

UART2Ethernet Manual

Công ty: R&P

Nội dung: Sử dụng thiết bị UART2Ethernet(U2E)

Phiên bản Rev.02

MICROSOFT WORD

Sơ lược:

UART2Ethernet là thiết bị cho phép bất kỳ một ứng dụng hay thiết bị nào có cổng Serial(UART) có thể kết nối vào mạng Ethernet và Internet, truyền hay nhận dữ liệu từ Ethernet thông qua Serial. Nó là một thiết bị đặc lực khi xây dựng ứng dụng có sử dụng truyền thông qua mạng, bạn có thể kết nối tới ứng dụng của mình từ bất kỳ nơi nào trên thế giới thông qua Internet hay tạo một kết nối đến Server của bạn từ U2E hoặc tạo kết nối giữa 2 thiết bị U2E với nhau qua mạng Ethernet hay Internet. Thiết bị U2E dễ dàng được cấu hình thông qua trang Web, hay qua cổng Serial. Bạn có thể sử dụng DHCP(tự động lấy IP) hay gán IP tĩnh cho U2E.

1. Đặc điểm sơ bộ

- ✓ Nguồn cấp 3.1V-3.6V@450mA
- ✓ UART 1200~115200 Baud, mặc định 19200 Baud
- ✓ Kích thước 940x1500 mil (238x381mm)
- ✓ Ngõ vào có thể cấp áp tới 5.5V



2. Sơ đồ chân

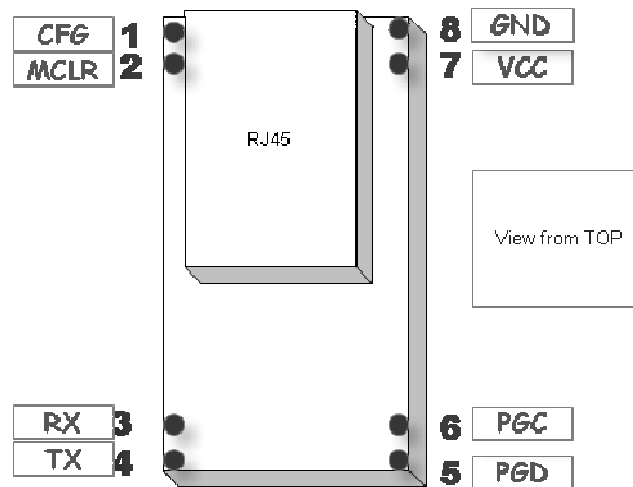


Figure 1 U2E Pin Configuration

PIN 1	CFG	Chân cấu hình, Chạm xuống vào mode cấu hình, mức 0 duy trì mode cấu hình, mức 1 trở lại mode hoạt động
PIN 2	MCLR	Giao diện ICSP
PIN 3	RX	UART Rx
PIN 4	TX	UART Tx
PIN 5	PGD	Giao diện ICSP/Trạng thái kết nối
PIN 6	PGC	Giao diện ICSP/RESET (Nối xuống 0 trong 5s để Reset U2E – Bình thường phải được nối lên mức 1)
PIN 7	VCC	Nguồn VCC. 3.3V
PIN 8	GND	Nguồn GND

3. Cấu hình thông qua trang Web

Mỗi một thiết bị U2E có một Netbios Name, được in trên nhãn của nó. Trang web cấu hình của U2E được truy xuất bằng chính tên này.

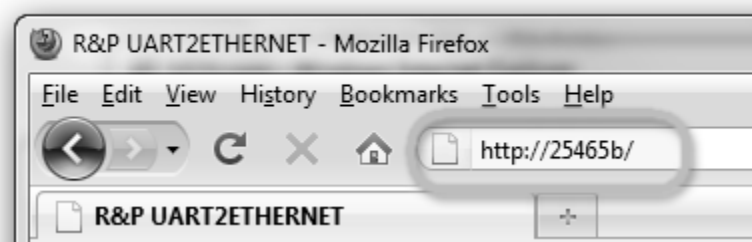


Figure 2 Truy xuất Web cấu hình

Mặc định tên đăng nhập và mật khẩu của U2E là **admin** và **admin**

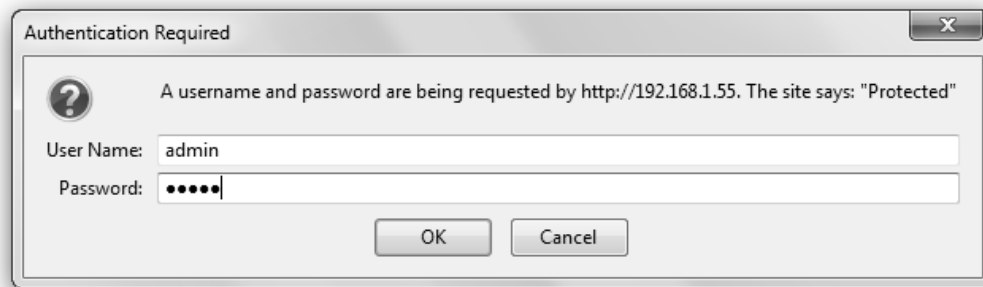


Figure 3 Username "admin", Password "admin" để đăng nhập vào Web cấu hình cho U2E

Navigation tabs: Status (selected), Network, UART, U2E, Password, Reboot

Server Information

	Thông tin
IP Address	192.168.1.55
DHCP	<input type="checkbox"/>
Gateway	192.168.1.1
Subnet Mask	255.255.255.0
Primary DNS Server	192.168.1.1
Secondary DNS Server	0.0.0.0
MAC Address	00:04:A3:25:9B:BE
Host name	http://259BBE
Using Dynamic DNS	<input checked="" type="checkbox"/>
DDNS Service	http://noip.info
DDNS User	youraccount
Domain	yourhost.no-ip.info
UART Socket Type	Server
Connect/Listen to	192.168.1.26
UART Baud rate	19200

COPYRIGHT © 2009 R&P TRADING AND FORWARDING CO. LTD. ALL RIGHTS RESERVED. WEBSITE: HTTP://RPC.VN

Figure 4 Status, thông tin cấu hình hiện hành của U2E

Status	Network	UART	U2E	Password	Reboot
Network Configuration					
		Information			
DHCP	<input type="checkbox"/>				
IP Address	<input type="text" value="192.168.1.55"/>				
Gateway	<input type="text" value="192.168.1.1"/>				
Subnet Mask	<input type="text" value="255.255.255.0"/>				
Primary DNS Server	<input type="text" value="192.168.1.1"/>				
Secondary DNS Server	<input type="text" value="0.0.0.0"/>				
MAC Address	<input type="text" value="00:04:A3:25:9B:BE"/>				
Host name	<input type="text" value="http://259BBE"/>				
<input type="button" value="Save Config"/>					

COPYRIGHT © 2009 R&P TRADING AND FORWARDING CO. LTD. ALL RIGHTS RESERVED. WEBSITE: HTTP://RPC.VN

Figure 5 Network, cấu hình dùng DHCP, IP tĩnh(LAN)...

Status	Network	UART	U2E	Password	Reboot
UART Configuration					
		Option			
Baud Rate	<input type="text" value="19200"/>				
<input type="button" value="Save"/>					

COPYRIGHT © 2009 R&P TRADING AND FORWARDING CO. LTD. ALL RIGHTS RESERVED. WEBSITE: HTTP://RPC.VN

Figure 6 UART, Chọn UART Baudrate (mặc định 19200, 1 bit stop, không flow control, không parity, data bit 8)

Uart To Ethernet Configuration		
	Option	Notes
UART Socket Type	<input type="radio"/> Client <input checked="" type="radio"/> Server	
Use DDNS	<input checked="" type="checkbox"/>	Current Status: The IP has not changed since the last update.<ok>
DDNS Service	http://noip.info	
Username	youraccount	
Password	••••••••	
Host	yourhost.no-ip.info	Ex: uart2ethernet.no-ip.com
Ethernet To UART		
Connect To Server	192.168.1.26	Ex: uart2ethernet.no-ip.info (Internet) hay 192.168.1.200 (LAN)
Port	1234	Valid value range 1000 to 9999
<input type="button" value="Save Config"/>		

COPYRIGHT © 2009 R&P TRADING AND FORWARDING CO. LTD. ALL RIGHTS RESERVED. WEBSITE: HTTP://RPC.VN

Figure 7 U2E, cấu hình kiểu kết nối, sử dụng dịch vụ DDNS

Change Password (Default Username: Admin)		
New Password	<input type="text"/>	Max 6 character
Type again	<input type="text"/>	
<input type="button" value="Change Password"/>		

COPYRIGHT © 2009 R&P TRADING AND FORWARDING CO. LTD. ALL RIGHTS RESERVED. WEBSITE: HTTP://RPC.VN

Figure 8 Password, thay đổi password

Reboot		
<input type="button" value="Save + Reboot"/>		

COPYRIGHT © 2009 R&P TRADING AND FORWARDING CO. LTD. ALL RIGHTS RESERVED. WEBSITE: HTTP://RPC.VN

Figure 9 Reboot, khởi động U2E để tải cấu hình mới

4. Cấu hình qua UART

a. UART

- Baud rate: 19200
- Data: 8bits
- Stop: 1 bit
- Parity: None
- FlowControl: None
- Chân CFG phải ở mức 0 khi cấu hình bằng UART, mức 1 khi hoạt động bình thường
- Chân RESET ở mức thấp khi khởi động sẽ reset U2E về cấu hình mặc định

b. Khung dữ liệu

❖ Gói từ thiết bị khác đến U2E

HEADER	LEN	CMD	DATA
--------	-----	-----	------

- **HEADER:** Bắt đầu 1 khung dữ liệu, 1 byte, từ thiết bị khác tới U2E, giá trị: 0xBA
- **LEN:** Chiều dài khung dữ liệu, bao gồm CMD và DATA
- **CMD:** Mã điều khiển, 1 byte
- **DATA:** Dữ liệu, chiều dài phụ thuộc vào mã điều khiển

❖ Gói từ U2E đến thiết bị khác

HEADER	LEN	STATUS	DATA
--------	-----	--------	------

- **HEADER:** Bắt đầu 1 khung dữ liệu, 1 byte, từ U2E đến thiết bị khác, giá trị: 0xBD
- **LEN:** Chiều dài khung dữ liệu, bao gồm CMD và DATA
- **STATUS:** Trạng thái làm việc, 1 byte
- **DATA:** Dữ liệu, chiều dài phụ thuộc vào mã điều khiển

❖ Sơ lược về mã điều khiển

CMD	Mô tả
0x01	Đọc địa chỉ IP
0x81	Ghi địa chỉ IP
0x02	Đọc Subnet Mask
0x82	Ghi Subnet Mask

0x03	Đọc DNS Chính
0x83	Ghi DNS Chính
0x04	Đọc DNS Phụ
0x84	Ghi DNS Phụ
0x05	Đọc Gateway
0x85	Ghi Gateway
0x06	Đọc trạng thái DHCP Client
0x86	Tắt/Mở DHCP Client
0x07	Đọc Mật khẩu hiện tại – mật khẩu truy cập vào Web cấu hình
0x87	Ghi Mật khẩu mới – mật khẩu truy cập vào Web cấu hình
0x08	Đọc UART Baud hiện thời
0x88	Ghi UART Baud mới
0x09	Đọc trạng thái cho phép cập nhật Dynamic DNS
0x89	Tắt/Mở cho phép cập nhật Dynamic DNS
0x0A	Đọc Dynamic DNS Server hiện tại
0x8A	Ghi/Sử dụng dịch vụ Dynamic DNS Server
0x0B	Đọc Dynamic DNS Username
0x8B	Ghi Dynamic DNS Username
0x0C	Đọc Dynamic DNS Password
0x8C	Ghi Dynamic DNS Password
0x0D	Đọc Dynamic DNS Hostname
0x8D	Ghi Dynamic DNS Hostname
0x0E	Đọc kiểu làm việc của U2E - U2E Socket Type (Client/Server)
0x8E	Ghi kiểu làm việc của U2E - U2E Socket Type (Client/Server)
0x0F	Đọc địa chỉ Server U2E sẽ kết nối tới khi làm việc ở chế độ Client (U2E Remote Server Address)
0x8F	Ghi địa chỉ Server U2E sẽ kết nối tới khi làm việc ở chế độ Client (U2E Remote Server)
0x11	Đọc giá trị Port mà U2E lắng nghe/ Kết nối (U2E Port)
0x91	Ghi giá trị Port mà U2E lắng nghe/Kết nối (U2E Port)
0x12	Lưu & Khởi động lại
0x13	Lấy lại cấu hình mặc định
0x14	Đọc chuỗi U2E Netbios Name
0xAA	Gửi HTTP Request tới HTTP Server

❖ Sơ lược về trạng thái

Status	Mô tả
0x00	Không có lỗi, lệnh thực hiện thành công
0x01	Lệnh thực hiện bị lỗi
0x02	Lệnh này không tồn tại

c. Danh sách chi tiết các lệnh điều khiển

❖ Đọc địa chỉ IP, Subnet Mask, Gateway, DNS1, DNS2

HEADER	LEN	CMD
0xBA	0x01	cmd

- cmd:

- 0x01: Đọc địa chỉ IP
- 0x02: Đọc Subnet Mask
- 0x03: Đọc DNS chính
- 0x04: Đọc DNS phụ
- 0x05: Đọc Gateway

- Trả về

HEADER	LEN	STATUS	DATA
0xBD	len	status	data

- Status:

- 0x00 : Lệnh thực hiện thành công
- 0x01: Lệnh thực hiện bị lỗi

- Data:

- Nếu status = 0x00, Data gồm 4 Byte, là IP/DNS/Gateway/Subnetmask được đọc về
- Ví dụ: trả về IP: 192.168.1.10

HEADER	LEN	STATUS	DATA			
0xBD	0x05	0x00	0xC0	0xA8	0x01	0x0A

❖ Ghi địa chỉ IP, DNS, Gateway, Subnet Mask

HEADER	LEN	CMD	DATA			
0xBA	0x05	cmd	Byte3	Byte2	Byte1	Byte0

- Cmd:

- 0x81: Ghi địa chỉ IP
- 0x82: Ghi Subnet Mask
- 0x83: Ghi DNS chính
- 0x84: Ghi DNS phụ
- 0x85: Ghi Gateway

- Trả về

HEADER	LEN	STATUS
0xBD	0x01	status

- Status:

- 0x00 : Lệnh thực hiện thành công
- 0x01: Lệnh thực hiện bị lỗi
- Ví dụ: Ghi IP: 192.168.1.11

HEADER	LEN	CMD	DATA			
0xBA	0x05	0x81	0xC0	0xA8	0x01	0x0B

❖ **Đọc trạng thái DHCP/DDNS/U2E Socket Type**

HEADER	LEN	CMD
0xBA	0x01	cmd

- Cmd:

- 0x06: Đọc trạng thái DHCP có được cho phép không
- 0x09: Đọc trạng thái cập nhật Dynamic DNS có được cho phép hay không
- 0x0E: Đọc trạng thái hiện tại U2E Socket Type (Server hay Client)

- Trả về

HEADER	LEN	STATUS	DATA
0xBD	len	status	data

- Status:

- 0x00 : Lệnh thực hiện thành công
- 0x01: Lệnh thực hiện bị lỗi

- Data:

- 0x00: DHCP, DDNS đang tắt/ không được cho phép, U2E Socket Type Server
- 0x01: DHCP, DDNS đang bật/ được cho phép, U2E Socket Type Client

❖ **Cho phép DHCP, Dynamic DNS và thay đổi U2E Socket Type**

HEADER	LEN	CMD	DATA
0xBA	0x02	cmd	data

- Cmd:

- 0x86: DHCP
- 0x89: DDNS
- 0x8E: U2E Socket Type

- Data:

0x00: DHCP, DDNS đang tắt/ không được cho phép, U2E Socket Type Server
 0x01: DHCP, DDNS đang bật/ được cho phép, U2E Socket Type Client

- Trả về

HEADER	LEN	STATUS
0xBD	0x01	status

- Status:

- 0x00 : Lệnh thực hiện thành công

- 0x01: Lệnh thực hiện bị lỗi

-

❖ **Đọc mật khẩu hiện tại (mật khẩu truy cập Web cấu hình), DDNS Username, DDNS Password, DDNS Hostname, U2E Remote Server Address, U2E Netbios Name**

HEADER	LEN	CMD
0xBA	0x01	cmd

- Cmd:

- 0x07: Đọc mật khẩu hiện tại
- 0x0B: Đọc DDNS Username
- 0x0C: Đọc DDNS Password
- 0x0D: Đọc DDNS Hostname
- 0x0F: Đọc U2E Remote Server Address
- 0x14: Đọc chuỗi U2E Netbios Name

- Trả về

HEADER	LEN	STATUS	DATA
0xBD	len	status	data

- Status:

- 0x00 : Lệnh thực hiện thành công
- 0x01: Lệnh thực hiện bị lỗi

- Data:

- Chuỗi ký tự

- Len: Chiều dài của data bao gồm status

- Ví dụ: trả về mật khẩu mật định "admin"

HEADER	LEN	STATUS	DATA				
0xBD	0x06	0x00	0x61	0x64	0x6D	0x69	0x6e

❖ **Ghi mật khẩu mới(mật khẩu truy cập Web cấu hình) ,DDNS Username, DDNS Password, DDNS Hostname, U2E Remote Server Address**

HEADER	LEN	CMD	DATA
0xBA	len	cmd	data

- Cmd:

- 0x87: Ghi mật khẩu hiện tại
- 0x8B: Ghi DDNS Username
- 0x8C: Ghi DDNS Password
- 0x8D: Ghi DDNS Hostname
- 0x8F: Ghi U2E Remote Server Address

- Len: Chiều dài của data bao gồm CMD

- Data:

- Mật khẩu mới: Tối đa là 6 ký tự, tối thiểu là 2 ký tự
- DDNS Username: Tối đa 24 ký tự, tối thiểu là 3 ký tự
- DDNS Password: Tối đa 10 ký tự và tối thiểu là 3 ký tự
- DDNS Hostname: Tối đa 24 ký tự và tối thiểu 3 ký tự
- U2E Remote Server Address: Domain hay IP(chuỗi – “xxx.xxx.xxx.xxx”) tối đa 24 tối thiểu 3 ký tự

- Trả về

HEADER	LEN	STATUS
0xBD	0x01	status

- Status:

- 0x00 : Lệnh thực hiện thành công
- 0x01: Lệnh thực hiện bị lỗi

- Ví dụ: đặt mật khẩu mới “admin”

HEADER	LEN	CMD	DATA				
0xBA	0x06	0x87	0x61	0x64	0x6D	0x69	0x6e

❖ **Đọc tốc độ Baud hiện tại**

HEADER	LEN	CMD
0xBA	0x01	0x08

- Trả về

HEADER	LEN	STATUS	DATA
0xBD	0x02	status	data

- Status:
 - 0x00 : Lệnh thực hiện thành công
 - 0x01: Lệnh thực hiện bị lỗi
- Data:
 - 0x01: 1200 baud
 - 0x02: 2400 baud
 - 0x03: 4800 baud
 - 0x04: 9600 baud
 - 0x05: 19200 baud
 - 0x06: 57600 baud
 - 0x07: 115200 baud

❖ Ghi tốc độ Baud mới

HEADER	LEN	CMD	DATA
0xBA	0x02	0x88	data

- Data:
 - 0x01: 1200 baud
 - 0x02: 2400 baud
 - 0x03: 4800 baud
 - 0x04: 9600 baud
 - 0x05: 19200 baud
 - 0x06: 57600 baud
 - 0x07: 115200 baud

- Trả về

HEADER	LEN	STATUS
0xBD	0x01	status

- Status:
 - 0x00 : Lệnh thực hiện thành công
 - 0x01: Lệnh thực hiện bị lỗi

❖ Đọc dịch vụ DDNS đang được sử dụng

HEADER	LEN	CMD
0xBA	0x01	0x0A

- Trả về

HEADER	LEN	STATUS	DATA
0xBD	0x02	status	data

- Status:
 - 0x00 : Lệnh thực hiện thành công
 - 0x01: Lệnh thực hiện bị lỗi
- Data:
 - 0x00: dịch vụ của "dyndns.org"
 - 0x01: dịch vụ của "no-ip.info"
 - 0x02: dịch vụ của "dnsomatic.com"

▪ Ghi dịch vụ DDNS nào sẽ sử dụng

HEADER	LEN	CMD	DATA
0xBA	0x02	0x0A	data

- Data:
 - 0x00: dịch vụ của "dyndns.org"
 - 0x01: dịch vụ của "no-ip.info"
 - 0x02: dịch vụ của "dnsomatic.com"

- Trả về

HEADER	LEN	STATUS
0xBD	0x01	status

- Status:
 - 0x00 : Lệnh thực hiện thành công
 - 0x01: Lệnh thực hiện bị lỗi

❖ **Đọc U2E Port**

HEADER	LEN	CMD
0xBA	0x01	0x11

- Trả về

HEADER	LEN	STATUS	DATA
0xBD	0x03	status	data

- Status:
 - 0x00 : Lệnh thực hiện thành công
 - 0x01: Lệnh thực hiện bị lỗi
- Data:
 - Giá trị 16 bit Port
- Ví dụ, trả về PORT 9999(DEC) = 0x270F

HEADER	LEN	STATUS	DATA	
0xBD	0x03	0x00	0x27	0x0F

❖ Ghi U2E Port

HEADER	LEN	CMD	DATA
0xBA	0x02	0x91	data

- Data:

- Giá trị 16 bit Port, Byte cao trước, byte thấp sau

- Trả về

HEADER	LEN	STATUS
0xBD	0x03	status

- Status:

- 0x00 : Lệnh thực hiện thành công
- 0x01: Lệnh thực hiện bị lỗi

- Ví dụ, ghi giá trị PORT 9999(DEC) = 0x270F

HEADER	LEN	CMD	DATA
0xBA	0x03	0x91	0x27 0x0F

❖ Khởi động lại , lấy cấu hình mặc định

HEADER	LEN	CMD
0xBA	0x01	cmd

- cmd:

- 0x12: Khởi động lại
- 0x13: Lấy lại cấu hình mặc định

❖ Gửi HTTP Request tới Server (domain/IP – chuỗi)

HEADER	LEN	CMD	DATA
0xBA	len	0xAA	data

- data:

- Chuỗi ký tự là domain hay ip address, ví dụ: "www.google.com.vn/?q=abc" hay "192.168.1.10"

- Trả về : Chuỗi dữ liệu từ Website, bao gồm thông tin HTTP Header

5. Ứng dụng U2E

- Hai U2E kết nối với nhau qua mạng LAN, chế độ Cầu nối Serial

U2E_A, Socket Type Client

Uart To Ethernet Configuration		
Uart Socket Type	Option	Notes
	<input checked="" type="radio"/> Client <input type="radio"/> Server	
Use DDNS	<input type="checkbox"/>	Current Status: The IP has not changed since the last update.<ok>
DDNS Service	http://noip.info	
Username	YourUserName	
Password	••••••••	
Host	yourdomain.no-ip.info	Ex: uart2ethernet.no-ip.com
Ethernet To UART		
Connect To Server	192.168.1.55	Ex: uart2ethernet.no-ip.info (Internet) hay 192.168.1.200 (LAN)
Port	9999	Valid value range 1000 to 9999
<input type="button" value="Save Config"/>		

COPYRIGHT © 2009 R&P TRADING AND FORWARDING CO. LTD. ALL RIGHTS RESERVED. WEBSITE: HTTP://RPC.VN

Figure 10 U2E_A running like TCP Client, Connect To U2E_B

Server Information	
	Thông tin
IP Address	192.168.1.26
DHCP	<input checked="" type="checkbox"/>
Gateway	192.168.1.1
Subnet Mask	255.255.255.0
Primary DNS Server	192.168.1.1
Secondary DNS Server	0.0.0.0
MAC Address	00:04:A3:25:46:5B
Host name	http://25465B
Using Dynamic DNS	<input type="checkbox"/>
DDNS Service	http://noip.info
DDNS User	YourUserName
Domain	yourdomain.no-ip.info
UART Socket Type	Client
Connect/Listen to	192.168.1.55
UART Baud rate	19200

COPYRIGHT © 2009 R&P TRADING AND FORWARDING CO. LTD. ALL RIGHTS RESERVED. WEBSITE: HTTP://RPC.VN

Figure 11 U2E Current configuration

U2E_B, Socket Type Server

Uart To Ethernet Configuration		
	Option	Notes
UART Socket Type	<input type="radio"/> Client <input checked="" type="radio"/> Server	
Use DDNS	<input type="checkbox"/>	Current Status: The IP has not changed since the last update.<ok>
DDNS Service	http://noip.info	
Username	youraccount	
Password	••••••••	
Host	yourhost.no-ip.info	Ex: uart2ethernet.no-ip.com
Ethernet To UART		
Connect To Server	yourserver.com	Ex: uart2ethernet.no-ip.info (Internet) hay 192.168.1.200 (LAN)
Port	9999	Valid value range 1000 to 9999
<input type="button" value="Save Config"/>		

COPYRIGHT © 2009 R&P TRADING AND FORWARDING CO. LTD. ALL RIGHTS RESERVED. WEBSITE: HTTP://RPC.VN

Figure 12 This U2E_B running like TCP Server and listen in its IP Address is 192.168.1.55

Server Information	
	Thông tin
IP Address	192.168.1.55
DHCP	<input type="checkbox"/>
Gateway	192.168.1.1
Subnet Mask	255.255.255.0
Primary DNS Server	192.168.1.1
Secondary DNS Server	0.0.0.0
MAC Address	00:04:A3:25:9B:BE
Host name	http://259BBE
Using Dynamic DNS	<input type="checkbox"/>
DDNS Service	http://noip.info
DDNS User	youraccount
Domain	yourhost.no-ip.info
UART Socket Type	Server
Connect/Listen to	yourserver.com
UART Baud rate	19200

COPYRIGHT © 2009 R&P TRADING AND FORWARDING CO. LTD. ALL RIGHTS RESERVED. WEBSITE: HTTP://RPC.VN

Figure 13 U2E_B Current Configuration

6. Một số ứng dụng khác

a) Ứng dụng trong giám sát điều khiển

Giả sử chúng ta cần một hệ thống giám sát và điều khiển, hệ thống này hoạt động với khả năng cập nhật dữ liệu và điều khiển liên tục. Thông tin thu thập được có thể truy cập từ nhiều hướng, nhiều kết nối. Mô hình của nó như sau.

Mỗi một U2E sẽ là một TCP Client, kết nối đến Server và truyền/nhận tất cả những thông tin từ Thiết bị giám sát/điều khiển đến Server.

Mỗi máy tính PC có thể kết nối với Server thông qua bất kỳ giao thức nào, HTTP, TCP, Telnet... để điều khiển quá trình, monitor hệ thống ...

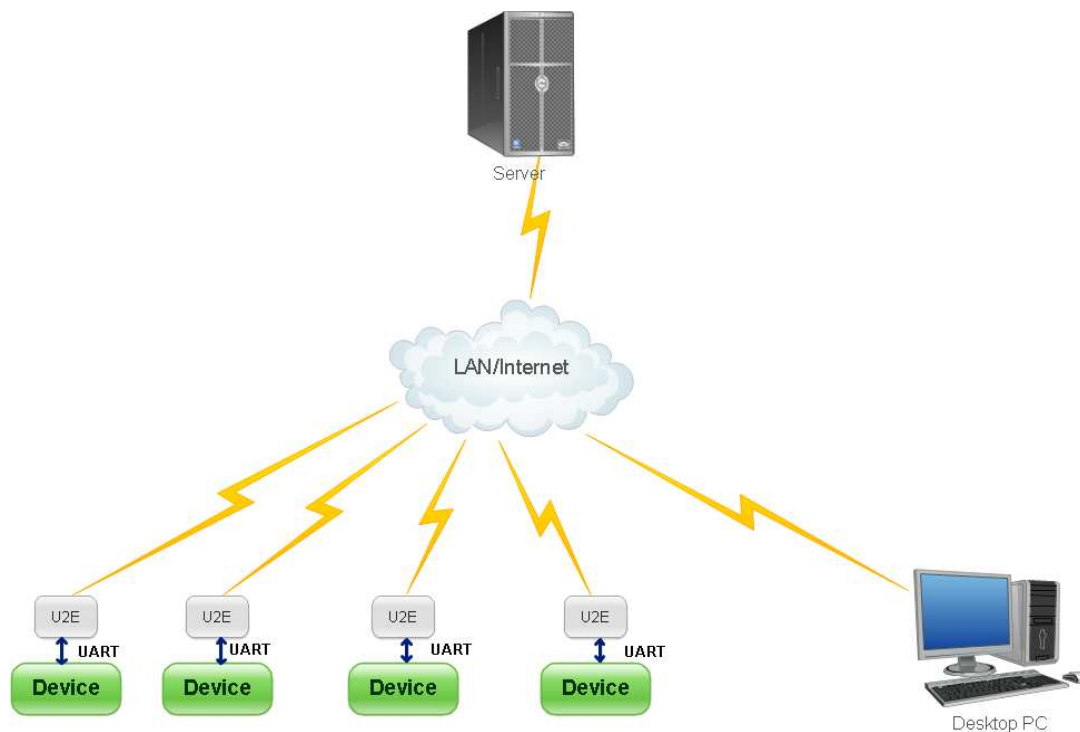
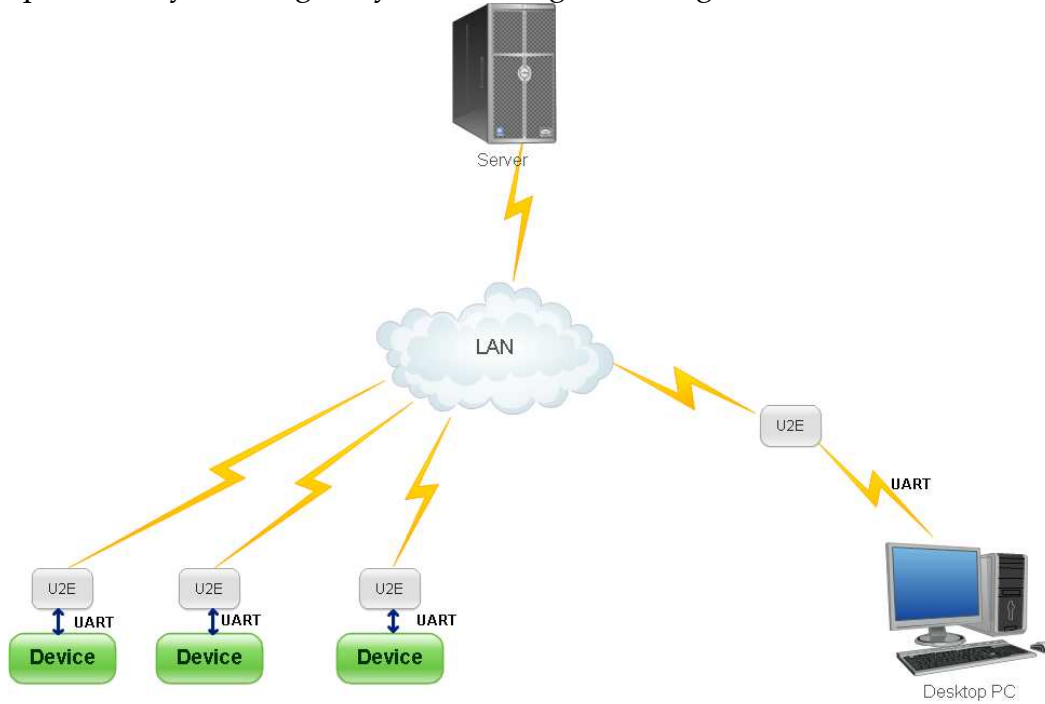


Figure 14 Mô hình dùng U2E giám sát điều khiển

b) Ứng dụng trao đổi dữ liệu trong mạng LAN

Tương tự như ứng dụng trên, tuy nhiên PC kết nối với LAN thông qua 1 U2E, đơn giản trong lập trình truyền thống, thay thế dễ dàng cho mạng RS485 cũ



c) Ứng dụng lấy thông tin từ Internet

Giả sử một ứng dụng cụ thể là Quang báo, việc cập nhật thông tin cho bảng chứng khoán hay Ngân hàng cho hàng chục Bảng Quang báo một lúc thực hiện rất đơn giản với U2E.

Mô hình của nó như sau:

Thiết bị điều khiển Quang báo kết nối truyền nhận với U2E qua UART, WWW Server chứa thông tin được trích xuất từ những Website khác

PC có thể được sử dụng để điều khiển định hướng dữ liệu cho U2E, nâng cấp phần mềm trên Server (yourserver.com). Hoặc dùng để điều khiển gián tiếp U2E thông qua Server.

Cấu hình cho WWW Server:

Cần hiểu biết về một trong những Server Script, để dùng những script này tách lọc thông tin từ các Server khác và gởi về U2E. Cần một domain để trở tới WWW Server này (có thể sử dụng IP)

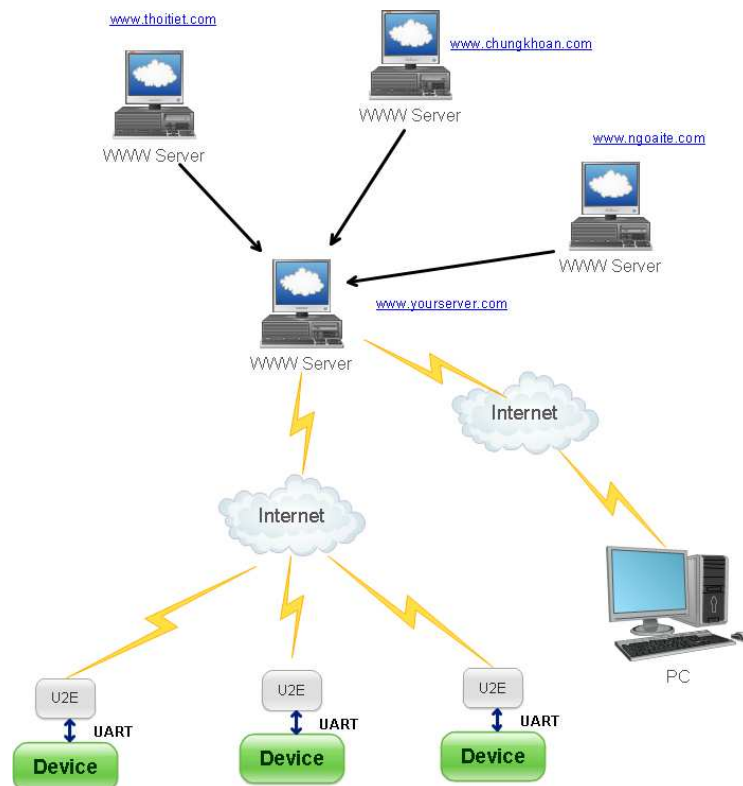


Figure 15 Mô hình lấy thông tin từ Internet

d) Ứng dụng trong truyền nhận dữ liệu khoảng cách xa

Những ứng dụng trước đây của bạn truyền UART với khoảng cách gần, tuy nhiên, do yêu cầu nào đó bạn cần tăng khoảng cách kết nối, với một số lựa chọn thì RS485 hàng đầu, tuy nhiên với một khoảng cách khá xa thì U2E có lẽ là sự lựa chọn tốt nhất, chỉ với 2 U2E, một cái cho mỗi thiết bị, firmware cũ của bạn không cần thay đổi, thay vì kết nối trực tiếp từ thiết bị A sang thiết bị B, bạn chỉ cần nối thiết bị A với U2E, thiết bị B với U2E và một số cấu hình đơn giản cho U2E thì khoảng cách kết nối của bạn là Internet.

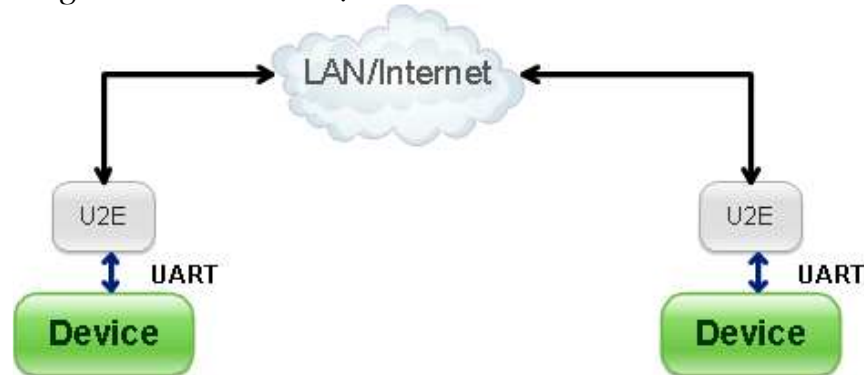


Figure 16 Mô hình truyền nhận dữ liệu khoảng cách xa
Cấu hình cho U2E như mục 7 - Ứng dụng U2E

7. Mạch ví dụ

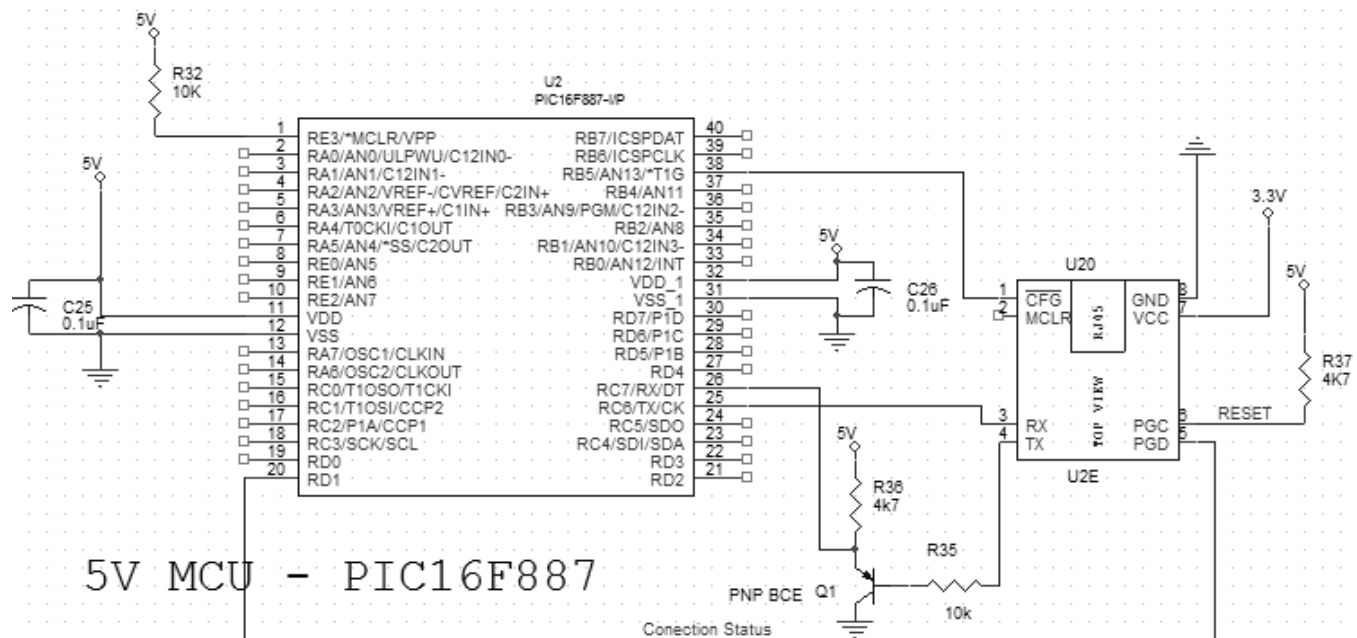


Figure 17 Kết nối với PIC16F887

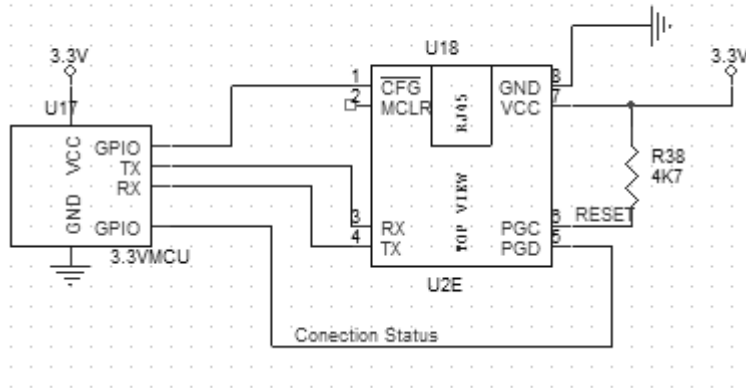


Figure 18 Kết nối với MCU hoạt động ở 3.3V

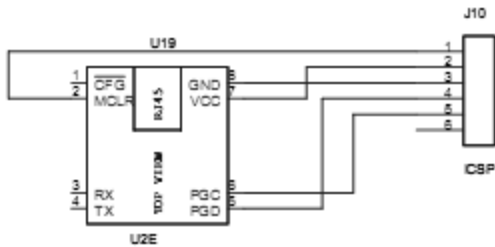
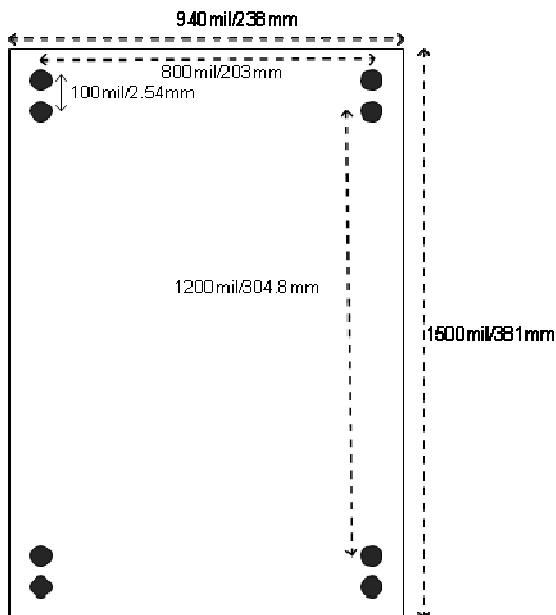


Figure 19 Nạp lại cho PIC18F67J60/PIC18F66J65 trong U2E

8. Kích thước Board



9. Phiên bản

Phiên bản hiện tại U2E Rev.02, những thay đổi như sau:

- + Cấu hình thông qua UART bằng frame dữ liệu định nghĩa như trên
- + Chân số 1 – CFG chỉ có 1 chức năng duy nhất – vào mode cấu hình

- + Chân số 6 – PGC/RESET, chức năng RESET lại cấu hình mặc định khi giữ ở mức thấp trong vòng 5s lúc khởi động, **bình thường phải được kéo lên mức 1 bằng 1 điện trở kéo lên.**
- + Thêm module ICMP, có thể PING U2E trong mạng

10. Code mẫu (PIC16F887 – Trình biên dịch HiTech PIC C)

❖ Cấu hình U2E Client, kết nối tới Server có IP: 192.168.1.10

```
#include<htc.h>

__CONFIG(HS & WDTDIS & UNPROTECT & LVPDIS & BORDIS);
// Clock 20Mhz

#define      U2ESTAT          RB1
#define      U2ECFG          RB0
#define      U2ERESSET       RB2

char getch();
void putch(char c);
void PutFrame(const char *f);
char GetFrame(char *f);
char IsU2EConfig();
char FrmRecv[48];

/*Write U2E Socket Type, Client*/
const char FrmU2EServ[] = {0xBA, 0x02, 0x8E, 0x00}; //0xBA: from PIC to U2E

    //0x02: Data len, containt CMD

    //0x0E: Command set U2E Socket Type

    //0x00: Client Type
/*Read current config U2E Socket Type*/
const char ReadU2EServ[] = {0xBA, 0x01, 0x0E};

/* Write Remote Server address U2E will be connect to, 192.168.1.10 - IP or domain as
string*/
const char FrmU2ERemoteServ[] = {0xBA, 0x0D, 0x8F,
'1', '9', '2', '.', '1', '6', '8', '.', '1', '.', '1', '0'};

    //0x0D: Data len, containt CMD

    //0x8F: Command set U2E Remote Server will be connect

    //Data: Ip Address as string: 192.168.1.11
/* Read current config Server address*/
const char ReadU2ERemoteServ[] = {0xBA, 0x01, 0x0F};

/* Write Server Remote Port Format HighByte:LowByte */
const char FrmU2ERemotePort[] = {0xBA, 0x03, 0x91, 0x27, 0x0F };

    //0x91: Command set U2E Remote Port

    //Data: Port: 0x270F = 9999
/*Read current config Server Remote Port */
const char ReadU2ERemotePort[] = {0xBA, 0x01, 0x11};

/* Command Save and Reboot, do it after send command config*/
```

```

const char SaveReboot[] = {0xBA, 0x01, 0x12};

void main() {
    char c;
    /* Hardware initialization */

    TRISC6 = 0;           // UART TX pin
    TRISC7 = 1;           // UART RX pin
    TRISB = 0;            // PORTB Output
    TRISB1 = 1;           // And config only RB2 as input
    ANSEL = 0;            //All PIN as Digital IO
    ANSELH = 0;

    SYNC=0;               //asynchronous
    SPEN=1;               //enable serial port pins
    CREN=1;               //enable reception
    SREN=0;               //no effect

    TX9=0;                //8-bit transmission
    RX9=0;                //8-bit reception
    TXEN=0;               //reset transmitter
    TXEN=1;               //enable the transmitter
    BRG16 = 1;            //High speed baudrate
    BRGH = 1;
    SPBRG = 3;
    SPBRGH = 1;           //19200
    RCIF = 0;             //Clear RX Flag
    RCIE = 0;             //Disable interrupts
    PEIE = 0;
    GIE = 0;

    U2ECFG = 1;           //Set U2E Config pin High
    U2ERESSET = 1;        //Set U2E Reset pin High (Low(0) 5s when U2E start
will load U2E default setting)

    _delay(100000);       //wait for u2e ready

    //check if U2E was not configuration, config it
    if(!IsU2EConfig()) {

        /* Config U2E Socket type, Client */
        FrmRecv[0] = 0xFF;
        while(FrmRecv[0] != 0x00) {
            PutFrame(FrmU2EServ); //Put frame config U2E Socket Type Client
            while(!GetFrame(FrmRecv)); //Read Response Frame with Status
        }

        /* Config U2E Remote Port 9999 = 0x270F*/
        FrmRecv[0] = 0xFF;
        while(FrmRecv[0] != 0x00) {
            PutFrame(FrmU2ERemotePort); //Put frame config U2E Remote Port
            while(!GetFrame(FrmRecv)); //Read Response Frame with Status
        }

        /* Config U2E Remote Server Address*/
        FrmRecv[0] = 0xFF;
        while(FrmRecv[0] != 0x00) {
            PutFrame(FrmU2ERemoteServ); //Config U2E Remote Server Address
            while(!GetFrame(FrmRecv)); //Read Response Frame with Status
        }

        PutFrame(SaveReboot); //Put frame config U2E Remote Server Address
        _delay(10000);
    }
}

```

```
        /* Save and reboot to load new setting */
    }

    /* Config PIN High to enter operation mode */
    U2ECFG = 1;
    U2ERESSET = 1;
    /* wait until U2E Connected to Server */
    while(!U2ESTAT);

    while(1) {
        c = getch();
        if(c > 0x60 && c<0x7B) {
            c-= 0x20;    //convert to UPCASE
            putchar(c);
        }
    }
}

/* get 1 byte from uart */
char getch() {
    while(!RCIF);
    while(!RCIF);
    return RCREG;
}

/* put 1 byte from uart */
void putchar(char c) {
    while(!TXIF);
    TXREG = c;
}

void PutFrame(const char *f) {
    unsigned char len;
    U2ECFG = 1;    // Pulse CFG PIN
    U2ECFG = 0;    //Low to enter config mode
    _delay(10);
    for(len = 0; len < *(f+1)+2; len++){
        while(!TXIF);
        TXREG = *(f+len);
    }
}

char GetFrame(char *f) {
    static unsigned char gState = 0, frmLen;
    unsigned char ret = 0, i;
    switch(gState) {

        //header, must 0xBD
        case 0:
            if(getch() == 0xBD) {
                gState ++;
            }
            break;

        //Data Len, must more than 1
        case 1:
            frmLen = getch();
            if(frmLen > 0)
                gState ++;
            else gState = 0;
            break;

        //Data receive, contain status
        case 2:
            for(i=0; i<frmLen; i++) {
```



```
                *(f+i) = getch();
            }
            ret = 1;
            gState = 0;
            break;
        }
        return ret;
    }
}
char IsU2EConfig() {
    unsigned char i;
    /******
    //Read U2E Socket Type
    FrmRecv[0] = 0xFF;
    while(FrmRecv[0] != 0x00) {
        PutFrame(ReadU2EServ);
        //Read Response Frame with Status
        while(!GetFrame(FrmRecv));
    }
    //check if Socket Type Server, break
    if(FrmRecv[1]) return 0;

    /******
    //Read U2E Socket Type
    FrmRecv[0] = 0xFF;
    while(FrmRecv[0] != 0x00) {
        PutFrame(ReadU2ERemoteServ);
        //Read Response Frame with Status
        while(!GetFrame(FrmRecv));
    }

    for(i=1; i<13; i++) {
        if(FrmRecv[i] != FrmU2ERemoteServ[i+2]) return 0;
    }

    /******
    FrmRecv[0] = 0xFF;
    while(FrmRecv[0] != 0x00) {
        PutFrame(ReadU2ERemotePort);
        //Read Response Frame with Status
        while(!GetFrame(FrmRecv));
    }
    if(FrmRecv[1] != 0x27 || FrmRecv[2] != 0x0F) return 0;

    return 1;
}
```

❖ Gởi yêu cầu tới một Website và hiển thị thông tin ra LCD (Cập nhật sau)