



UNIVERSITATEA TEHNICĂ "GHEORGHE ASACHI" IAȘI
FACULTATEA DE AUTOMATICĂ ȘI CALCULATOARE
SPECIALIZAREA CALCULATOARE ȘI TEHNOLOGIA INFORMAȚIEI
DISCIPLINA REȚELE DE CALCULATOARE PROIECT

Aplicație pentru monitorizarea resurselor SO Client Modbus TCP

Coordonator: Botezatu Nicolae-Alexandru

Studenti: Cucu Cătălin-Ionuț, Grădinaru Cezar-Nicolae

Grupă: 1308B

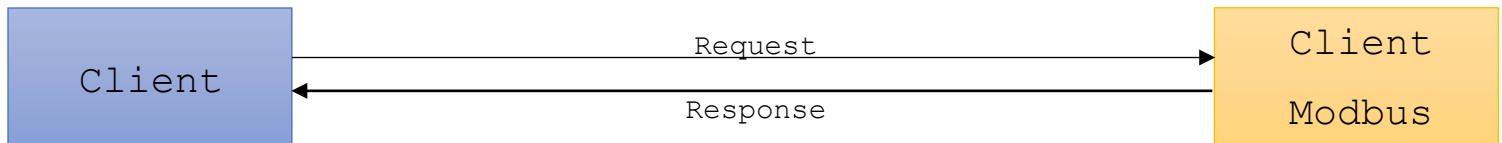
2020-2021

Scopul aplicației

Poriectare și implemenarea unei aplicații de monitorizare a resurselor sistemului de operare prin intermediul unei comunicații Modbus TCP.

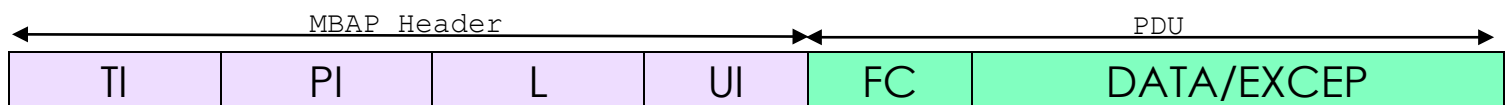
Despre Modbus TCP

1.Interacțiunea Client/Server(Client Modbus):



Modbus TCP asigură comunicarea client/server conectate la o rețea TCP/IP prin mesaje - ADU. Clientul trimite un mesaj, request, către server care oferă înapoi un mesaj, response, cu datele obținute în urma aplicării funcției cerute în request sau un mesaj de excepție.

2.ADU(Application Data Unit):



a.MBAP Header(Modbus Application Header):

- **Transaction Identifier(TI)** [2 bytes]
→codul tranzacției, unic, setat de client;
- **Protocol Identifier(PI)** [2 bytes]
→mereu 0;
- **Length(L)** [2 byte]
→numărul de biți care urmeaza după el la dreapta : 1+1+n;
- **Unit Identifier(UI)** [1 byte]
→indentifică un server care nu este conectat la o rețea TCP/IP;

b.PDU(Protocol Data Unit):

- **Request/Respond PDU**(ReqPDU/RspPDU)
→Function Code(FC)[1 byte]: 1, 2, 3, 4, 5, 6, 15, 16
→Request Data(DATA)[n bytes]: valori, offseturi
- **Exception response PDU**(ExcepRspPDU)
→Function Code(FC)[1 byte]: 1, 2, 3, 4, 5, 6, 15, 16 + 0x80;
→Exception Code(EXCEP)[1 byte]: 1, 2, 3, 4, 6;

3.Funcții:

•Read Coil Status(0x01)

→Rol:

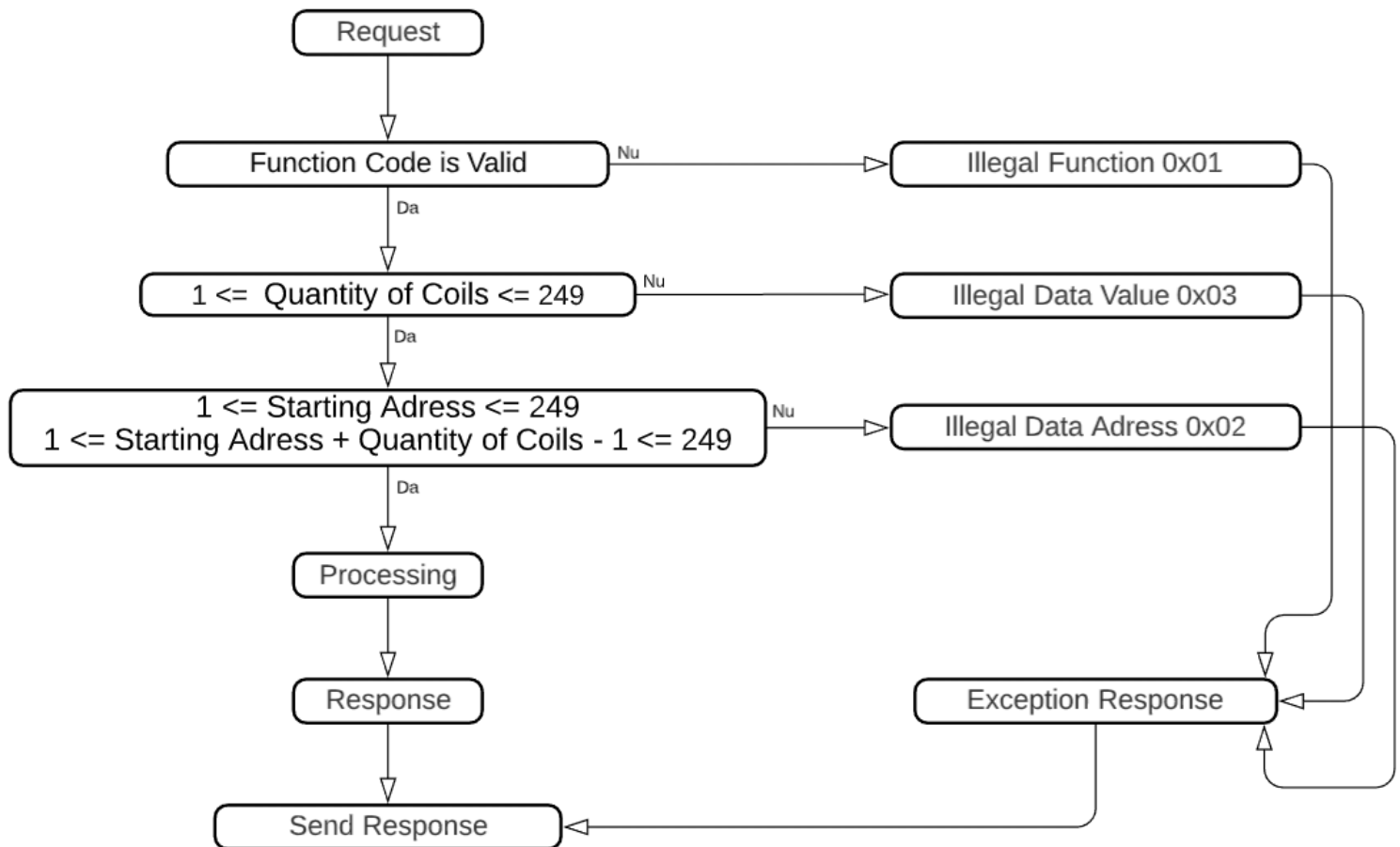
Funcție ce va returna sub formă de bytes valorile fiecărui coil(1 bit) de la o adresă specificată.

→ Structură mesaje:

Request
Function Code
Starting Adress
Quantity of Coils

Response
Function Code
Byte Count
Status of Coils

→Diagrama de funcționare:



•Read Discrete Inputs(0x02)

->Rol:

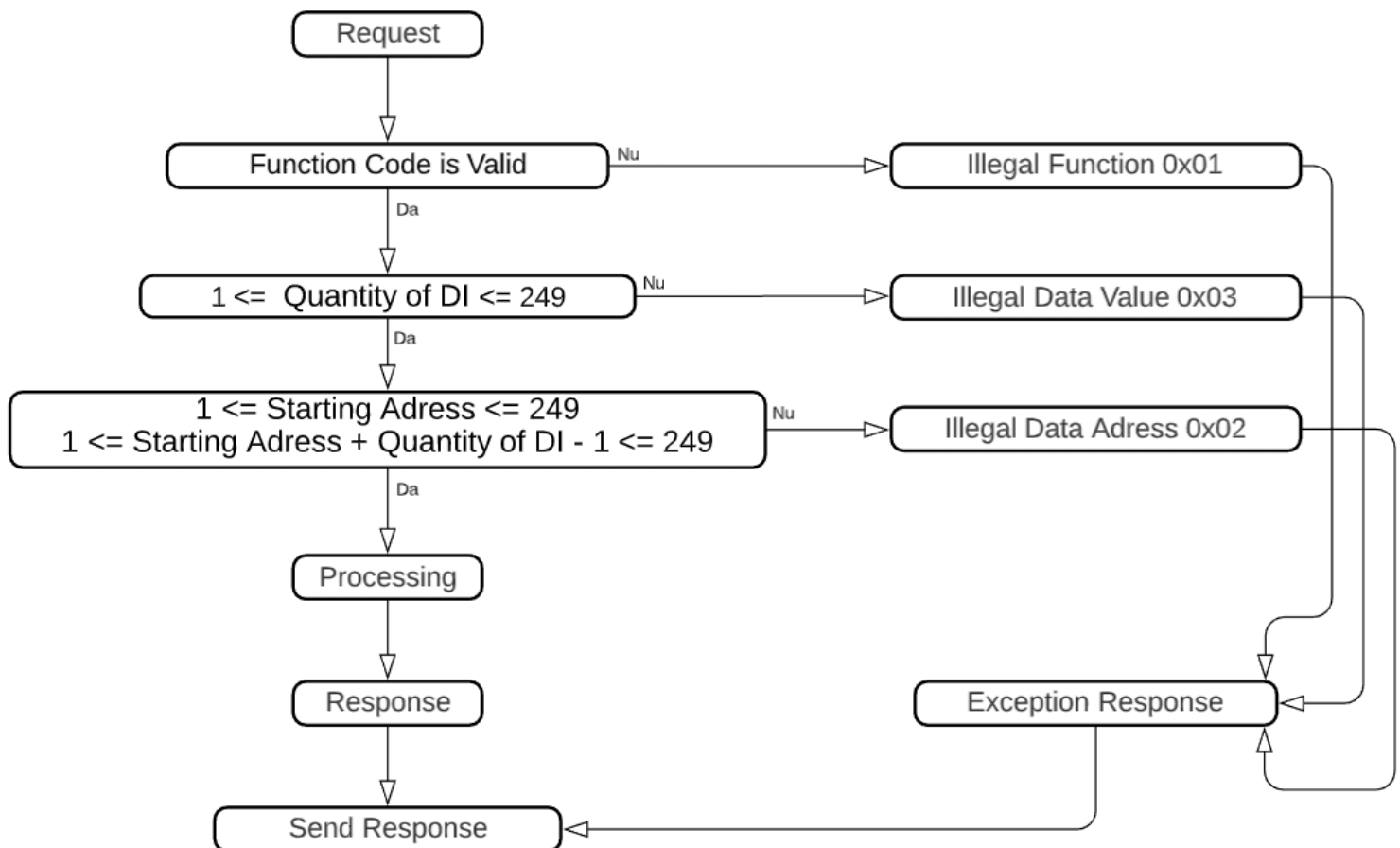
Funcție ce va returna sub formă de bytes valorile fiecărui discrete input(1 bit) - DI de la o adresă specificată.

-> Structură mesaje:

Request
Function Code
Starting Adress
Quantity of DI

Response
Function Code
Byte Count
Status of DI

->Diagrama de funcționare:



•Read Holding Registers(0x03)

->Rol:

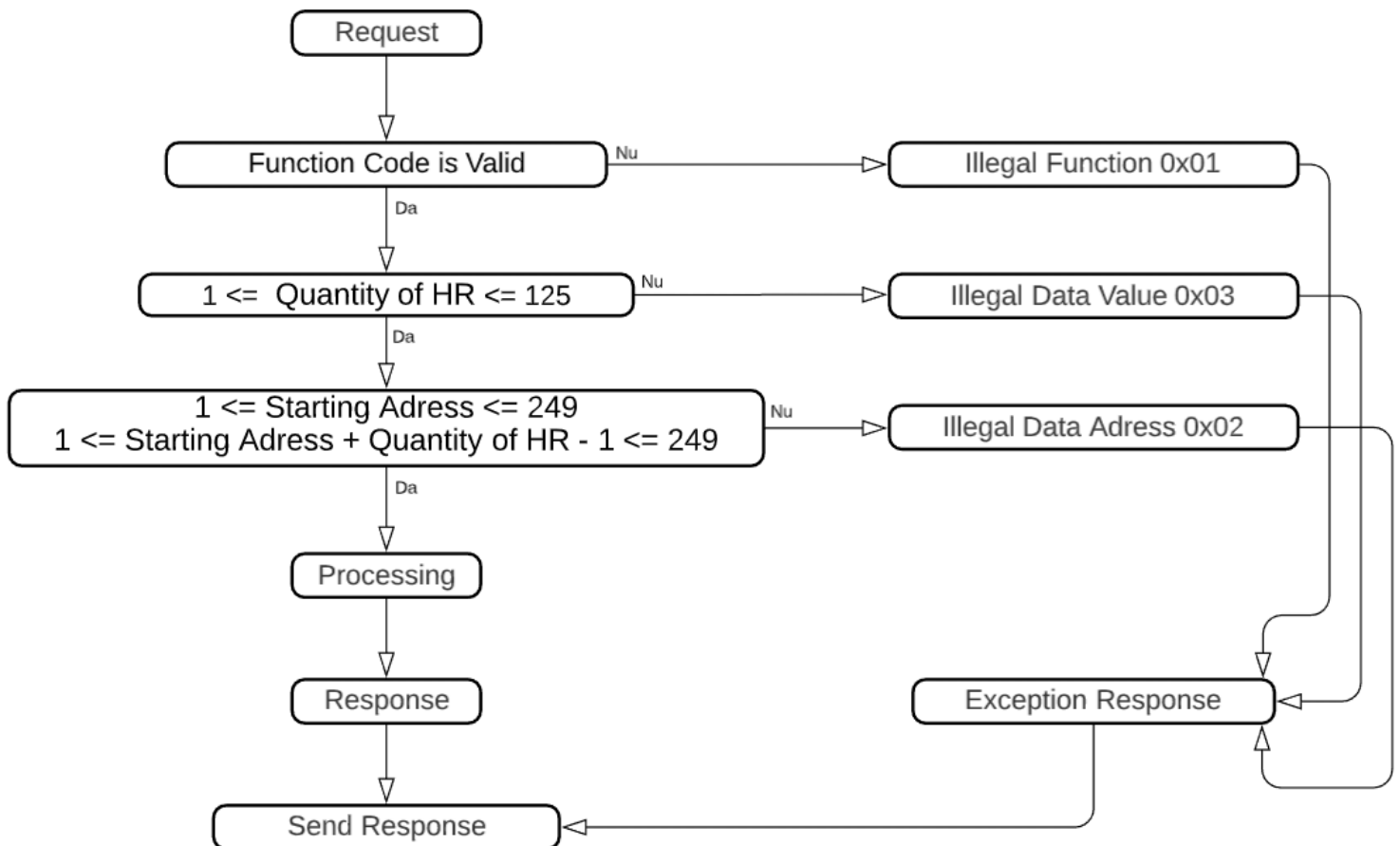
Funcție ce va returna valorile fiecărui holding register(1 byte) - HR de la o adresă specificată.

-> Structură mesaje:

Request
Function Code
Starting Adress
Quantity of HR

Response
Function Code
Byte Count
Value of HR

->Diagrama de funcționare:



•Read Input Registers(0x04)

->Rol:

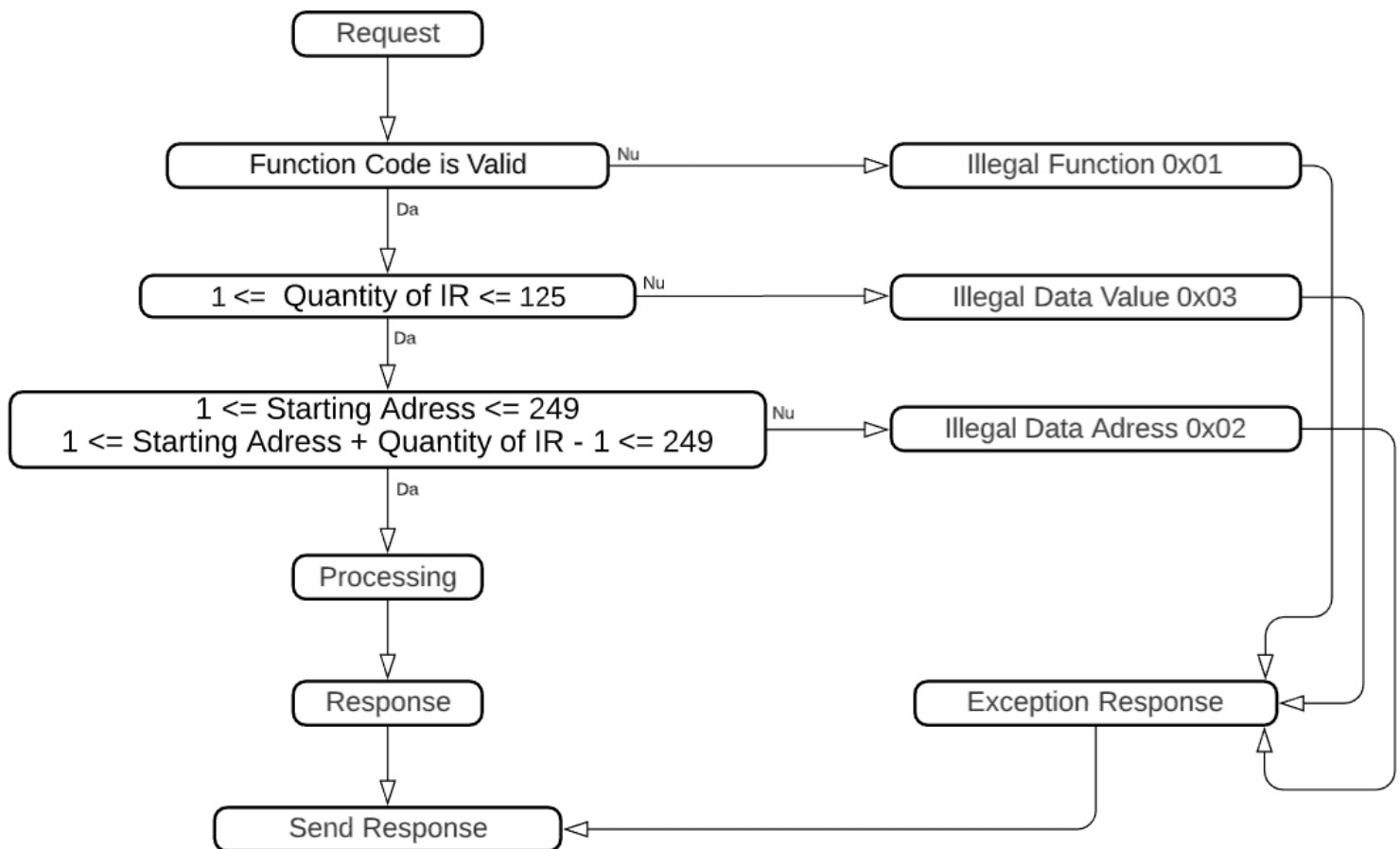
Funcție ce va returna valorile fiecărui input register(1 byte) – IR de la o adresă funcționare.

-> Structură mesaje:

Request
Function Code
Starting Address
Quantity of IR

Response
Function Code
Byte Count
Value of IR

->Diagrama de functionare:



•Force Single Coil(0x05)

->Rol:

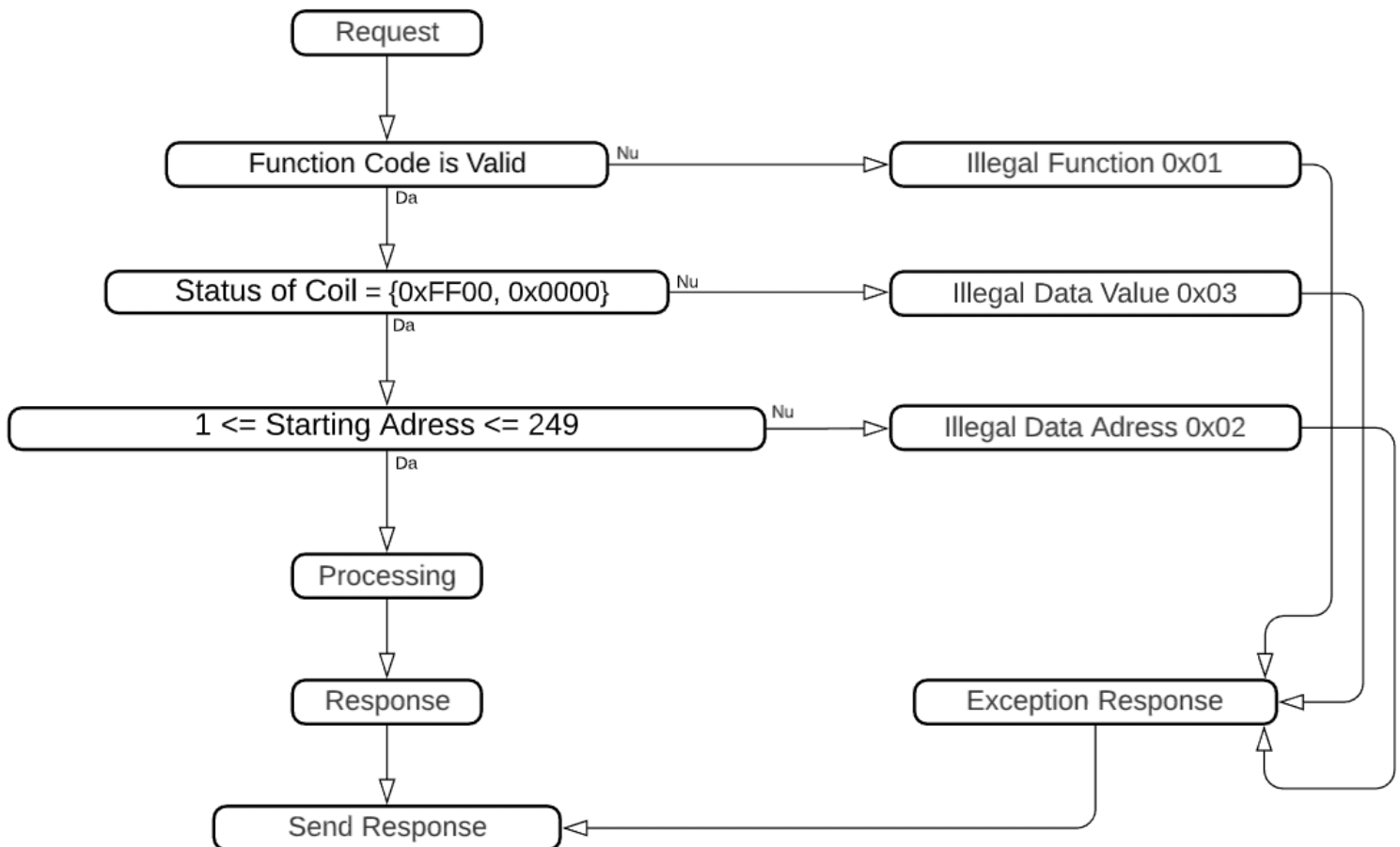
Funcție ce va scrie valoarea unui coil(1 bit) la o adresă specificată.

-> Structură mesaje:

Request
Function Code
Coil Adress
Status of Coil

Response
Function Code
Coil Adress
Status of Coil

->Diagrama de funcționare:



• Write Single Register(0x06)

-> Rol:

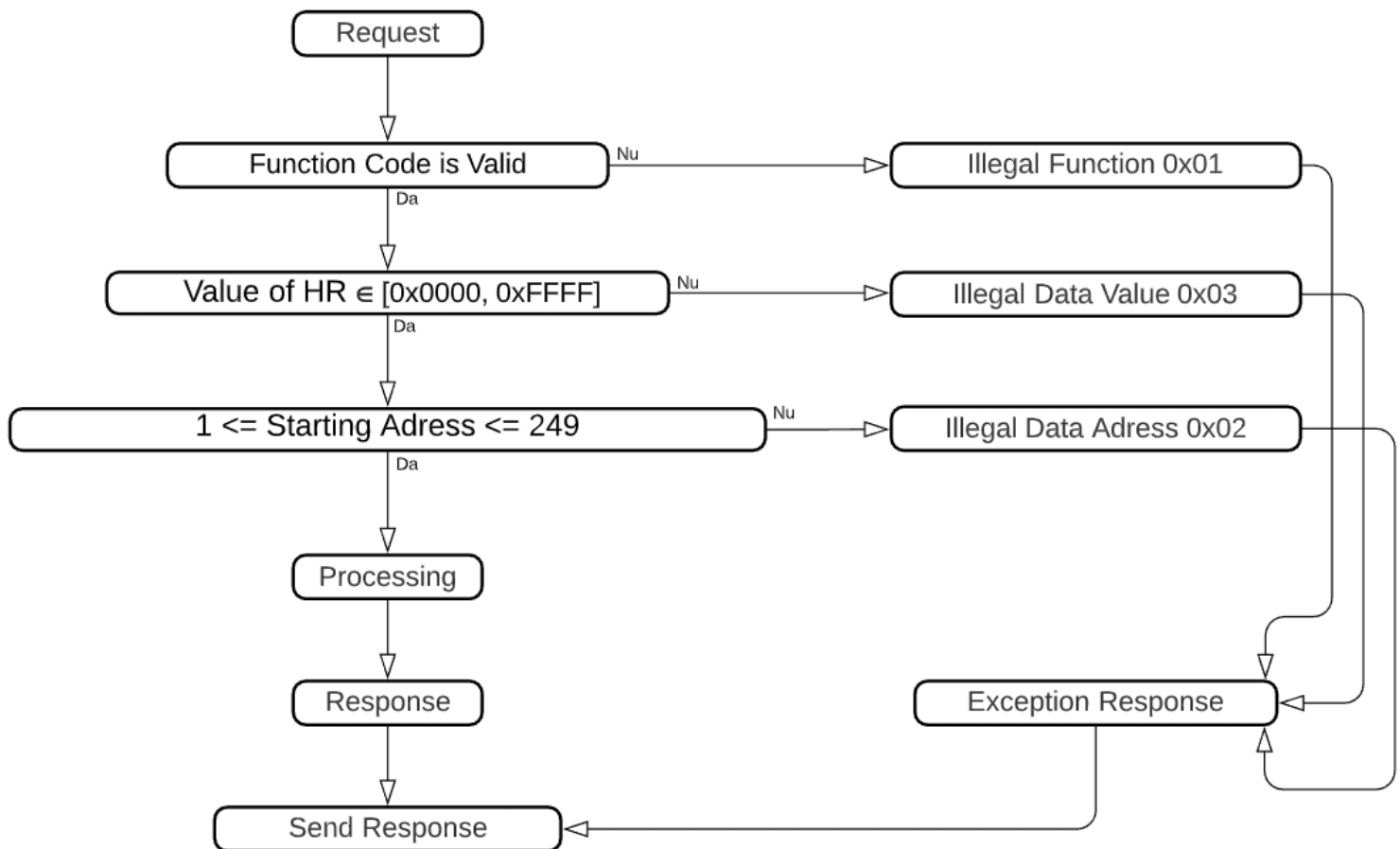
Funcție ce va scrie valoarea unui holding register(1 byte) – HR la o adresă specificată.

-> Structură mesaje:

Request
Function Code
Register Address
Value of HR

Response
Function Code
Register Address
Value of HR

->Diagrama de funcționare:



•Force Multiple Coils(0x0F)

->Rol:

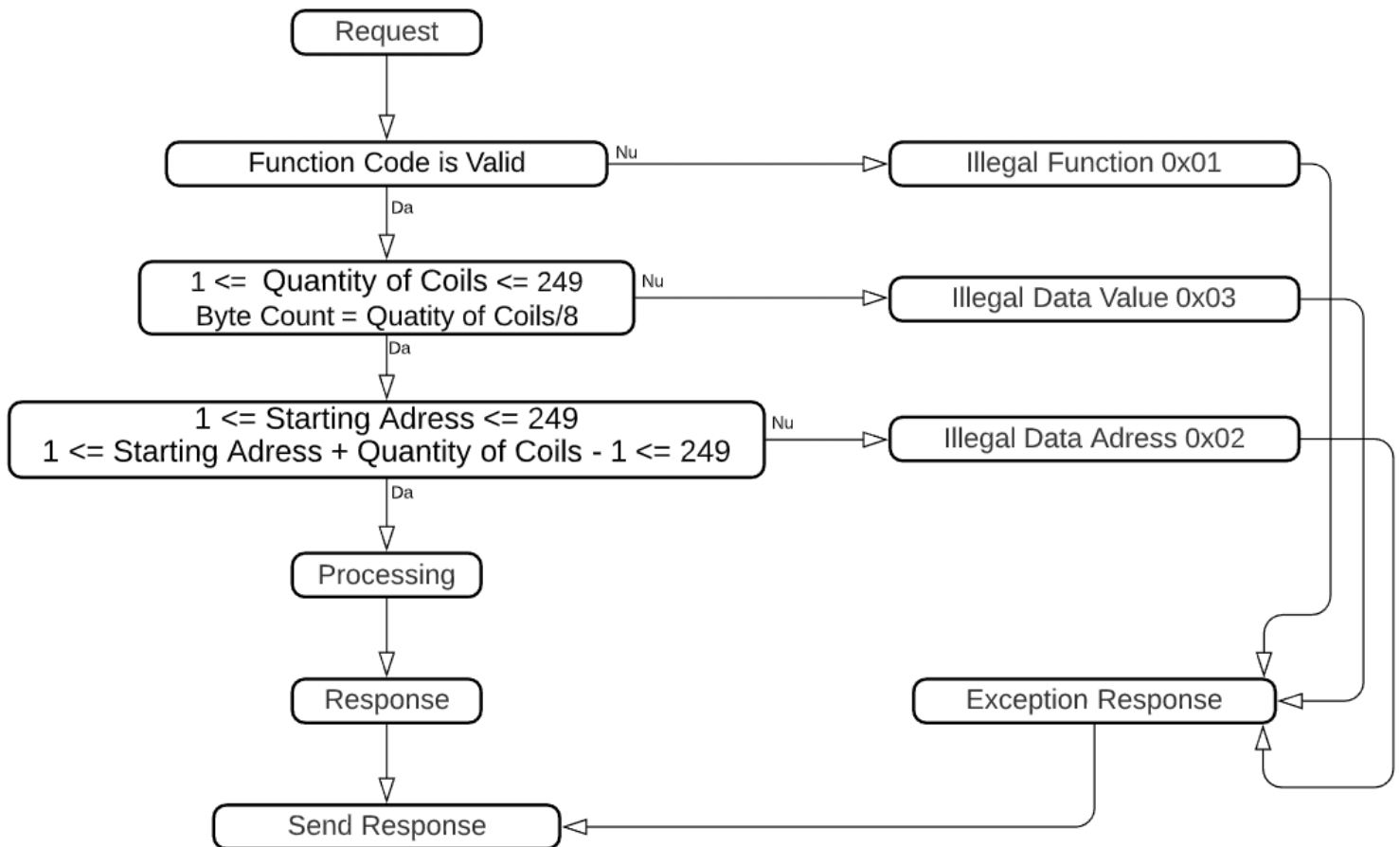
Funcție ce va scrie valoarea fiecarui coil(1 bit) de la o adresă specificată dintr-o secvență de bytes dată.

-> Structură mesaje:

Request
Function Code
Starting Address
Quantity of Coils
Byte Count
Status of Coils

Response
Function Code
Starting Address
Quantity of Coils

->Diagrama de funcționare:



•Write Multiple Registers(0x10)

->Rol:

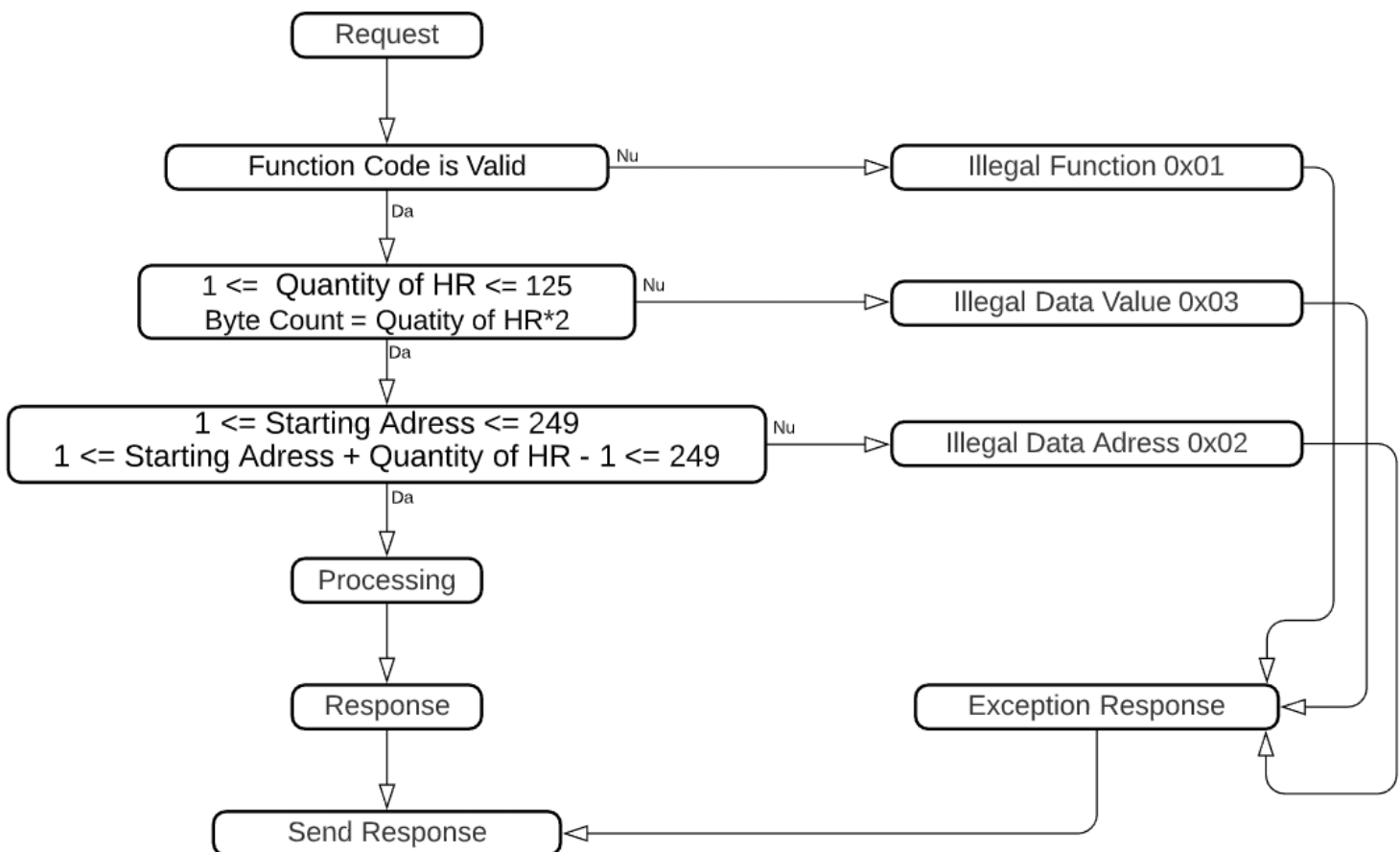
Funcție ce va scrie valoarea fiecarui holding register(1 byte) – HR de la o adresă specificată dintr-o secvență de bytes dată.

-> Structură mesaje:

Request
Function Code
Starting Adress
Quantity of HR
Byte Count
Value of HR

Response
Function Code
Starting Adress
Quantity of HR

->Diagrama de funcționare:



4.Excepții:

- **Illegal Function**(0x01)

Se pot apela doar funcțiile 1, 2, 3, 4, 5, 6, 15, 16.

- **Illegal Data Adress**(0x02)

Este descrisă în diagrama de funcționare a fiecărei funcții.

- **Illegal Data Value**(0x03)

Este descrisă în diagrama de funcționare a fiecărei funcții.

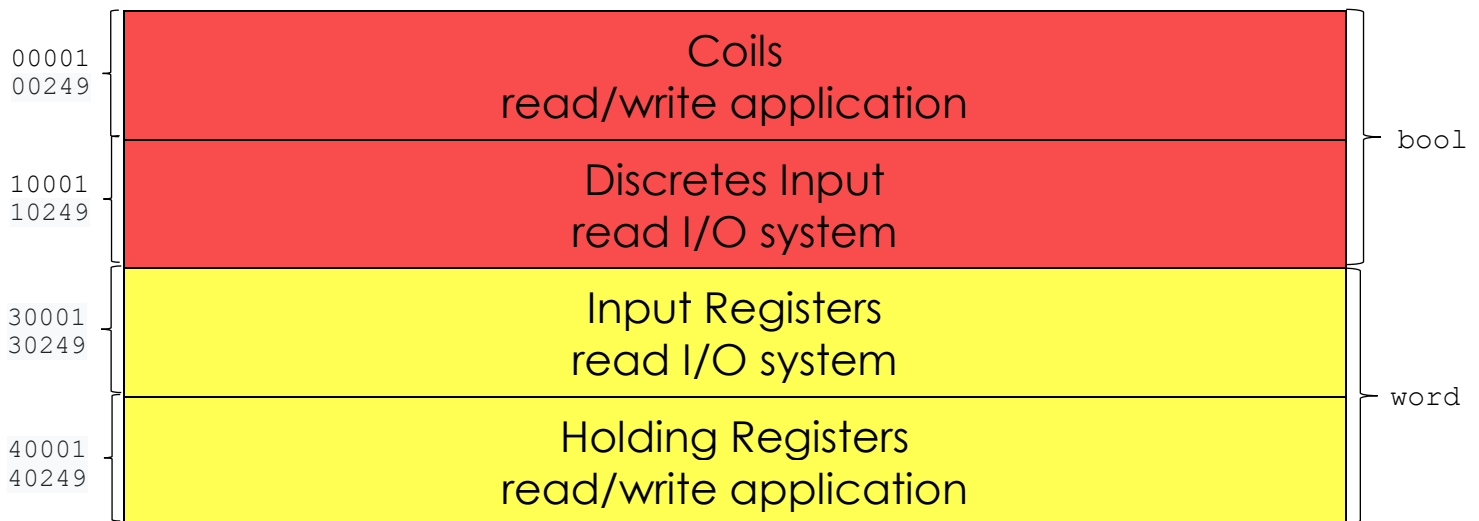
- **Server Failure**(0x04)

Va fi activat din interfață.

- **Server Busy**(0x06)

Va fi activat din interfață.

Maparea memoriei



În Discretes Input de la adresa 10001 se va specifica starea unui program adăugat din interfață (0 – închis, 1 - deschis) în ordinea în care au fost adăugate.

În Input Registers la adresa 30001 se va afla procentul de CPU utilizat, la adresa 30002 totalul de memorie RAM, la adresa 30003 procentul de memorie RAM utilizată, de la adresele 30004 și 30005 se vor stoca perechile (memorie totală parte, procent de memorie utilizată parte), la adresa 30099 se va stoca numărul de conturi create, iar de la adresa 30100 se vor stoca conturile (IP-urile și parolele asociate).

Toate informațiile sunt stocate într-o bază de date MySQL.

Interfața aplicației

A fost creată cu ajutorul librăriei python Tkinter.

Prezintă un panou în care se va activa una dintre excepții Server Failure sau Server Busy, un panou în care se poate crea conturi print-un IP și o parolă, un panou care afișează procentul de utilizare a procesorului și a memoriei RAM precum și o intrare pentru a adauga programe care vor avea activitatea urmărită și un panou care va afișa comunicația dintre aplicație și clienții care sunt conectați.

The screenshot shows a window titled "Client ModbusTCP" with a dark background. The interface is divided into several sections:

- For The Next Request:** Two buttons, "Server Failure" (light blue) and "Server Busy" (pink), are displayed side-by-side.
- Create An Account:** This section contains two input fields: "Enter The IP" and "Enter The Password". Below these fields is a red "Create" button.
- Resources:** This section displays two boxes: "CPU Usage" showing "1.7%" and "RAM Usage" showing "32.6%". Below these is an input field labeled "Enter A Process" and another red "Create" button.
- Listening Area:** A large rectangular area at the bottom, outlined in gray, with the text "[LISTENING]" in the top-left corner.