UNESP – UNIVERSIDADE ESTADUAL PAULISTA

CAMPUS DE SOROCABA

***ENGENHARIA DE CONTROLE E AUTOMAÇÃO***

# ***Disciplina: Redes Industriais de Comunicação***

# *Prática 7 – Comunicação entre Servidor OPC UA ESP32 e Cliente (App e PC)*

Orientador: Prof. Eduardo Paciência Godoy
Aluno: Daniel Augusto Carneiro de Souza

Versão 1

Sorocaba

2021

## Introdução

Esta prática tem como finalidade demonstrar as funcionalidades do padrão OPC UA através de um servidor OPC UA desenvolvido no ESP32, bem como realizar conexões com diversos dispositivos através do mecanismo de transporte de dados TCP/IP do OPC UA.

## Material necessário

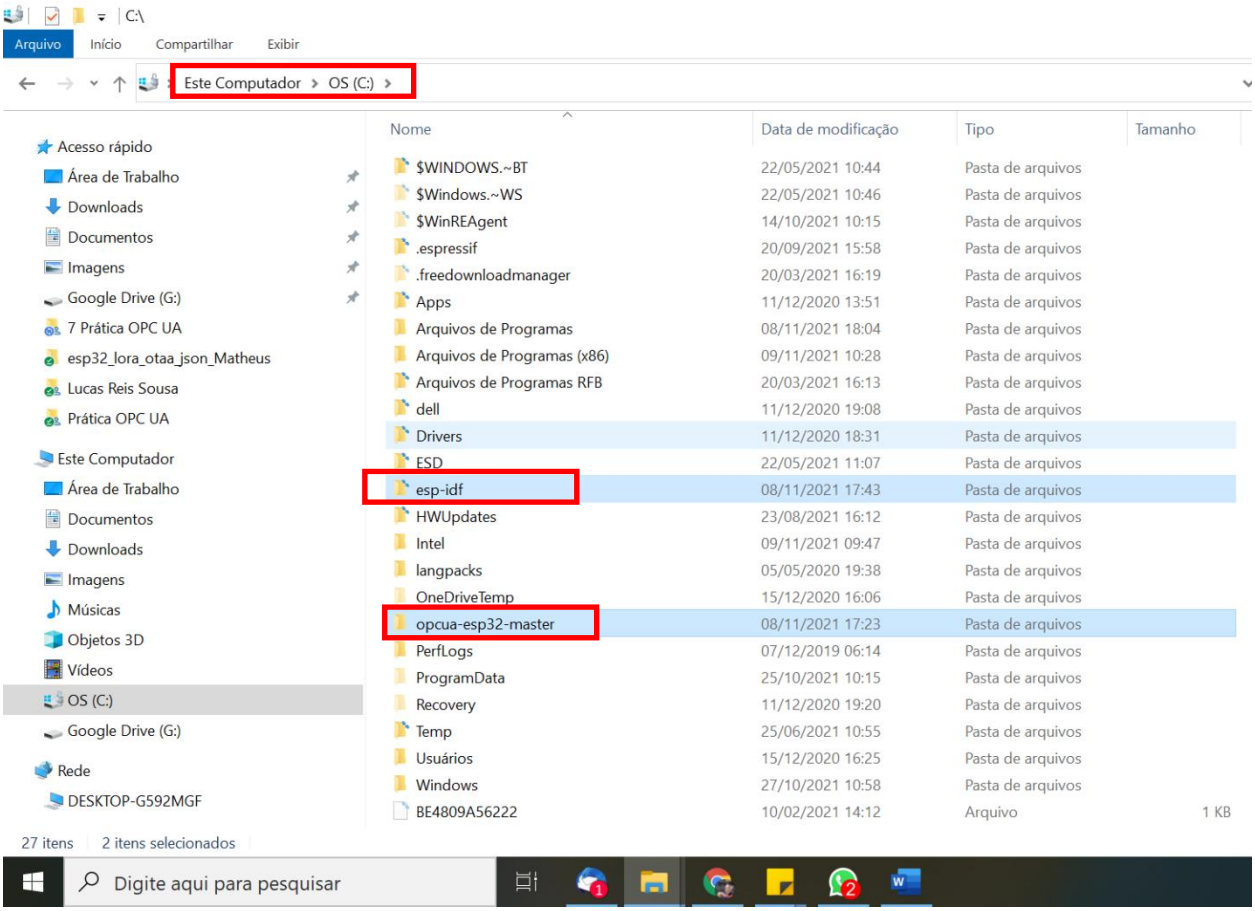
- UAEperxt (Unified Automation) (instalador disponível na pasta de arquivos da Prática): https://www.unified-automation.com/downloads/opc-ua-clients.html;
- Aplicativo Prosys OPC UA Client (versão para Android: https://play.google.com/store/apps/details?id=com.prosysopc.ua.android2&hl=en_US&gl=US )

## 1. Configuração do Servidor OPC UA do ESP32

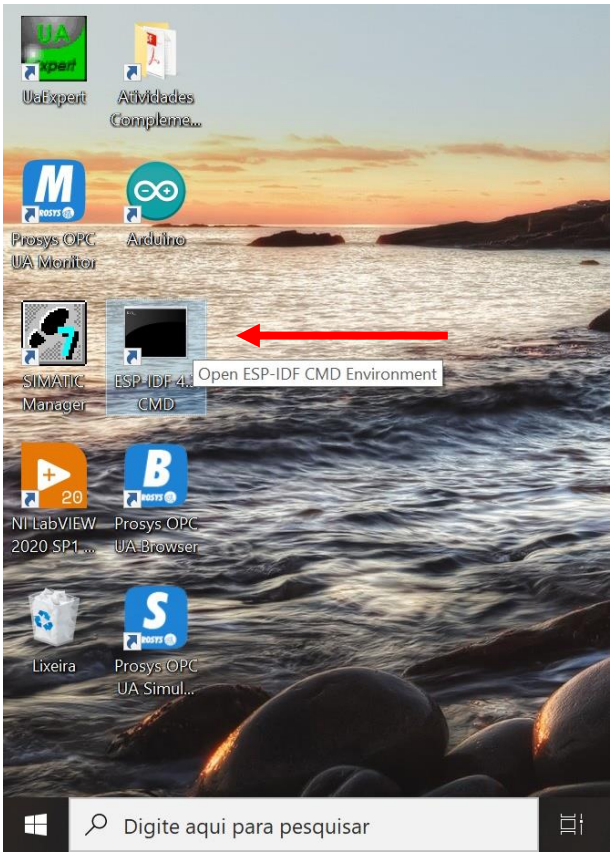
- Para criar o servidor, usaremos um código desenvolvido pelo professor e o aluno Daniel Souza em seu trabalho de TG. O código está pronto para ser compilado e gravado na ESP32, porém foi desenvolvido com o ambiente de desenvolvimento da Espressif (fabricante do ESP32), chamado de ESP-IDF;

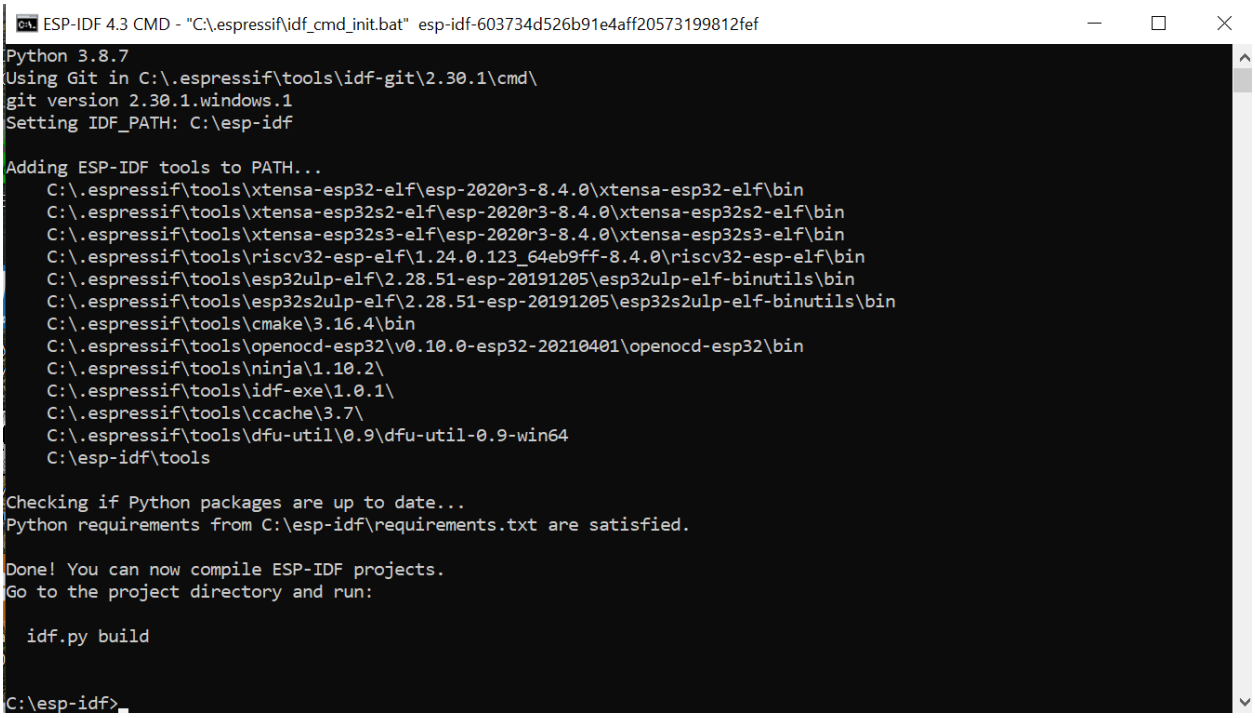
- Para a instalação desse ambiente ESP-IDF, acessar procedimento detalhado no link: https://youtu.be/N5BnTShl_RY;
- Instalar arquivo **“esp-idf-tools-setup-2.3.exe”** disponível na pasta compartilhada da prática;

- Após a instalação, mover a pasta **“esp-idf”** criada na área de trabalho do seu computador para a raiz C: do seu computador **(C:\esp-idf)**;
- Extrair a pasta compactada **“opc-esp32-master.rar”** disponível na pasta compartilhada da prática também para a raiz C: do seu computador **(C:\opcua-esp32-master)**;

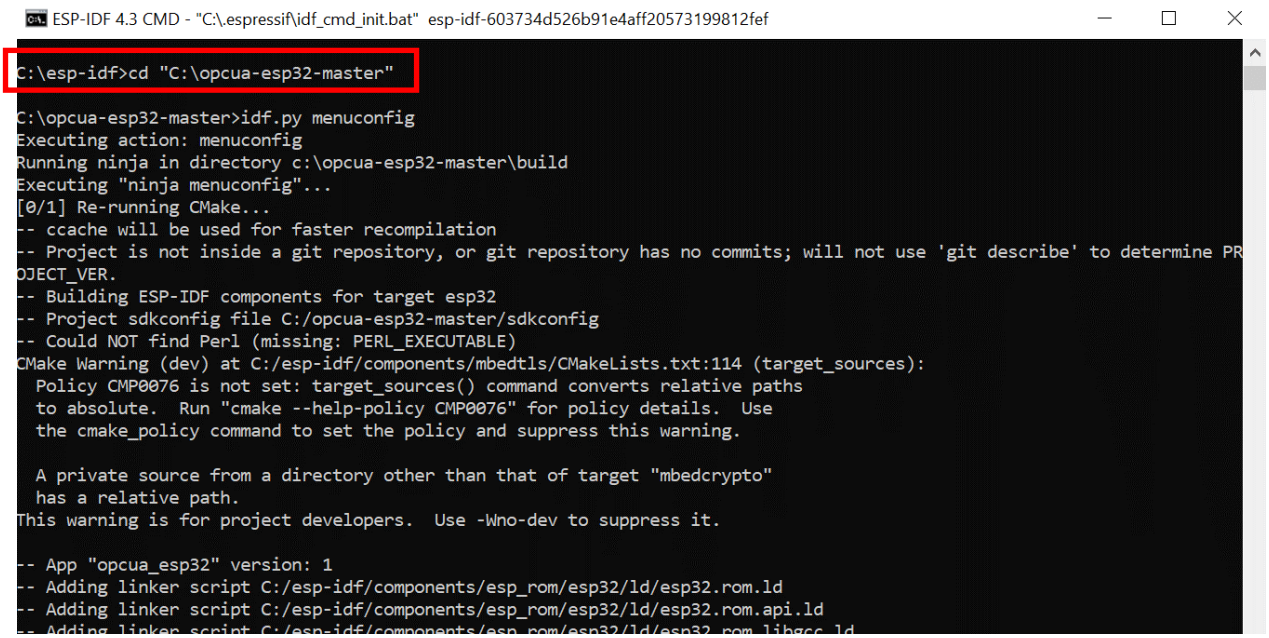



- Abrir ESP-IDF usando ícone criado na área de trabalho do seu computador

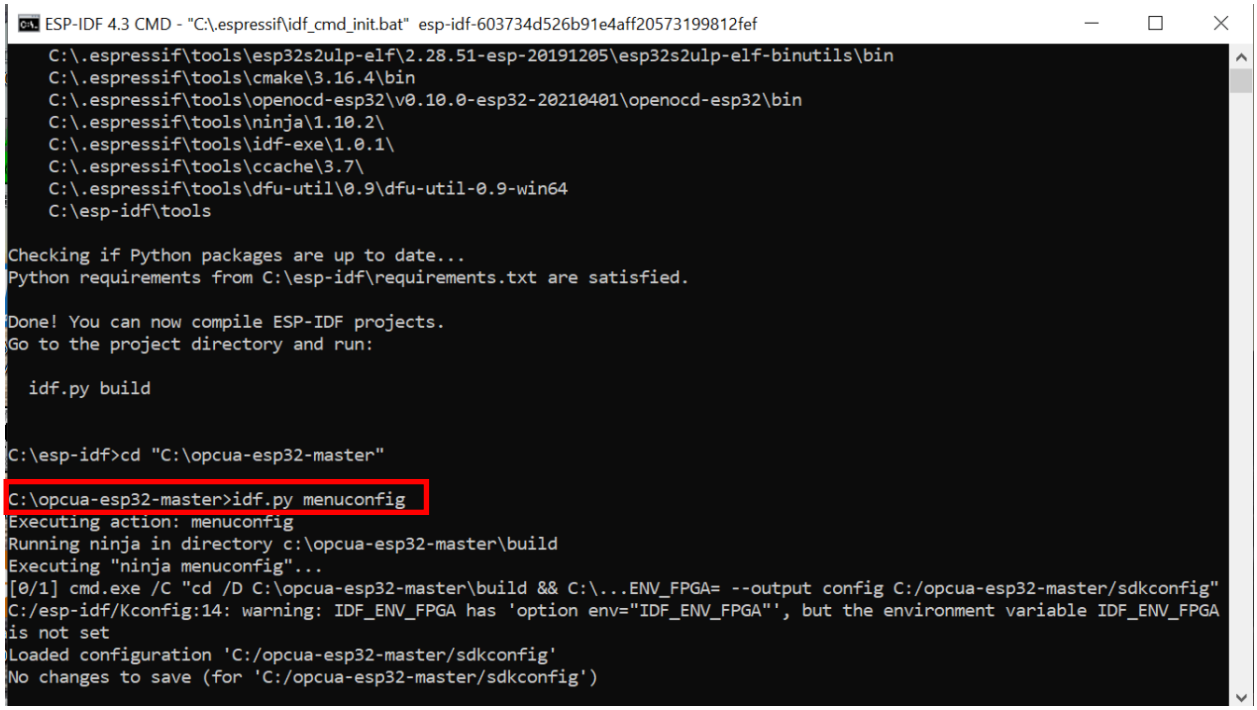




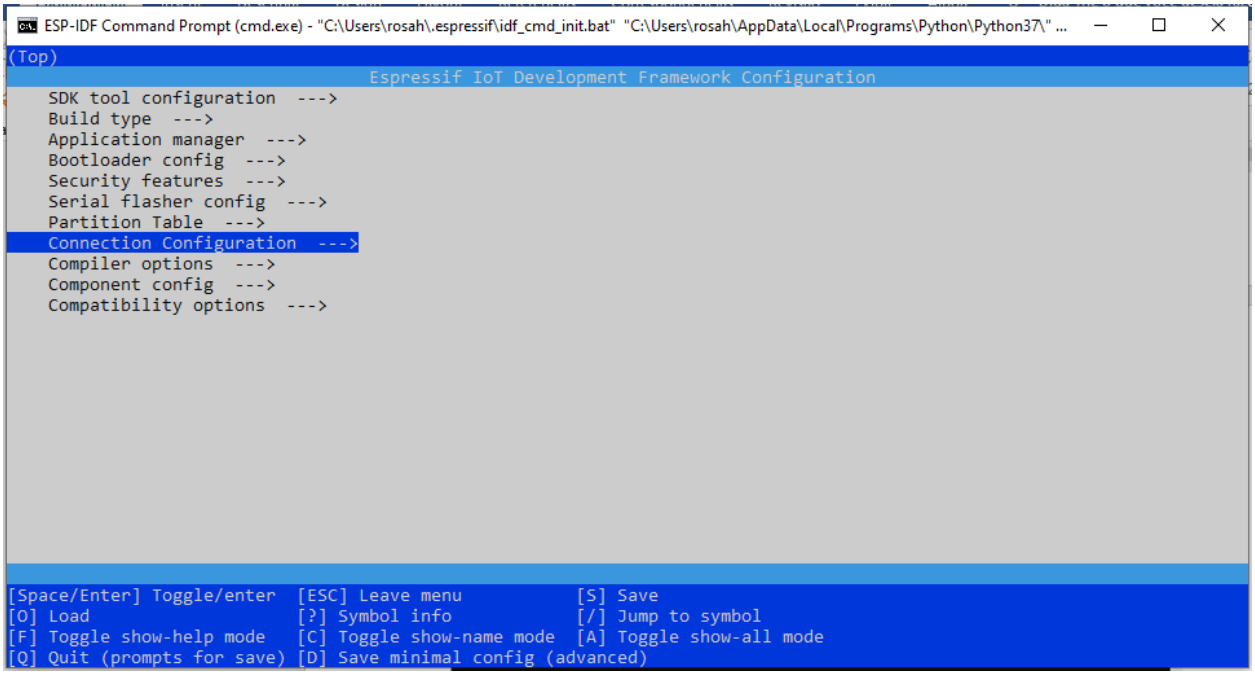

- Inserir comando para acessar pasta do arquivo: **cd “endereço do arquivo”**. No nosso caso digitar: **cd “C:\opcua-esp32-master”**



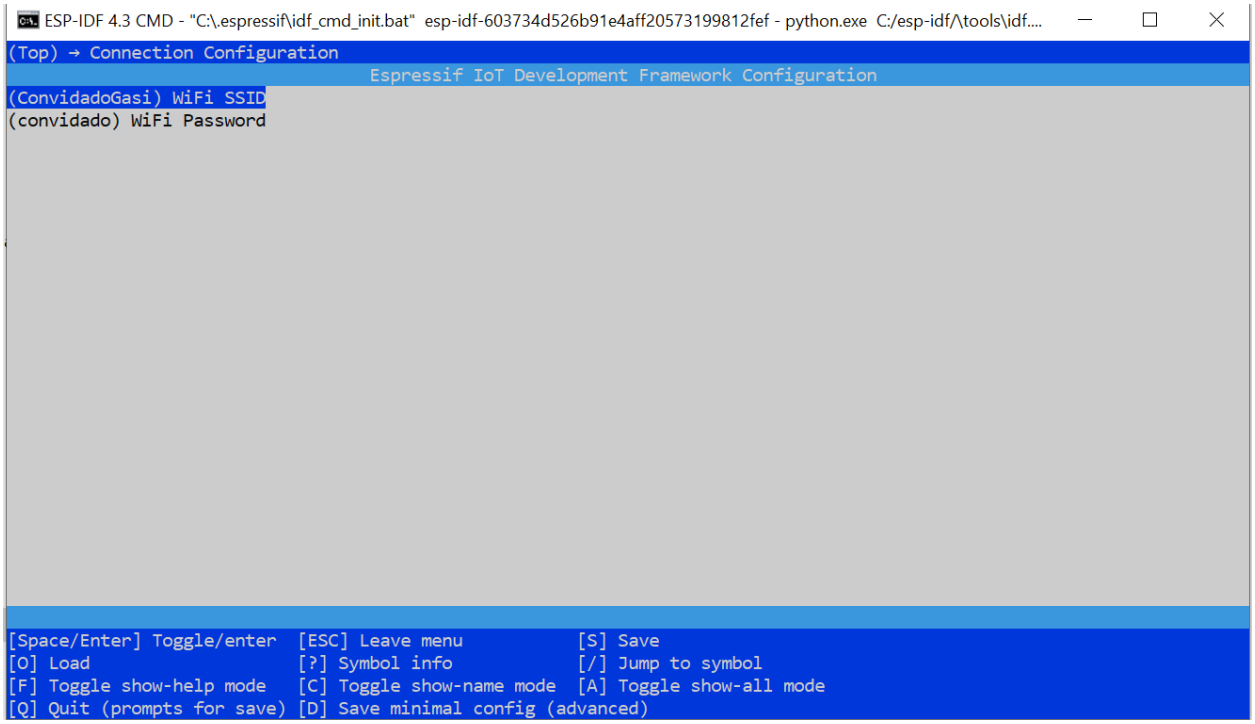

- Acessar menu de configuração do ESP32 na ESP=IDF com o comando: **idf.py menuconfig**

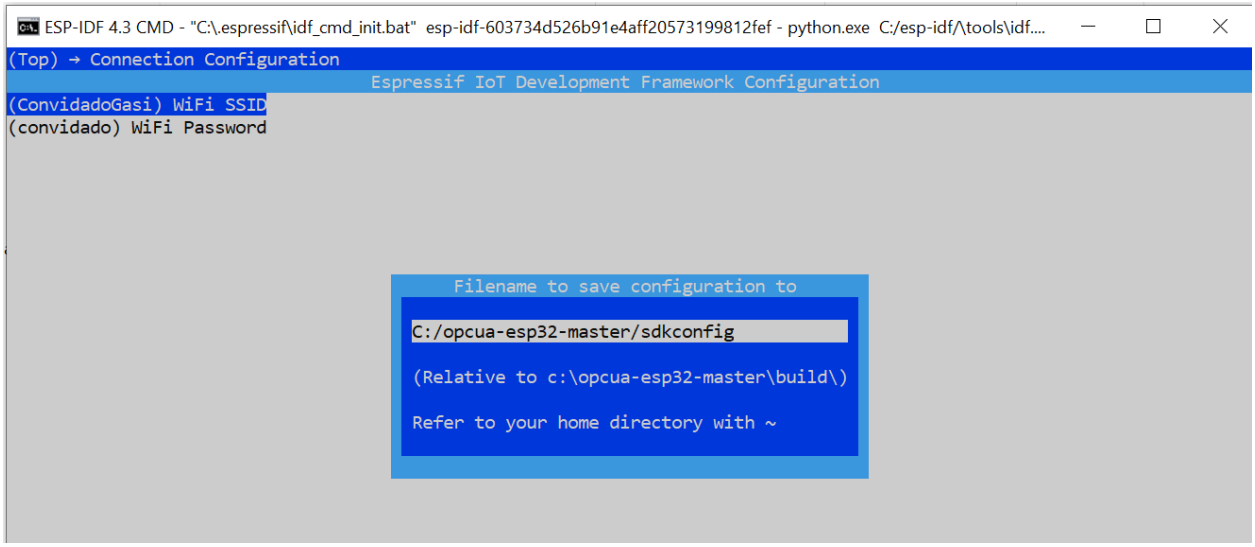


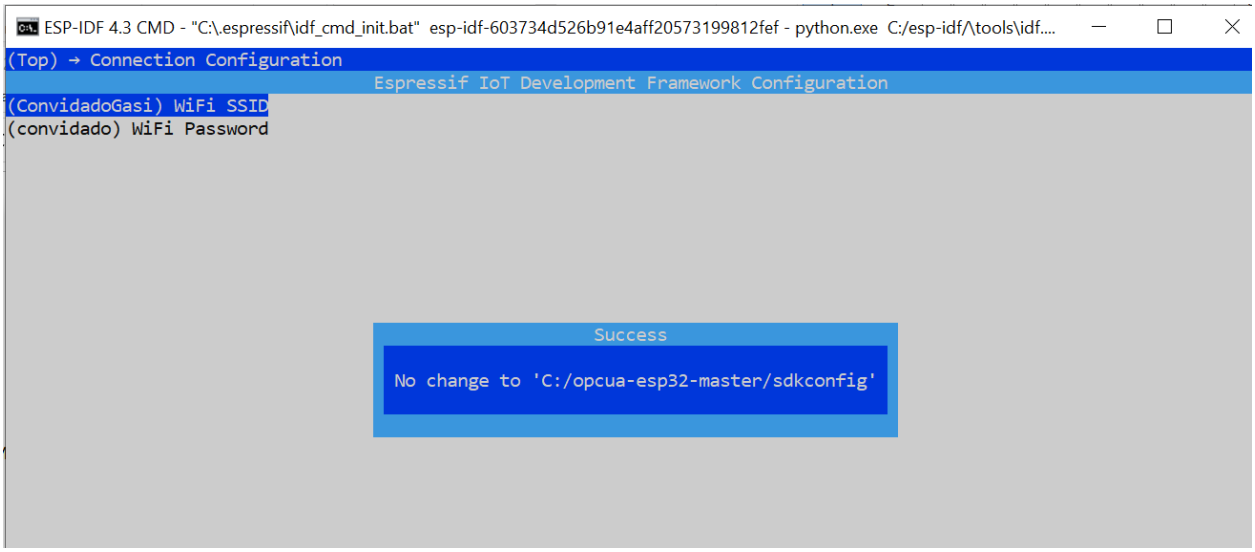

- Com a tela de configuração aberta, acessar "*Connection Configuration*" para conectar-se a sua rede Wi-FI (atualmente o ESP só suporta redes 2,4 GHz);

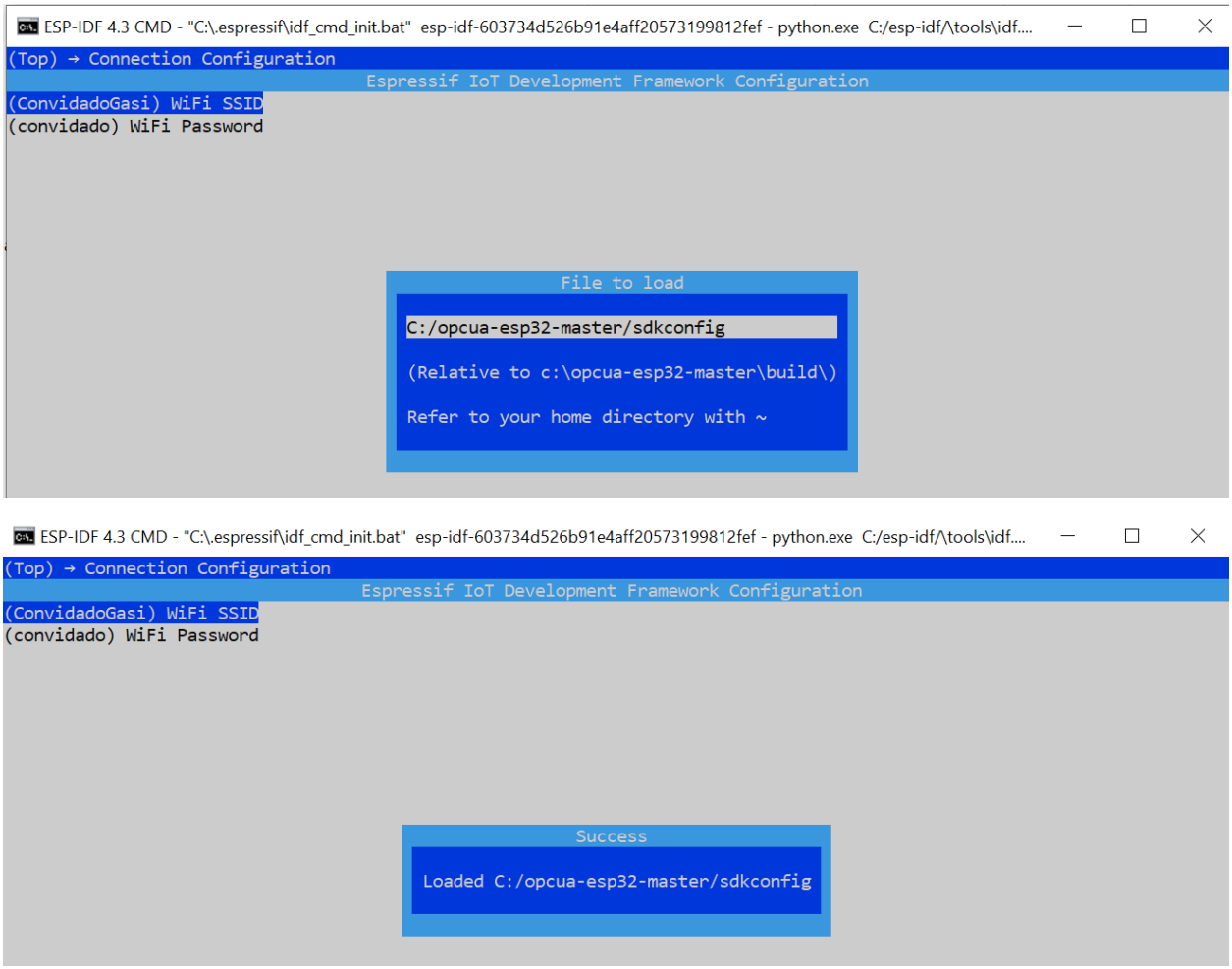



- Digite nome e senha da sua rede. Depois salve e carregue no ESP;

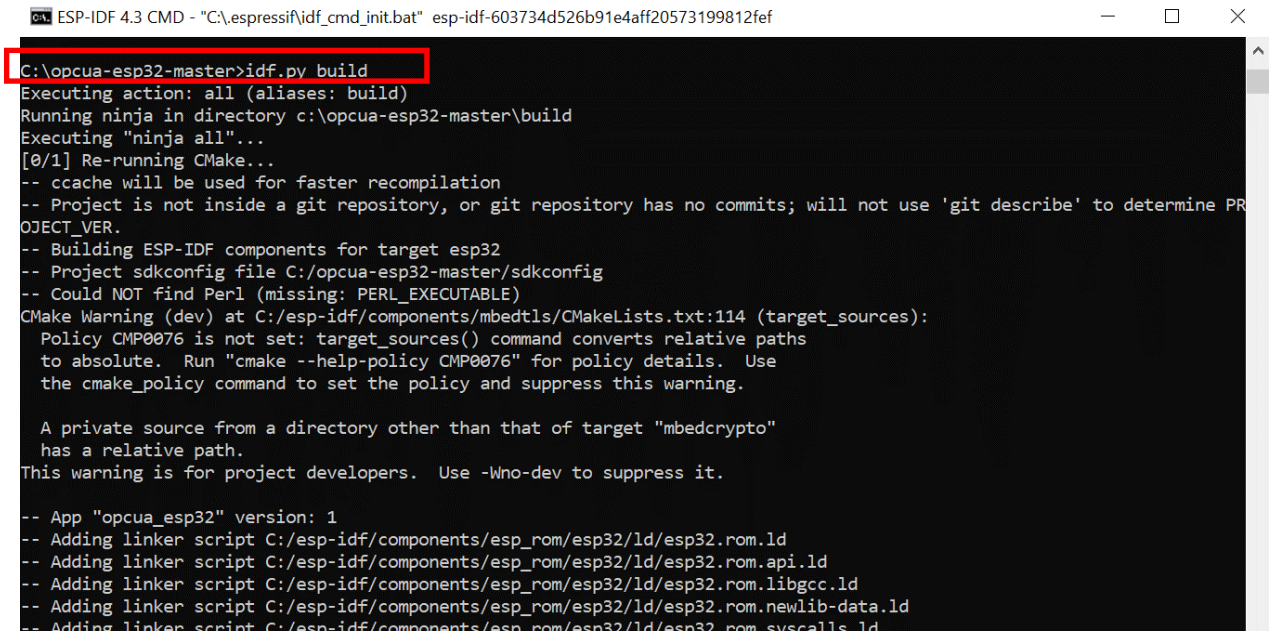

ESP-IDF 4.3 CMD - "C:\.espressif\idf_cmd_init.bat" esp-idf-603734d526b91e4aff20573199812fef - python.exe C:/esp-idf/\tools\idf....
(Top) → Connection Configuration
Espressif IoT Development Framework Configuration
(ConvidadoGasi) WiFi SSID
(convidado) WiFi Password
[Space/Enter] Toggle/enter [ESC] Leave menu [S] Save
[O] Load [?] Symbol info [/] Jump to symbol
[F] Toggle show-help mode [C] Toggle show-name mode [A] Toggle show-all mode
[Q] Quit (prompts for save) [D] Save minimal config (advanced)


ESP-IDF 4.3 CMD - "C:\.espressif\idf_cmd_init.bat" esp-idf-603734d526b91e4aff20573199812fef - python.exe C:/esp-idf/\tools\idf....
(Top) → Connection Configuration
Espressif IoT Development Framework Configuration
(ConvidadoGasi) WiFi SSID
(convidado) WiFi Password
Filename to save configuration to
C:/opcua-esp32-master/sdkconfig
(Relative to c:\opcua-esp32-master\build\)
Refer to your home directory with ~

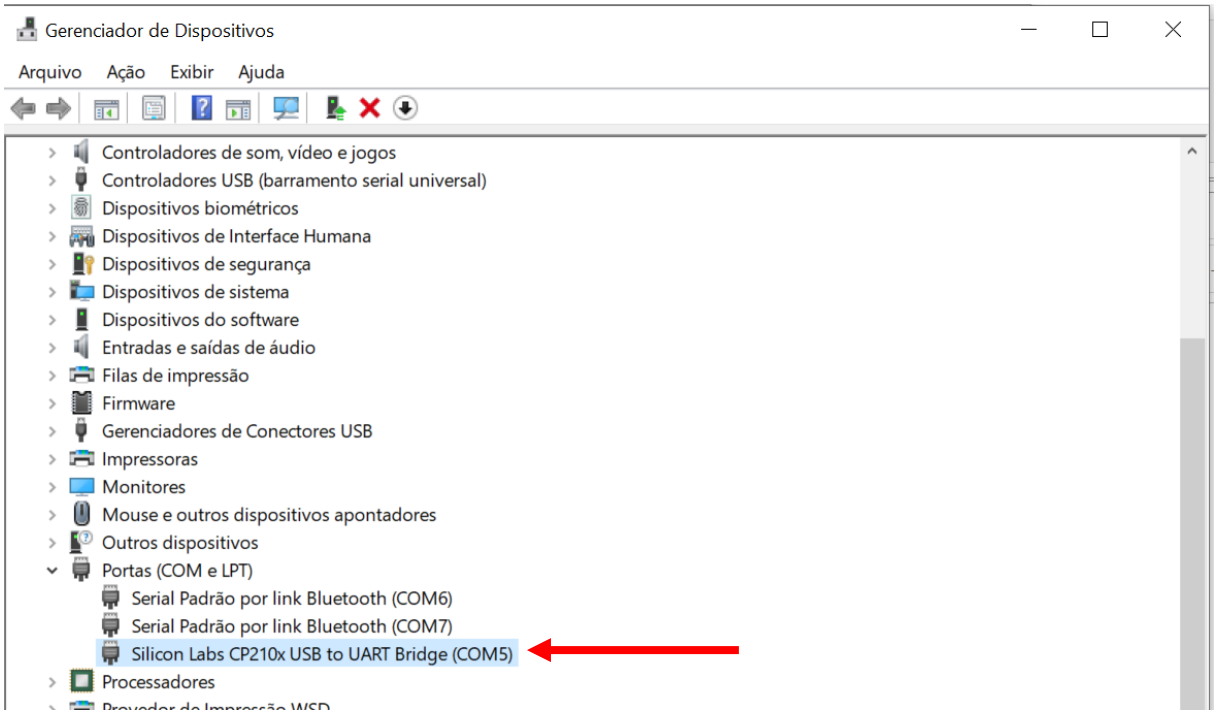

ESP-IDF 4.3 CMD - "C:\.espressif\idf_cmd_init.bat" esp-idf-603734d526b91e4aff20573199812fef - python.exe C:/esp-idf/\tools\idf....
(Top) → Connection Configuration
Espressif IoT Development Framework Configuration
(ConvidadoGasi) WiFi SSID
(convidado) WiFi Password
Success
No change to 'C:/opcua-esp32-master/sdkconfig'

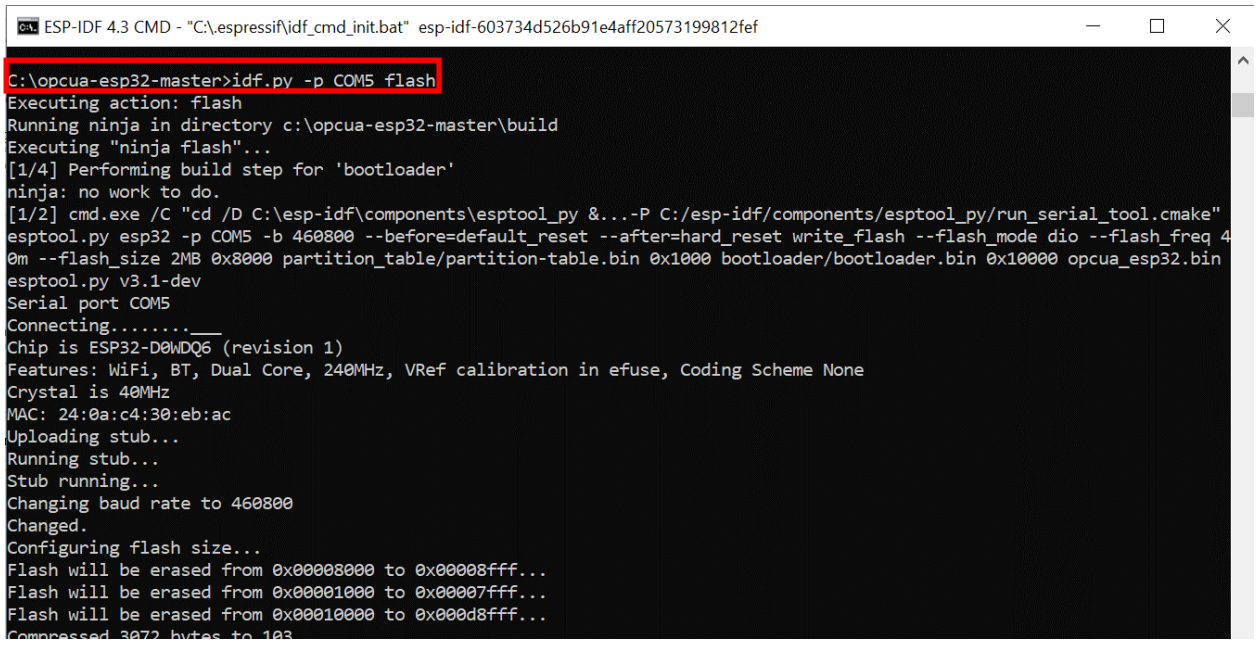



- Com as configurações realizadas, aperte “ESC” até sair do menu.
- Para compilar o programa, digite o comando: **idf.py build**

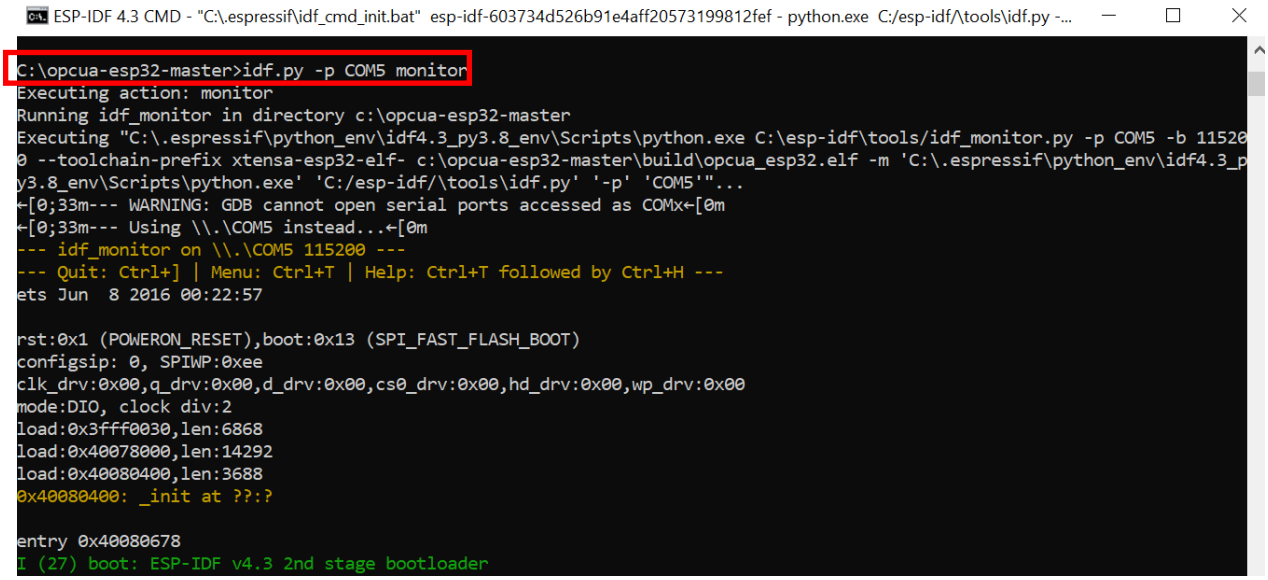


- Após compilação, para gravação do código compilado na ESP32, digite o comando: **idf.py -p COMX flash.** A porta COMX representa sua porta COM criada após a conexão do ESP32 na USB no computador. Acesse o Gerenciador de Dispositivos do seu Windows para verificar qual a porta COM.

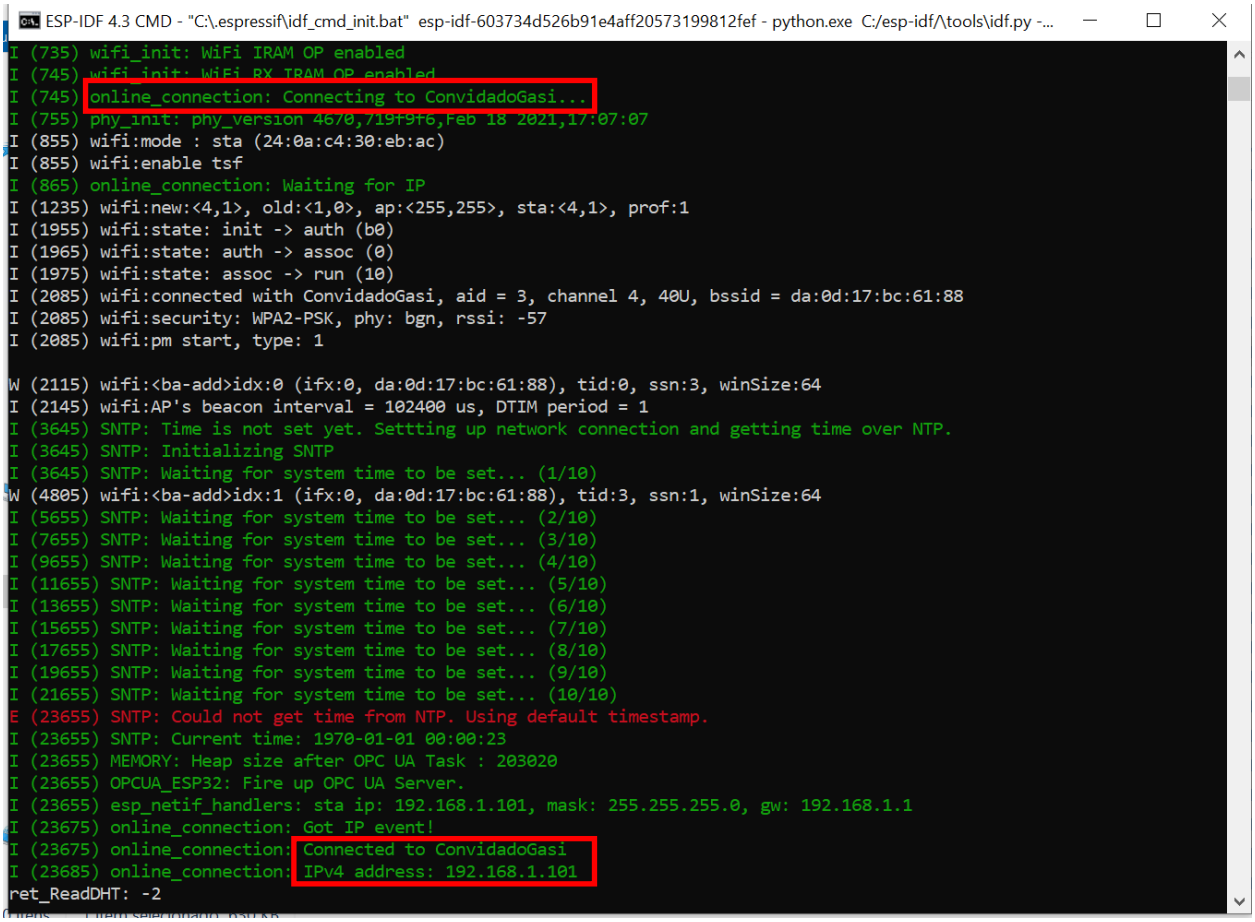



- Após a programação do ESP32, para verificar a operação, digite o comando: **idf.py -p COMX monitor.** A porta COMX representa sua porta COM criada após a conexão do ESP32 na USB no computador. Caso falhe a conexão automática entre o IDF e a ESP32, tente realizar o comando novamente e apertar o botão BOOT/PRG da ESP32 se aparecer a mensagem "*Conecting*..." na tela;

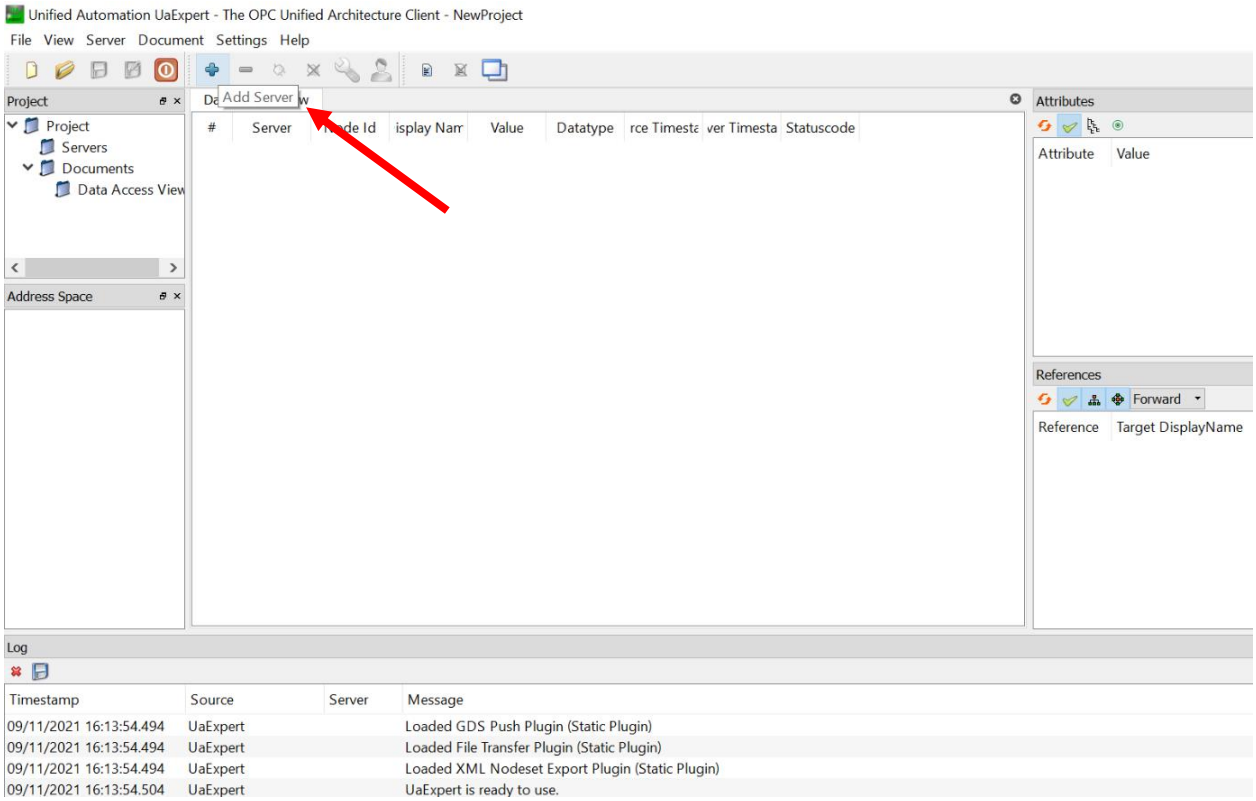



- Com a comunicação estabelecida, diversas informações são apresentadas, entre elas o endereço IP do servidor OPC UA da ESP32 que deverá ser usado para conexão:

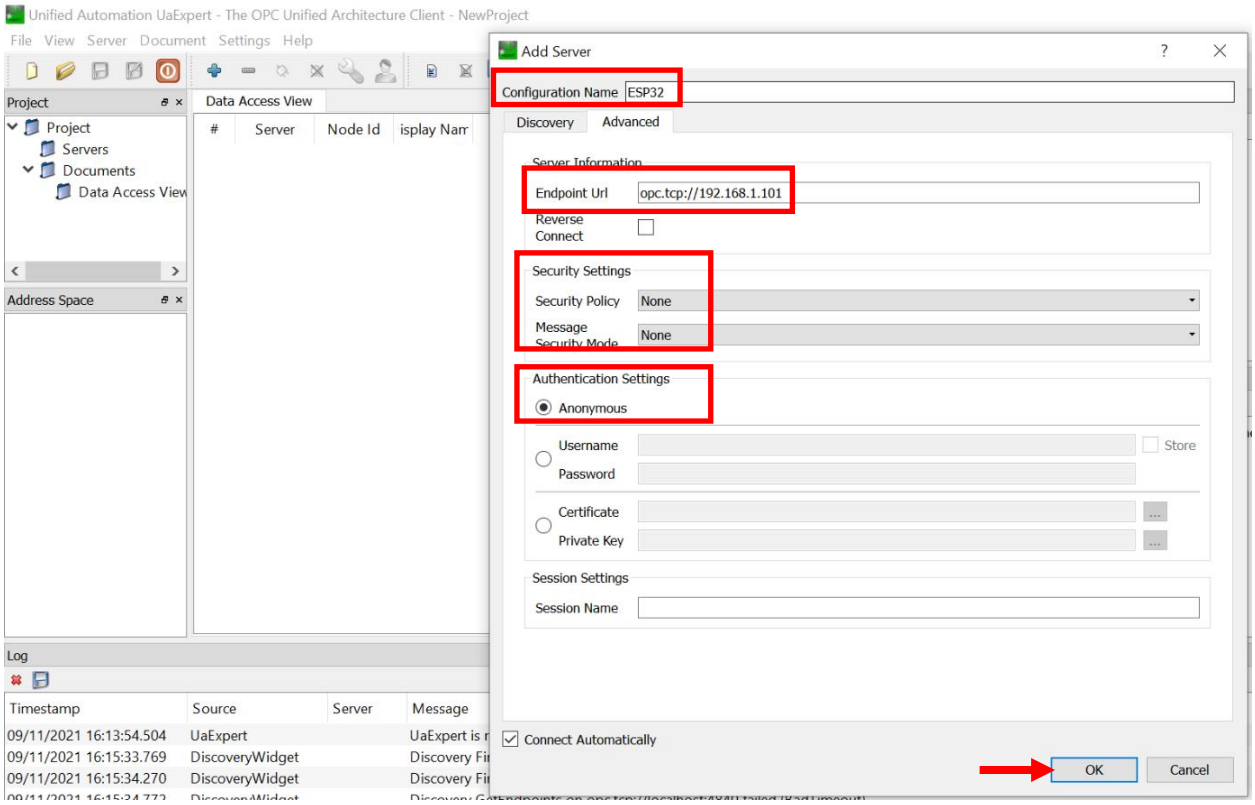



## 2. Configuração do Cliente OPC UA

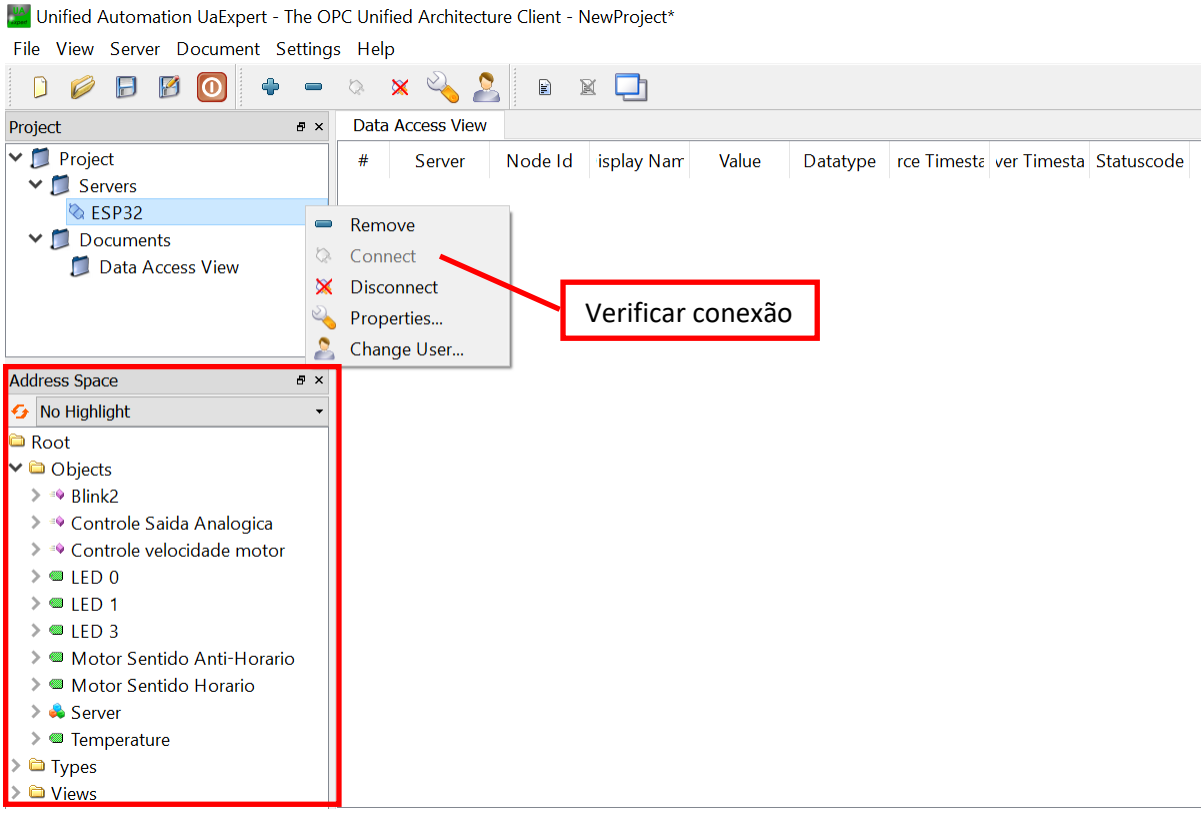
- Abra o aplicativo UAExpert e clique no ícone “+” para adicionar um servidor OPC UA;



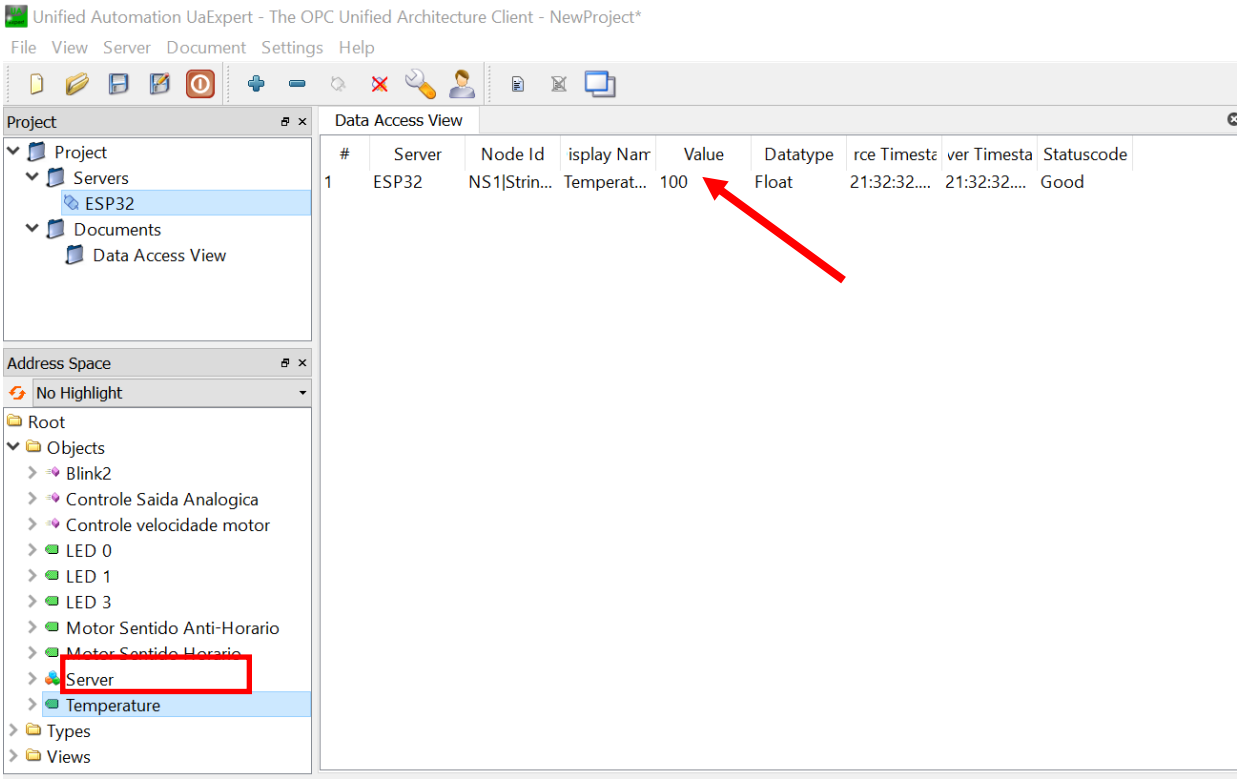

- Selecione a aba "*Advanced*" de configuração e digite os dados do servidor OPC UA do ESP32. Clique em OK para confirmar.



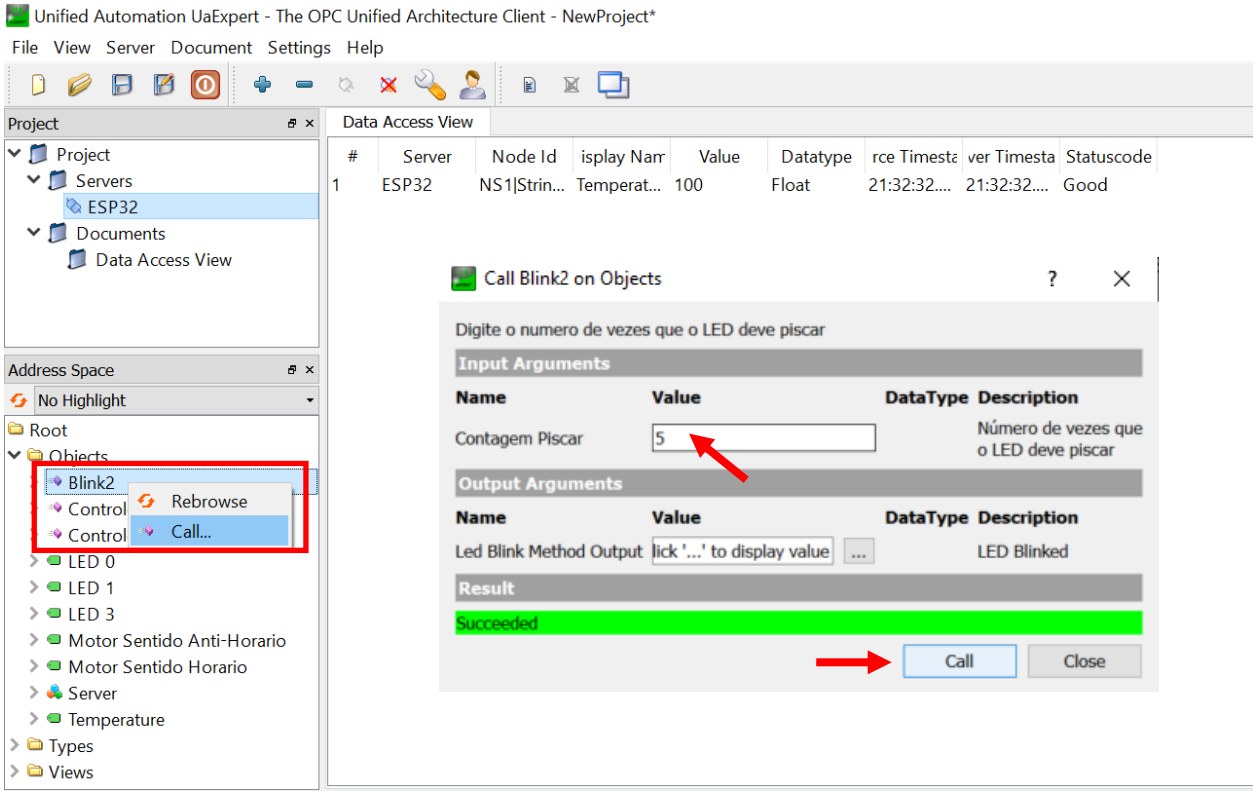
- O Cliente UAExpert deve se conectar ao servidor OPC UA do ESP32. Visualize as informações do Espaço de Endereço (variáveis Tag) do servidor OPC UA:




- Para monitorar (fazer a leitura de uma entrada) de uma Tag, clique sobre a variável na área do Espeço de Endereço e puxe a variável para a área do “Data Access View”. Por exemplo, o valor da *Temperature* pode ser visualizado no campo “*Value*”;



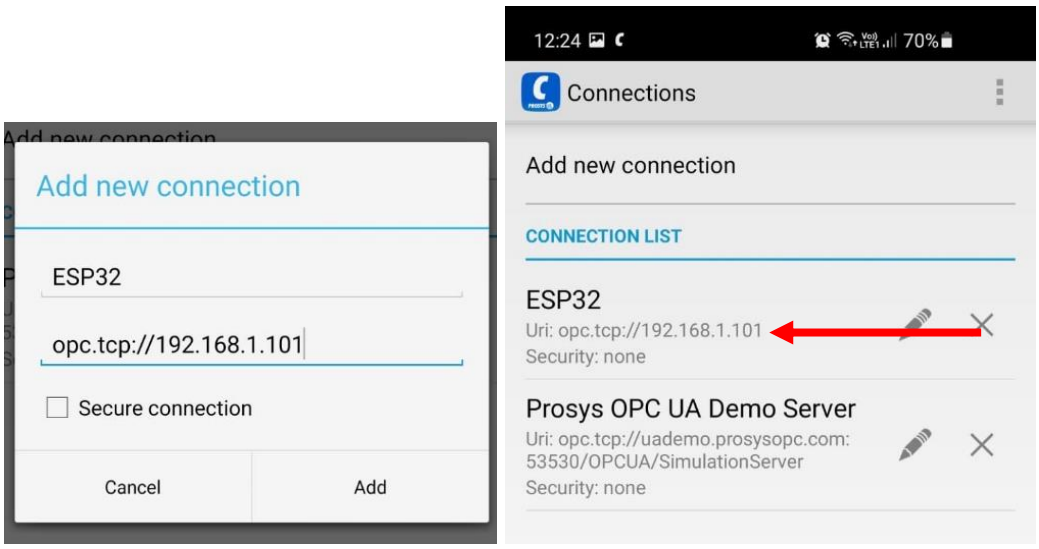
- Para escrever (atualizar uma saída) de uma Tag, clique sobre o nome da variável no Espaço de Endereço e selecione “Call”. Por exemplo, para escrever um valor no *Tag* “Blink2” para comandar a piscagem (acender/apagar) do LED interno do ESP32, selecione “Call” nessa variável. Na caixa de diálogo digite um novo valor (campo Contagem Piscar) e clique em “Call”:

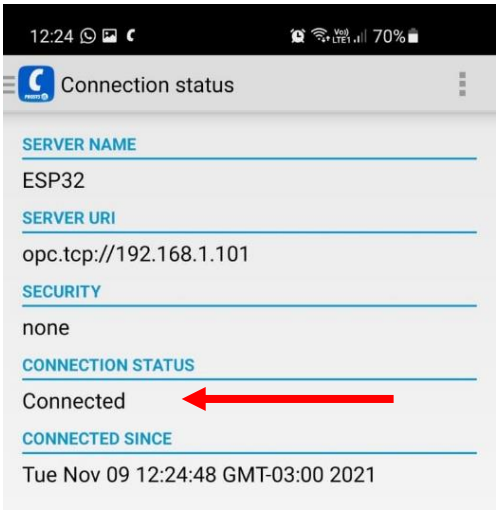



Após o comando, verifique que o LED interno do ESP32 pisca conforme comandado.

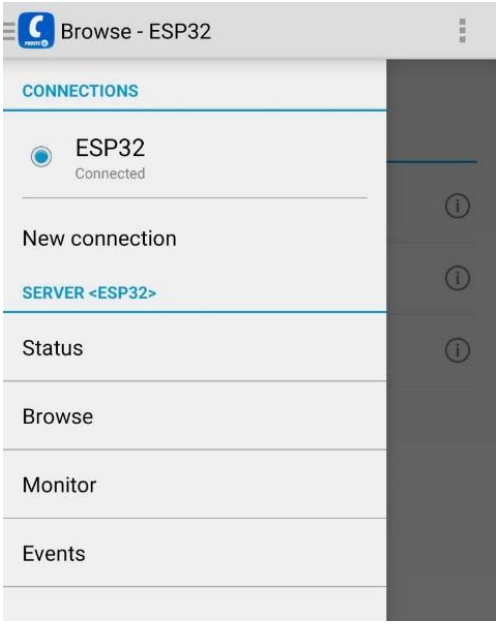
## 3. Aplicativo Cliente OPC UA para celular

- No celular, abra o aplicativo Prosys OPC UA Client e selecione a opção “*Add new connection*”;
- Digite o endereço do servidor (o mesmo utilizado no UAExpert)
- Verifique a conexão entre o Cliente e Servidor;

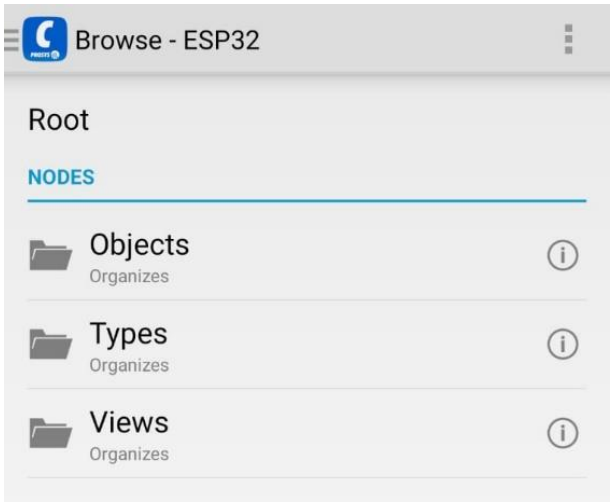


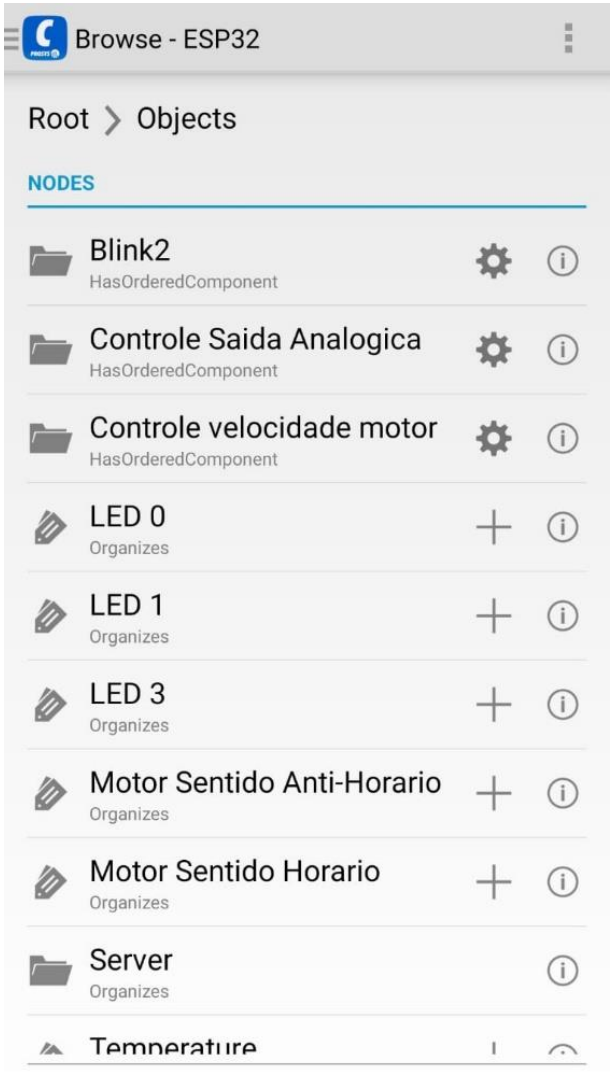




- Clique sobre o botão "*Connection Status*" no canto superior esquerdo e em seguida selecione a opção "*Browse*":

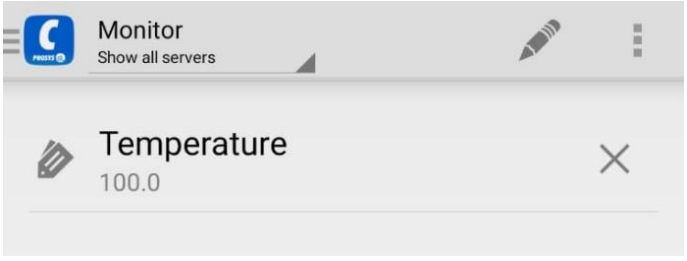


- Acesse as pastas *Objects* para ver os *Tags* disponíveis:

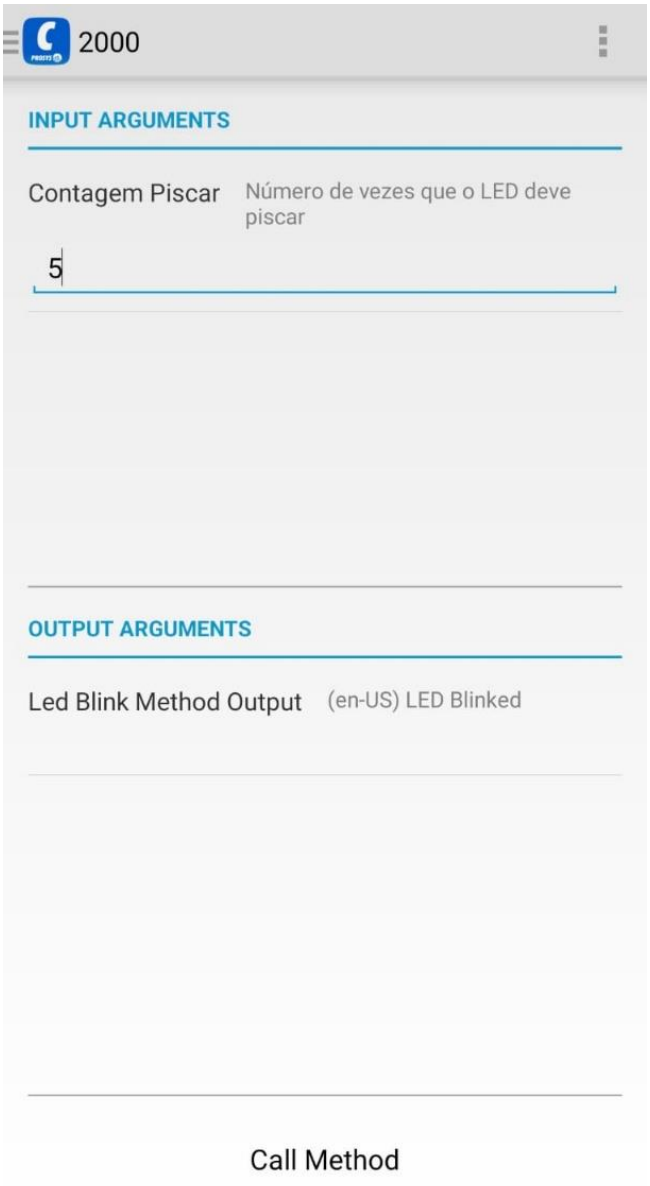



- Para monitorar (fazer a leitura de uma entrada) de uma Tag, clique no sinal "+" ao lado da variável para habilitar o monitoramento. Por exemplo, clique ao lado da variável *Temperature*;
- Abra novamente o menu (canto superior esquerdo) e clique em "*Monitor*" e observe o valor do *Tag* "Temperature";



- Para escrever (atualizar uma saída) de uma Tag, clique sobre o ícone da Engrenagem de uma das variáveis de saída. Por exemplo, para escrever um valor no *Tag* “Blink2” para comandar a piscagem (acender/apagar) do LED interno do ESP32, dê um clique sobre a engrenagem ao lado do nome do *Tag* e digite o novo valor na caixa de diálogo e clique em “Call Method”:




Após o comando, verifique que o LED interno do ESP32 pisca conforme comandado.