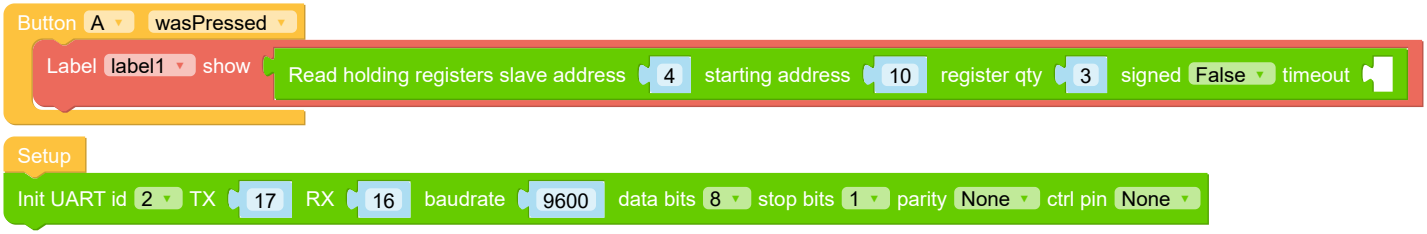
Modbus Master

Example

o implement the Modbus master station, press button A to trigger the reading of the hold register data of the slave station address 4 and display it on the screen




```
from m5stack import *
from m5ui import *
from uiflow import *
from modbus.master.uModBusSerial import uModBusSerial

setScreenColor(0x222222)

title0 = M5Title(title="Master Read Holding Register", x=60, fgcolor=0xFFFFFF, bgcolor=0xff0078)
label0 = M5TextBox(16, 212, "Press A Send Data", lcd.FONT_Default, 0xFFFFFF, rotate=0)
label1 = M5TextBox(15, 69, "Text", lcd.FONT_DejaVu24, 0xe2ff00, rotate=0)

def buttonA_wasPressed():
  # global params
  label1.setText(str(modbus.read_holding_registers(4, 10, 3, False, timeout=)))
  pass
btnA.wasPressed(buttonA_wasPressed)


modbus = uModBusSerial(2, tx=17, rx=16, baudrate=9600, data_bits=8, stop_bits=1, parity=None, ctrl_pin=
```

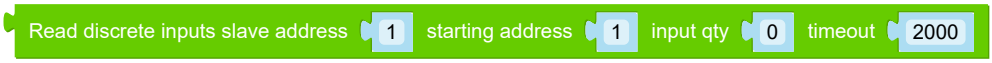
API


```
modbus = uModBusSerial(1, tx=0, rx=0, baudrate=9600, data_bits=8, stop_bits=1, parity=None, ctrl_pin=No
```

- UART initialization configuration

```
print((str('address:') + str((modbus.read_coils(1, 1, 0, timeout=2000)))))
```

- Read coil status




```
print((str('address:') + str((modbus.read_discrete_inputs(1, 1, 0, timeout=2000)))))
```

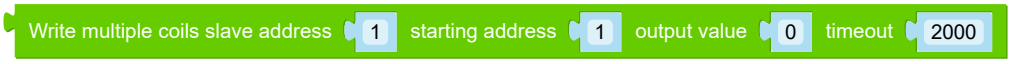
- Read discrete inputs


```
print((str('address:') + str((modbus.read_holding_registers(1, 1, 0, True, timeout=2000)))))
```

- Read holding registers


```
print((str('address:') + str((modbus.read_input_registers(1, 1, 0, True, timeout=2000)))))
```

- Read input registers

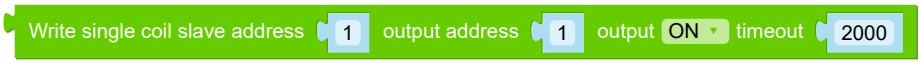



```
print((str('write address:') + str((modbus.write_multiple_registers(1, 1, 0, True, timeout=2000)))))
```

- Write multiple coils


```
print((str('write address:') + str((modbus.write_multiple_coils(1, 1, 0, timeout=2000)))))
```

- Write multiple registers




```
print((str('write address:') + str((modbus.write_single_coil(1, 1, 0xFF00, timeout=2000)))))
```

- Write single coil

```
print((str('write address:') + str((modbus.write_single_register(1, 1, 0, True, timeout=2000)))))
```

- Write single register