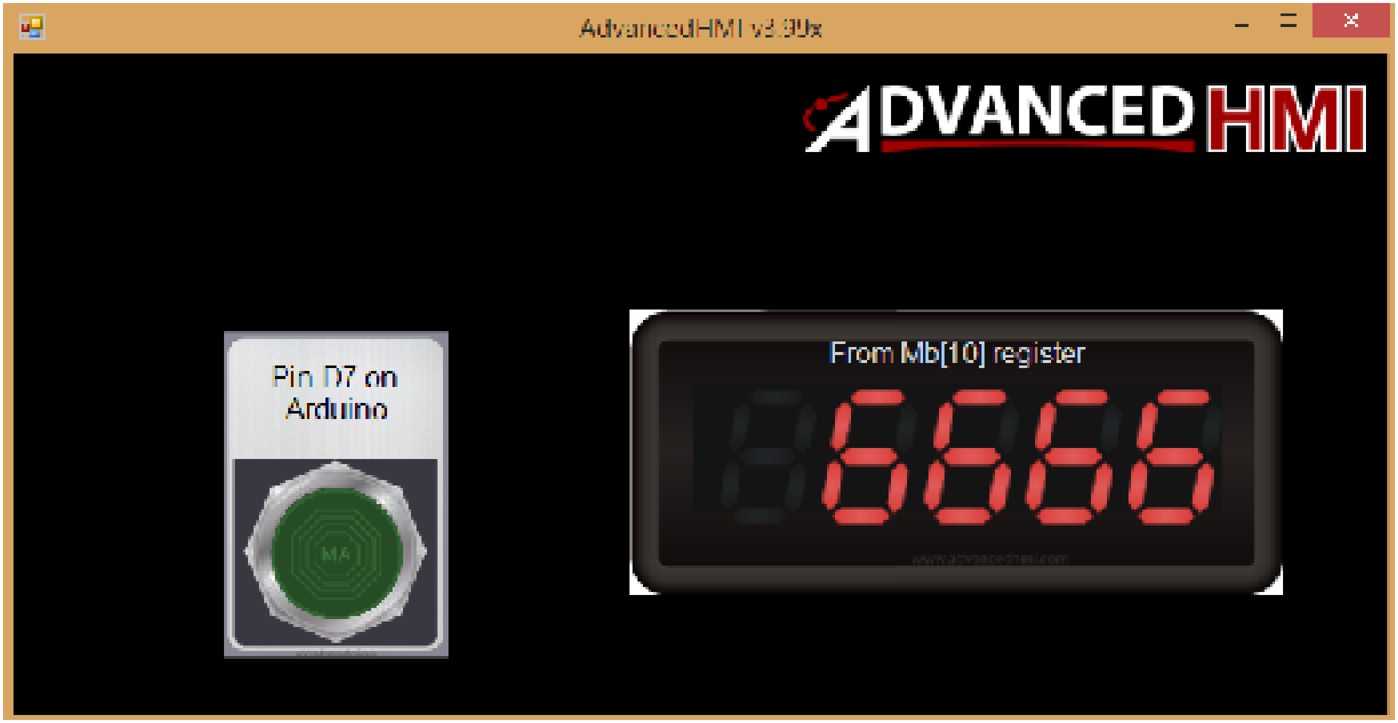
ERTHERNET SHIELDS V2

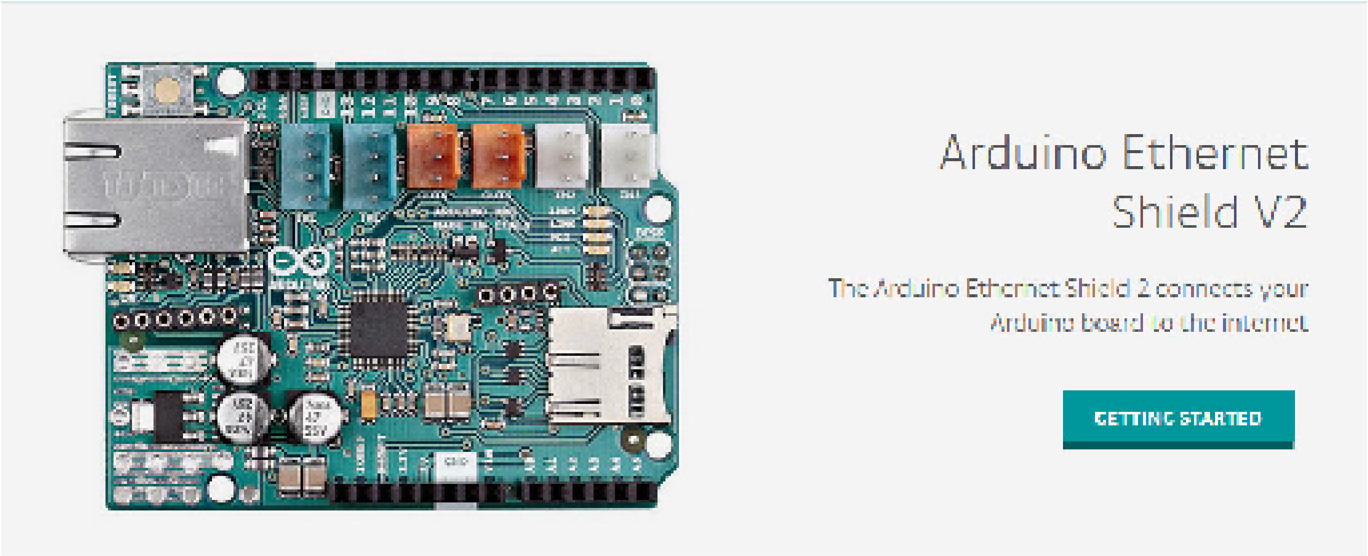
0-Settings of AdvancedHMI:

I used AdvancedHMI to check the Modbus TCP communication. The pushbutton has the address 40010 (Mb[9] register on Arduino board) and light on pin D7 on Arduino board. The digitalPanelmeter has the address 40011 (Mb[10] register on Arduino board). The Ethernet shield has the IP Address 192.168.1.144, MSK /24, GTW 192.168.1.1.



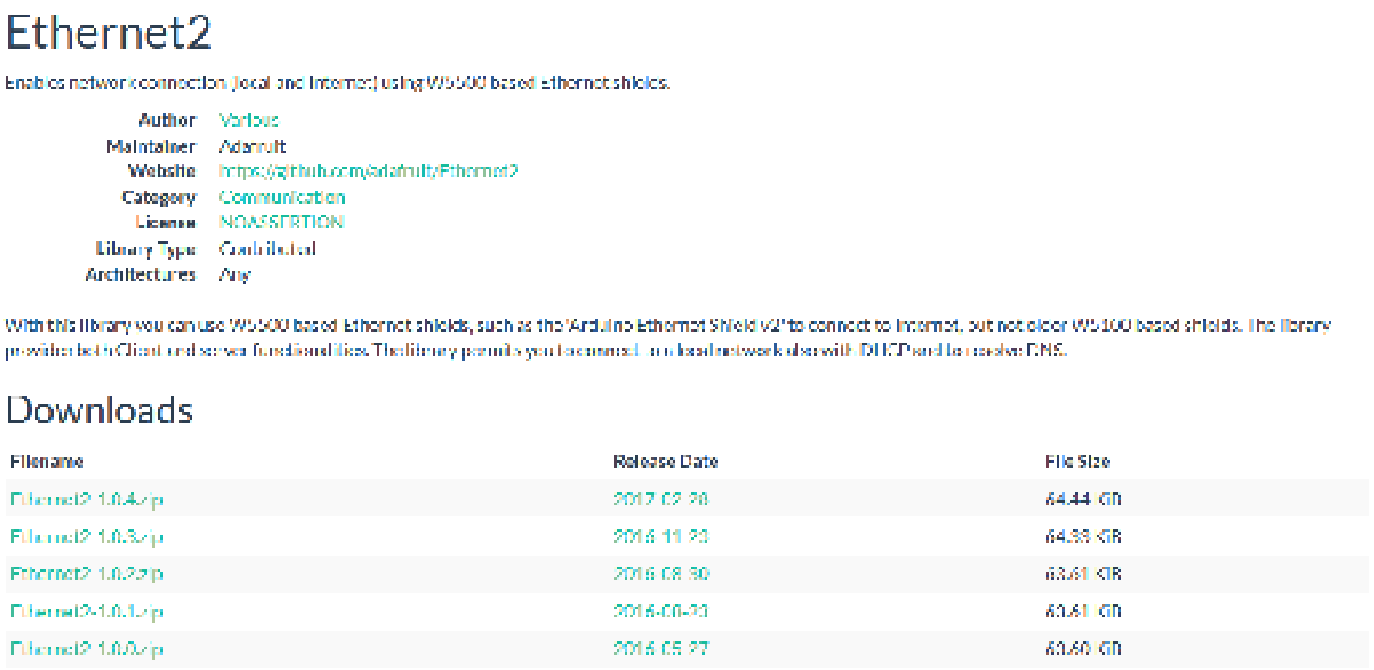
1-Arduino Ethernet shield V2 based on W5500

On the site : https://www.arduino.cc/en/Main/ArduinoEthernetShieldV2


Arduino Ethernet Shield V2

The Arduino Ethernet Shield 2 connects your Arduino board to the internet

The library to use: https://www.arduinolibraries.info/libraries/ethernet2



Ethernet2

Enables network connection (local and Internet) using W5500 based Ethernet shields.

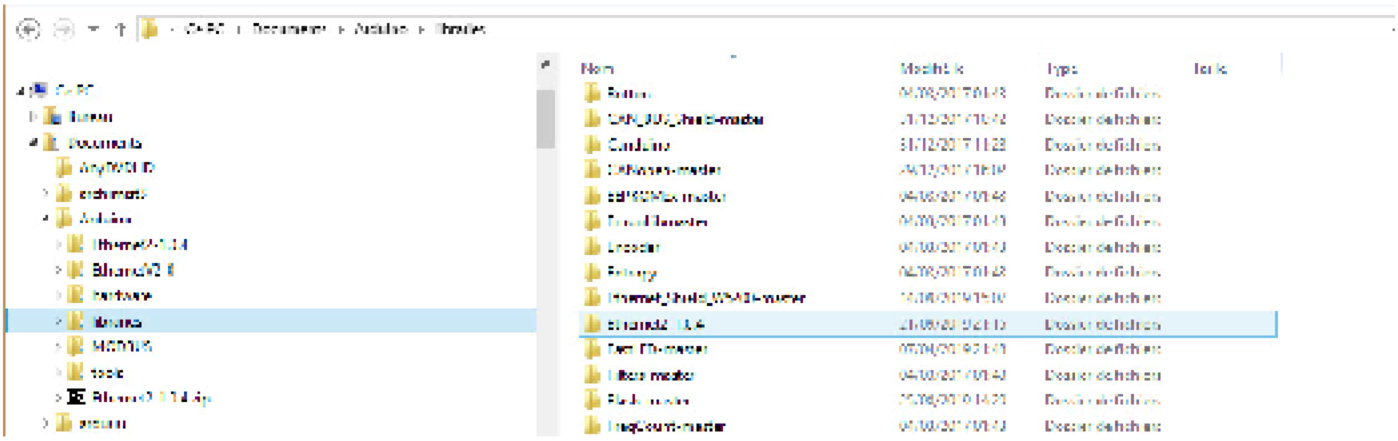
Author	Various
Maintainer	Adafruit
Website	https://github.com/adafruit/Ethernet2
Category	Communication
License	NOASSERTION
Library Type	Contributed
Architectures	Any

With this library you can use W5500 based Ethernet shields, such as the 'Arduino Ethernet Shield v2' to connect to Internet, but not older W5100 based shields. The library provides both Client and server functionalities. The library permits you to connect to a local network also with DHCP and to resolve DNS.

Downloads

Filename	Release Date	File Size
Ethernet2-1.0.4.zip	2017-02-20	64.44 KB
Ethernet2-1.0.3.zip	2016-11-23	64.33 KB
Ethernet2-1.0.2.zip	2016-08-30	63.61 KB
Ethernet2-1.0.1.zip	2016-06-23	63.61 KB
Ethernet2-1.0.0.zip	2016-05-27	63.60 KB

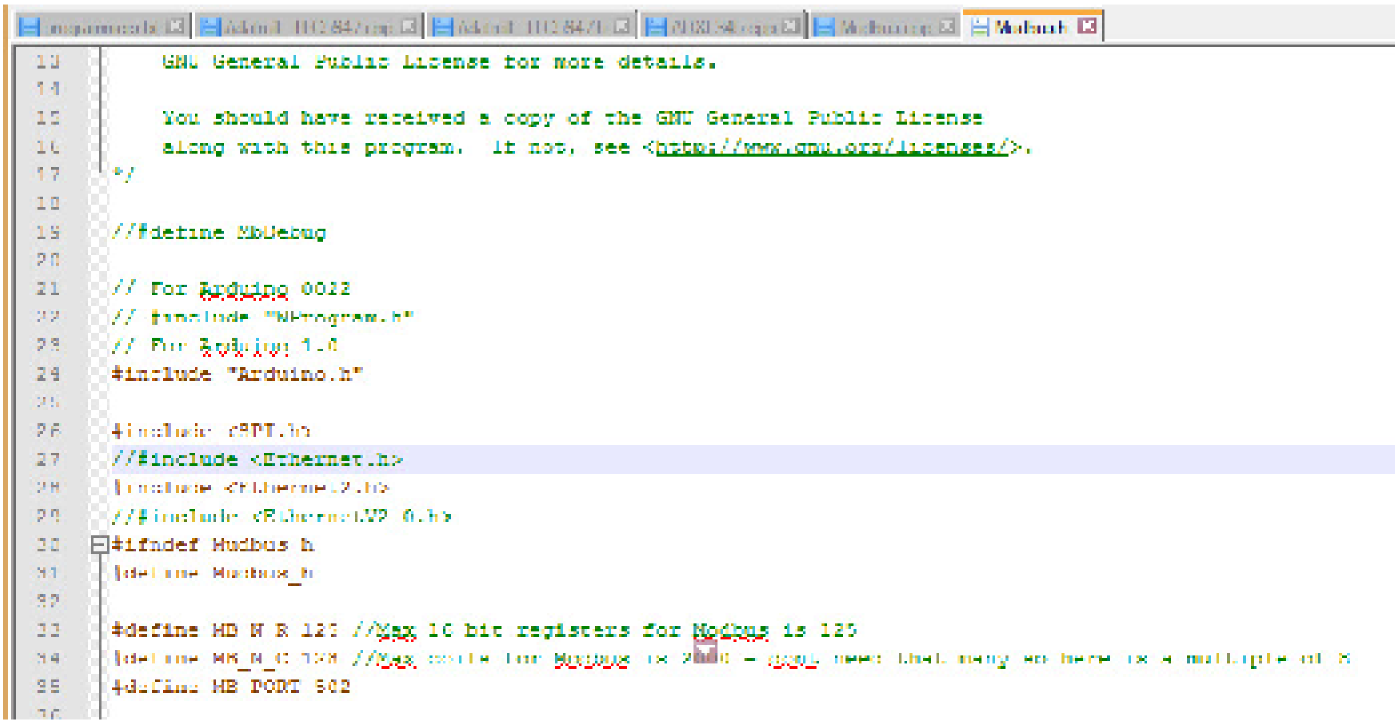
Put it here:



Modify the mudbus lib like this: change in mudbus.h

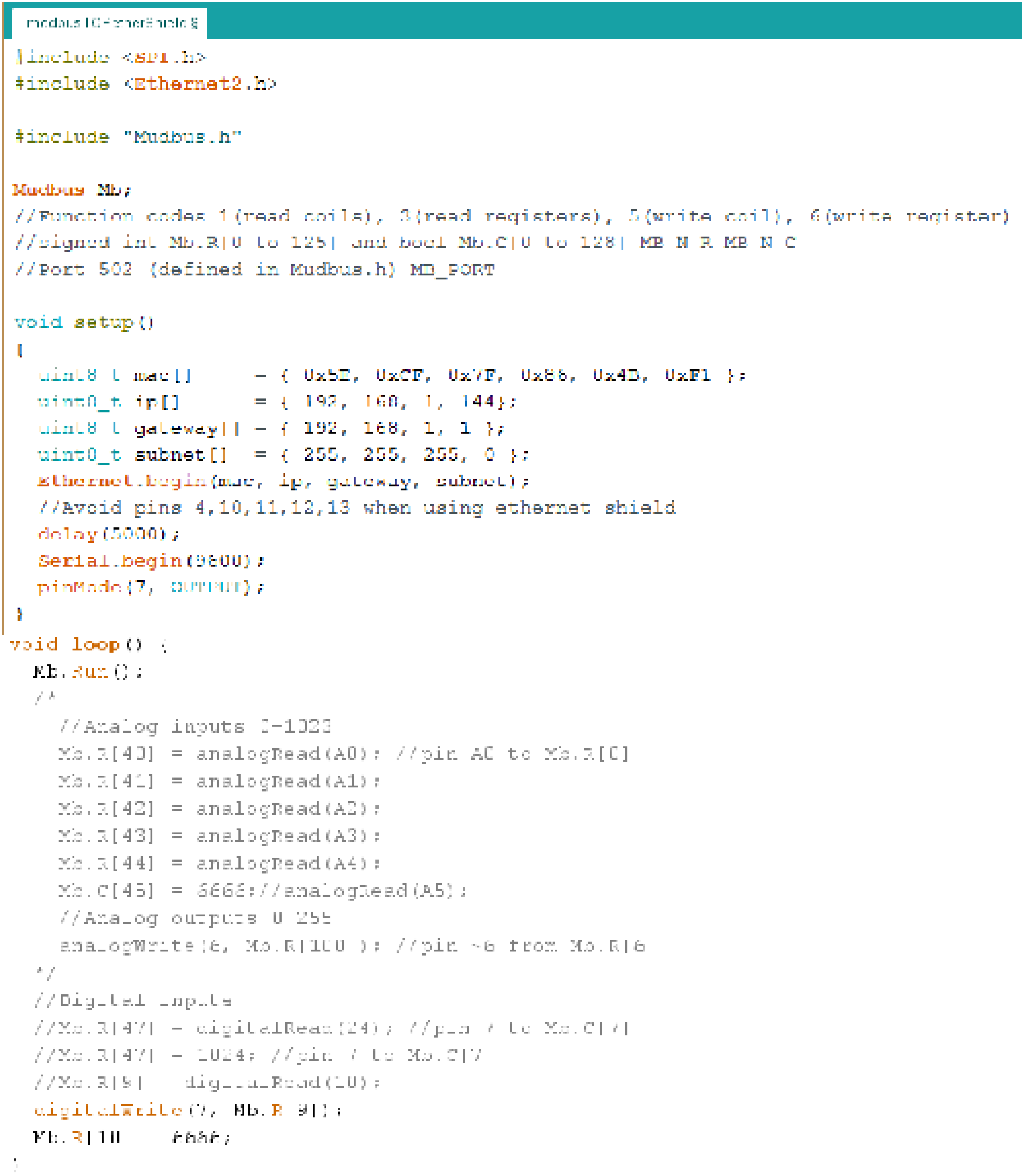
Available in https://github.com/luizcantoni/mudbus

Take a look at: http://blog.industrialshields.com/en/modbus-tcpip-library-for-arduino-based-plcs/



```
    GNU General Public License for more details.

    You should have received a copy of the GNU General Public License
    along with this program.  If not, see <https://www.gnu.org/licenses/>.
*/

//#define MbDebug

// For Arduino 0022
// #include "WProgram.h"
// For Arduino 1.0
#include "Arduino.h"

#include <SPI.h>
//#include <Ethernet.h>
#include <Ethernet2.h>
//#include <EthernetV2_0.h>
#ifndef Mudbus_h
#define Mudbus_h

#define MB_N_R 125 //Max 16 bit registers for Modbus is 125
#define MB_N_C 128 //Max coils for Modbus is 2000 - dont need that many so here is a multiple of 8
#define MB_PORT 502
```

The exemple of sketch:



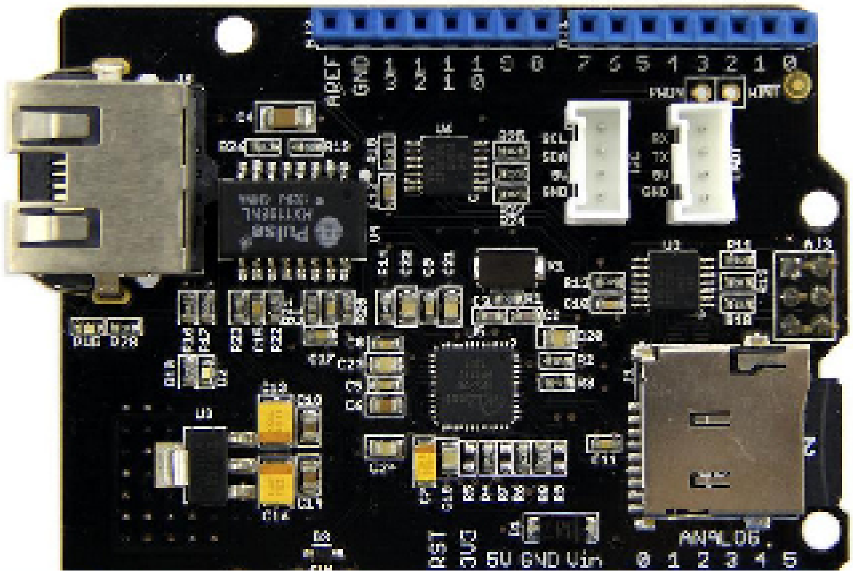
modbusTCPethernetShield §

```
#include <SPI.h>
#include <Ethernet2.h>

#include "Mudbus.h"

Mudbus Mb;
//Function codes 1(read coils), 3(read registers), 5(write coil), 6(write register)
//signed int Mb.R[0 to 125] and bool Mb.C[0 to 128] MB_N_R MB_N_C
//Port 502 (defined in Mudbus.h) MB_PORT

void setup()
{
  uint8_t mac[]     = { 0x5E, 0xCF, 0x7F, 0x86, 0x4D, 0xF1 };
  uint8_t ip[]      = { 192, 168, 1, 144};
  uint8_t gateway[] = { 192, 168, 1, 1 };
  uint8_t subnet[]  = { 255, 255, 255, 0 };
  Ethernet.begin(mac, ip, gateway, subnet);
  //Avoid pins 4,10,11,12,13 when using ethernet shield
  delay(5000);
  Serial.begin(9600);
  pinMode(7, OUTPUT);
}
void loop() {
  Mb.Run();
  /*
    //Analog inputs 0-1023
    Mb.R[40] = analogRead(A0); //pin A0 to Mb.R[0]
    Mb.R[41] = analogRead(A1);
    Mb.R[42] = analogRead(A2);
    Mb.R[43] = analogRead(A3);
    Mb.R[44] = analogRead(A4);
    Mb.C[45] = 6666;//analogRead(A5);
    //Analog outputs 0-255
    analogWrite(6, Mb.R[100 ]); //pin ~6 from Mb.R[6
  */
  //Digital inputs
  //Mb.R[47] = digitalRead(24); //pin 7 to Mb.C[7]
  //Mb.R[47] = 1024; //pin 7 to Mb.C[7
  //Mb.R[9] = digitalRead(10);
  digitalWrite(7, Mb.R[9]);
  Mb.R[10] = 6666;
}
```

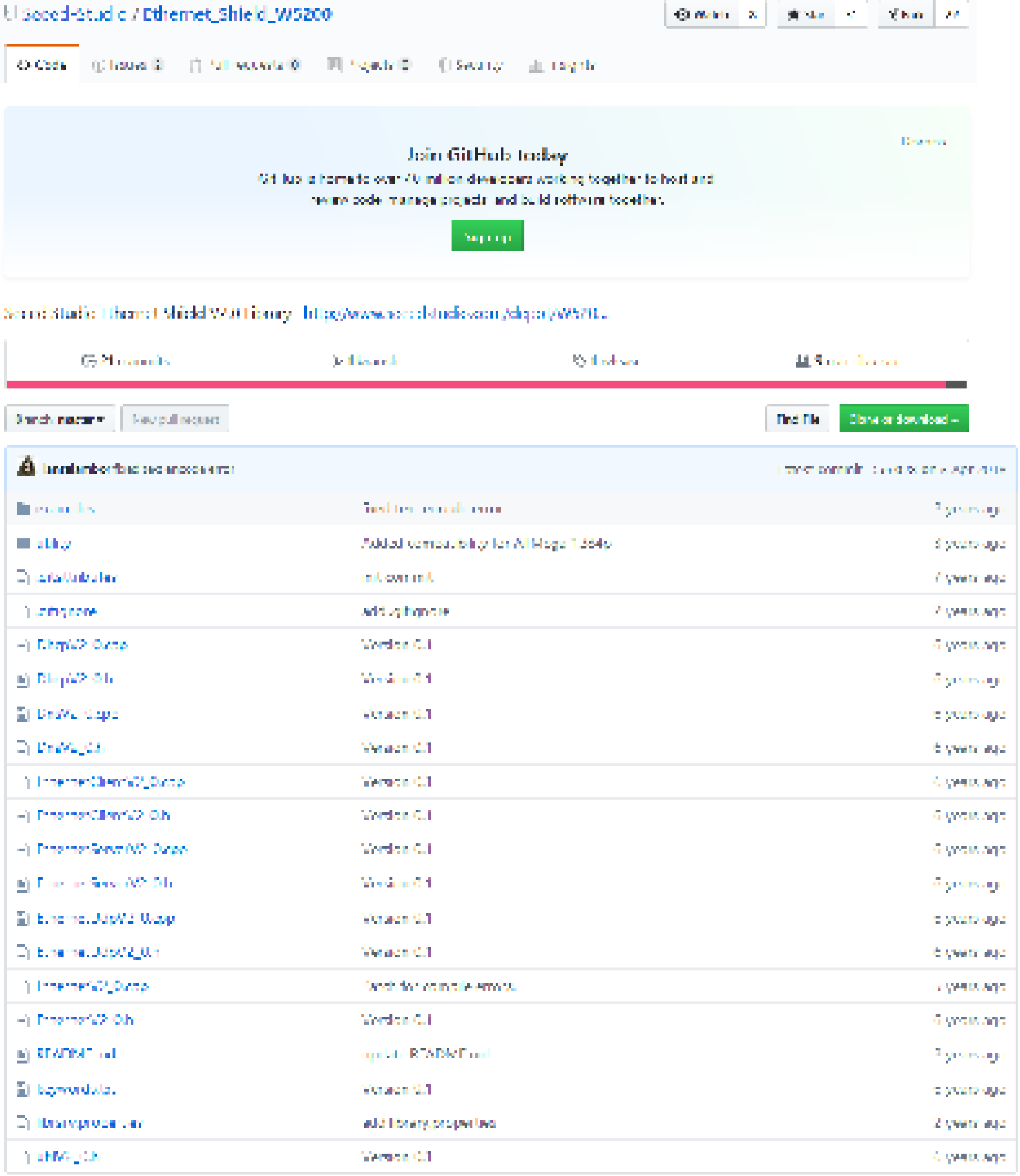
2-Arduino Ethernet shield V2.0 Seeeduino based on W5200

On the siwebsite: http://wiki.seeedstudio.com/Ethernet_Shield_V2.0/

Ethernet Shield V2.0



The library to use: https://github.com/Seeed-Studio/Ethernet_Shield_W5200



Seeed-Studio / Ethernet_Shield_W5200

Join GitHub today

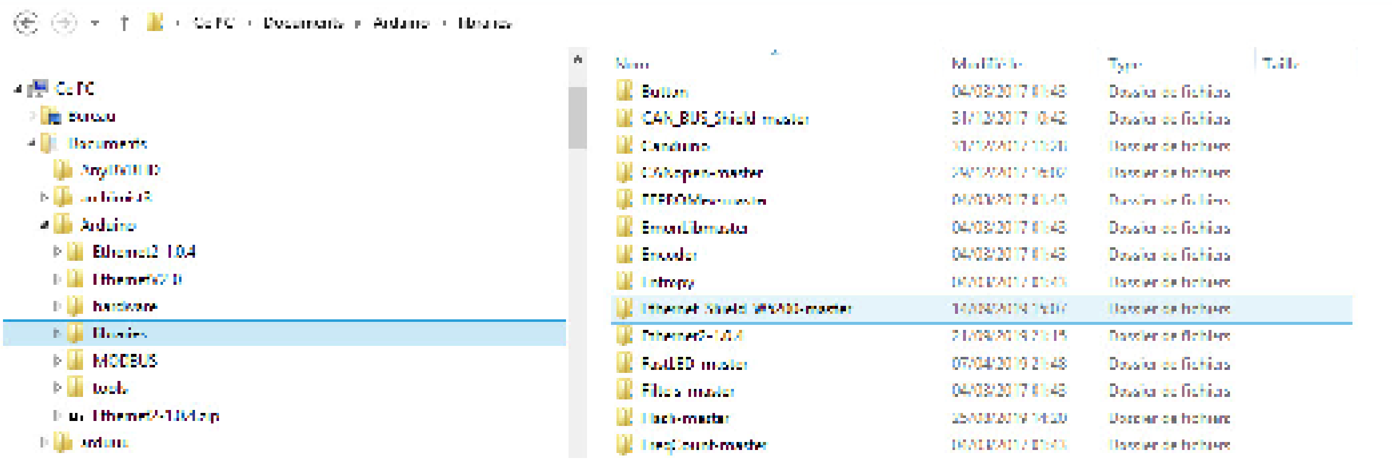
GitHub is home to over 40 million developers working together to host and review code, manage projects, and build software together.

Seeed Studio Ethernet Shield V2.0 Library http://www.seeedstudio.com/depot/W5200...

lanselambor fixed compile errors — Latest commit ... on 4 Apr 2016

examples	fixed for compile errors	3 years ago
utility	Added compatibility for ATMega 1284p	5 years ago
.gitattributes	init commit	7 years ago
.gitignore	add .gitignore	7 years ago
DhcpV2_0.cpp	Version 2.1	5 years ago
DhcpV2_0.h	Version 2.1	5 years ago
DnsV2_0.cpp	Version 2.1	5 years ago
DnsV2_0.h	Version 2.1	5 years ago
EthernetClientV2_0.cpp	Version 2.1	5 years ago
EthernetClientV2_0.h	Version 2.1	5 years ago
EthernetServerV2_0.cpp	Version 2.1	5 years ago
EthernetServerV2_0.h	Version 2.1	5 years ago
EthernetUdpV2_0.cpp	Version 2.1	5 years ago
EthernetUdpV2_0.h	Version 2.1	5 years ago
EthernetV2_0.cpp	fixed for compile errors	3 years ago
EthernetV2_0.h	Version 2.1	5 years ago
README.md	update README.md	3 years ago
keywords.txt	Version 2.1	5 years ago
library.properties	add library.properties	4 years ago
utilV2_0.h	Version 2.1	5 years ago

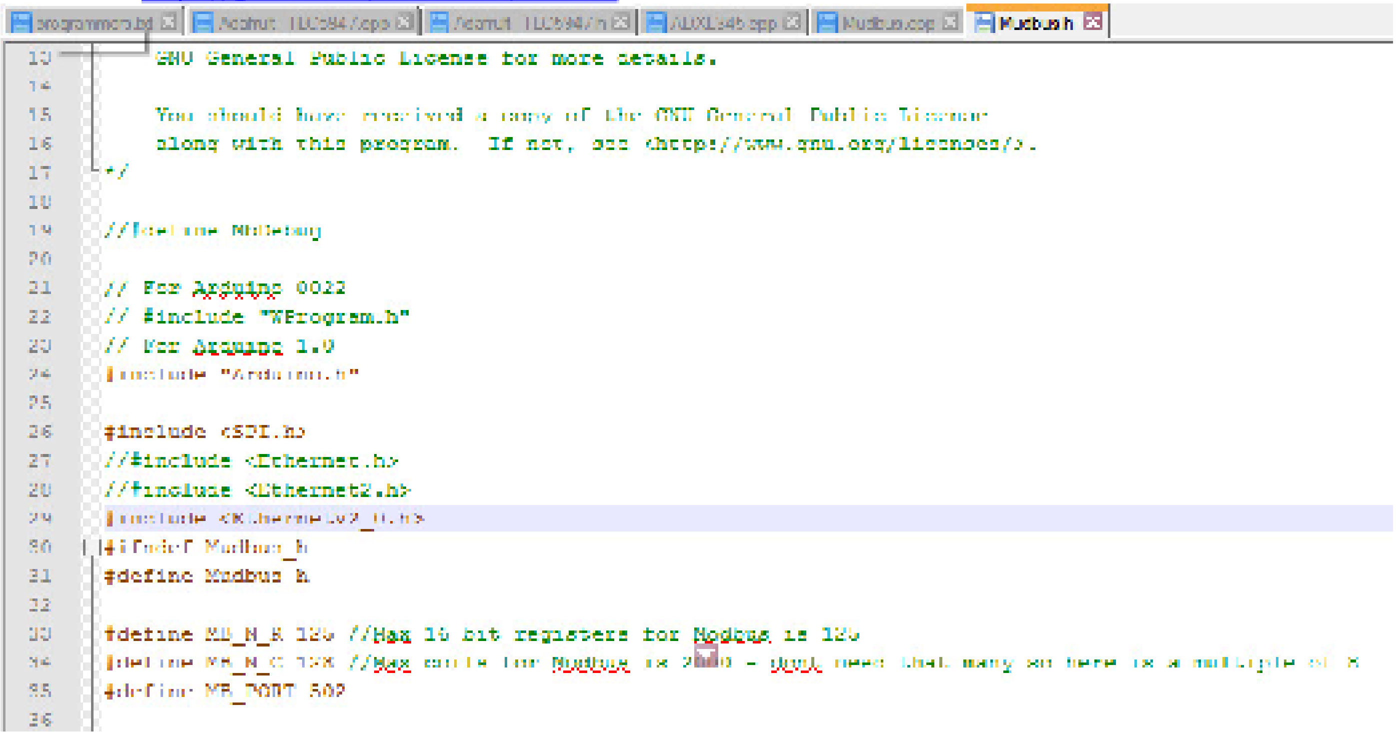
Put it here:



Nom	Modifié le	Type	Taille
Button	04/03/2017 0:43	Dossier de fichiers	
CAN_BUS_Shield-master	31/12/2017 0:42	Dossier de fichiers	
Canduino	[illegible]	Dossier de fichiers	
CANopen-master	[illegible]	Dossier de fichiers	
EEPROMEx-master	04/03/2017 0:43	Dossier de fichiers	
EmonLib-master	04/03/2017 0:43	Dossier de fichiers	
Encoder	04/03/2017 0:43	Dossier de fichiers	
Entropy	[illegible]	Dossier de fichiers	
Ethernet Shield W5500-master	[illegible]	Dossier de fichiers	
Ethernet2-1.0.4	21/08/2019 21:15	Dossier de fichiers	
FastLED-master	07/04/2019 2:48	Dossier de fichiers	
Filters-master	04/03/2017 0:43	Dossier de fichiers	
Flash-master	[illegible]	Dossier de fichiers	
FreqCount-master	[illegible]	Dossier de fichiers	

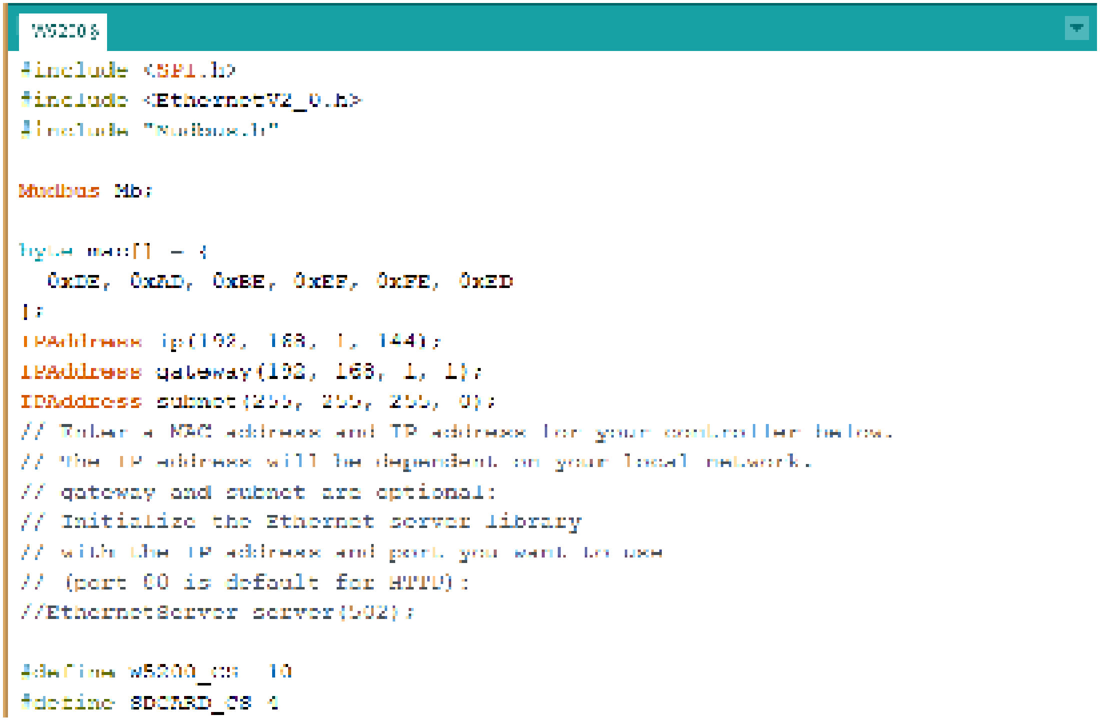
Modify the mudbus lib like this: change in mudbus.h

Available in https://github.com/luizcantoni/mudbus

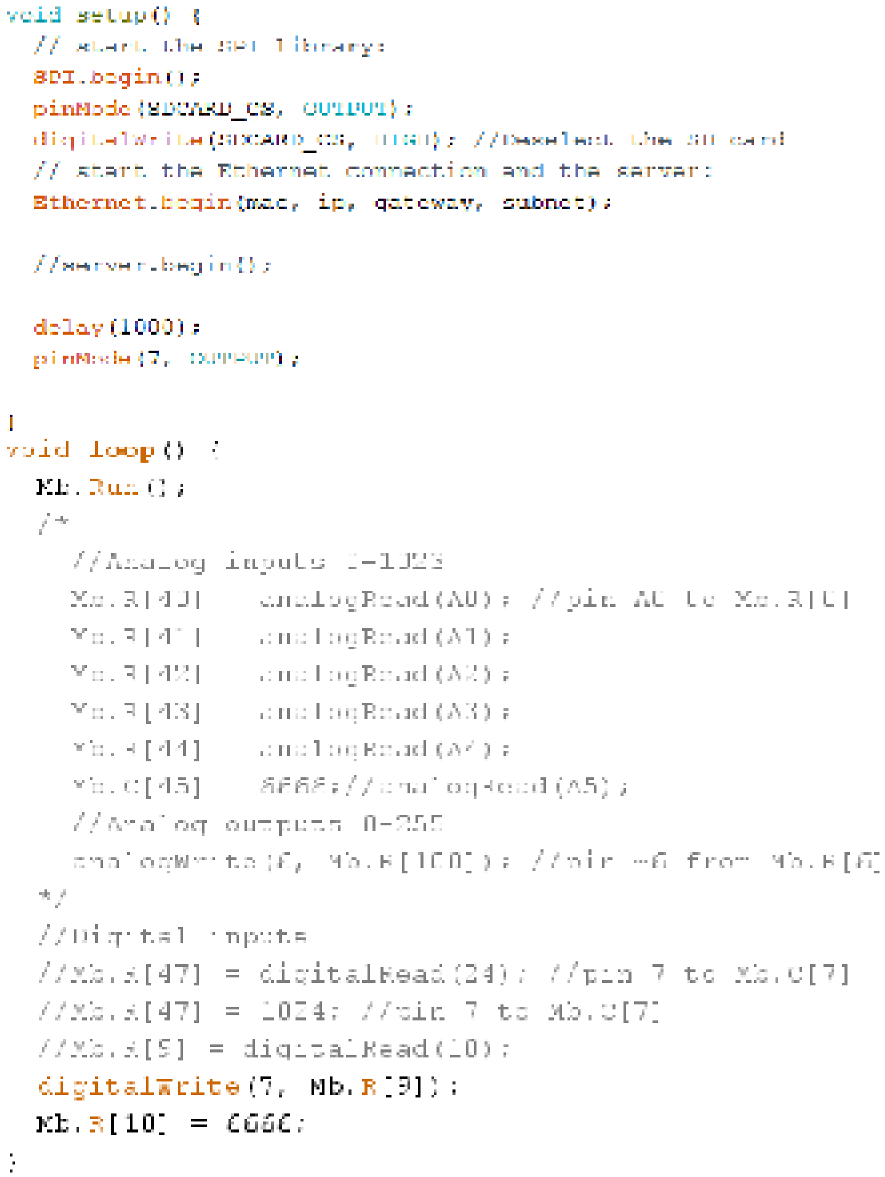


```
    GNU General Public License for more details.

    You should have received a copy of the GNU General Public License
    along with this program.  If not, see <http://www.gnu.org/licenses/>.
*/

//#define MbDebug

// For Arduino 0022
// #include "WProgram.h"
// For Arduino 1.0
#include "Arduino.h"

#include <SPI.h>
//#include <Ethernet.h>
//#include <Ethernet2.h>
#include <EthernetV2_0.h>
#ifndef Mudbus_h
#define Mudbus_h

#define MB_N_R 125 //Max 16 bit registers for Modbus is 125
#define MB_N_C 128 //Max coils for Modbus is 2000 - dont need that many so here is a multiple of 8
#define MB_PORT 502

```

The exemple of sketch:



```
#include <SPI.h>
#include <EthernetV2_0.h>
#include "Mudbus.h"

Mudbus Mb;

byte mac[] = {
  0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED
};
IPAddress ip(192, 168, 1, 144);
IPAddress gateway(192, 168, 1, 1);
IPAddress subnet(255, 255, 255, 0);
// Enter a MAC address and IP address for your controller below.
// The IP address will be dependent on your local network.
// gateway and subnet are optional:
// Initialize the Ethernet server library
// with the IP address and port you want to use
// (port 80 is default for HTTP):
//EthernetServer server(502);

#define W5200_CS  10
#define SDCARD_CS 4
```



```
void setup() {
  // start the SPI library:
  SPI.begin();
  pinMode(SDCARD_CS, OUTPUT);
  digitalWrite(SDCARD_CS, HIGH); //Deselect the SD card
  // start the Ethernet connection and the server:
  Ethernet.begin(mac, ip, gateway, subnet);

  //server.begin();

  delay(1000);
  pinMode(7, OUTPUT);

}
void loop() {
  Mb.Run();
  /*
    //Analog inputs 0-1023
    Mb.R[40] = analogRead(A0); //pin A0 to Mb.R[0]
    Mb.R[41] = analogRead(A1);
    Mb.R[42] = analogRead(A2);
    Mb.R[43] = analogRead(A3);
    Mb.R[44] = analogRead(A4);
    Mb.R[45] = 6666;//analogRead(A5);
    //Analog outputs 0-255
    analogWrite(6, Mb.R[100]); //pin ~6 from Mb.R[6]
  */
  //Digital inputs
  //Mb.R[47] = digitalRead(24); //pin 7 to Mb.C[7]
  //Mb.R[47] = 1024; //pin 7 to Mb.C[7]
  //Mb.R[9] = digitalRead(10);
  digitalWrite(7, Mb.R[9]);
  Mb.R[10] = 6666;
}
```