of your control interface
struct {

// input variables
uint8_t W; // =1 if button pressed, else =0
uint8_t S; // =1 if button pressed, else =0
uint8_t A; // =1 if button pressed, else =0
uint8_t D; // =1 if button pressed, else =0

// other variable
uint8_t connect_flag; // =1 if wire connected, else =0
```

```

} RemoteXY;
#pragma pack(pop)

////////////////////////////////////
//                               END RemoteXY include                               //
////////////////////////////////////

void Motors_Forward ()
{
    Left_Motor_Forward();
    Right_Motor_Forward();
}

void Motors_Backward ()
{
    Left_Motor_Backward();
    Right_Motor_Backward();
}

void Left_Motor_Forward ()
{
    analogWrite(5,200);
    digitalWrite(6,1);
    digitalWrite(7,0);
}

void Left_Motor_Backward ()
{
    analogWrite(5,200);
    digitalWrite(6,0);
    digitalWrite(7,1);
}

void Left_motor_OFF ()
{
    analogWrite(5,0);
    digitalWrite(6,0);
    digitalWrite(7,0);
}

void Right_Motor_Forward ()
{
    analogWrite(11,200);
    digitalWrite(12,1);
    digitalWrite(13,0);
}

void Right_Motor_Backward ()
{
    analogWrite(11,200);
    digitalWrite(12,0);
    digitalWrite(13,1);
}

```

```

void Right_Motor_OFF ()
{
  analogWrite(11,0);
  digitalWrite(12,0);
  digitalWrite(13,0);
}

void Motors_OFF ()
{
  Left_motor_OFF();
  Right_Motor_OFF();
}

void turn_left ()
{
  Right_Motor_Forward();
  Left_motor_OFF();
}

void turn_right ()
{
  Left_Motor_Forward();
  Right_Motor_OFF();
}

void _delay(float seconds)
{
  long endTime = millis() + seconds * 1000;
  while(millis() < endTime) _loop();
}

void setup()
{
  pinMode(5,OUTPUT);
  pinMode(6,OUTPUT);
  pinMode(7,OUTPUT);
  pinMode(11,OUTPUT);
  pinMode(12,OUTPUT);
  pinMode(13,OUTPUT);
}

void _loop()
{
}

void loop()
{
  _delay(1);
  Motors_Forward();
  _delay(1);
  Motors_Backward();
  _delay(1);
  Motors_OFF();
  _loop();
}

```

Pēc tam kopējiet pārējās komandas:

- RemoteXY Init () iestatīšanas / setup funkcijai. Šī funkcija inicializē lietotāja interfeisu.
- RemoteXY Handler () uz cilpas / loop funkciju, kas pārbauda pogu stāvokli, vai tās tika nospiestas.

Visām pogām ir savi mainīgie / variables (skatiet avota koda daļu zem ievades mainīgo komentāra), kas ir vienāds ar 1, nospiežot pogu. Šajā gadījumā „

mainīgos sauc par W, S, A un D. Tālāk norādītais avota kods pārbauda katra mainīgā vērtību. Nospiežot pogu, tiek palaista atbilstošā komanda:

```
...  
  
void setup()  
{  
  pinMode(5,OUTPUT);  
  pinMode(6,OUTPUT);  
  pinMode(7,OUTPUT);  
  pinMode(11,OUTPUT);  
  pinMode(12,OUTPUT);  
  pinMode(13,OUTPUT);  
  
  RemoteXY_Init();  
}  
  
void _loop()  
{  
}  
  
void loop()  
{  
  RemoteXY_Handler();  
  
  if(RemoteXY.W) Motors_Forward();  
  if(RemoteXY.S) Motors_Backward();  
  if(RemoteXY.A) turn_left();  
  if(RemoteXY.D) turn_right();  
  
  delay(1);  
  Motors_OFF();  
  
  loop();  
}
```

Noderīgas funkcijas:

- `#include "bibliotēkas nosaukums"` - šī rinda ļauj avota kodam pievienot bibliotēkas galveni,
- `analogWrite (tapa, darba cikls / pin, duty cycle)` - šī funkcija ļauj ģenerēt PWM (Pulse Width Modulation) uz izvēlēta kontakta numura un ar izvēlēto darba ciklu,
- `digitalWrite (kontakts, vērtība / pin, value)` - šī funkcija uz digitālā kontakta iestata izvēlēto vērtību (HIGH un LOW vai 0 un 1),
- `milis ()` - atgriež pagājušo laiku pēc koda palaišanas milisekundēs.

Laiks amatniecībai!

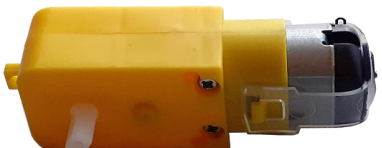
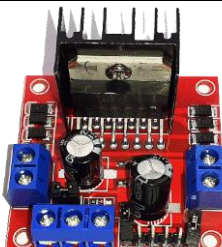

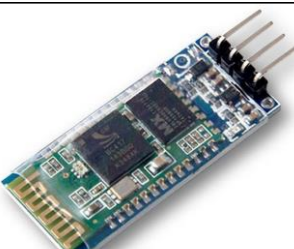
Sāciet strādāt pie DIY automašīnas dizaina, izmantojot pieejamos izgatavošanas materiālus.

Tagad jūs varat palaist savu kodu DIY automašīnā. Pēdējais solis ir lietotnes RemoteXY instalēšana savā tālrunī. Šo lietojumprogrammu varat atrast Google Play veikalā. Pēc tam savienojiet pārī savu mobilo tālruni ar Bluetooth moduli, kura nosaukums būs HC-06. Parole ir 1234.

Pēc tam savā mobilajā tālrunī atveriet lietotni RemoteXY. Noklikšķiniet uz pluszīmes labajā augšējā stūrī, izvēlieties Savienot ar Bluetooth ierīci / *Connect to Bluetooth device*, un izvēlieties ierīci HC-06 šādi. Pēc tam noklikšķiniet uz pievienotās ierīces un tagad varat attālināti vadīt savu DIY automašīnu. Izbaudi!

Elektriskie komponenti

Šajā tabulā ir uzskaitītas visi galvenie elektriskie komponenti, kas nepieciešami, lai īstenotu šo projektu.

	DC Motor
	L298n driver
	Arduino Sensor Shield
	Bluetooth module HC-06

ROBOSCIENTISTS PROJEKTS

Motivating secondary school students towards STEM careers through robotic artefact making

Robotikas artefaktu veidošana vidusskolēnu motivēšanai STEM karjeru izvēlei

Erasmus+ KA2 2018-1PL01-KA201-051129

Autori

Angelika Tefelska (WUT)

Informācija

Šis ziņojums ir sagatavots projekta ROBOSCIENTISTS ietvarā. Ja ir izmantoti citi publicēti un nepublicēti avoti, tie ir atzīti.

Autortiesības

© Copyright 2018 - 2021 the Roboscientists Consortium

All rights reserved.



Šis dokuments ir licencēts saskaņā ar Creative Commons Attribution- nekomerciāls-ShareAlike 4.0 starptautisko licenci.

Finansējums

Šis projekts ir finansēts ar Eiropas Komisijas atbalstu. Šis paziņojums atspoguļo tikai autora uzskatus, un Komisija nav atbildīga par jebkādu tajā ietvertās informācijas izmantošanu.