Contents

[PI PICO CODE 2](#_Toc90133143)

[PCB: 2](#_Toc90133144)

[SETUP: 2](#_Toc90133145)

[LIBRARY: 2](#_Toc90133146)

[CAT24C512 2](#_Toc90133147)

[QMA7981 2](#_Toc90133148)

[INA3221 3](#_Toc90133149)

[MAX3485ESA 3](#_Toc90133150)

[MAX7219 3](#_Toc90133151)

[W5100S 4](#_Toc90133152)

[Modbus RTU 4](#_Toc90133153)

[Modbus TCP 4](#_Toc90133154)

[Modbus TCP 4](#_Toc90133155)

[Read Status of P0-2 4](#_Toc90133156)

[SENSOR Status 4](#_Toc90133157)

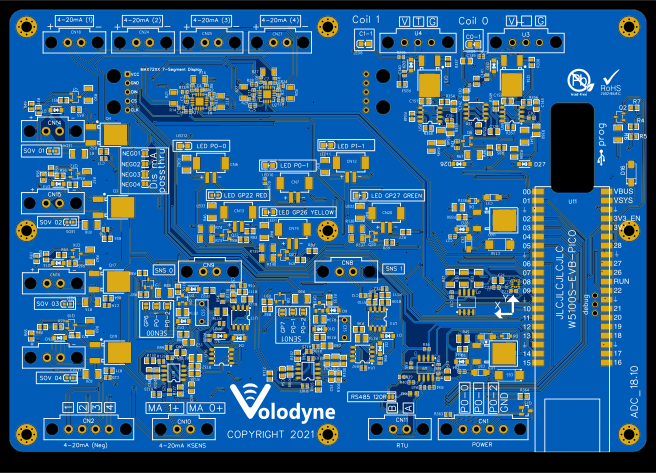
[Solenoid drivers 5](#_Toc90133158)

[LED Display 5](#_Toc90133159)

# PI PICO CODE

## PCB:

For address details see schematic from Easy EDA.



## SETUP:

Language: CPP using VS Code and PI PICO code chain, Cmake, GCC Compiler.

See: <https://www.youtube.com/watch?v=Q1Kfg8k54jM>

## LIBRARY:

Default settings shall be documented and set in the setup main file.

Libraries shall be customized for the RP2040.

Format and print values to serial

### CAT24C512

EPROMM

Serial: I2C, Address 0x50

Provide standard library.

Search for headers and locate header files and cpp file and samples.

Provide Write and Read function for writing and reading Integers (10^9),

i.e.

int A = epprom.read(“A1”)

epprom.write(“A1”, a);

There are libraries that take care of addressing so that variables can be named, saved and retrieved.

Format and print values to serial

## MPU-6050 / GY-521

GYRO

Serial: I2C, Address 0x68

Format and print values to serial

Provide function for retrieving all data and temp from IC and the data type.

Format and print values to serial

### INA3221

Current Sensor, 3 channel

Serial: I2C, Address 0x40 and 0x41

0x40, 1 -> mA3 , 2 -> mA2, 3 -> mA4

0x41, 1 -> Sns0 , 2 -> Sns1, 3 -> mA1

Format and print values to serial

Wear leveling shall be applied to writing data across multiple addresses.

Provide read functions to return the mA value based on the Shunt resistor used in the circuit.

### MAX3485ESA

Modbus RTU

GP4 -> RO

GP5 -> DI

GP14 -> DG

Implement Modbus library, assign placeholders for read bits and read float over RS485 (see Ethernet too).

Format and print values to serial

### MAX7219

8 digit 7 segment display

SPI1, Serial2

Implement Modbus library, Write 8 digit Integer to display, if integer is less than 8 digits then only illuminate the required digits.

Format and print values to serial

### W5100S

Ethernet, part of the W5100S-EVB-PICO board.

Implement static IP ethernet library.

In main file set MAC Address, IP Address and subnet mask.

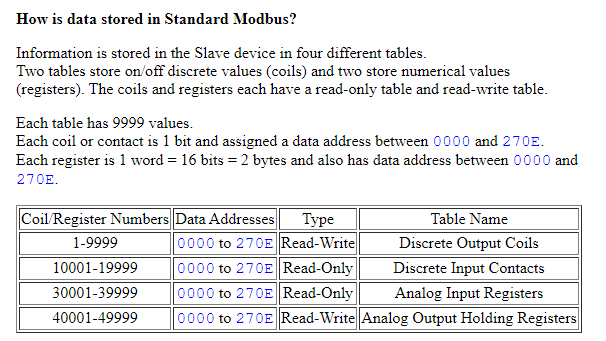
### Modbus RTU

Implement RTU over the RS3485 chip.

### Modbus TCP

Implement Modbus to write library for reading from Modbus TCP, and for Reading from Modbus Address.

Test coils and register using modbus scan software, i.e. modscan or vinci



https://the-vinci.com/vinci-software

Implement Modbus to write library for reading from Modbus TCP, and for Reading from Modbus Address.

Write to registry the current from the INA219 single channel, note on PCB 18.10 the INA is 3 channel and 3 channel needs to be implemented.

### Read Status of P0-2

Read status of GP28, this could be interrupt.

### SENSOR Status

Sensor 0 and 1 can be reset by pulling pins GP6 and GP7 up (+3V3).

### Solenoid drivers

Solenoids can be actuated by pulling pins GP0, GP1, GP2 and GP3 up.

### LED Display

LED’s can be illuminated by pulling pins GP22, GP26, and GP27 up.

# Mapping

In loop update;

1. registers for outputs values (current, gyro, Eprom IGN Variable).
2. registry write for RST0 and RST1.
3. Coils (i.e. P0-2).
4. Coils for output led.
5. Coils for SOV.

Eprom increment is performed on P0-0 when permissive from sensors is true.

# MQTT:

Report variables from Sensors or from Algorithm over MQTT to MQTT bridge over Ethernet.

# LOOP:

Data processing is needs from signals, the client will provide input to the coder once the first half of the working function has been delivered.