

MyScan - Modbus Scanner GUI (Python + PyQt5)

🔍 Proje Hakkında

MyScan, ModScan benzeri, modern ve kullanıcı dostu bir Modbus tarayıcı masaüstü uygulamasıdır. Bu proje, teknik kullanıcıların Modbus TCP ile veri okuması yapmasına olanak tanıyan profesyonel görünümlü bir GUI ile geliştirilmiştir. Uygulama Python ve PyQt5 kütüphanesi kullanılarak yazılmıştır.

Özellikler

- IP, Port, Unit ID, Başlangıç Adresi ve Uzunluk gibi temel ayarların girilebilmesi
- Register tipi (Holding, Input, Coil, Discrete) seçimi
- Veri tipi (Int, UInt, Float, Double, Hex, Binary, ASCII) seçimi
- Modern, renklendirilmiş ve düzenli kullanıcı arayüzü
- Okunan verilerin tablo şeklinde görüntülenmesi

📁 Kullanılan Kütüphaneler

```
from PyQt5.QtWidgets import (  
    QApplication, QMainWindow, QWidget, QLabel, QLineEdit, QPushButton,  
    QComboBox, QVBoxLayout, QHBoxLayout, QTableWidgetItem,  
    QTableWidgetItem  
)
```

Açıklama: PyQt5 ile arayüz oluşturmak için gerekli tüm bileşenleri içeren kütüphaneler yüklenmiştir.

- **QApplication** : Uygulamanın ana olay döngüsünü başlatır
- **QMainWindow** : Ana pencere bileşeni
- **QWidget** : Arayüz taşıyıcısıdır
- **QLabel** : Metin göstermek için
- **QLineEdit** : Kullanıcıdan metin (örneğin IP, adres) almak için
- **QPushButton** : Tıklanabilir düğmeler
- **QComboBox** : Açılır seçim kutuları (register ve data type seçimi)
- **QTableWidgetItem** : Veri sonuçlarını tabloda göstermek için

GUI Elemanları ve Açıklamaları

IP ve Port Alanı (Sağ Üst Köşe)

```
ip_label = QLabel("IP:")
self.ip_input = QLineEdit("127.0.0.1")
port_label = QLabel("Port:")
self.port_input = QLineEdit("502")
```

Kullanıcıdan Modbus cihazına ait IP ve port bilgisi istenir.

Connect Butonu

```
self.connect_button = QPushButton("Connect")
self.connect_button.setObjectName("connect_button")
```

Bağlantıyı başlatmak için yeşil renkli bir buton. GUI'de sağ üst köşede görünür.

Başlangıç Adresi ve Uzunluk Alanı (Sol Üst)

```
self.address_input = QLineEdit("0")
self.count_input = QLineEdit("10")
```

Okumaya başlanacak adres ve kaç adet register okunacağı bilgileri.

Register Tipi ve Veri Tipi Seçimi (Orta Alan)

```
self.regtype_combo.addItem("Holding")
self.regtype_combo.addItem("Input")
self.regtype_combo.addItem("Coil")
self.regtype_combo.addItem("Discrete")
self.datatype_combo.addItem("Int")
self.datatype_combo.addItem("UInt")
self.datatype_combo.addItem("Float")
self.datatype_combo.addItem("Double")
self.datatype_combo.addItem("Hex")
self.datatype_combo.addItem("Binary")
self.datatype_combo.addItem("ASCII")
```

- Register Type : Hangi tür register okunacak (03, 04, 01, 02) - Data Type : Okunan veri nasıl yorumlanacak (tam sayı, float, binary vb.)

Start Read ve Stop Butonları

```
self.read_button = QPushButton("Start Read")
self.read_button.setObjectName("read_button")
self.stop_button = QPushButton("Stop")
self.stop_button.setObjectName("stop_button")
```

Start Read okuma başlatmak için, **Stop** ise durdurmak için kullanılır.

Veri Gösterim Tablosu (Alt Kısım)

```
self.table = QTableWidgetItem()
self.table.setColumnCount(2)
self.table.setHorizontalHeaderLabels(["Address", "Value"])
```

Okunan register değerlerini tablo halinde gösterir. Her satırda adres ve değeri yer alır.

Görsel Tasarım & Modern Stil

```
self.setStyleSheet("""
    QMainWindow {
        background-color: #f0f2f5;
    }
    QLabel {
        font-weight: bold;
    }
    QLineEdit, QComboBox {
        padding: 5px;
        border: 1px solid #ccc;
        border-radius: 4px;
        background-color: #ffffff;
    }
    QPushButton {
        padding: 6px;
        font-weight: bold;
        border-radius: 5px;
    }
    QPushButton#connect_button {
        background-color: #4CAF50;
        color: white;
    }
    QPushButton#stop_button {
        background-color: #e74c3c;
        color: white;
    }
    QPushButton#read_button {
        background-color: #3498db;
        color: white;
    }
})
```

```

QWidget {
    background-color: white;
    alternate-background-color: #f2f2f2;
    gridline-color: #dcdcdc;
}
QHeaderView::section {
    background-color: #dfe4ea;
    font-weight: bold;
    border: 1px solid #cfd8dc;
    padding: 4px;
}
"""
)

```

Arayüzü modernleştirmek için CSS benzeri `setStyleSheet()` metodu kullanıldı.

- `#connect_button` : Yeşil bağlan butonu
- `#stop_button` : Kırmızı durdurma butonu
- `#read_button` : Mavi okuma başlatma butonu

Gelecek Adımlar

- `Connect` butonuna basıldığında IP/Port üzerinden bağlantı kurulacak
- `Start Read` tıklanınca seçilen parametrelere göre veri okuma başlayacak
- RS485 desteği ikinci aşamada eklenecek

Kullanıcının Özel Talimatları

Bu proje, kullanıcının aşağıdaki görsel ve fonksiyonel isteklerine göre şekillendirilmiştir:

- IP ve Port alanları sağ üstte hizalı olacak
- Başlangıç adresi ve uzunluk sol üstte
- Register ve veri tipi ortada, teknik cihaz arayüzü mantığıyla hizalı
- Hex, Binary gibi tüm veri türleri dahil edildi
- Modern ve renkli arayüz teması uygulandı

Dosya Yapısı (Önerilen)

```

MyScan/
├── main.py          # PyQt5 GUI uygulaması

```

— README.md	# Bu belge
— modbus_handler.py	# (İleri adımda eklenecek - veri okuma fonksiyonları)

Gereksinimler

- Python 3.8+
- PyQt5 kurulumu:

```
pip install pyqt5
```

Destek / Katkı

İleride bu uygulamayı open-source hale getirmek istersen, katkı yapanlar için açık bir [CONTRIBUTING.md](#) de oluşturulabilir.

Herhangi bir yardım ihtiyacında bu proje ChatGPT ile yürütülmüştür ve güncellemeler buradan takip edilmiştir.