TÀI LIỆU HƯỚNG DẪN SỬ DỤNG

BỘ ĐIỀU KHIỂN

AI-CO688

Địa chỉ : 69 Nguyễn Thị Rành, xã Tân An Hội, Tp. Hồ Chí Minh **Hotline**
Email : info@aitech.com.vn **0949 491 355**
Mã số thuế: 0311008280

MỤC LỤC

1. Thông tin về thiết bị..... 1

1.1 Ứng dụng trong nhà máy – tự động hóa công nghiệp 1

1.2 Ứng dụng trong BMS – Tòa nhà thông minh..... 1

1.3 Ứng dụng trong giao thông – hạ tầng kỹ thuật..... 2

1.4 Ứng dụng trong IoT công nghiệp & giám sát từ xa 2

1.5 Thông số kỹ thuật..... 3

2. Lưu ý quan trọng..... 5

2.1 Đối với giao thức truyền thông RS232 5

2.2 Đối với RS485 (thiết bị hổ trợ Half duplex – truyền nhận 2 dây) 6

2.3 Thuật toán RSA..... 6

3. Thông tin cấu hình ban đầu của thiết bị..... 6

4. Cấu hình cơ bản của thiết bị..... 9

4.1 Cấu hình port 9

4.2 Cấu hình TCP/IP các kết nối 10

4.3 Bảo mật kết nối socket (Secure socket) 15

5. Chức năng điều khiển DI/DO từ máy tính.....16

5.1 Điều khiển và giám sát DI/DO theo giao thức AI COMMAND 17

5.1.1 Giao thức khung lệnh TCP (Frame Structure) 18

5.1.2 Điều khiển đóng (ON) Output 18

5.1.3 Điều khiển hở (OFF) Output 19

5.1.4 Điều khiển Output xuất xung 300 ms (điều khiển: Barrier ...) 19

5.1.5 Phản hồi từ thiết bị 19

5.2 Điều khiển và giám sát DI/DO theo giao thức Modbus TCP 20

5.3 Các kết nối khác. 20

6. Phần mềm hỗ trợ kèm theo 20

6.1 Phầm mềm tìm thông tin thiết bị 20

6.2 Phần mềm chuyển đổi dữ liệu TCP sang COM ảo VCOM2TCP 21

6.3 Phần mềm tạo cặp cổng COM ảo.....21

6.4 Phần mềm tạo Khóa RSA.....22

7. Source demo kết nối thiết bị. 22

8. Thông tin công ty TNHH Công nghệ A.I 23

1. Thông tin về thiết bị

Thiết bị giao tiếp và điều khiển công nghiệp đa năng được thiết kế nhằm đáp ứng nhu cầu kết nối, giám sát và điều khiển tập trung trong các hệ thống hiện đại. Với khả năng tích hợp RS485/RS232, Relay Input/Output và Ethernet TCP/IP trong một thiết bị nhỏ gọn, sản phẩm giúp đơn giản hóa kiến trúc hệ thống, nâng cao độ ổn định và khả năng mở rộng cho các ứng dụng công nghiệp và IoT.

- Tích hợp đa giao tiếp – đa tín hiệu trong một thiết bị duy nhất.
- Giảm chi phí phần cứng, đơn giản hóa hệ thống.
- Dễ mở rộng, dễ tích hợp với hệ thống hiện có.
- Vận hành ổn định, phù hợp cho hệ thống 24/7.

1.1 Ứng dụng trong nhà máy – tự động hóa công nghiệp

Thiết bị đóng vai trò gateway thu thập và điều khiển tín hiệu hiện trường, kết nối trực tiếp các thiết bị sử dụng RS485/RS232 như PLC, biến tần, đồng hồ đo điện, cảm biến công nghiệp... về hệ thống giám sát trung tâm qua mạng Ethernet TCP/IP.

Với 8 kênh Relay Input và 8 kênh Relay Output, thiết bị cho phép giám sát trạng thái máy móc (ON/OFF, báo lỗi) và điều khiển thiết bị công suất (bơm, quạt, van, đèn tín hiệu). Thiết kế công nghiệp, công suất thấp, vận hành ổn định giúp hệ thống hoạt động liên tục 24/7 trong môi trường nhà xưởng.

1.2 Ứng dụng trong BMS – Tòa nhà thông minh

Trong hệ thống BMS, thiết bị được sử dụng để kết nối và chuẩn hóa dữ liệu từ các hệ thống con như: điều hòa, chiếu sáng, thang máy, báo cháy, cấp thoát nước...

Các cổng RS485/RS232 cho phép tích hợp thiết bị của nhiều hãng khác nhau, trong khi Relay Input/Output giúp điều khiển trực tiếp các mạch chiếu sáng, quạt thông gió, bơm nước. Thông qua Ethernet, dữ liệu được truyền về máy chủ BMS/SCADA, cho phép vận hành, giám sát và điều khiển tập trung từ phòng kỹ thuật.

1.3 Ứng dụng trong giao thông – hạ tầng kỹ thuật

Thiết bị phù hợp triển khai tại tủ điều khiển giao thông, bãi đỗ xe, trạm thu phí, trạm cân, biển báo điện tử...

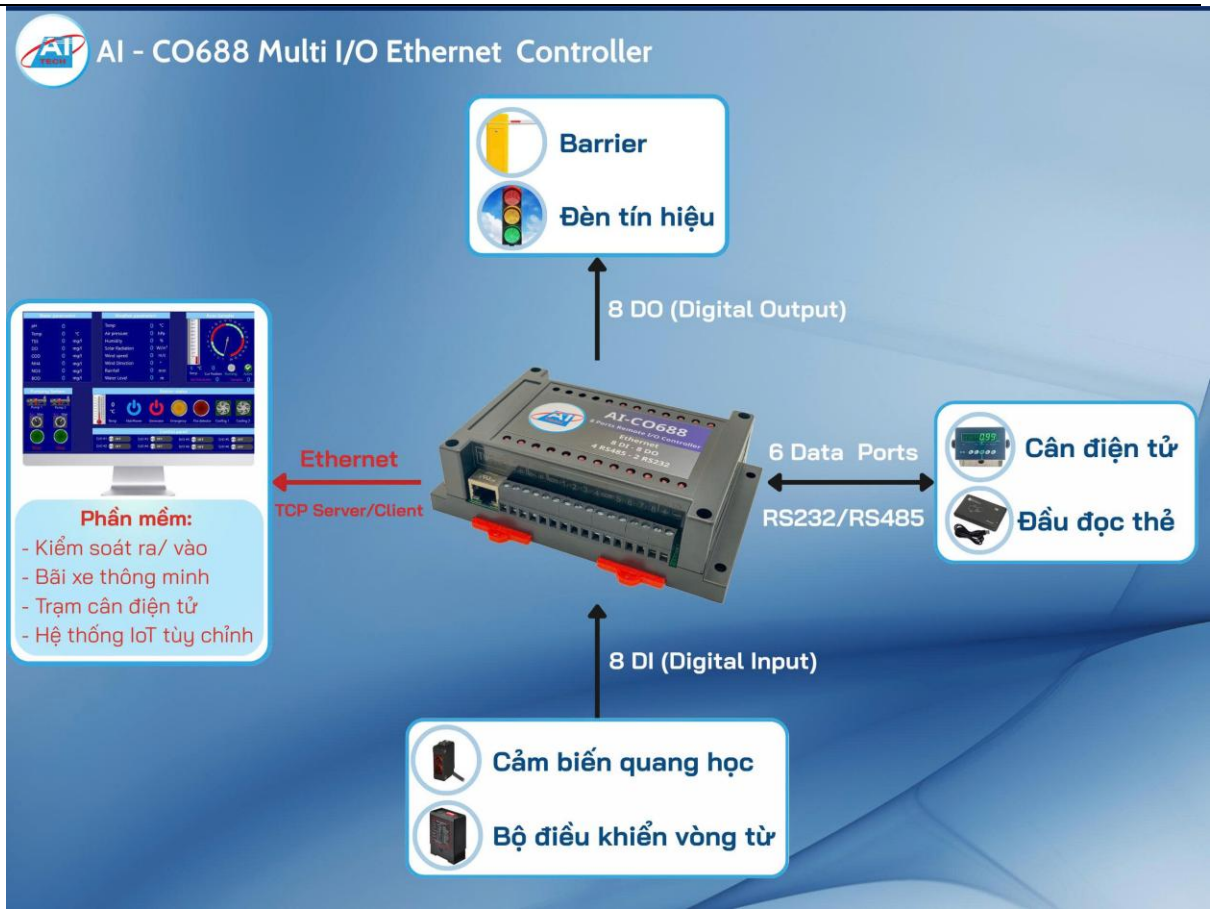
Relay Input dùng để nhận tín hiệu trạng thái (cảm biến xe, barrier, công tắc an toàn), trong khi Relay Output thực hiện đóng/cắt thiết bị điều khiển. Giao tiếp RS485/RS232 giúp kết nối các bộ điều khiển hiện trường, dữ liệu được truyền về trung tâm qua mạng

Ethernet để giám sát và điều hành từ xa, giảm chi phí vận hành và bảo trì.

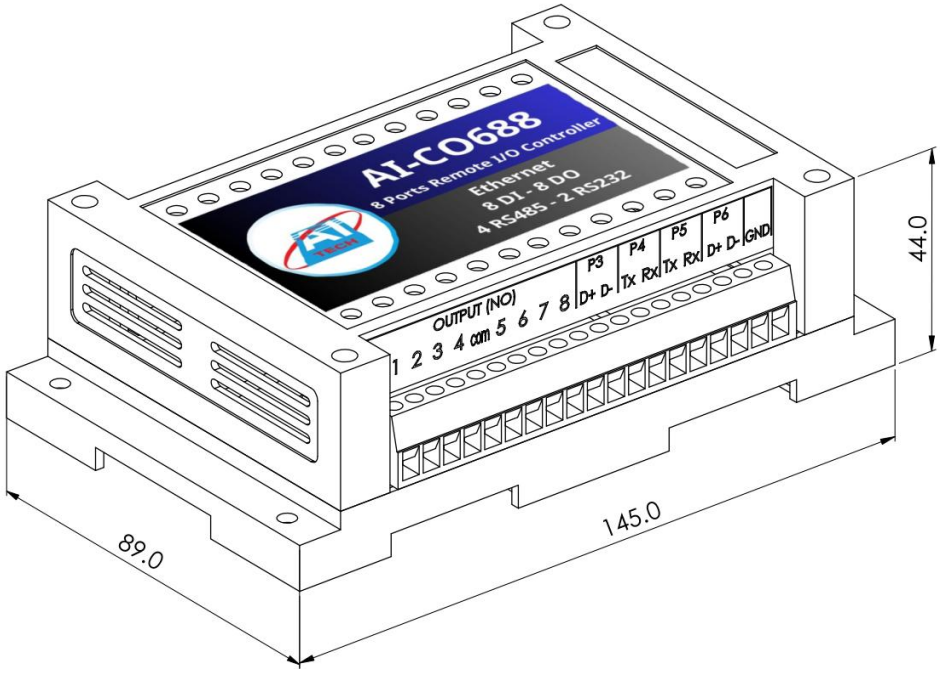
1.4 Ứng dụng trong IoT công nghiệp & giám sát từ xa

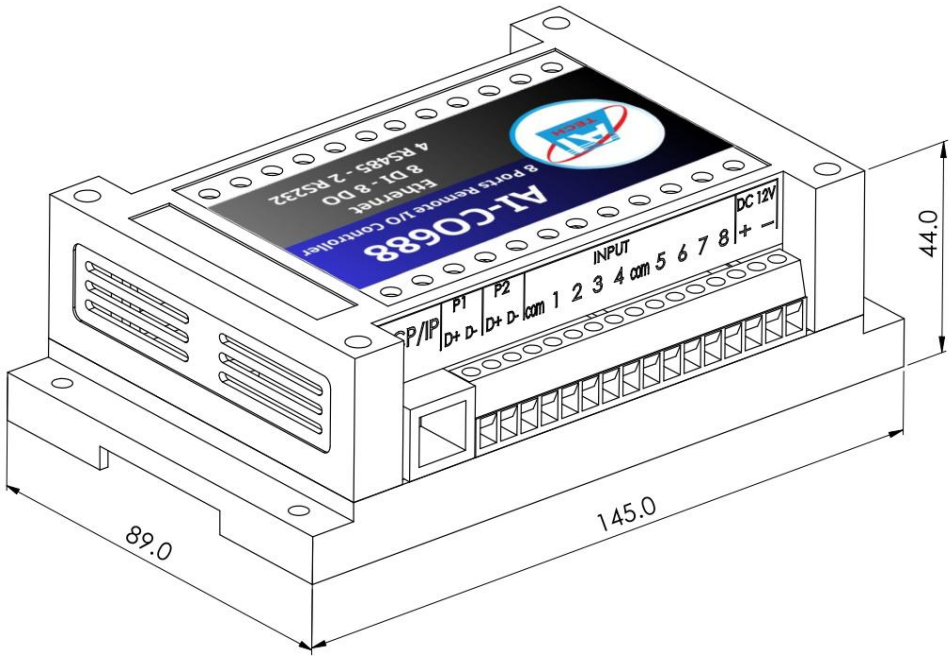
Thiết bị hoạt động như một IoT Edge Gateway, thu thập dữ liệu từ cảm biến và thiết bị hiện trường, sau đó truyền về server hoặc cloud thông qua TCP/IP.

Khả năng kết nối đồng thời nhiều client cho phép tích hợp linh hoạt với phần mềm giám sát, dashboard web, hoặc hệ thống phân tích dữ liệu. Với kích thước nhỏ gọn và tiêu thụ điện năng thấp, thiết bị đặc biệt phù hợp cho các hệ thống IoT phân tán, trạm giám sát không người trực, hoặc các dự án cần triển khai số lượng lớn.



1.5 Thông số kỹ thuật





BẢNG THÔNG SỐ KỸ THUẬT	
ETHERNET	
Số cổng	1
Giao diện	RJ45
Tốc độ	10/100 Mbp
Giao thức	IP, TCP, UDP, DHCP, DNS, HTTP
Bảo mật	Xác thực User Mã hóa và giải mã bằng thuật toán RSA
SERIAL (RS232, RS485)	
Số cổng	- 2 cổng RS232 (Rx, Tx, GND) - 4 cổng RS485 (D+ , D-) (half duplex) - Các cổng cùng lúc hoạt động
Baud rate	600 bps – 115k bps
Data bits	7, 8
Stop bit	1, 2
Check bit	None, Even, Odd

Folow control	None
Digital Input / Digital Output	
Digital Input	8
Digital Output	8 (NO – Thường hở)
Phạm vi điện áp	0-48 v (có thể điều chỉnh theo đầu vào)
PHẦN MỀM	
Giao thức kết nối	Socket TCP hoặc Modbus TCP
Source demo	C#, VB.net, python, Note.js, C++ ...
Cấu hình	Webpage, phần mềm
THÔNG SỐ KHÁC	
Kích thước	Dài x Rộng x Cao: 145 x89 x 44 (mm)
Nhiệt độ hoạt động	0 °C – 70°C
NGUỒN CẤP	
Điện áp	DC 12V
Trạng thái làm việc	< 250mA
Công suất	< 3 w
PHỤ KIỆN	
Bộ nguồn	12V, 1A
Cáp mạng	Độ dài 1m, 2 đầu RJ85

2. Lưu ý quan trọng

2.1 Đối với giao thức truyền thông RS232

Độ Dài Cáp: Độ dài cáp RS232 thường bị giới hạn ở khoảng 15 mét (khoảng 50 feet), tuy nhiên độ dài thực tế có thể bị ảnh hưởng bởi tốc độ truyền và chất lượng cáp. Đối với khoảng cách lớn hơn, bạn có thể cần sử dụng các bộ khuếch đại tín hiệu hoặc chuyển sang công nghệ truyền thông khác như RS485.

Cáp chéo và cáp thẳng: Tùy thuộc vào ứng dụng (ví dụ: kết nối máy tính với máy tính hoặc máy tính với thiết bị ngoại vi), bạn có thể cần sử dụng cáp chéo (null modem) hoặc cáp thẳng.

Cấu hình Phần mềm: Cấu hình phần mềm cho kết nối RS232, bao gồm tốc độ baud (bit mỗi giây), bit dừng, bit dữ liệu và kiểm tra chẵn lẻ (parity), cần được thực hiện cẩn thận để khớp với cấu hình của thiết bị đối diện.

Nhiễu: RS232 có thể bị ảnh hưởng bởi nhiễu điện từ, đặc biệt là ở tốc độ baud cao hoặc độ dài cáp lớn. Sử dụng cáp chống nhiễu hoặc giảm tốc độ truyền có thể giúp giảm thiểu vấn đề này

2.2 Đối với RS485 (thiết bị hổ trợ Half duplex – truyền nhận 2 dây)

Địa chỉ và giao thức: RS485 không định nghĩa một giao thức truyền thông; nó chỉ cung cấp lớp vật lý. Do đó, việc sử dụng RS485 đòi hỏi phải có một giao thức truyền thông phù hợp như Modbus hoặc Profibus để quản lý địa chỉ và đồng bộ hóa các thiết bị.

Chống nhiễu: Cáp chống nhiễu và đúng loại cáp là rất quan trọng để giảm thiểu tác động của nhiễu điện từ trong môi trường công nghiệp.

Cân nhắc về địa chỉ: Trong một hệ thống RS485, mỗi thiết bị cần có một địa chỉ duy nhất để có thể giao tiếp đúng cách trên mạng.

Half-duplex: Chế độ 2 dây hỗ trợ truyền thông half-duplex, nghĩa là dữ liệu chỉ có thể truyền đi hoặc nhận về một lúc, không thể đồng thời.

Kết nối: Cùng một cặp dây được sử dụng để truyền và nhận dữ liệu. Điều này giúp giảm chi phí cáp và đơn giản hóa cấu trúc mạng.

Điện trở kết thúc: Một điện trở kết thúc nên được lắp ở cả hai đầu của mạng để giảm nhiễu và tín hiệu phản xạ, đặc biệt quan trọng trong các mạng dài và ở tốc độ baud cao.

2.3 Thuật toán RSA

Các thiết bị của công ty AITECH, sử dụng thuật toán RSA để đảm bảo an toàn trong giao tiếp và trao đổi dữ liệu. RSA là một thuật toán mã hóa khóa công khai phổ biến, được sử dụng để mã hóa dữ liệu, tạo chữ ký số, và xác minh tính toàn vẹn của thông tin. Trong bối cảnh của các thiết bị AITECH, RSA có thể được sử dụng để:

- **Mã hóa dữ liệu**: Đảm bảo rằng dữ liệu truyền đi qua mạng hoặc lưu trữ trên thiết bị được bảo vệ khỏi những người không có quyền.
- **Xác thực thiết bị**: Sử dụng chữ ký số dựa trên RSA để xác minh rằng thiết bị đang giao tiếp là hợp lệ và được ủy quyền.
- **Bảo mật kết nối:** Sử dụng RSA trong các thiết bị của AITECH giúp tăng cường bảo mật, ngăn chặn truy cập trái phép, và bảo vệ tính bí mật của dữ liệu.

3. Thông tin cấu hình ban đầu của thiết bị

Thông tin thiết bị khi xuất xưởng:

- Thông tin đăng nhập vào thiết bị:

Thông số	Giá trị
Địa chỉ IP	192.168.1.201
User	admin
Password	admin

- Thông tin kết nối socket:

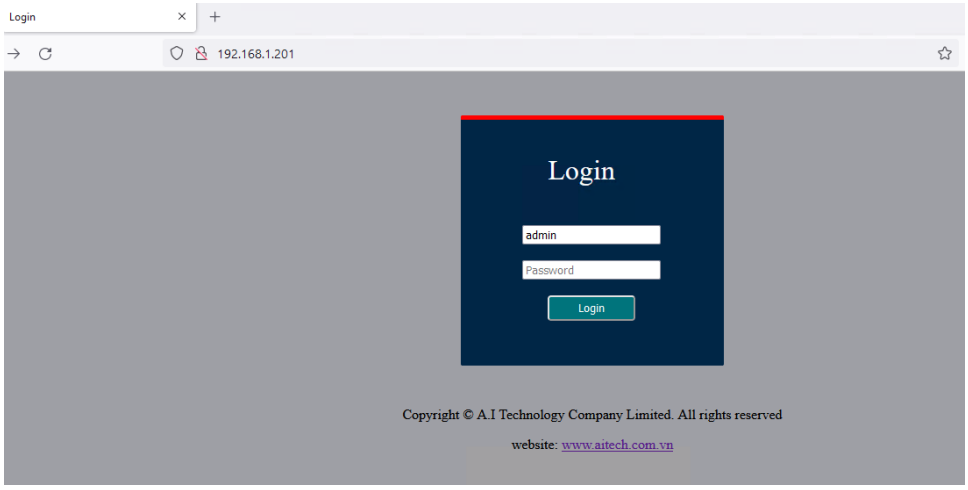
Thông số	Giá trị
Socket user	socketuser
Socket password	socketpassword

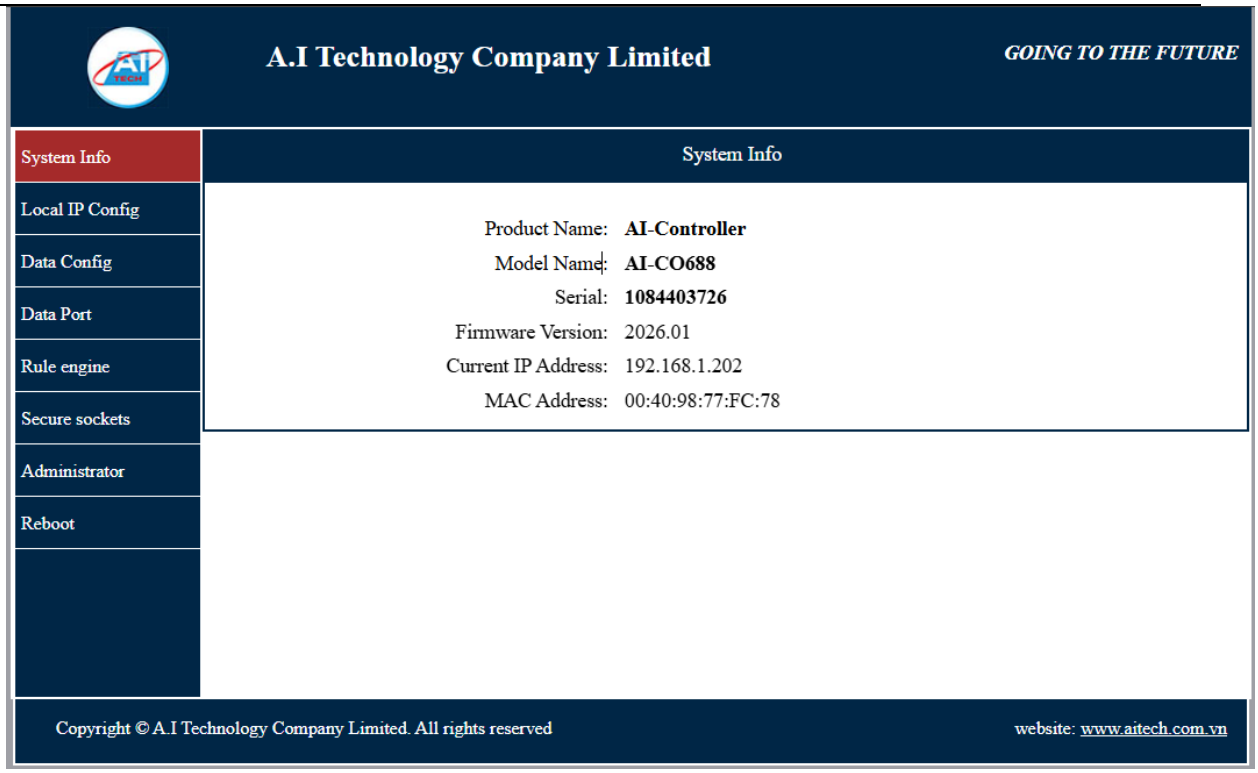
- Bảo mật RSA: Key mặc định

PUBLIC KEY	PRIVATE KEY
MÃ DÙNG TRÊN THIẾT BỊ	
modulus=B70FE4BCADE88781E71ACF8E83BEEA2F522D41B22E3941075B9881C372288BBDBD805B70C41347F3A5AA4A07ED5186D8E0F0C14DF7708ADE8DC729389EC67ABB0AA469A8CC11B2672D187D958A4541C16E4B89689BAF2A7045DB9E9179E2D1EB092E3145439AF1316D0FAA160E778DEAA3E57CFEBEFDB38A1B09FC5A788D4C1D exponent=010001	modulus=B70FE4BCADE88781E71ACF8E83BEEA2F522D41B22E3941075B9881C372288BBDBD805B70C41347F3A5AA4A07ED5186D8E0F0C14DF7708ADE8DC729389EC67ABB0AA469A8CC11B2672D187D958A4541C16E4B89689BAF2A7045DB9E9179E2D1EB092E3145439AF1316D0FAA160E778DEAA3E57CFEBEFDB38A1B09FC5A788D4C1D exponent=135CE700DE143E610B24D6D76830056D43EFED86FF44A710342C096DB6E17B0672C8E4E3249A357E1C03CB18D38D500A18A0DE91749EBED193858162AFC8A6AA7C3F20C33D7525A03606C333BDB3C353824EDC257CA6660102C8753AA2EA1EDC919286BA9261BFF9D378D9895CF43B3D9DC8297A77B6F3111F406911B83D8D31
MÃ DẠNG XML DÙNG TRÊN PHẦN MỀM	
<RSAKeyValue> <Modulus>tw/kvK3oh4HnGs+Og77qL1ltQbIuOUEHW5iBw3loi729gFtwxBNH86WqSgftUYbY4PDBTfdwit6Nxyk4nsZ6uwqkaajMEbJnLRh9lYpFQcFuS4lom68qcEXbnpF54tHrCS4xRUOa8TFtD6oWDneN6qPlfP6+/bOKGwn8WniNTB0=</Modulus>	<RSAKeyValue> <Modulus>tw/kvK3oh4HnGs+Og77qL1ltQbIuOUEHW5iBw3loi729gFtwxBNH86WqSgftUYbY4PDBTfdwit6Nxyk4nsZ6uwqkaajMEbJnLRh9lYpFQcFuS4lom68qcEXbnpF54tHrCS4xRUOa8TFtD6oWDneN6qPlfP6+/bOKGwn8WniNTB0=</Modulus>

<Exponent>AQAB</Exponent> </RSAKeyValue>	<Exponent>AQAB</Exponent> <P>0T6r6y1LlMjWMqWhkIIu5MQOPQLUctHdEL/900H1f8TQTdmJX2sMLy5SA3by8Pl9hkp9yNmEceMopGhdaq0pZw==</P> <Q>3/eBFndRcz8SioG9SRs5l3iqX5ak+ABChdtecyvJNiF6pT5JEkOaIUls9UZiE9JpkcvFpmpetjN9yd5cbvR32w==</Q> <DP>m7/4m+91SORNfBmJr1UXrrS4nHVUybqowGUIfI2mPVq0ap44AMxfn2btKSMLJopJfYJJKo2uZk9d2zLMBjGqlQ==</DP> <DQ>Bc2B2LUqtROzRUdrQhOkgFot/+AFUi+fiZxPbCh8RBzF8rYUapa5zZb7bbdvHxhah96BtngXEB1xauOJ/Y3z9Q==</DQ> <InverseQ>pTC/KBGVO9anMSLZ3cv43/v6Jm+oCgnu9D5+4nXnTr5eKx4By0AKT8hzdZ6GMDmg2kBkzfXrPwa+ALjU5jPn5g==</InverseQ> <D>E1znAN4UPmELJNbXaDAFbUPv7Yb/RKcQNCwJbbbhewZyyOTjJJo1fhwDyxjTjVAKGKDekXSevtGThYFir8imqnw/IMM9dSWgNgbDM72zw1OCTtwlfKZmAQLIdTqi6h7ckZKGupJhv/nTeNmJXPQ7PZ3IKXp3tvMRH0BpEbg9jTE=</D> </RSAKeyValue>

Dùng trình duyệt web để đăng nhập vào thiết bị:



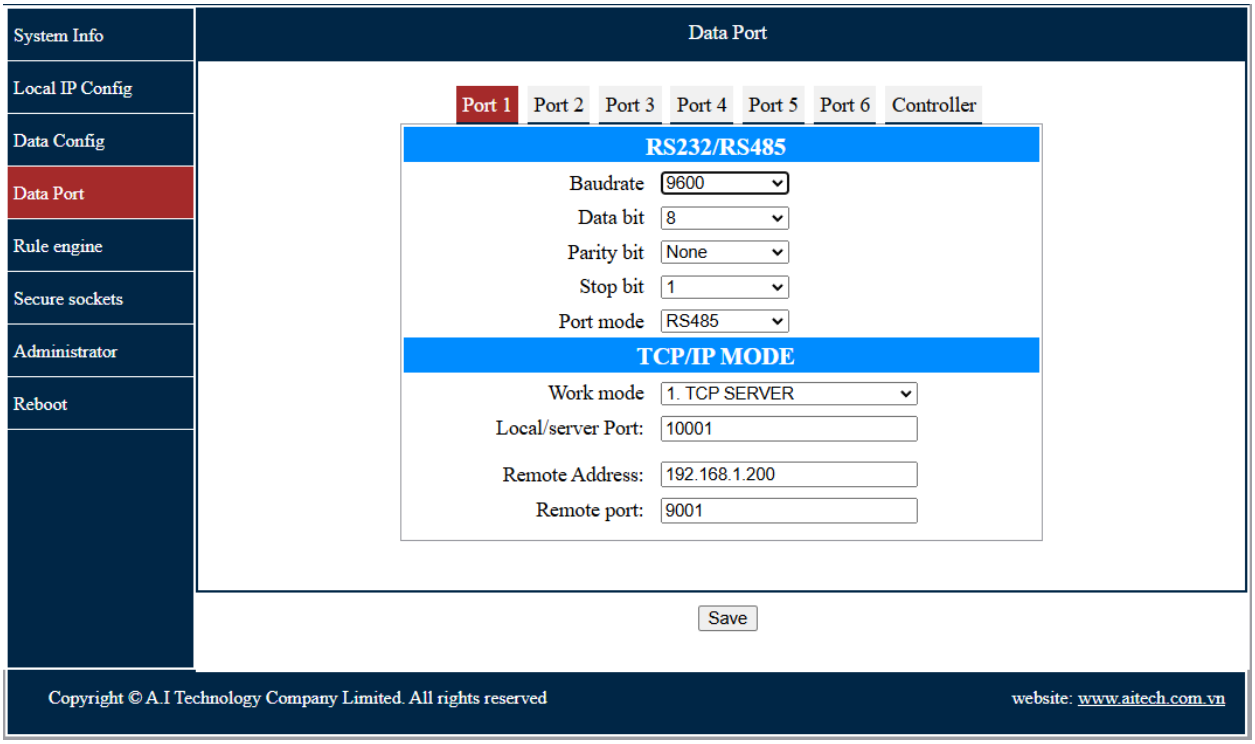


4. Cấu hình cơ bản của thiết bị.

4.1 Cấu hình port

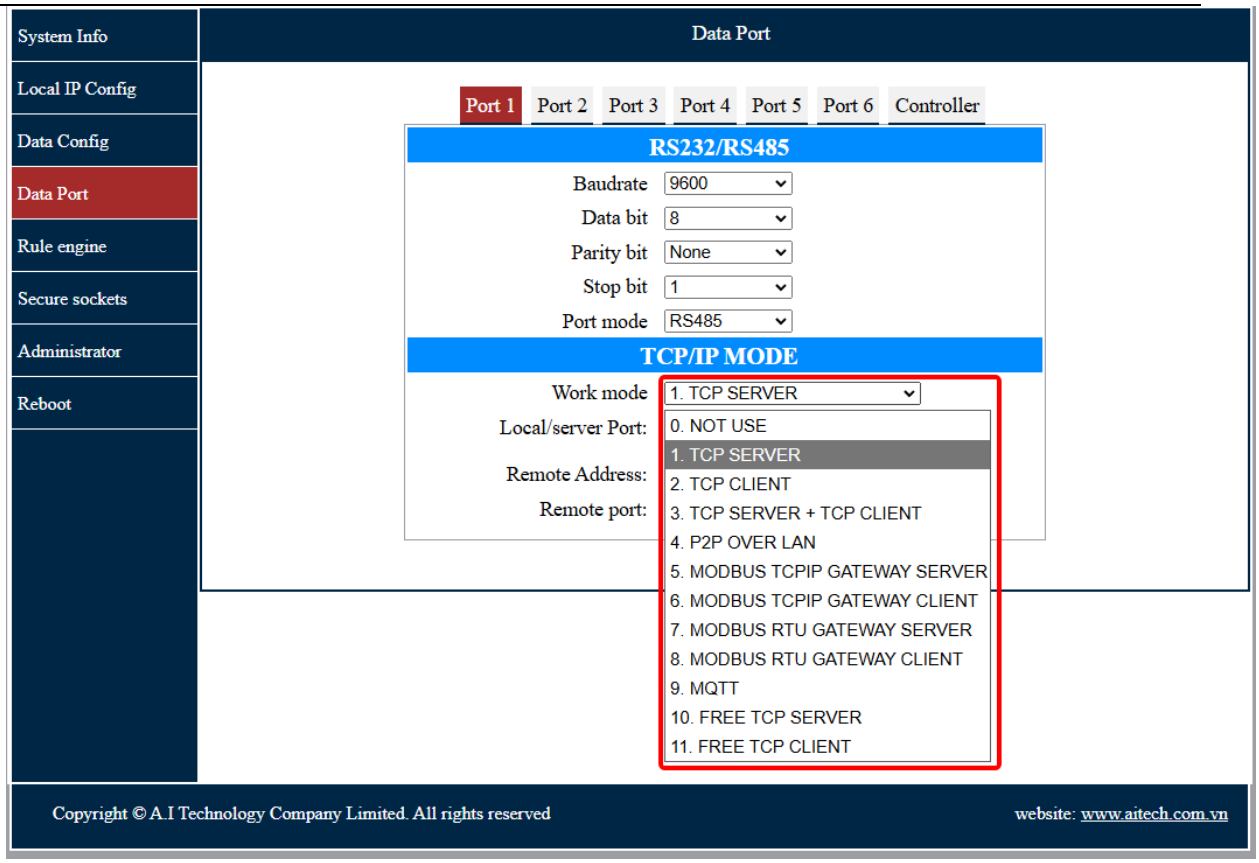
- Mỗi cổng (Port 1, Port 2, Port 3, Port 4, Port 6) tương ứng với mỗi port vật lý trên thiết bị.
- Mỗi cổng RS232 và RS485 sẽ có vị trí vật lý khác nhau trên thiết bị, lưu ý vị trí kết nối vật lý khi sử dụng thiết bị và cấu hình loại kết nối thiết bị phải giống nhau, đối với thiết bị có mã AI-CO688 thì:
    - **Port 1, Port 2, Port 3, Port 6: Giao tiếp RS485**
    - **Port 4, Port 5: Giao tiếp RS232**
    - **Controller: Cấu hình điều khiển DI/DO**
- Cần cấu hình đúng với kết nối đang sử dụng các thông tin sau:
    - Baudrate
    - Data bit
    - Stop bit

- Port mode (*cần cấu hình đúng với cổng vật lý trên thiết bị*)



4.2 Cấu hình TCP/IP các kết nối

Kết nối điều khiển I/O, dữ liệu từng cổng trên thiết bị hoạt động độc lập nhau trên TCP port khác nhau.



Thiết bị có thể hoạt động với nhiều chế độ khác nhau gồm:

- **0. NOT USE**: Không sử dụng cổng này

Nhóm 1: Chế độ TCP Tiêu Chuẩn

- **1. TCP SERVER**
    - **Nguyên lý:** Thiết bị ngoại vi (hardware) sẽ mở một cổng mạng (Port) cụ thể và chuyển sang trạng thái "Lắng nghe" (Listen). Nó thụ động chờ các thiết bị khác (Client) kết nối đến. Khi có kết nối, dữ liệu nhận từ cổng Serial (như RS232/RS485) sẽ được đẩy trực tiếp lên kết nối TCP này.
    - **Ứng dụng thực tế:** Phù hợp trong mạng LAN nội bộ. Ví dụ: Một service trên máy chủ nội bộ chủ động mở socket kết nối đến địa chỉ IP của đầu đọc RFID UHF tại cổng kiểm soát để liên tục lấy mã thẻ quẹt.
- **2. TCP CLIENT**

- **Nguyên lý:** Thiết bị phần cứng sẽ chủ động đi tìm và khởi tạo kết nối đến một địa chỉ IP và Port của máy chủ (Server) đã được khai báo trước. Nếu kết nối đứt, thiết bị sẽ liên tục thử kết nối lại (reconnect).
- **Ứng dụng thực tế:** Phù hợp cho thiết bị nằm ở các vị trí phân tán, qua mạng NAT hoặc 4G/5G. Ví dụ: Vi điều khiển đọc dữ liệu từ cảm biến nhiệt độ, độ ẩm hoặc đồng hồ thời gian thực và chủ động kết nối TCP để đẩy dữ liệu telemetry về một máy chủ đặt ở trung tâm dữ liệu.

- **3. TCP SERVER + TCP CLIENT**
  - **Nguyên lý:** Thiết bị chạy song song cả hai luồng mạng. Vừa duy trì một kết nối chủ động (Client) đẩy dữ liệu về trung tâm, vừa mở một cổng (Server) để cho phép kết nối cục bộ.
  - **Ứng dụng thực tế:** Dùng trong các hệ thống yêu cầu độ dự phòng cao. Dữ liệu chính vẫn liên tục đẩy về server trung tâm (Client mode), nhưng khi cần bảo trì, kỹ thuật viên có thể cắm máy tính vào mạng LAN trạm và dùng phần mềm kết nối trực tiếp vào thiết bị (Server mode) để xem log hoặc chẩn đoán.
- **4. P2P OVER LAN**
  - **Nguyên lý:** Mạng ngang hàng (Peer-to-Peer). Các thiết bị tự động phát hiện nhau trong cùng một dải mạng LAN và truyền dữ liệu trực tiếp mà không cần một máy chủ TCP trung tâm.
  - **Ứng dụng thực tế:** Liên kết các cụm phần cứng với nhau. Ví dụ: Hai bo mạch điều khiển cổng barie ở hai đầu trạm cân tự động đồng bộ trạng thái đóng/mở cho nhau.
- **5. MODBUS TCPIP GATEWAY SERVER**
  - **Nguyên lý:** Thiết bị hoạt động như một bộ dịch ngôn ngữ (Protocol Converter). Khi nhận được bản tin chuẩn **Modbus TCP** từ mạng Ethernet, nó sẽ tháo bỏ vỏ bọc TCP, trích xuất dữ liệu và chuyển thành chuẩn **Modbus RTU** gửi qua cổng Serial (RS485/RS232) tới thiết bị cuối. Khi thiết bị cuối phản hồi, nó làm ngược lại. Chế độ

Server nghĩa là nó chờ phần mềm trung tâm kết nối vào để hỏi dữ liệu.

- **Ứng dụng thực tế:** Tích hợp các cảm biến công nghiệp thế hệ cũ (chỉ có RS485/Modbus RTU) lên hệ thống mạng IP hiện đại để SCADA hoặc phần mềm quản lý có thể đọc được dễ dàng.

- **6. MODBUS TCPIP GATEWAY CLIENT**
  - **Nguyên lý:** Tương tự như (5) về mặt dịch thuật giao thức, nhưng ở chế độ này, bộ Gateway sẽ **chủ động** kết nối TCP tới máy chủ trung tâm.
  - **Ứng dụng thực tế:** Vượt qua các tường lửa hoặc mạng NAT. Máy chủ trung tâm không thể "nhìn thấy" IP của Gateway để kết nối vào (như mode Server), nên Gateway phải chủ động mang dữ liệu Modbus TCP đi tìm máy chủ.
- **7. MODBUS RTU GATEWAY SERVER**
  - **Nguyên lý:** Không dịch thuật giao thức. Nó nhận dữ liệu của frame Modbus RTU từ cổng Serial và truyền thẳng vào phần tải (payload) của gói tin TCP/IP (gọi là Modbus RTU over TCP).
  - **Ứng dụng thực tế:** Dùng khi phần mềm quản lý trên máy tính được lập trình sẵn để xử lý trực tiếp các frame Modbus RTU thô (raw bytes) nhưng lại muốn truyền qua đường cáp mạng thay vì cáp COM/Serial truyền thống.
- **8. MODBUS RTU GATEWAY CLIENT**
  - **Nguyên lý:** Giống hệt (7) là không dịch thuật (chỉ bọc dữ liệu), nhưng thiết bị chủ động khởi tạo kết nối mạng lên máy chủ định trước.

Nhóm 3: Chế độ Giao Thức IoT Tối Ưu

- **9. MQTT**
  - **Nguyên lý:** Thiết bị hoạt động như một MQTT Client, kết nối tới một MQTT Broker tập trung (như EMQX, Mosquitto). Dữ liệu được gửi đi theo dạng "Publish" lên các chủ đề (Topic), và nhận lệnh điều khiển thông qua "Subscribe" các chủ đề đó.

- **Ứng dụng thực tế:** Giao thức hoàn hảo cho IoT vì cực kỳ nhẹ, tiết kiệm băng thông và có cơ chế giữ kết nối tốt. Rất phù hợp để theo dõi các thông số thời gian thực (giám sát phần cứng CPU, RAM, Temp từ xa) hoặc đẩy log hệ thống về một Dashboard giám sát tập trung mà không lo bị nghẽn mạng.

Nhóm 4: Chế độ TCP Tùy Biến

- **10. FREE TCP SERVER**
  - **Nguyên lý:** Một biến thể của TCP Server tiêu chuẩn nhưng được nhà sản xuất module mở rộng thêm tính năng. Hỗ trợ cấu hình sâu hơn về Heartbeat packet (gói tin giữ nhịp), khả năng chịu tải nhiều Client kết nối cùng lúc hơn, hoặc tự động ngắt kết nối ảo (zombie connections).
- **11. FREE TCP CLIENT**
  - **Nguyên lý:** Biến thể của TCP Client, cho phép cấu hình linh hoạt việc gửi chuỗi "Registration packet" (gói tin đăng ký) ngay khi vừa kết nối thành công tới Server.
  - **Ứng dụng thực tế:** Giúp máy chủ phân biệt được thiết bị nào vừa kết nối tới (dựa vào MAC hoặc ID đính kèm trong gói đăng ký), cực kỳ hữu ích khi quản lý hàng nghìn thiết bị đẩy dữ liệu về cùng một IP.

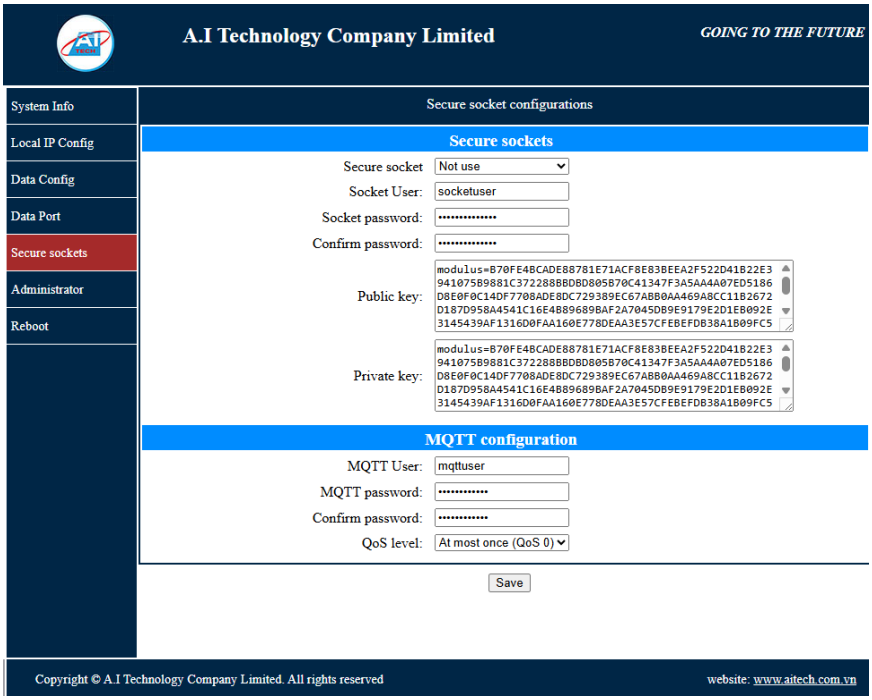
- **1. TCP SERVER**: Cổng hoạt động chế độ **TCP server** đi qua cổng **Ethernet**; các thiết bị **phần mềm khác ở chế độ TCP Client** kết nối tới địa chỉ IP của thiết bị và port Local/server Port của thiết bị.
- **2. TCP CLIENT**: Cổng hoạt động ở chế độ TCP Client **đi qua cổng Ethernet**; thiết bị kết nối tới TCP Server của thiết bị hoặc phần mềm từ thông tin: **Remote Address và Remote Port**
- **3. TCP SERVER + TCP CLIENT**: Thiết bị hoạt động cả chế độ **TCP Server và TCP Client đi qua cổng Ethernet**
- **4. P2P OVER LAN:** Point-to-Point hoặc UDP Broadcast/Multicast. Dùng để hai thiết bị phần cứng tự động tìm và truyền dữ liệu trực tiếp cho nhau trong cùng một mạng LAN mà không cần thông qua server trung gian.

- **5. MODBUS TCPIP GATEWAY SERVER :** Thiết bị đóng vai trò bộ chuyển đổi giao thức (Protocol Converter). Nó sẽ nhận bản tin Modbus RTU (qua cổng Serial), dịch thành chuẩn **Modbus TCP** (có header TCP) và gửi qua mạng internet.
- **6. MODBUS TCPIP GATEWAY CLIENT:** Thiết bị đóng vai trò bộ chuyển đổi giao thức (Protocol Converter). Nó sẽ nhận bản tin Modbus RTU (qua cổng Serial), dịch thành chuẩn **Modbus TCP** (có header TCP) và gửi qua mạng internet.
- **7. MODBUS RTU GATEWAY SERVER / 8. CLIENT:** Thiết bị nhận bản tin Modbus RTU và đóng gói thẳng vào payload của gói tin TCP/IP (Modbus RTU over TCP) mà không thay đổi cấu trúc frame Modbus. Các phần mềm backend xử lý theo dạng truyền raw byte thông thường.
- **9. MQTT:** Giao thức Publish/Subscribe cực kỳ nhẹ và ổn định. Thiết bị sẽ đóng vai trò là MQTT Client, kết nối đến một MQTT Broker (như Mosquitto, EMQX). Chế độ này lý tưởng cho việc truyền dữ liệu telemetry có tần suất liên tục nhưng dung lượng nhỏ, ví dụ như gửi thông số nhiệt độ, độ ẩm (từ DHT11), hoặc trạng thái phần cứng (CPU, RAM, Temp) về một dashboard giám sát tập trung.
- **10. FREE TCP SERVER / 11. FREE TCP CLIENT:** Các chế độ cho phép người dùng lập trình Ngôn ngữ Rule bytecode lên 2048 , có thể lập trình lên 500 dòng lệnh để controller xử lý các tự động các thao tác nhẹ, ví dụ kiểm tra nhiệt độ và bật relay mà không cần xử lý phần mềm

4.3 Bảo mật kết nối socket (Secure socket)

- **Not use**: Không sử dụng cơ chế bảo mật xác thực, các kết nối thực hiện như các kết nối thông thường khác, triển khai dễ dàng. Thường được triển khai trong mạng LAN có độ tin tưởng cao đã được bảo mật.
- **Send device information:** Xác thực thông tin kết nối; Thiết bị sẽ gởi thông tin kết nối của socket nhưng không mã hóa như: ID device, socket user, socket password.
- **Encrypt device information**: Xác thực thông tin kết nối; Thiết bị sẽ gởi thông tin kết nối của socket và mã hóa thông tin gởi: ID device, socket user, socket password. Nhà sản xuất khuyến cáo nên sử dụng chế độ này.

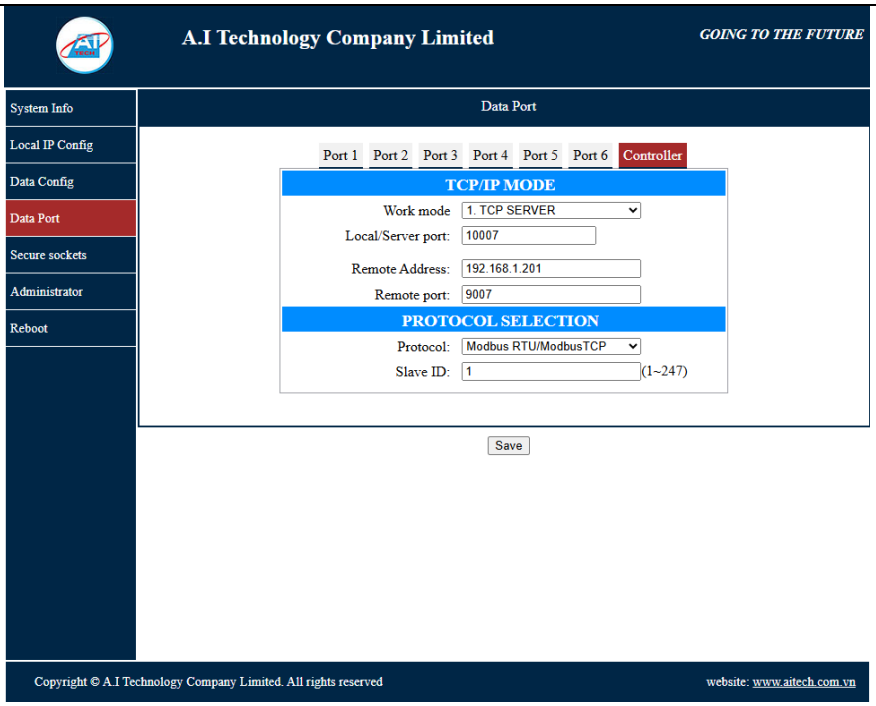
- **Encrypt data:** Xác thực thông tin kết nối; và mã hóa dữ liệu gởi giữa các thiết bị và kết nối TCP.



Có thể thay đổi RSA key bằng phần mềm tạo RSA đi kèm, socket user, socket password; việc này tăng tính bảo mật và đảm bảo thiết bị của bảo thiết bị của bạn không bị truy cập trái phép.

5. Chức năng điều khiển DI/DO từ máy tính

Giao thức kết nối: Socket TCP hoặc Modbus TCP



5.1 Điều khiển và giám sát DI/DO theo giao thức AI COMMAND

Thiết bị AI-CO688 hỗ trợ giao thức TCP/IP tùy chỉnh AI COMMAND cho phép:

- Giám sát trạng thái Input theo thời gian thực
- Phát sinh sự kiện ON/OFF
- Điều khiển Relay Output (ON / OFF / Pulse)
- Tự động kết nối lại khi mất mạng
- Hoạt động ổn định trong chế độ scan liên tục 24/7

Chức năng này đặc biệt phù hợp cho:

- Phần mềm giám sát custom (C#, VB.NET, C++, python..., SCADA riêng)
- Hệ thống cần phản ứng nhanh theo sự kiện.
- IoT Edge / Gateway không phụ thuộc Modbus

5.1.1 Giao thức khung lệnh TCP (Frame Structure)

- Cấu trúc gói tin:

[STX] [CMD] [SUB] [CHANNEL] [CHK] [ETX]

Trường	Giá trị (hex)
STX	24
CMD	43
SUB	30: Xuất xung điều khiển 54: Đọc trạng thái input 50: Điều khiển ON 70: Điều khiển OFF
CHANNEL	1 - 8 : từ kênh 1 đến kênh 8 [0]: Khi đọc trạng thái input
CHK	XOR(STX, CMD, SUB, DATA)
ETX	23

Đọc trạng thái input:

Request command (Hex)	Response	Note
24 43 54 00 33 23	24 43 54 **05** 32 23	**05h** = 0000 0101b Input1 = ON Input2 = OFF Input3 = ON Input4 = OFF Input5 = OFF Input6 = OFF Input7 = OFF Input8 = OFF

5.1.2 Điều khiển đóng (ON) Output

Kênh	Request command (Hex)	Response
1	24 43 50 **01** 36 23	24 41 43 4B 23
2	24 43 50 **02** 35 23	24 41 43 4B 23
3	24 43 50 **03** 34 23	24 41 43 4B 23
4	24 43 50 **04** 33 23	24 41 43 4B 23
5	24 43 50 **05** 32 23	24 41 43 4B 23

6	24 43 50 **06** 31 23	24 41 43 4B 23
7	24 43 50 **07** 30 23	24 41 43 4B 23
8	24 43 50 **08** 3F 23	24 41 43 4B 23

5.1.3 Điều khiển hở (OFF) Output

Kênh	Request command (Hex)	Response
1	24 43 70 **01** 16 23	24 41 43 4B 23
2	24 43 70 **02** 15 23	24 41 43 4B 23
3	24 43 70 **03** 14 23	24 41 43 4B 23
4	24 43 70 **04** 13 23	24 41 43 4B 23
5	24 43 70 **05** 12 23	24 41 43 4B 23
6	24 43 70 **06** 11 23	24 41 43 4B 23
7	24 43 70 **07** 10 23	24 41 43 4B 23
8	24 43 70 **08** 1F 23	24 41 43 4B 23

5.1.4 Điều khiển Output xuất xung 300 ms (điều khiển: Barrier ...)

Kênh	Request command (Hex)	Response
1	24 43 30 **01** 56 23	24 41 43 4B 23
2	24 43 30 **02** 55 23	24 41 43 4B 23
3	24 43 30 **03** 54 23	24 41 43 4B 23
4	24 43 30 **04** 53 23	24 41 43 4B 23
5	24 43 30 **05** 52 23	24 41 43 4B 23
6	24 43 30 **06** 51 23	24 41 43 4B 23
7	24 43 30 **07** 50 23	24 41 43 4B 23
8	24 43 30 **08** 5F 23	24 41 43 4B 23

5.1.5 Phản hồi từ thiết bị

Thực hiện thành công	Lệnh không hợp lệ / lỗi
24 41 43 4B 23	24 41 43 4B 23
$ACK#	$NAK#

5.2 Điều khiển và giám sát DI/DO theo giao thức Modbus TCP

- Giao thức: Modbus TCP/IP
- Chế độ: Slave
- Địa chỉ IP mặc định: 192.168.1.201
- Port cài đặt ban đầu: 10007
- Slave ID cài đặt ban đầu: 1
- Kiểu dữ liệu Big Endian
- Thiết bị luôn dùng địa chỉ logic 0-based (KHÔNG dùng offset 40001 / 00001)
- Modbus Function Map:

Function	FC	Quyền	Kiểu DL	Địa chỉ	Giải thích
Read Discrete Inputs	02	Read	Bool	0 → 7	Trạng thái DI1 → DI8
Read Coils	01	Read	Bool	0 → 7	Trạng thái DO1 → DO8
Read Holding Registers	03	Read	Int16	0 → 7	ON Counter của DI1 → DI8
Read Holding Registers	03	Read	Int16	8 → 15	OFF Counter của DI1 → DI8
Write Single Coils	05	Write	Bool	0 → 7	Điều khiển DO1 → DO8 (ON / OFF)
Write Single Register	06	Write	Int16	0 → 7	Ghi giá trị ON Counter DI1 → DI8
Write Single Register	06	Write	Int16	8 → 15	Ghi giá trị OFF Counter DI1 → DI8

5.3 Các kết nối khác.

- (Đang cập nhật nội dung)

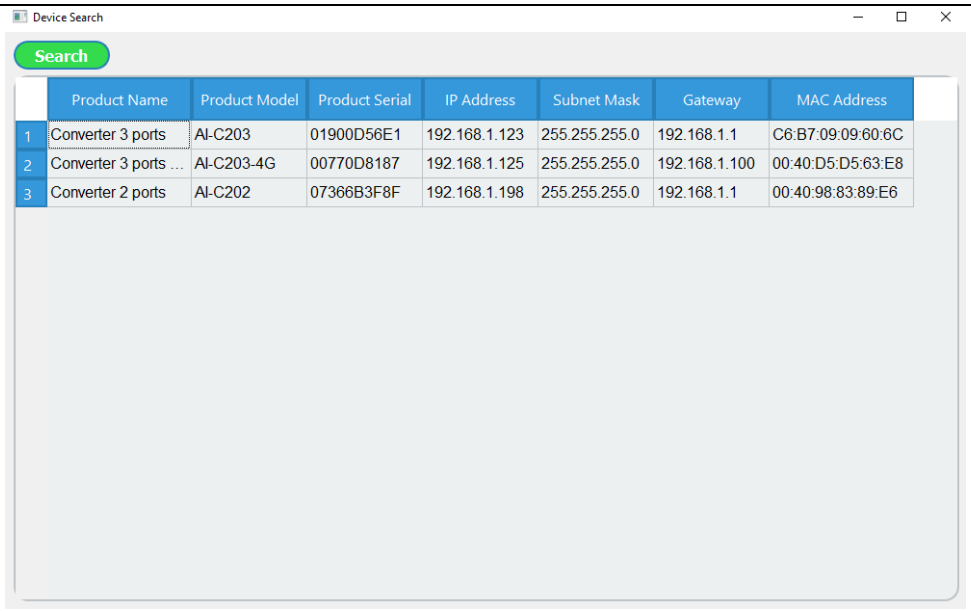
6. Phần mềm hỗ trợ kèm theo

Thiết bị có các phần mềm hỗ trợ kèm theo.

Tải phần mềm tại link: https://aitech.com.vn/cong-cu/

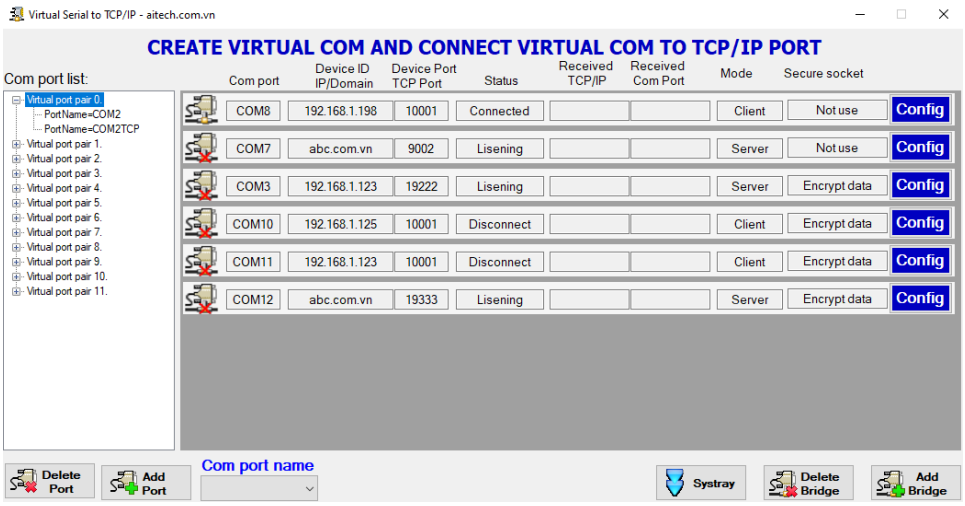
6.1 Phầm mềm tìm thông tin thiết bị

Phần mềm này tìm thông tin thiết bị trong mạng LAN



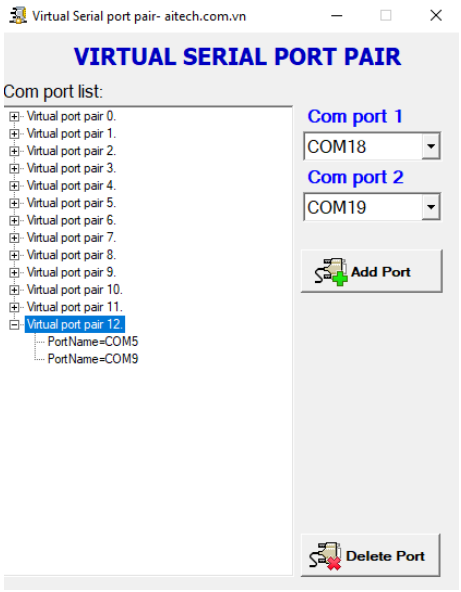
6.2 Phần mềm chuyển đổi dữ liệu TCP sang COM ảo VCOM2TCP

Phần mềm này tạo cổng COM ảo, kết nối TCP với thiết bị và chuyển dữ liệu từ cổng serial vật lý qua cổng COM ảo trên máy tính.



6.3 Phần mềm tạo cặp cổng COM ảo

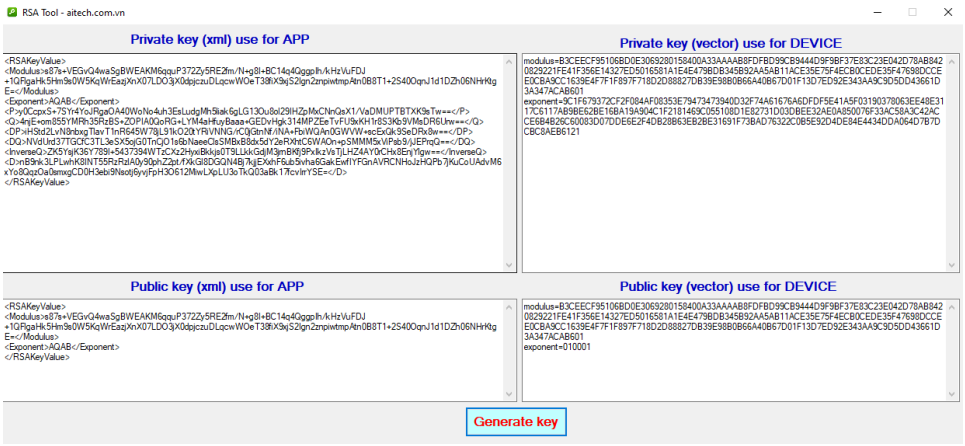
Phần mềm tạo cặp cổng COM ảo có kết nối cáp chéo (null modem) với nhau.



6.4 Phần mềm tạo Khóa RSA

Phần mềm tạo khóa RSA được thiết kế để tạo cặp khóa RSA trên thiết bị và định dạng XML để sử dụng trong phần mềm.

Phần mềm sinh ra một cặp khóa gồm một khóa công khai và một khóa riêng tư. Độ dài của khóa 1024 byte, độ dài dữ liệu có thể mã hóa tối đa 117 byte.



7. Source demo kết nối thiết bị.

Tải source demo tại link: https://aitech.com.vn/source-code/

- (Đang cập nhật nội dung)

8. Thông tin công ty TNHH Công nghệ A.I

Địa chỉ: 69 Nguyễn Thị Rành, ấp Bàu Tre 2, xã Tân An Hội, Tp. Hồ Chí Minh

Hotline: 0949491355

Email: info@aitech.com.vn

Website: https://aitech.com.vn

Fanpage: https://www.facebook.com/AITechVietnam