

Эксперимент № 8. Обмен данными через COM-порт Bluetooth HC-06

Вам понадобится

1. [Arduino IDE](#)
2. <http://ai2.appinventor.mit.edu>
3. [Delphi \(DSPack Library, CPortLibrary\(TComPort\)\)](#)

Список деталей для эксперимента

- Bluetooth модуль HC-06
- 1 светодиод
- 1 мотор или сервопривод
- 1 резистор номиналом 220 Ом
- 1 беспаячная макетная плата
- провода «папа-мама»
- Arduino UNO R3

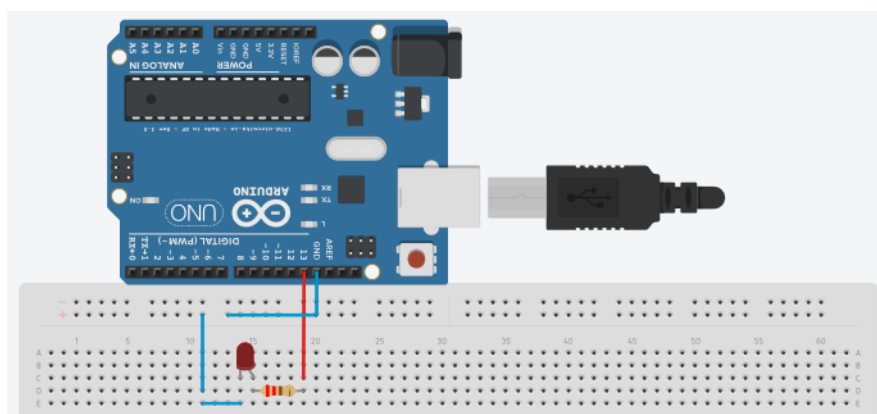
Повторение

- управление моторами. Arduino, L293D и фотоэлементы
- объектно-ориентированное программирование

Пошаговое руководство для проведения эксперимента

Иногда в проектах возникает необходимость в дистанционном управлении или передачи данных с ваших телефонных гаджетов. Один из самых популярных и распространенных методов обмена данными посредством Bluetooth. Рассмотрим простой пример как можно подключить Bluetooth модуль к Arduino и настроить дистанционное управление включением и выключением светодиода с телефона, или включением/выключением двигателя для управления игрушечным автомобилем.

Повторение. Разместите на макетной плате компоненты как показано на рисунке ниже. Для соединения деталей между собой воспользуйтесь нажатиями левой кнопки мыши.



Важно! Подключите к Arduino (рисунок которой показан выше) Bluetooth модуль HC-06 в соответствии с приведенной ниже схемой подключения. Подключать Bluetooth модуль к микроконтроллеру Arduino удобнее всего с помощью проводов.

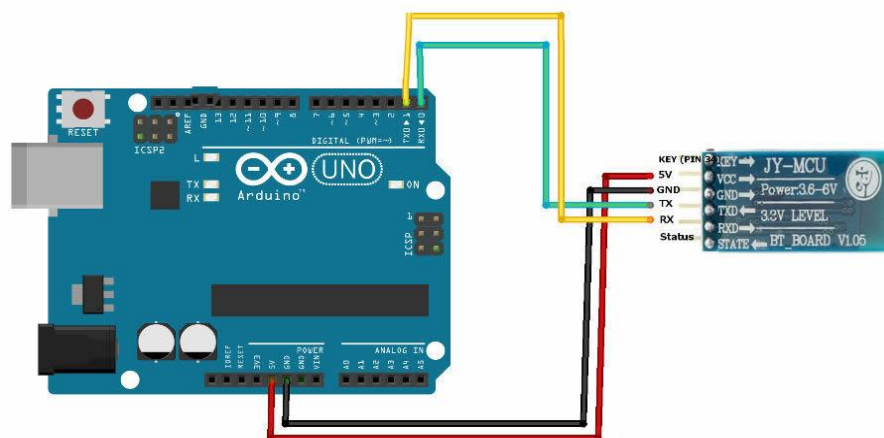
Arduino	Bluetooth
Pin 1 (TX)	RXD

Pin 0 (RX)	TXD
GND	GND
5V	VCC

Теперь необходимо записать пробный код программы:

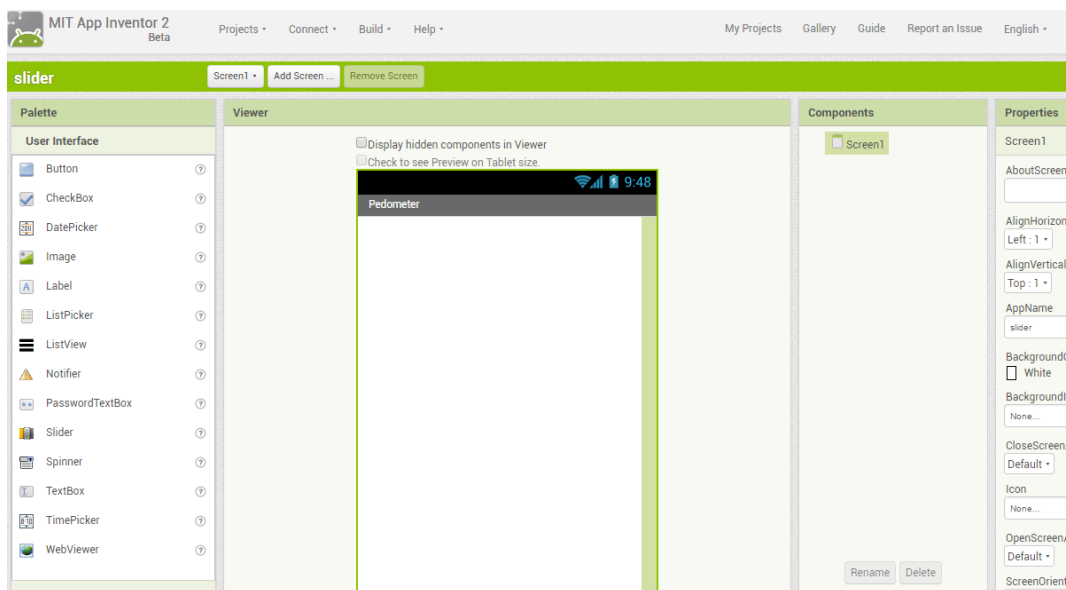
Важно. Во время загрузки скетча необходимо что бы Bluetooth модуль был отключен от микроконтроллера Arduino. В противном случае скетч не запишется, потому что связь с Bluetooth модулем происходит по одному и тому же порту RX и TX, что и USB.

```
int val;
int LED = 13;
void setup()
{
  Serial.begin(9600);
  pinMode(LED, OUTPUT);
  digitalWrite(LED, HIGH);
}
void loop()
{
  if (Serial.available())
  {
    val = Serial.read();
    // При символе "1" включаем светодиод
    if (val == '1')
    {
      digitalWrite(LED, HIGH);
    }
    // При символе "0" выключаем светодиод
    if (val == '0')
    {
      digitalWrite(LED, LOW);
    }
  }
}
```

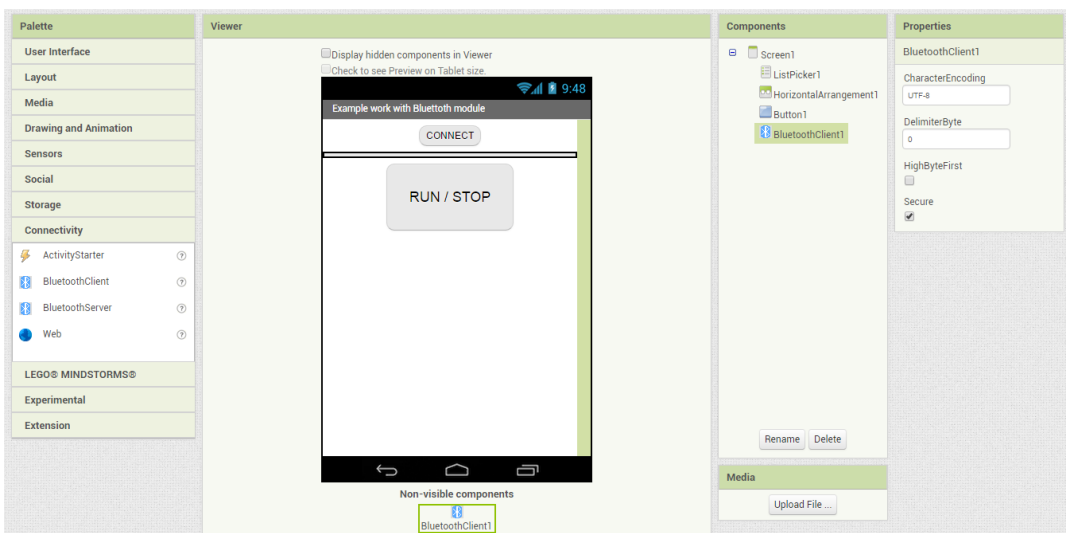


После того как скетч записан и Bluetooth модуль подключен к Arduino, можно перейти к следующему шагу. Откройте сайт <http://ai2.appinventor.mit.edu/>. Этот сайт является платформой для разработки мобильных приложений на платформе Android. Войдите на сайт или

зарегистрируйтесь на сайте как новый пользователь. После успешной авторизации на сайте ваш браузер перенаправит страницу в режим разработчика приложения



Разместите компоненты как показано на рисунке ниже



Для перехода в режим разработки кода программы перейдите к разделу Blocks
Используйте спрайты для разработки программы приведенной ниже.



```
when Button1 . Click
do
  call BluetoothClient1 . SendText
  text " 1 "
```

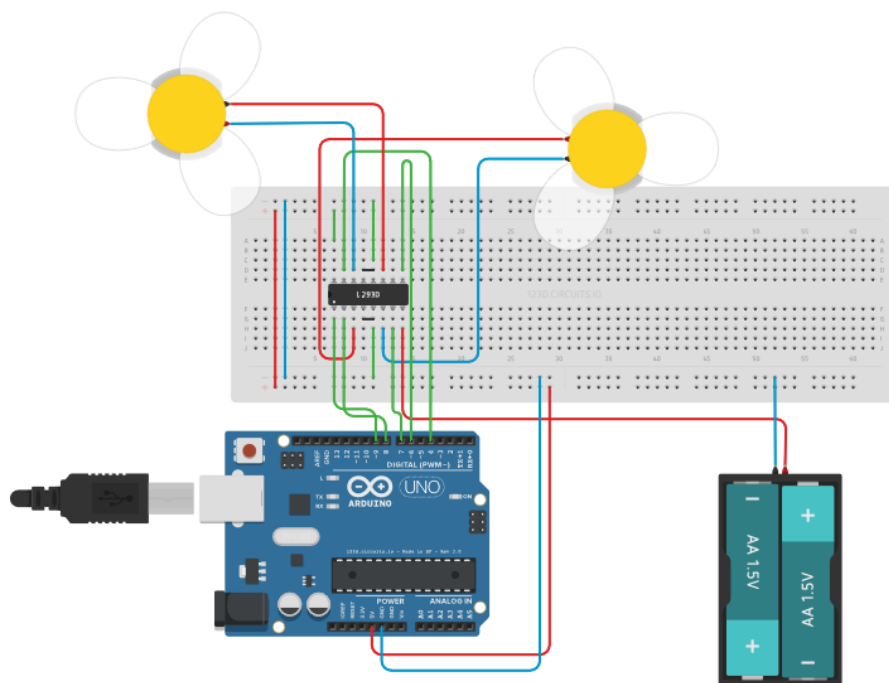
```
when ListPicker1 . BeforePicking
do
  set ListPicker1 . Elements to BluetoothClient1 . AddressesAndNames
```

```
when Button1 . LongClick
do
  call BluetoothClient1 . SendText
  text " 0 "
```

```
when ListPicker1 . AfterPicking
do
  if
    call BluetoothClient1 . Connect
    address ListPicker1 . Selection
  then
    set ListPicker1 . Text to ListPicker1 . Selection
  else
    set ListPicker1 . Text to " DISCONNECT "
```

Код и комментарии к голосовому управлению с Android, моторами подключенными к Arduino

Повторение эксперимента с L293D. Управление моторами можно осуществлять с отладочной платы путем подключения двигателей через микросхему L293D.



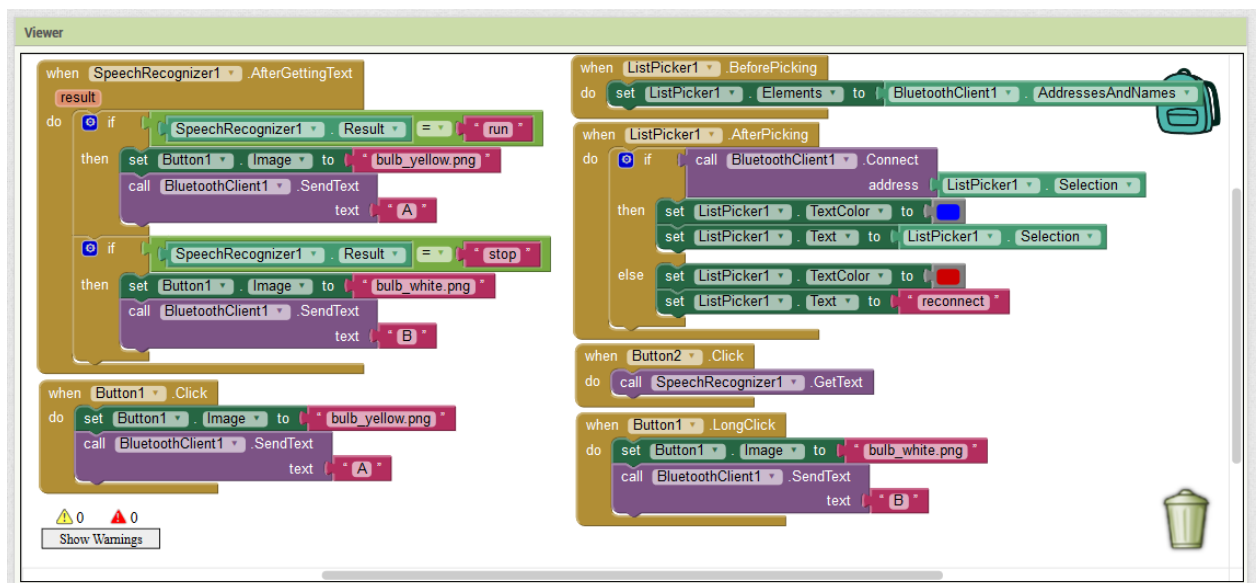
```
int IN1 = 8; //input1 подключен к выводу 8
int IN2 = 7;
int IN3 = 2;
int IN4 = 4;
int EN1 = 9;
int EN2 = 6;
int i;
void setup()
{
  pinMode (EN1, OUTPUT);
  pinMode (IN1, OUTPUT);
  pinMode (IN2, OUTPUT);
  pinMode (EN2, OUTPUT);
  pinMode (IN4, OUTPUT);
  pinMode (IN3, OUTPUT);
}
void loop()
{
  digitalWrite (IN2, HIGH);
  digitalWrite (IN1, LOW);
  digitalWrite (IN4, HIGH);
  digitalWrite (IN3, LOW);
  for (i = 50; i <= 180; ++i)
  {
    analogWrite(EN1, i);
    analogWrite(EN2, i);
```

```

    delay(30);
}
analogWrite (EN1, 0);
analogWrite (EN2, 0);
delay(500);
digitalWrite (IN1, HIGH);
digitalWrite (IN2, LOW);
digitalWrite (IN3, HIGH);
digitalWrite (IN4, LOW);
for (i = 50; i <= 180; ++i)
{
    analogWrite(EN1, i);
    analogWrite(EN2, i);
    delay(30);
}
analogWrite (EN1, 0);
analogWrite (EN2, 0);
delay(8000);
}

```

Ниже представлен фрагмент программы для голосового управления включением/выключением моторов с помощью приложения разработанного в среде AppInventor, а так же скетч для управления моторами с использованием модуля Bluetooth.



Задание. Подключите модуль Bluetooth к отладочной плате как показано на рисунке выше. Подключите моторы к отладочной плате, или используйте Motor Shield если он у вас имеется. Примените полученные выше знания изменив текст примера L293D для голосового управления моторами.

```

const int motor = 9; // порт Ардуино занятый первым мотором
const int motor1 = 10; // порт Ардуино занятый вторым мотором
char incom;
int n1;
void setup() {

Serial.begin(9600);
pinMode(motor, OUTPUT);

```

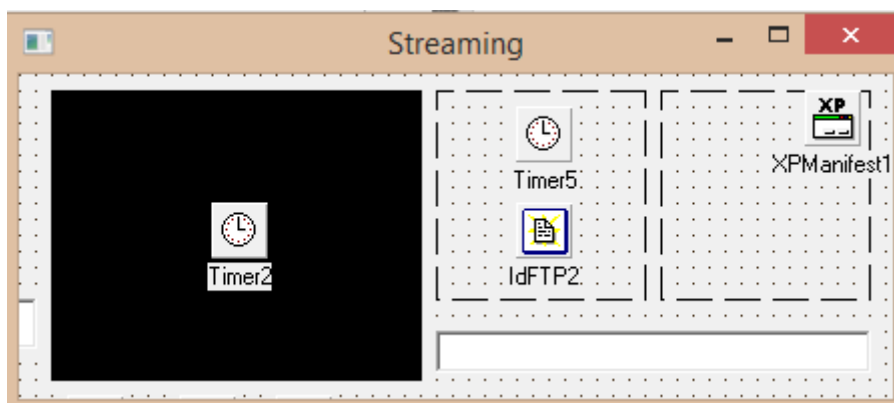
```

}
void loop()
{

if (Serial.available()>0) { // проверка входящих данных
  { n1 = Serial.read();
  if (n1 == 65) // 65 – символьное значение буквы А
  {
    Serial.println("65");
    for (int i=0; i<256; i++) // плавное увеличение скорости моторов
    { analogWrite(motor,i);
      analogWrite(motor1,i);
      delay(10); }
    } else {
  if (n1 == 66) 65 – символьное значение буквы В
  {
    Serial.println("66");
    for (int i=255; i>=0; i--) // плавное уменьшение скорости моторов
    { analogWrite(motor,i);
      analogWrite(motor1,i);
      delay(10); }
    }
  } } }
}

```

Кроме приложения Windows для удаленного управления компьютерами и электронными устройствами в локальной и глобальной сети реализована мобильная часть приложения для ОС Android предназначенная для просмотра снимка веб-камеры стороны сервера с использованием **WebView** и управления одним или несколькими устройств с использованием приема-передачи файлов по протоколу передачи файлов. Серверная часть приложения реализует элементы компьютерного зрения и осуществляет попиксельную проверку изображений с веб-камеры. Количество изменений в кадре изображения записывается в текстовый файл in.txt и загружается по протоколу FTP вместе с изображением с камеры. Приложение-клиент принимает информацию с FTP сервера о попиксельном изменении в кадре на сервере в текстовом формате для удаленного управления электрических и электронных устройств. Реализация управления устройствами осуществляется по беспроводной связи Bluetooth с использованием передачи информации через COM-порт на управляемое устройство.



```
uses
... IdBaseComponent, IdComponent, IdTCPConnection, IdTCPClient, IdFTP, CPort, DSUtil, DirectShow9,
jpeg, ToolWin, ComCtrls, ComObj, XPMAn;
```

```
public
{ Public declarations }
CamItem: TSysDevEnum;
end;
```

```
var
Form1: TForm1;
i:integer; f:textfile; s:string; j:byte;
```

```
implementation
```

```
function capCreateCaptureWindowA(lpszWindowName: PCHAR;
dwStyle: longint; x: integer; y: integer;
nWidth: integer; nHeight: integer; ParentWin: HWND;
nId: integer): HWND; stdcall external 'AVICAP32.DLL';
```

```
procedure TForm1.FormCreate(Sender: TObject);
```

```
var
```

```
i: integer; k:char; tsr : tsearchrec;
```

```
begin
```

```
CamItem:= TSysDevEnum.Create(CLSID_VideoInputDeviceCategory);
```

```
if CamItem.CountFilters > 0 then
```

```
begin
```

```
FilterGraph1.ClearGraph;
```

```
FilterGraph1.Active:=false;
```

```
//Задаем устройство с которого будем получать изображение
```

```
//О это индекс устройства общее количество можно узнать при помощи CamItem.CountFilters
```

```
Filter1.BaseFilter.Moniker:=CamItem.GetMoniker(0);
```

```
FilterGraph1.Active:=true;
```

```
//Откуда получать и куда показывать.
```

```
with FilterGraph1 as ICaptureGraphBuilder2 do
```

```
RenderStream(@PIN_CATEGORY_PREVIEW, nil, Filter1 as IBaseFilter, SampleGrabber1 as IBaseFilter,
```

```
VideoWindow1 as IbaseFilter);
```

```
//Выводим изображение с камеры.
```

```
FilterGraph1.Play;
```

```
end;
```

```
begin
```

```
if FindFirst('C:\www\cloud\' + '*.*',faAnyFile,tsr) = 0 then
```

```
repeat
```

```
Listbox1.Items.Add(tsr.name);
```

```
until FindNext(tsr) <> 0;
```

```
FindClose(tsr);
```

```
end;
```

```
end;
```

```
procedure TForm1.Timer2Timer(Sender: TObject);
```

```
var
```

```
//координата пикселя по горизонтали
```

```
i:integer;
```

```
//координата пикселя по вертикали
```

```

j:integer;
//переменные количества различий
k:integer; r1,g1,b1:Byte; r2,g2,b2:Byte; FirstColor,SecondColor:Integer;
Color:TColor; ChangeDetect:byte;
begin
//делаем 1-ый снимок
if Timer2.Tag=0 then
begin
SampleGrabber1.GetBitmap(Image1.Picture.Bitmap);
Timer2.Tag:=1;
exit;
end;
//через некоторое время - второй, с которым будем сверять
SampleGrabber1.GetBitmap(Image2.Picture.Bitmap);
Timer2.Tag:=0;
k:=0;
//начинаем попиксельное сравнение
for i := 1 to Image1.Picture.Bitmap.Height do
begin
for j := 1 to Image1.Picture.Bitmap.Width do
begin
ChangeDetect:=0;
//получаем цвет текущего пикселя первой картинки
FirstColor:=Image1.Picture.Bitmap.Canvas.Pixels[i,j];
//получаем составляющие RGB
r1:=GetRValue(FirstColor);
g1:=GetGValue(FirstColor);
b1:=GetBValue(FirstColor);
SecondColor:=Image2.Picture.Bitmap.Canvas.Pixels[i,j];
r2:=GetRValue(SecondColor);
g2:=GetGValue(SecondColor);
b2:=GetBValue(SecondColor);
//Начало проверки между 2-мя картинками
if Abs(r1-r2)>20 then inc(ChangeDetect);
if Abs(g1-g2)>20 then inc(ChangeDetect);
if Abs(b1-b2)>20 then inc(ChangeDetect);
//Если изменения существенные то увеличиваем счетчик
if ChangeDetect=3 then k:=k+1;
Application.ProcessMessages;
Edit2.Text:=inttostr(k);
end; end; end;

```

```

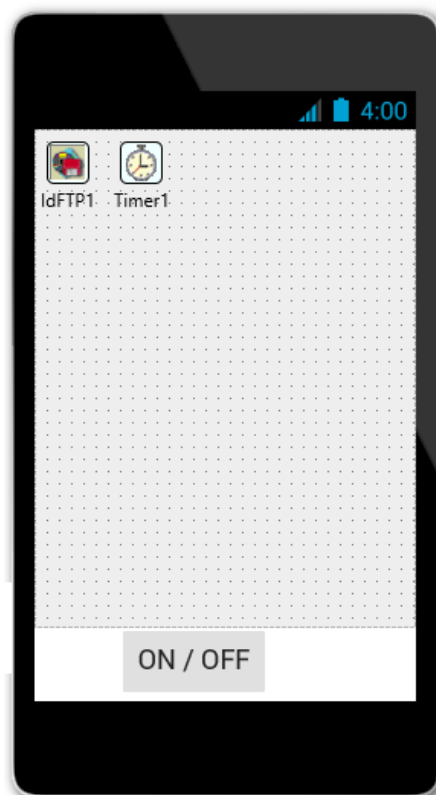
procedure TForm1.Timer5Timer(Sender: TObject);
var
jpg_img: TJPEGImage; f:textfile; s:string;
begin
assignfile(f,'in.txt');
rewrite(f);
writeln(f,Edit2.Text);
closefile(f);
jpg_img := TJPEGImage.Create;
jpg_img.Assign(Image2.Picture.Bitmap);
jpg_img.CompressionQuality := 20;
jpg_img.Compress;

```

```

jpg_img.SaveToFile('C:\www\2.jpg');
jpg_img.Free;
//ftp://oalex-sherzer:21Ozat2012@alex-sherzer.narod.ru
idFTP2.Host:='alex-sherzer.narod.ru'; //FTP-сервер
idFTP2.Port:=21; //FTP-server
idFTP2.Username:='oalex-sherzer';
idFTP2.Password:='21Ozat2012';
idFTP2.Connect;
idFTP2.put('C:\www\2.jpg','2.jpg',false);
idFTP2.put('C:\www\Detects\in.txt','in.txt',false);
idFTP2.Disconnect;
end;
end.

```



```

{$R *.fmx}
{$R *.NmXhdpiPh.fmx ANDROID}
{$R *.SmXhdpiPh.fmx ANDROID}

procedure TForm2.Button1Click(Sender: TObject);
var
f:textfile; s:string;
begin
//ftp://oalex-sherzer:21Ozat2012@alex-sherzer.narod.ru
assignfile(f,'/storage/emulated/0/bd/txt.txt');
rewrite(f);

```

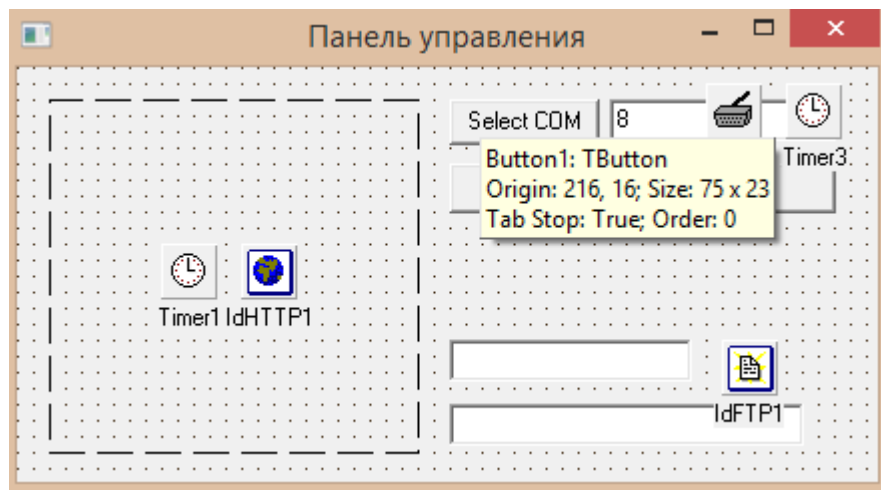
```

writeln(f,form2.Edit1.Text);
closefile(f);
idFTP1.Host:='alex-sherzer.narod.ru';
idFTP1.Port:=21; //FTP-server
idFTP1.Username:='oalex-sherzer';
idFTP1.Password:='21Ozat2012';
idFTP1.Connect;
idFTP1.Put('/storage/emulated/0/bd/txt.txt','txt.txt',false);
idFTP1.Disconnect;
end;

procedure TForm2.FormCreate(Sender: TObject);
begin
form2.WebBrowser1.Navigate('http://alex-sherzer.narod.ru/2.html');
end;

procedure TForm2.Timer1Timer(Sender: TObject);
begin
form2.WebBrowser1.Reload;
end;

```



uses
...jpeg, IdBaseComponent, IdComponent, IdTCPConnection, IdTCPClient, IdHTTP, StdCtrls, XPMAN, CPort, DSPack, IdFTP;

```

procedure TForm1.Timer1Timer(Sender: TObject);
var
strStream: String; memStream: TMemoryStream;
jpegimg: TJPEGImage; f:textfile; s:string;
begin
//ftp://oalex-sherzer:21Ozat2012@alex-sherzer.narod.ru
idFTP1.Host:='alex-sherzer.narod.ru'; //FTP-?????
idFTP1.Port:=21; //FTP-server
idFTP1.Username:='oalex-sherzer';
idFTP1.Password:='21Ozat2012';
idFTP1.Connect;
idFTP1.Get('in.txt',extractfilepath(application.exename)+'in.txt',true);
idFTP1.Disconnect;

```

```

assignfile(f,'in.txt');
reset(f);
readln(f,s);
Edit3.Text:=s;
closefile(f);
if (strtoint(s)>30) then edit2.text:='A' else edit2.text:='B';
//*****//
try
  strStream := idhttp1.Get ('http://alex-sherzer.narod.ru/2.jpg');
except
  Exit;
end;
memStream := TMemoryStream.Create;
jpegimg := TJPEGImage.Create;
try
  memStream.Write(strStream[1], Length(strStream));
  memStream.Position := 0;
  jpegimg.LoadFromStream(memStream);
  image1.Picture.Assign(jpegimg);
finally
  memStream.Free;
  jpegimg.Free;
end;
end;

procedure TForm1.Button1Click(Sender: TObject);
begin
form1.ComPort1.Port:='COM'+form1.Edit1.Text;
Timer3.Enabled:=true;
end;

procedure TForm1.Button2Click(Sender: TObject);
begin
form1.ComPort1.ShowSetupDialog;
end;

procedure TForm1.Timer3Timer(Sender: TObject);
var
d:char;
c:integer;
begin
ComPort1.Open;
ComPort1.Connected:=True;
c:=Ord(Edit1.Text[1]);
ComPort1.Write(c,1);
end;

procedure TForm1.ComPort1RxChar(Sender: TObject; Count: Integer);
var
Str: String;
serin: String;
begin
ComPort1.ReadStr(Str, Count);
Edit2.Text:=Str;

```

end;
end.

Примечание. В папке с примерами вы найдете папку с названием **detect**. Изучите внимательно содержимое папки. В данном каталоге вы найдете необходимые библиотеки для работы приложения, а так же исходный код приложения Windows для управления двигателями за счет считывания изменений в кадре. Для работы с файлами типа *.dpr потребуется установка языка программирования Delphi (рекомендуется установка Delphi 7). Исходный код может быть изменен вами или дополнен на ваше усмотрение или потребности для реализации иного проекта.