

EM-485NET

V1.0 (2019)

Network (LAN-RJ45) to RS485



EM-485NET คือบอร์ดแปลงจาก Network แบบสาย Lan (RJ45) มาเป็น RS485 หรือ RS232-TTL พร้อมทั้งมีขั้ว I/O ต่าง ๆ ให้ใช้งานได้ด้วย มี LED แสดงสถานะการทำงาน และมีปุ่มกด 2 ตัวบนบอร์ด เพื่อการตั้งค่าพิเศษ ด้าน Lan จะสื่อสารด้วย TCP ทั้งแบบ Full คือมีการเชื่อมต่อระหว่าง Client ไปยัง Server ก่อน แล้วจึงส่งข้อความไปให้ Server และรอ Server ตอบกลับมายัง Client อีกที จึงจะถือว่าจบการสื่อสาร หรือแบบ Half คือมีการเชื่อมต่อระหว่าง Client ไปยัง Server จากนั้นก็ส่งข้อความไปให้ Server แล้วถือว่าจบการสื่อสารเลย EM-485NET มีโหมดรองรับหลากหลายดังนี้

โหมด SAC (Smart Ascii Command) ... ใช้เพื่อการตั้งค่า Config ต่าง ๆ ผ่านทาง RS485 ร่วมกับโปรแกรม Terminal ขณะเดียวกันก็ใช้เป็น Client เพื่อการเชื่อมต่อกับ Cloud แบบ Http (Web-Page) ได้ด้วย

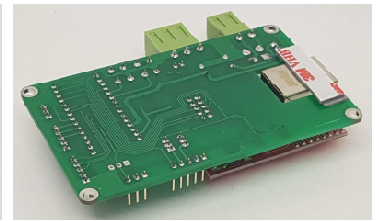
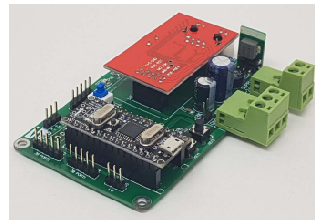
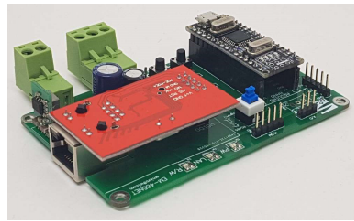
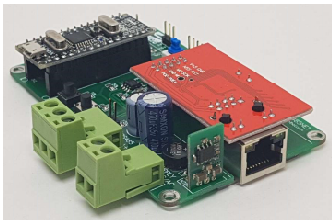
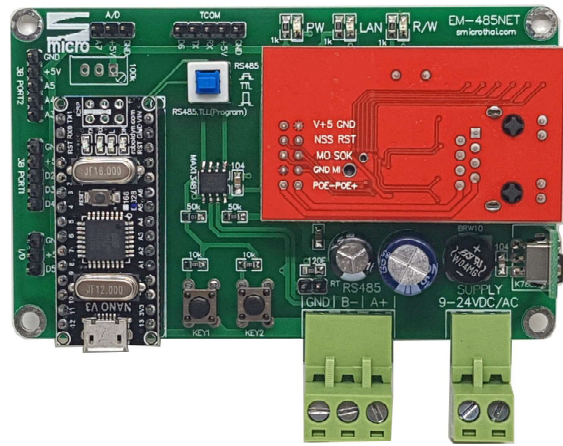
โหมด Client/Server หรือ Client หรือ Server คือเป็นตัวแปลง Lan เป็น RS485 นั้นเอง เสมือนเชื่อมต่ออุปกรณ์ RS485 เข้าด้วยกันโดยไม่ต้องเดินสายจริง แต่ผ่าน Lan แทน ทั้งนี้จะต้องสื่อสารในรูปแบบ Ascii ที่ลงท้ายด้วย \r เท่านั้น

โหมด Server-I/O คือเป็นเหมือน I/O Node ในระบบ Lan ที่มี Digital I/O 8 Port ให้ใช้งานได้ และยังมี Analog-Input (A/D 10 Bit) อีก 1 Port พร้อมทั้งมองปุ่มกด 2 ตัวเป็น Input ได้ อีกทั้งยังส่งข้อมูลออกทาง RS485 ได้ด้วย

คุณสมบัติ

- MCU แบบโมดูล Arduino-Nano (ATmega328 Clock 16 MHz)
- มีขั้ว RS485 (A,B,Gnd) และขั้ว RX,TX แบบ TTL (5 Pin) เลือกลงใช้งานด้วยปุ่มกด (สีฟ้า)
- คุณสมบัติการสื่อสารคือ Speed=9600 Parity=None Data=8 Stop=1
- มี LED แสดงการสื่อสาร (สีเหลือง) คือ LAN และ RW สำหรับด้าน RS485
- มีขั้ว Digital I/O 3B-Port (5 Pin) = 2 และ I/O-Port (3 Pin) = 1 และ I/O-Port อยู่กับขั้ว Tcom (232-TTL)
- มีขั้ว Analog Input แบบ A/D 10 Bit (3 Pin) = 1
- มีชุดคำสั่ง 6 คำสั่ง ในโหมด SAC สำหรับการตั้งค่าและดูสถานะต่าง ๆ ใช้งานผ่านโปรแกรม Terminal
- มีชุดคำสั่ง 5 คำสั่ง ในโหมด SIO สำหรับการสั่งงานผ่าน Computer
- การเชื่อมต่อ Lan รองรับการทำงานรับค่า IP แบบ DHCP หรือจะตั้งค่าแบบ Fix IP ก็ได้
- ใช้ไฟเลี้ยง 9-24VDC/AC กินกระแส 150 mA
- ขนาดบอร์ด 97 x 60 mm (ไม่รวมส่วนยื่นของขั้วไขว้น็อต)

ภาพแสดงบอร์ด



การใช้งานพื้นฐาน

- 1> เมื่อจ่ายไฟเลี้ยงเข้าบอร์ด LED สีเหลืองทั้ง 2 ดวง (LAN,R/W) จะกะพริบ 2 ครั้ง จากนั้นจะทำการตั้งคุณสมบัติทางด้าน LAN และเชื่อมต่อ ถ้าทุกอย่างเรียบร้อย LED-LAN ก็จะสว่าง และจะไปทำงานตามโหมดที่ตั้งไว้ทันที
- 2> กรณีที่ตั้ง Fix IP ตัว LED-LAN จะสว่าง และพร้อมทำงาน ถึงแม้จะยังไม่ได้เสียบสาย Lan ก็ตาม
- 3> ในช่วงเปิดเครื่อง ถ้ากดปุ่ม Key1 หรือ Key2 ค้างไว้ก่อน จะเป็นการเลือกเข้าสู่โหมด SAC โดยตรง ไม่ว่าจะตั้งโหมดใด ๆ ไว้ก็ตาม เพื่อให้เข้าถึงการตั้งค่าได้อย่างสะดวก ดังนี้

Key1 ... เข้าสู่โหมด SAC

Key2 ... ทำการล้างค่าตัวแปรต่าง ๆ เป็นค่า Default และเข้าสู่โหมด SAC

- 4> ในการทำงานปกติ LED-R/W จะกะพริบตามจังหวะที่มีการสื่อสาร ทั้งทางด้าน RS485 หรือ Lan
- 5> ปุ่มกดสีฟ้า ... กดลง = ใช้งาน RS485 ทางขั้วเขียว ... กดขึ้น = ใช้งาน RS232-TTL ทางขั้ว 5 Pin
- 6> ปุ่มเล็ก ๆ สีดำบนโมดูล ... กดเพื่อ Reset ระบบ และเริ่มทำงานใหม่
- 7> VR100K ต่อกับ A7 (Analog A/D) จะเป็น Option กรณีต้องการใช้ก็สามารถใส่เพิ่มเติมได้

ทำความเข้าใจกับ TCP กับ EM-485NET

TCP คือรูปแบบการสื่อสารที่นิยมมากในโลกของ Network ทั้งหลาย ไม่ว่าจะเป็นแบบภายใน Local หรือแบบ Cloud (Internet) ที่เชื่อมกันทั่วโลก หลักการเชื่อมต่อของ TCP ระหว่าง Client กับ Server จะเป็นดังนี้

- 1> Client เชื่อมต่อไปยัง Server ตามหมายเลข IP และ Port
- 2> Client ส่งข้อมูลไปให้ Server
- 3> Server ส่งข้อมูลตอบกลับมายัง Client
- 4> Client จบการสื่อสาร

EM-485NET จะใช้หลักการ TCP นี้ทั้งแบบ Full และแบบ Half คือตัดขั้นตอนที่ (3) ออกไป คือการที่ Server ตอบกลับมายัง Client เพื่อไม่ให้เกิดจังหวะการรอนั่นเอง ผสมผสานกับโหมดการใช้งานต่าง ๆ ซึ่งทำให้เกิดการประยุกต์ใช้งานได้หลากหลาย สรุปได้จากทั้ง 2 ตารางต่อไปนี้

TCP-Full

โหมด	คำสั่ง	สื่อสาร	สถานะ	ตัวอย่างประยุกต์ใช้งาน
SAC	:T :S	Http	Client	# ใช้เป็นตัวลูกส่งข้อมูลเข้าสู่ Web-Server
CLI		Ascii	Client	# ใช้แปลง Lan-RS485 เชื่อมกับอีกตัวที่ตั้งเป็น SER # ใช้แปลง Lan-RS485 จากหลายตัว ไปยังอีกตัวที่ตั้งเป็น SER หรือเข้า Computer
SER		Ascii	Server	# ใช้แปลง Lan-RS485 เชื่อมกับอีกตัวที่ตั้งเป็น CLI # ใช้ Computer เชื่อมต่อกับอุปกรณ์ RS485 ได้หลายตัว โดยส่งข้อมูลแบบไปกลับ
SIO		Ascii	Server	# ใช้ Computer เชื่อมต่อกับ I/O ของบอร์ดได้หลายตัว

TCP-Half

โหมด	คำสั่ง	สื่อสาร	สถานะ	ตัวอย่างประยุกต์ใช้งาน
SAC	:A	Ascii	Client	# ใช้ทดสอบการสื่อสารผ่าน Lan
SAC	=xxx	Ascii	Server	# ใช้ทดสอบการสื่อสารผ่าน Lan
C/S		Ascii	Client Server	# ใช้แปลง Lan-RS485 เชื่อมกับอีกตัวที่ตั้งเป็น C/S # ใช้แปลง Lan-RS485 จากหลายตัว ส่งข้อมูลเข้า Computer # ใช้ Computer ส่งข้อมูลไปยังอุปกรณ์ RS485 ได้หลายตัว

โหมด SAC (Smart Ascii Command)

ในโหมดนี้ EM-485NET รับคำสั่งทาง RS485 (ขั้วเขียว) หรือ RS232-TTL (ขั้ว 5 Pin) โดยจะเชื่อมต่อกับบอร์ดไมโครคอนโทรลเลอร์ใดๆ คุณสมบัติคือ Speed=9600 Data=8 Stop=1 Parity=No ชุดคำสั่งที่ใช้จะเป็นแบบ Ascii Command โดยเมื่อส่งคำสั่งไปอย่างถูกต้อง เครื่องก็จะตอบกลับมาเสมอ สามารถทดลองคำสั่งได้โดยใช้โปรแกรมสื่อสารแบบ Terminal ที่ทำงานบนเครื่องคอมพิวเตอร์ เพราะมีรูปแบบเป็นตัวอักษร Ascii ทั้งหมด คำสั่งจะมีรูปแบบดังนี้...

:clr

: คือรหัสนำหน้า (0x3A)

c คือคำสั่งเป็นตัวเลข หรือตัวอักษรตัวเดียว และอาจมีหรือไม่มีข้อมูลติดตามมาก็ได้

\r คือรหัสลงท้าย (0x0D)

ชุดคำสั่งจะสรุปได้ตามตารางต่อไปนี้ ในวงเล็บ (R) หมายถึงคำสั่งที่ส่งไป และ (T) คือข้อมูลที่ตอบกลับ โดยจะแสดงเป็นตัวอักษรสมมุติเพื่อให้เข้าใจ ตัวคำสั่งจะแสดงแบบไม่ใส่รหัสลงท้าย \r เพื่อให้ดูสบายตา ส่วนค่าภายใน [] ต่อท้ายคำสั่ง หมายถึง Option คือใส่หรือไม่ใส่ก็ได้ ซึ่งถ้าใส่จะให้ความหมายทำงานของ “เขียนค่า” (write) และการไม่ใส่คือ “อ่านค่า” (read) ส่วนการตอบกลับก็จะแยกเป็น 2 กรณีด้วย คือแบบปกติ กับภายใน [] อีกที

รูปแบบคำสั่ง	การใช้งาน
(R) :Z (T) EM-485NET vX.X	Check (Show Model) แสดงรุ่นของสินค้าและหมายเลข Version ของ Firmware
(R) :W (T) 192.168.4.2 192.168.4.1 255.255.255.0	Work Status คือแสดงสถานะของ Lan คือ IP, Gateway และ Subnet Mask
(R) :T (T) ข้อความมากมาย ...	Test Cloud สำหรับทดสอบการเชื่อมต่อกับโลกของ Cloud (Internet) โดยจะเชื่อมไปยัง www.smicrothai.com และมีคำว่า OK-Check ตอบกลับมา ข้อความต่าง ๆ ที่แสดง จะเป็นการสื่อสารที่เกิดขึ้นจริงในรูปแบบ Http Protocol โดยเพิ่มอักขรนำ # สำหรับข้อความที่ส่งไปยัง Server และ = สำหรับข้อความที่ Server ตอบกลับมา เพื่อแยกแยะให้ดูง่ายขึ้น
(R) :Sxxx (T) ข้อความมากมาย ... หรือ ER,NH,NP,CE	Send (Http) to Cloud ... xxx คือข้อความใด ๆ สำหรับส่งข้อมูลไปยัง Cloud (Internet) ทำนองเดียวกับคำสั่ง :T โดยสามารถกำหนดข้อความได้ตามต้องการ และทั้งนี้จะต้องตั้งตัวแปรในคำสั่ง :C ด้วย คือตัวแปร Host และ Port ... ทำความเข้าใจกับตัวอย่างนี้ ตั้งค่าตัวแปร Host=www.smicrothai.com และ Port=80 และใช้คำสั่งคือ :s/_link/test.php?a=4 ซึ่งสุดท้ายจะได้ข้อความว่า Data From Server = 20 (คือ Server รับตัวเลข 4 นำไปคูณกับ 5 และส่งค่า 20 กลับมาให้)

รูปแบบคำสั่ง

การใช้งาน

(R) :Axxx Send Ascii ... xxx คือข้อความใด ๆ
 (T) ข้อความตอบกลับ ... สำหรับส่งข้อมูลไปเป้าหมาย โดยอาจจะเป็น Cloud (Internet) หรือ Network ภายใน
 หรือ ER,NH,NP,CE ก็ได้ เป็นข้อความ Ascii ลงท้ายด้วย \r (เป็น Protocal อีสระ) และทั้งนี้จะต้องตั้งตัวแปร
 ในคำสั่ง :C ด้วย คือตัวแปร Host และ Port ด้วย

คำสั่ง :A จะคล้าย ๆ กับการทำงานในโหมด C/S (Client/Server) นั่นเอง คือถ้าทดสอบ
 ด้วยคำสั่ง :A แล้วทำงานได้ปกติ ก็จะใช้งานในโหมด C/S ได้ด้วย ในโหมด SAC นี้เมื่อ
 มีข้อความส่งเข้ามา ก็จะได้รับข้อความเหล่านั้น และส่งออกมาทาง RS485 ด้วยเช่นกัน โดย
 จะนำหน้าด้วย = เพื่อให้ทราบว่ามาจากภายนอก ทำความเข้าใจจากตัวอย่างนี้

ใช้ EM-485NET จำนวน 2 ตัวกำหนด Fix IP เป็น 192.168.1.1 และ 192.168.1.2
 และตั้งตัวแปร Host ของแต่ละตัวให้สลับกัน (เช่น ตัวที่มี IP 192.168.1.1 ให้ตั้ง Host
 เป็น 192.168.1.2) ส่วนตัวแปร Port ให้ตั้งเป็น 1470 ไว้ ... เข้าใช้งานในโหมด SAC
 แล้วใช้คำสั่ง :Atest-hello จากตัวหนึ่ง ข้อความ =test-hello ก็จะไปปรากฏที่อีกตัวหนึ่ง
 นั้นแสดงว่าทั้ง 2 ตัวสามารถสื่อสารกันได้แล้ว

ความหมายการตอบกลับของคำสั่ง :S และ :A
 ER = Error (ไม่ได้ใส่ข้อความต่อท้าย)
 NH = No-Host (ไม่ได้กำหนดตัวแปร Host ไว้)
 NP = No-Port (ไม่ได้กำหนดตัวแปร Port ไว้)
 CE = Connection Error (ไม่สามารถเชื่อมต่อกับ Host ได้)

(R) :Cn[xx...x] Config สำหรับดูและตั้งค่าตัวแปรต่าง ๆ
 (T) <xx...x>[OK] หมายเลข n คือค่า 0-6 ถ้าใส่ [xx...x] จะหมายถึงตั้งค่าตามหมายเลขนั้น ๆ
 แต่ถ้าไม่ใส่ ก็หมายถึงขอดูค่านั้น ๆ ผู้ใช้สามารถใช้เพียงแค่ :C เพื่อดูค่า
 ทุกหมายเลขได้ในคราวเดียวเลย รายละเอียดของแต่ละหมายเลขเป็นดังนี้

0<222.173.190.239.254.201> ... Mac Address (6 Byte)
 1<192.168.1.2> ... IP Address
 2<192.168.1.1> ... Gateway
 3<255.255.255.0> ... Subnet Mask
 4<www.smicrothai.com> ... Host
 5<80> ... Port
 6<sac> ... Mode (SAC,C/S,SIO,CLI,SER)

ค่าใน < > คือตัวแปรที่ตั้ง ส่วนข้อความหลักจาก ... คือความหมายของมัน กรณีต้องการ
 ลบค่าใด ๆ สามารถใส่เครื่องหมาย - ได้ เช่น :C4- หมายถึงลบค่า Host ออกไป ... ตัวแปร
 IP,Gateway และ Subnet ทั้ง 3 ตัวนี้ ถ้าไม่ใส่เลย จะหมายถึงให้ดึง IP จากระบบโดย
 อัตโนมัติ (DHCP) ... ตัวแปร Mac Address ถ้าใช้งาน EM-485NET หลาย ๆ ตัว จะ
 ต้องตั้งไม่ให้ซ้ำกันเลย แนะนำให้เปลี่ยนตัวท้ายสุดจาก 201,202,203 ไปเรื่อย ๆ ถึง 254

โหมด C/S (Client/Server) โหมด CLI (Client) โหมด SER (Server)

โหมด C/S นี้จะทำหน้าที่เหมือนเป็นตัวแปลง Lan เป็น RS485 โดยเป็นแบบ TCP-Half และทำงานผสมผสานกันระหว่างเป็น Client และ Server ไปพร้อมกัน วิธีการนี้ทำให้ไม่ต้องเกิดจังหวะรอคำตอบจาก Server ลองทำความเข้าใจกับตัวอย่างต่อไปนี้ ซึ่งเป็นการใช้งาน แบบ Point-to-Point

กำหนดให้ A คือ EM-485NET ที่มี Fix IP=192.168.1.1 และตั้ง Host=192.168.1.2 และ Port=1470

กำหนดให้ B คือ EM-485NET ที่มี Fix IP=192.168.1.2 และตั้ง Host=192.168.1.1 และ Port=1470

เมื่อมีข้อมูล Ascii ลงท้ายด้วย \r มายัง RS485 ของ A

A (Client) จะทำการเชื่อมต่อไปยัง B (Server) แล้วส่งข้อมูลให้ (โดยไม่รอตอบกลับ)

B ได้รับข้อมูลแล้วก็จะส่งต่อออกไปยัง RS485

กรณี B ต่อ RS485 กับอุปกรณ์ใด ๆ ที่มีการตอบกลับ

อุปกรณ์เหล่านั้นก็จะตอบกลับมาเป็น Ascii ลงท้ายด้วย \r มายัง B

B (Client) จะทำการเชื่อมต่อไปยัง A (Server) แล้วส่งข้อมูลให้ (โดยไม่รอตอบกลับ)

A ได้รับข้อมูลแล้วก็จะส่งต่อออกไปยัง RS485 ต่อไป

สำหรับการใช้งานแบบ Computer-to-MultiPoint หรือ MultiPoint-to-Computer ก็จะเป็นไปในทำนองเดียวกัน โดยที่ EM-485NET ทุกตัวจะตั้ง Host ไปยังหมายเลข IP ของ Computer ส่วนทางด้าน Computer ก็สามารถเลือกสื่อสารกับแต่ละตัวตาม Fix IP ได้อยู่แล้ว วิธีนี้สามารถประยุกต์ใช้ทั้งการสื่อสารแบบไปกลับ หรือแบบด้านเดียวจะเป็นขาไป หรือขากลับก็ได้

อย่างไรก็ตาม บางครั้งลักษณะการเขียนโปรแกรมบน Computer หรือเกี่ยวกับลักษณะของการสื่อสารทาง RS485 ที่แตกต่างกันไป อาจจำเป็นต้องใช้งานแบบ TCP-Full ไปเลย ซึ่งก็สามารถเลือกใช้เป็นโหมด CLI หรือ SER ก็ได้ตามความเหมาะสม ทั้งนี้ตัว EM-485NET ก็จะทำหน้าที่เป็น Client หรือ Server เพียงแบบเดียว ลองทำความเข้าใจกับตัวอย่างต่อไปนี้ ซึ่งเป็นการใช้งาน แบบ Point-to-Point (ข้อกำหนดเหมือนตัวอย่างที่แล้ว)

เมื่อมีข้อมูล Ascii ลงท้ายด้วย \r มายัง RS485 ของ A ซึ่งตั้งโหมดเป็น CLI (Client)

A จะทำการเชื่อมต่อไปยัง B ซึ่งตั้งโหมดเป็น SER (Server) แล้วส่งข้อมูลให้

B ได้รับข้อมูลแล้วก็จะส่งต่อออกไปยัง RS485

กรณี B ต่อ RS485 กับอุปกรณ์ใด ๆ ที่มีการตอบกลับ (B จะรอไม่เกิน 1 วินาที)

อุปกรณ์เหล่านั้นก็จะตอบกลับมาเป็น Ascii ลงท้ายด้วย \r มายัง B

B จะส่งข้อมูลกลับไปให้ A

A ได้รับข้อมูลแล้วก็จะส่งกลับออกไปยัง RS485 ต่อไป (A จะรอไม่เกิน 1 วินาทีเช่นกัน)

โหมด SIO (Server I/O)

โหมด SIO นี้จะเป็นเหมือน I/O Node ในระบบ Lan ที่มี Digital Port ต่าง ๆ ให้เลือกใช้งานได้ โดยเขียนสั่งการมาจาก Computer อื่นๆ หรืออาจจะมาจาก EM-485NET อีกตัวที่หน้าที่เป็น Client ก็ได้ และเขียนสั่งการด้วย Controller ผ่านทาง RS485 อีกที ชุดคำสั่งที่มาทาง Lan จะเป็นแบบ Ascii Command โดยเมื่อส่งคำสั่งไปอย่างถูกต้อง เครื่องก็จะตอบกลับมาเสมอ มีรูปแบบเป็นตัวอักษร Ascii ทั้งหมด คำสั่งจะมีรูปแบบดังนี้...

```
cc[xxx]\r
```

cc คือคำสั่งอักษร 2 หลัก

[xxx] คือ Option ข้อมูลติดตาม ซึ่งอาจจะมีหรือไม่มีตามมาก็ได้

\r คือรหัสลงท้าย (0x0D)

ชุดคำสั่งจะสรุปได้ตามตารางต่อไปนี้ ในวงเล็บ (R) หมายถึงคำสั่งที่ส่งไป และ (T) คือข้อมูลที่ตอบกลับ โดยจะแสดงเป็นตัวอักษรสมมุติเพื่อให้เข้าใจ ตัวคำสั่งจะแสดงแบบไม่ใส่รหัสลงท้าย \r เพื่อให้ดูสบายตา ส่วนค่าภายใน [] ต่อท้ายคำสั่ง หมายถึง Option คือใส่หรือไม่ใส่ก็ได้ ซึ่งถ้าใส่จะให้ความหมายทำนอง “เขียนค่า” (write) และการไม่ใส่คือ “อ่านค่า” (read) ส่วนการตอบกลับก็จะแยกเป็น 2 กรณีด้วย คือแบบปกติ กับภายใน [] อีกที

รูปแบบคำสั่ง	การใช้งาน
(R) ZZ (T) EM-485NET vX.X	Check (Show Model) แสดงรุ่นของสินค้าและหมายเลข Version ของ Firmware
(R) RW[xxxxyyyzz] (T) xxxxyyyzz[OK,ER]	Read/Write I/O อ่านหรือเขียนค่าให้กับ I/O Port คือค่า 0 หรือ 1 xxx คือ 3B-Port A5 A4 A3 yyy คือ 3B-Port D2 D3 D4 zz คือ I/O-Port D5 และที่ขั้ว Tcom คือ D6
(R) RA (T) nnnn	Read A/D อ่านค่าจาก Analog to Digital (A/D) จากขั้ว A7 nnnn ตัวเลขที่อ่านได้ตั้งแต่ 0-1023
(R) WSxxx (T) OK	Write Serial ส่งข้อมูลออกทาง RS485 หรือ Tcom (TX) xxx คือข้อความเป็น Ascii ที่ลงท้ายด้วย \r
(R) RK (T) kk	Read Key อ่านค่าจาก Key1 และ Key2 บนบอร์ด โดยเป็น 1 ถ้าไม่ได้กด และเป็น 0 เมื่อกด