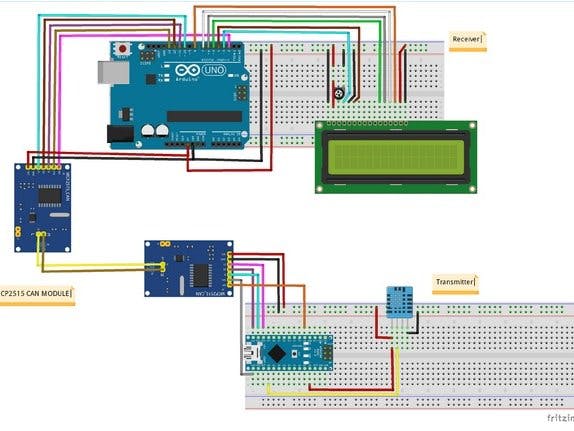
https://create.arduino.cc/projecthub/maurizfa-13216008-arthur-jogy-13216037-agha-maretha-13216095/can-bus-using-arduino-9ce7ba



**CAN Bus Using Arduino** [**© GPL3+**](http://opensource.org/licenses/GPL-3.0)

We implement CAN Bus communication with MCP2515 module to communicate between two Arduino for sending temperature data from DHT

* [communication](https://create.arduino.cc/projecthub/projects/tags/communication)
* [serial communication](https://create.arduino.cc/projecthub/projects/tags/serial+communication)
* 61,918 views
* 0 comments
* 15 respects

**Components and supplies**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| [Ph a000066 iso (1) ztbmubhmho](https://create.arduino.cc/projecthub/products/buy/41?s=BAhJIhMzMzc0NjUsUHJvamVjdAY6BkVG%0A) | |  | | --- | | [Arduino UNO](https://create.arduino.cc/projecthub/products/buy/41?s=BAhJIhMzMzc0NjUsUHJvamVjdAY6BkVG%0A) | |  | | [×](https://create.arduino.cc/projecthub/products/buy/41?s=BAhJIhMzMzc0NjUsUHJvamVjdAY6BkVG%0A) | [1](https://create.arduino.cc/projecthub/products/buy/41?s=BAhJIhMzMzc0NjUsUHJvamVjdAY6BkVG%0A) |  |
| [Ard nano](https://create.arduino.cc/projecthub/products/buy/19172?s=BAhJIhMzMzc0NjUsUHJvamVjdAY6BkVG%0A) | |  | | --- | | [Arduino Nano R3](https://create.arduino.cc/projecthub/products/buy/19172?s=BAhJIhMzMzc0NjUsUHJvamVjdAY6BkVG%0A) | |  | | [×](https://create.arduino.cc/projecthub/products/buy/19172?s=BAhJIhMzMzc0NjUsUHJvamVjdAY6BkVG%0A) | [1](https://create.arduino.cc/projecthub/products/buy/19172?s=BAhJIhMzMzc0NjUsUHJvamVjdAY6BkVG%0A) |  |
|  | |  | | --- | | [Microchip Technology MCP2515](https://create.arduino.cc/projecthub/products/buy/73314?s=BAhJIhMzMzc0NjUsUHJvamVjdAY6BkVG%0A) | |  | | [×](https://create.arduino.cc/projecthub/products/buy/73314?s=BAhJIhMzMzc0NjUsUHJvamVjdAY6BkVG%0A) | [2](https://create.arduino.cc/projecthub/products/buy/73314?s=BAhJIhMzMzc0NjUsUHJvamVjdAY6BkVG%0A) |  |
| [20120702130508859](https://create.arduino.cc/projecthub/products/buy/42753?s=BAhJIhMzMzc0NjUsUHJvamVjdAY6BkVG%0A) | |  | | --- | | [DHT11 Temperature & Humidity Sensor (4 pins)](https://create.arduino.cc/projecthub/products/buy/42753?s=BAhJIhMzMzc0NjUsUHJvamVjdAY6BkVG%0A) | |  | | [×](https://create.arduino.cc/projecthub/products/buy/42753?s=BAhJIhMzMzc0NjUsUHJvamVjdAY6BkVG%0A) | [1](https://create.arduino.cc/projecthub/products/buy/42753?s=BAhJIhMzMzc0NjUsUHJvamVjdAY6BkVG%0A) |  |
| [12002 04](https://create.arduino.cc/projecthub/products/buy/42749?s=BAhJIhMzMzc0NjUsUHJvamVjdAY6BkVG%0A) | |  | | --- | | [Breadboard (generic)](https://create.arduino.cc/projecthub/products/buy/42749?s=BAhJIhMzMzc0NjUsUHJvamVjdAY6BkVG%0A) | |  | | [×](https://create.arduino.cc/projecthub/products/buy/42749?s=BAhJIhMzMzc0NjUsUHJvamVjdAY6BkVG%0A) | [1](https://create.arduino.cc/projecthub/products/buy/42749?s=BAhJIhMzMzc0NjUsUHJvamVjdAY6BkVG%0A) |  |
| [11026 02](https://create.arduino.cc/projecthub/products/buy/42742?s=BAhJIhMzMzc0NjUsUHJvamVjdAY6BkVG%0A) | |  | | --- | | [Jumper wires (generic)](https://create.arduino.cc/projecthub/products/buy/42742?s=BAhJIhMzMzc0NjUsUHJvamVjdAY6BkVG%0A) | |  | | [×](https://create.arduino.cc/projecthub/products/buy/42742?s=BAhJIhMzMzc0NjUsUHJvamVjdAY6BkVG%0A) | [20](https://create.arduino.cc/projecthub/products/buy/42742?s=BAhJIhMzMzc0NjUsUHJvamVjdAY6BkVG%0A) |  |

**Apps and online services**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| [Ide web](https://www.arduino.cc/en/main/software) | |  | | --- | | [Arduino IDE](https://www.arduino.cc/en/main/software) | |  | |  |

**About this project**

**CAN Bus**

CAN Bus, stand for Controller Area Network, is one type of serial communication that usually used in industrial and automotive environments. The real example of CAN Bus application can be found in speed car data sensor that can be transferred to the rpm indicator.

CAN Bus is a message based protocol that can be used for multiple device communication. The figure below represents that when several CAN devices are connected together like a network, each device can communicate with other devices in the node. In general, CAN communication range is in range 50Kbps to 1Mbps, with the distance range is 40 meters (at 1 Mbps) to 1000 meters (at 50 kbps).

In this communication, data is transferred in a certain message format. Each message consists of many segments, but there are two main segments: Identifier and Data. Identifier or CAN ID, or known as Parameter Group Number (PGN) is used to identify the CAN device in the CAN network, usually in 11 (Standard CAN) or 29 bit (Extended CAN) length, based on the CAN protocol. While Data is the message content to be sent, usually in 0 to 8 bytes length.

CAN Protocol consists of two wires : CAN\_H and CAN\_L to send and receive the message. This two wires act as a differential line where CAN signal is represented with the potential difference between them. If the difference is positive and larger than a certain minimum voltage, then the signal is 1, and if the difference between them is negative, it will be 0.

Usually, in CAN communication, twisted pair cable is used. And at the ends of the CAN networks, a single 120-ohm resistor is used. It is because the line should be balanced and tied to the same potential.

In this project, we are going to implement CAN Bus communication with MCP2515 module to communicate between two Arduino for sending temperature data from DHT11 sensor. This module uses 5V as operating voltage and has pinout configuration as shown in the following table.

The result is as shown below:

Reference:<https://circuitdigest.com/microcontroller-projects/arduino-can-tutorial-interfacing-mcp2515-can-bus-module-with-arduino>

<https://www.allaboutcircuits.com/technical-articles/introduction-to-can-controller-area-network/>

**Code**

* [Transmitter](javascript:void(0))
* [Receiver](javascript:void(0))

**TransmitterC/C++**

#include <SPI.h> //Library for using SPI Communication

#include <mcp2515.h> //Library for using CAN Communication

#include <DHT.h> //Library for using DHT sensor

#define DHTPIN A0

#define DHTTYPE DHT11

struct can\_frame canMsg;

MCP2515 mcp2515(10);

DHT dht(DHTPIN, DHTTYPE); //initialize object dht for class DHT with DHT pin with STM32 and DHT type as DHT11

void setup(){

while (!Serial);

Serial.begin(9600);

SPI.begin(); //Begins SPI communication

dht.begin(); //Begins to read temperature & humidity sensor value

mcp2515.reset();

mcp2515.setBitrate(CAN\_500KBPS, MCP\_8MHZ); //Sets CAN at speed 500KBPS and Clock 8MHz

mcp2515.setNormalMode();

}

void loop(){

int h = dht.readHumidity(); //Gets Humidity value

int t = dht.readTemperature(); //Gets Temperature value

canMsg.can\_id = 0x036; //CAN id as 0x036

canMsg.can\_dlc = 8; //CAN data length as 8

canMsg.data[0] = h; //Update humidity value in [0]

canMsg.data[1] = t; //Update temperature value in [1]

canMsg.data[2] = 0x00; //Rest all with 0

canMsg.data[3] = 0x00;

canMsg.data[4] = 0x00;

canMsg.data[5] = 0x00;

canMsg.data[6] = 0x00;

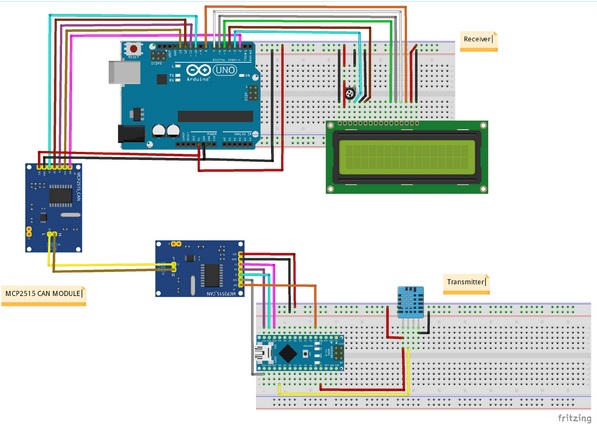
canMsg.data[7] = 0x00;

mcp2515.sendMessage(&canMsg); //Sends the CAN message

delay(1000);

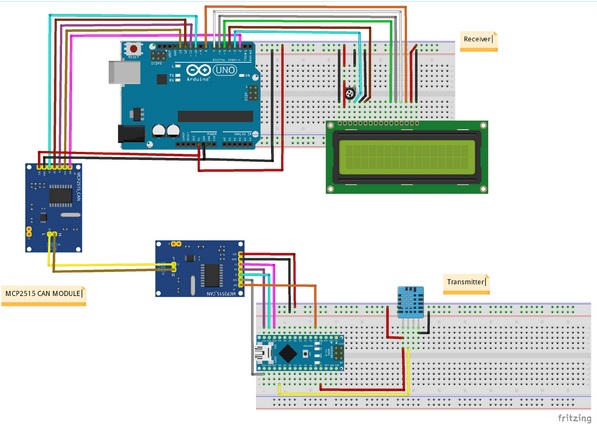
}

**Schematics**

****

**CAM Bus Circuit**

For Transmitter (Arduino 1)  
MCP2515 Module / DHT Sensor -> Arduino Nano  
MPC2515 – VCC -> +5V  
MPC2515 – GND -> GND  
MPC2515 – CS -> D10 (SPI\_SS)  
MPC2515 – SO -> D12 (SPI\_MISO)  
MPC2515 - S I -> D11 (SPI\_MOSI)  
MPC2515­ – SCK -> D13 (SPI\_SCK)  
MPC2515 – INT -> D2  
DHT11 – VCC -> +5V  
DHT11 – GND -> GND  
DHT11­ – OUT -> A0  
  
For Receiver (Arduino 2)  
MCP2515 Module -> Arduino Uno  
VCC -> +5V  
GND -> GND  
CS -> 10 (SPI\_SS)  
SO -> 12 (SPI\_MISO)  
SI -> 11 (SPI\_MOSI)  
SCK -> 13 (SPI\_SCK)  
INT -> 2  
  
Between two MCP2515  
MCP2515 on Arduino Nano (Arduino 1) -> MCP2515 on Arduino Nano (Arduino 2)  
H -> H  
L -> L



**Comments**

Please [log in](https://create.arduino.cc/projecthub/users/sign_in?id=337465&m=project&reason=comment&redirect_to=%2Fprojecthub%2Fmaurizfa-13216008-arthur-jogy-13216037-agha-maretha-13216095%2Fcan-bus-using-arduino-9ce7ba%23comments) or [sign up](https://create.arduino.cc/projecthub/users/sign_up?id=337465&m=project&reason=comment&redirect_to=%2Fprojecthub%2Fmaurizfa-13216008-arthur-jogy-13216037-agha-maretha-13216095%2Fcan-bus-using-arduino-9ce7ba%23comments&source=popup) to comment.

Be the first to comment!

Copy the code,

#include <SPI.h> //Library for using SPI Communication

#include <mcp2515.h> //Library for using CAN Communication

#include <DHT.h> //Library for using DHT sensor

#define DHTPIN A0

#define DHTTYPE DHT11

struct can\_frame canMsg;

MCP2515 mcp2515(10);

DHT dht(DHTPIN, DHTTYPE); //initialize object dht for class DHT with DHT pin with STM32 and DHT type as DHT11

void setup(){

while (!Serial);

Serial.begin(9600);

SPI.begin(); //Begins SPI communication

dht.begin(); //Begins to read temperature & humidity sensor value

mcp2515.reset();

mcp2515.setBitrate(CAN\_500KBPS, MCP\_8MHZ); //Sets CAN at speed 500KBPS and Clock 8MHz

mcp2515.setNormalMode();

}

void loop(){

int h = dht.readHumidity(); //Gets Humidity value

int t = dht.readTemperature(); //Gets Temperature value

canMsg.can\_id = 0x036; //CAN id as 0x036

canMsg.can\_dlc = 8; //CAN data length as 8

canMsg.data[0] = h; //Update humidity value in [0]

canMsg.data[1] = t; //Update temperature value in [1]

canMsg.data[2] = 0x00; //Rest all with 0

canMsg.data[3] = 0x00;

canMsg.data[4] = 0x00;

canMsg.data[5] = 0x00;

canMsg.data[6] = 0x00;

canMsg.data[7] = 0x00;

mcp2515.sendMessage(&canMsg); //Sends the CAN message

delay(1000);

}

## Receiver code

## [**Arduino\_MCP\_2515\_CAN\_Tutorial\_Recv.ino**](https://gist.github.com/elktros/87625c133d8fecbafc02aceca20e51a4#file-arduino_mcp_2515_can_tutorial_recv-ino)

|  |
| --- |
| #include <SPI.h> |
|  | #include "mcp\_can.h" |
|  |  |
|  | const int spiCSPin = 10; |
|  | const int ledPin = 2; |
|  | boolean ledON = 1; |
|  |  |
|  | MCP\_CAN CAN(spiCSPin); |
|  |  |
|  | void setup() |
|  | { |
|  | Serial.begin(115200); |
|  | pinMode(ledPin,OUTPUT); |
|  |  |
|  | while (CAN\_OK != CAN.begin(CAN\_500KBPS)) |
|  | { |
|  | Serial.println("CAN BUS Init Failed"); |
|  | delay(100); |
|  | } |
|  | Serial.println("CAN BUS Init OK!"); |
|  | } |
|  |  |
|  |  |
|  | void loop() |
|  | { |
|  | unsigned char len = 0; |
|  | unsigned char buf[8]; |
|  |  |
|  | if(CAN\_MSGAVAIL == CAN.checkReceive()) |
|  | { |
|  | CAN.readMsgBuf(&len, buf); |
|  |  |
|  | unsigned long canId = CAN.getCanId(); |
|  |  |
|  | Serial.println("-----------------------------"); |
|  | Serial.print("Data from ID: 0x"); |
|  | Serial.println(canId, HEX); |
|  |  |
|  | for(int i = 0; i<len; i++) |
|  | { |
|  | Serial.print(buf[i]); |
|  | Serial.print("\t"); |
|  | if(ledON && i==0) |
|  | { |
|  |  |
|  | digitalWrite(ledPin, buf[i]); |
|  | ledON = 0; |
|  | delay(500); |
|  | } |
|  | else if((!(ledON)) && i==4) |
|  | { |
|  |  |
|  | digitalWrite(ledPin, buf[i]); |
|  | ledON = 1; |
|  | } |
|  | } |
|  | Serial.println(); |
|  | } |
|  | } |

## Transmitter code

Transmitter code for Interfacing MCP2515 CAN Module with Arduino.

[Raw](https://gist.github.com/elktros/d7a64f7becc6b067e2d0dcb145392acf/raw/35a4e3e19f83a49c5b4d4766c1180ff01f6eb3ba/Arduino_MCP_2515_CAN_Tutorial_Tran.ino)

[**Arduino\_MCP\_2515\_CAN\_Tutorial\_Tran.ino**](https://gist.github.com/elktros/d7a64f7becc6b067e2d0dcb145392acf#file-arduino_mcp_2515_can_tutorial_tran-ino)

|  |  |
| --- | --- |
|  | #include <SPI.h> |
|  | #include <mcp\_can.h> |
|  |  |
|  | const int spiCSPin = 10; |
|  | int ledHIGH = 1; |
|  | int ledLOW = 0; |
|  |  |
|  | MCP\_CAN CAN(spiCSPin); |
|  |  |
|  | void setup() |
|  | { |
|  | Serial.begin(115200); |
|  |  |
|  | while (CAN\_OK != CAN.begin(CAN\_500KBPS)) |
|  | { |
|  | Serial.println("CAN BUS init Failed"); |
|  | delay(100); |
|  | } |
|  | Serial.println("CAN BUS Shield Init OK!"); |
|  | } |
|  |  |
|  | unsigned char stmp[8] = {ledHIGH, 1, 2, 3, ledLOW, 5, 6, 7}; |
|  |  |
|  | void loop() |
|  | { |
|  | Serial.println("In loop"); |
|  | CAN.sendMsgBuf(0x43, 0, 8, stmp); |
|  | delay(1000); |
|  | } |

[](https://gist.github.com/udayasreeM)

### [udayasreeM](https://gist.github.com/udayasreeM)commented [on Jun 2, 2020](https://gist.github.com/elktros/d7a64f7becc6b067e2d0dcb145392acf?permalink_comment_id=3326605#gistcomment-3326605)

I am doing same but i got an error (exit status 1  
mcp\_can.h: No such file or directory) like this. I have downloaded the libraries of MCP\_CAN 2515 related but still same error is coming. I am using arduino 1.8.1 version and the board is arduino mega 2560. please look at above code and where i am doing mistake please give me replay .

Top of Form

Bottom of Form

[](https://gist.github.com/aadithya2410)

### [aadithya2410](https://gist.github.com/aadithya2410)commented [on Mar 11, 2021](https://gist.github.com/elktros/d7a64f7becc6b067e2d0dcb145392acf?permalink_comment_id=3662180#gistcomment-3662180)

I am getting the same error as well

Top of Form

Bottom of Form

[Sign up for free](https://gist.github.com/join?source=comment-gist) **to join this conversation on GitHub**. Already have an account? [Sign in to comment](https://gist.github.com/login?return_to=https%3A%2F%2Fgist.github.com%2Felktros%2Fd7a64f7becc6b067e2d0dcb145392acf)

## Seeed github studio

## https://github.com/Seeed-Studio/Seeed\_Arduino\_CAN

## [GitHub - Seeed-Studio/Seeed\_Arduino\_CAN: Seeed Arduino CAN-BUS library - MCP2518FD&MCP2515&MCP2551](https://github.com/Seeed-Studio/Seeed_Arduino_CAN)