



ET-MINI CNT6 เป็นชุด Digital Programmable Counter ขนาดเล็ก โดยมีจอแสดงผลเป็นตัวเลขแสดงผลแบบ 7 Segment ขนาด 0.56 นิ้ว 7 หลักจำนวน 3 แถว สามารถนับและจดจำค่าการนับและคำนวณค่าผลต่างที่เกิดจากการนับได้เอง รองรับค่าการนับระหว่าง $\pm 999,999$ โดยมีสัญญาณ Input สำหรับควบคุมการนับจำนวน 3 ช่อง สำหรับ นับขึ้น นับลง และ รีเซ็ตค่าการนับ และมีเอาต์พุตแจ้งเตือนเป็นหน้าสัมผัสรีเลย์ (COMMON, NO, NC) สำหรับส่งสัญญาณแจ้งเตือน Alarm อีก 1 ช่อง

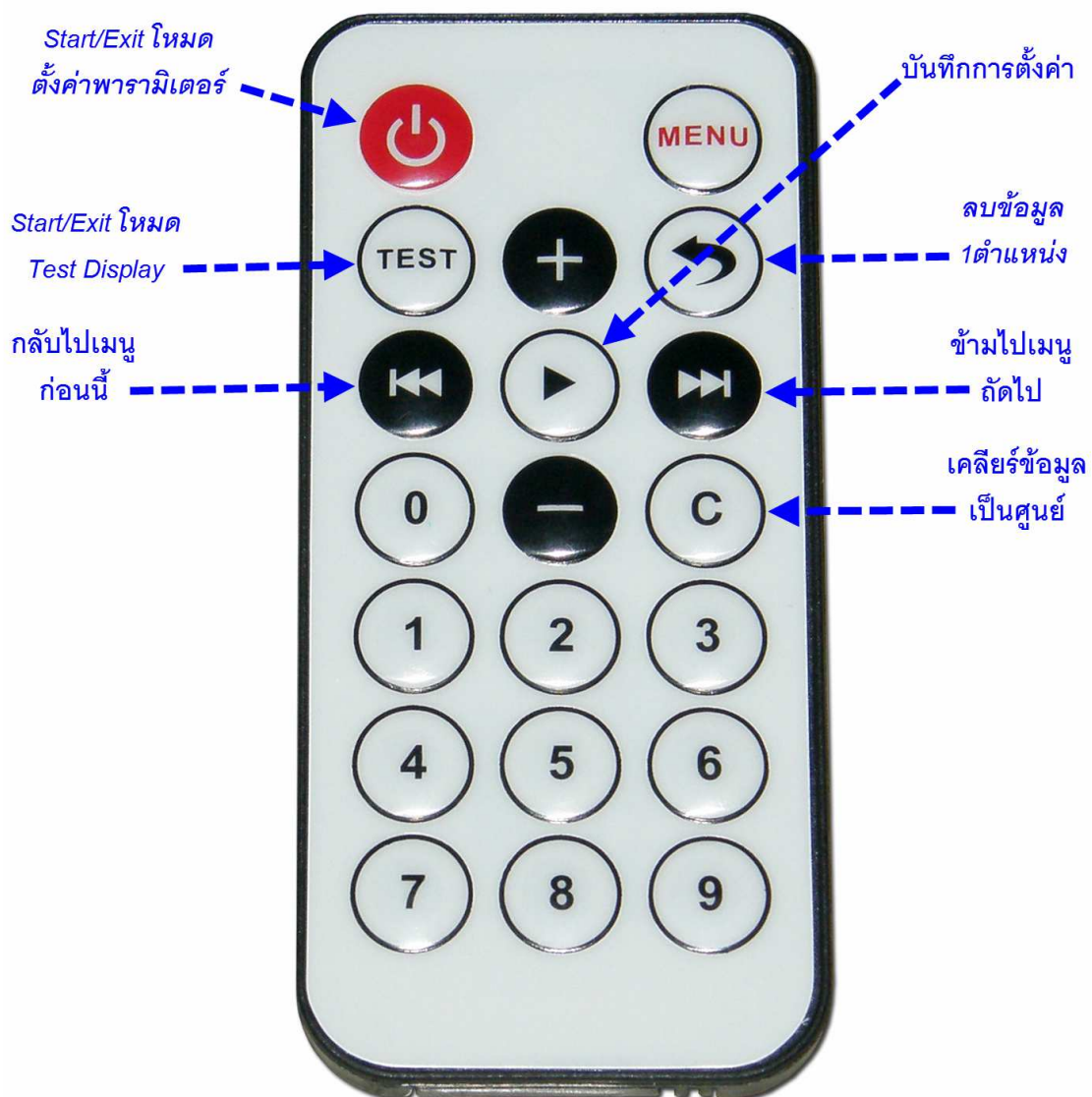
- TARGET ใช้แสดงค่าเป้าหมายหรือจำนวนชิ้นงานที่ต้องการนับ โดยค่า TARGET นี้สามารถกำหนดได้อิสระระหว่าง $\pm 999,999$
- ACTUAL ใช้แสดงค่าจำนวนชิ้นงานที่นับได้ มีค่าระหว่าง $\pm 999,999$ โดยค่าการนับจะเพิ่มขึ้น 1 ค่า เมื่อเกิดการกระตุ้นที่สัญญาณ Input Count Up 1 ครั้ง และค่าการนับจะลดลง 1 ค่า เมื่อเกิดการกระตุ้นที่สัญญาณ Input Count Down 1 ครั้ง โดยค่าการนับจะถูกเก็บสะสมไว้ตลอดไม่มีการสูญหายจนกว่าผู้ใช้งานจะมีการสั่งรีเซ็ตค่าการนับเอง โดยส่งสัญญาณกระตุ้นที่สัญญาณ Input Reset 1 ครั้ง ค่าการนับจึงจะกลับไปมีค่าเริ่มต้นเท่ากับค่า Reset Value ที่กำหนดไว้
- DIFFERENCE(DIFF.) ใช้แสดงค่าผลต่างของ TARGET และ ACTUAL โดยสามารถแสดงค่าระหว่าง $\pm 9,999,999$ โดยค่าผลต่างที่มีค่าต่ำกว่า $-999,999$ การแสดงผลจะใช้เครื่องหมาย จุดทศนิยมของ 7 Segment หลักทางขวาสุดในการแสดงเครื่องหมายลบแทน เช่น ถ้าค่าผลต่างมีค่าเป็น $-1,000,000$ ก็แสดงผลเป็น $1,000,000.$

คุณสมบัติ

- มีจอแสดงผลเป็น 7 Segment ขนาด 0.56 นิ้ว 7 หลัก จำนวน 3 แถว สำหรับแสดงค่าการนับ หรือใช้เป็นจอแสดงผลตัวเลข 7 หลัก อิสิระแบบเอนกประสงค์โดยรับคำสั่งผ่านทางพอร์ตสื่อสาร RS422/485
- มีวงจร Input แบบ Opto Isolate 3 ช่อง สำหรับรับสัญญาณ นับขึ้น นับลง และ รีเซ็ต โดยแต่ละช่องรองรับการใช้งานกับ Input แบบ หน้าสัมผัส และ NPN/PNP Sensor
- ความเร็วสูงสุดในการนับไม่รวมการ Debounce Input
 - 500Hz(2mS) ไม่ส่งข้อมูลการนับทาง RS485(SW[1]=OFF:RS485 Disable)
 - 40Hz(25mS) ส่งข้อมูลการนับทาง RS485 ด้วย Baudrate 19200BPS
- รองรับการจัดค่าพารามิเตอร์ต่างๆผ่านทาง IR Remote และ พอร์ตสื่อสาร RS422/485
- มีสัญญาณแจ้งเตือนเป็นหน้าสัมผัสรีเลย์(NO/NC/COMMON) จำนวน 1 ช่อง
- มีพอร์ตสื่อสารอนุกรม RS422/485 Full Duplex 4-Wire จำนวน 1 ช่อง
- รองรับการทำงานได้ทั้ง Programmable Counter Mode และ Terminal Display โดยใช้การสื่อสารผ่านทางพอร์ตสื่อสารอนุกรมแบบ RS422/485 Full Duplex 4-Wire พร้อม Dip Switch กำหนดรหัส ID Code ได้อิสิระ 8 รหัส
 - Terminal Display ASCII Code Command
 - Terminal Display Binary HEX Code Command
 - Terminal Display Counter Code Command
- สามารถใช้ ET-MINI CNT6 ทำหน้าที่เป็นชุดแสดงผลการนับ เพื่อเพิ่มจุดแสดงผลของ Counter หลายๆจุด ให้แสดงผลเหมือนชุด ET-MINI CNT6 ตัวหลักตัวเดียวได้
- ใช้กับแหล่งจ่าย 12-24VDC
- ขนาดกล่อง 5.25 x 12.00 x 17.50 cm.

การกำหนดค่าพารามิเตอร์ของ Counter

ในการกำหนดค่าพารามิเตอร์ของ Counter จะสามารถทำได้ 2 วิธีคือ ใช้ IR Remote ในการตั้งค่า หรือใช้โปรแกรม Setup จากคอมพิวเตอร์ PC ส่งคำสั่งตั้งค่าผ่านทาง RS422/485 (ต้องมีชุดแปลงสัญญาณ USB เป็น RS422/485 ในกรณีเครื่องคอมพิวเตอร์ PC ยังไม่มีพอร์ต RS232 หรือ RS422 อยู่ในเครื่อง หรือ ใช้ชุดแปลงสัญญาณ RS232 เป็น RS422/485 ในกรณีคอมพิวเตอร์ PC มีพอร์ต RS232 อยู่แล้ว) สำหรับการกำหนดค่าผ่านทาง IR Remote นั้นทำได้ดังนี้



รูปแสดง หน้าทีของคีย์ต่างๆที่ใช้ในการตั้งค่า ET-MINI CNT6

การตั้งค่าจาก IR Remote

1. กดคีย์ (Power) แล้วตามด้วย คีย์ตัวเลข (0) ถึง (7) ซึ่งเป็นค่าหมายเลข ID Code ของเครื่อง ตามลำดับ เพื่อเข้าสู่โหมดการตั้งค่า โดยเมื่อเข้าสู่โหมดการตั้งค่า การแสดงผลของตัวเลขทั้ง 3 บรรทัดจะเป็นดังนี้
 - บรรทัดที่1(บรรทัดบนสุด) จะแสดงค่าเมนูของการตั้งค่า ได้แก่ tARGET(Target) , ACtUAL(Actual Count) , PrESEt(Preset) , dLY-00(Debounce-In0:Count Up), dLY-01(Debounce-In1:Count Down) ,dLY-02(Debounce-In2:Reset) และ ALArM(Alarm)
 - บรรทัดที่2 จะแสดงค่าเดิมที่ถูกกำหนดไว้แล้วของพารามิเตอร์นั้นๆ
 - บรรทัดที่3 จะแสดงค่าที่ต้องการเปลี่ยนแปลงแก้ไขใหม่ของพารามิเตอร์นั้นๆ ซึ่งค่าการแสดงผลจะกระพริบอยู่ตลอดเวลาในระหว่างการแก้ไขค่า เมื่อได้ทำการป้อนค่าตัวเลขได้ตามต้องการแล้วให้กดคีย์ (>) เพื่อสั่งบันทึกการแก้ไข
2. กดคีย์ (>>) และ (<<) เพื่อเดินหน้า ถอยหลัง เลือกเมนูการตั้งค่าที่ต้องการ
3. กดคีย์ (>) เพื่อทำการบันทึกค่าที่ต้องการกำหนดใหม่
4. กดคีย์ (+) หรือ (-) เพื่อกำหนดค่าเครื่องหมายของตัวเลข
5. กดคีย์ (<) เพื่อลบค่าตัวเลขที่กำหนดไว้ในช่องแก้ไขค่าหลักสุดท้ายจำนวน 1หลัก
6. กดคีย์ (C) เพื่อเคลียร์ค่าตัวเลขที่กำหนดไว้ในช่องแก้ไขค่าให้เป็น 0
7. กดคีย์ (0) ถึง (9) เพื่อกำหนดค่าที่ต้องการ
8. กดคีย์ (Power) เพื่อออกจากโหมดการตั้งค่า
 - tArGET กำหนดค่า Target โดยสามารถกำหนดค่าได้ -999,999 ถึง +999,999
 - ACtUAL กำหนดค่า Actual โดยสามารถกำหนดค่าได้ -999,999 ถึง +999,999
 - PrESEt กำหนดค่า Preset ซึ่งเป็นค่าการนับเริ่มต้นเมื่อสั่ง Reset ค่าการนับของ Counter ซึ่งตามปกติแล้วจะกำหนดเป็น 0 ไว้เสมอ
 - dLY-00 เป็นการกำหนดค่า Debounce Input0(Count Up) ซึ่งผู้ใช้สามารถกำหนดค่าเวลา Debounce หน่วยเป็น mS ระหว่าง 0-999 ได้ตามต้องการ
 - dLY-01 เป็นการกำหนดค่า Debounce Input1(Count Down) ซึ่งผู้ใช้สามารถกำหนดค่าเวลา Debounce หน่วยเป็น mS ระหว่าง 0-999 ได้ตามต้องการ
 - dLY-02 เป็นการกำหนดค่า Debounce Input2(Reset) ซึ่งผู้ใช้สามารถกำหนดค่าเวลา Debounce หน่วยเป็น mS ระหว่าง 0-999 ได้ตามต้องการ
 - ALArM เป็นการกำหนดค่า Alarm ถ้าเป็น 0 คือ ยกเลิกการแจ้งเตือน(Disable), ถ้าเป็น 1 คือ เปิดการทำงานของแจ้งเตือน (Enable) หรือการแจ้งเตือน

การตั้งค่าโดยใช้คอมพิวเตอร์ PC

ET-MINI CNT6 SETUP

Com Port

Device Number: Device-0

Close

COM3

Display : ET-MINI CNT6 Counter

TARGET: 999999

ACTUAL: 0

DIFFERENCE: -999999

Configuration Device

<input checked="" type="checkbox"/> TARGET	+0999999
<input checked="" type="checkbox"/> ACTUAL	+0000000
<input checked="" type="checkbox"/> RESET	+0000000
<input checked="" type="checkbox"/> DEBOUNCE(0)	+0000050
<input checked="" type="checkbox"/> DEBOUNCE(1)	+0000050
<input checked="" type="checkbox"/> DEBOUNCE(2)	+0000999
<input checked="" type="checkbox"/> ALARM	+0000000

Read

Setup

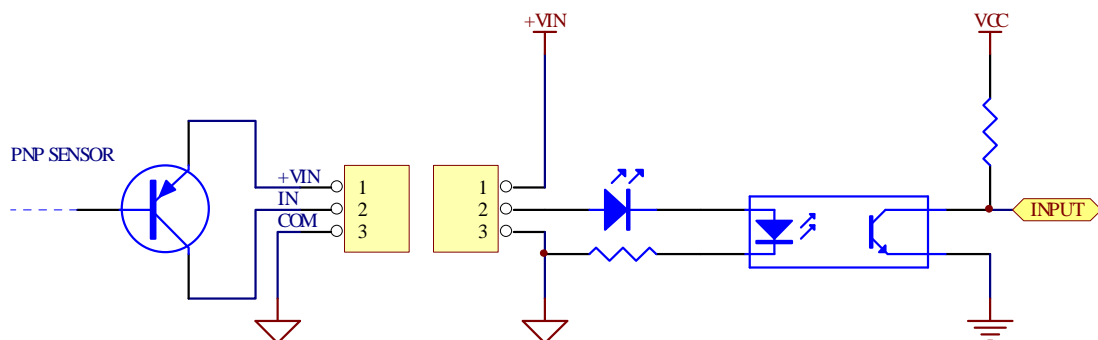
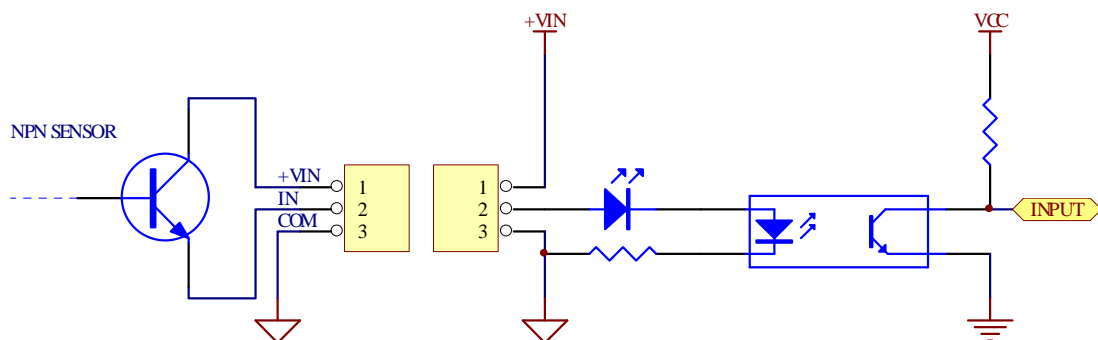
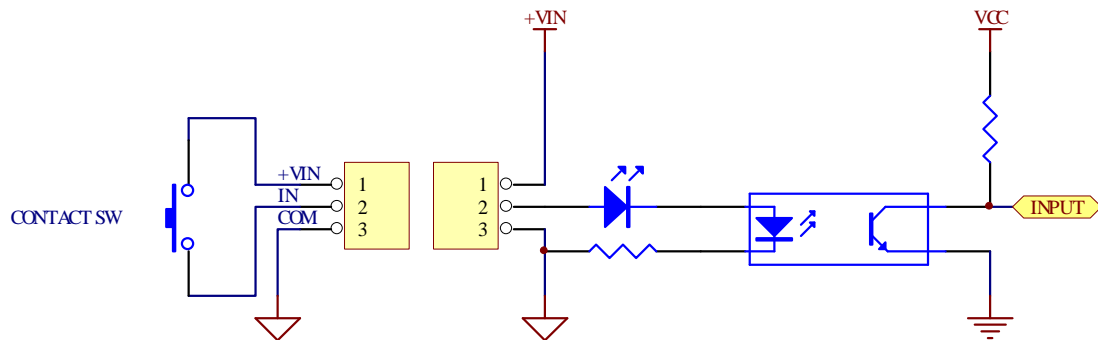
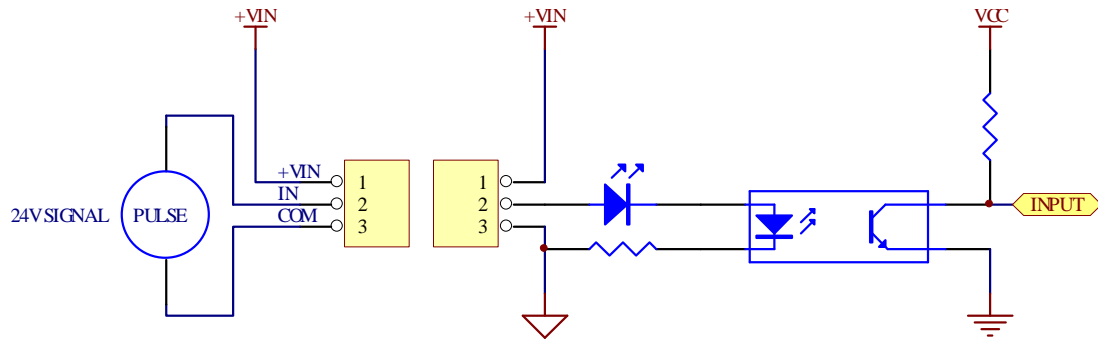
การต่อสัญญาณ Input

ET-MINI CNT6 จะมี Input จำนวน 3ช่อง คือ UP DOWN RESET โดยแต่ละ Input จะมี ขั้วต่อสัญญาณ 3 เส้น คือ +VIN IN และ COM ซึ่ง Input แต่ละช่อง สามารถใช้งานกับสัญญาณ Input ได้หลายแบบ ทั้งแบบ Volt Signal หรือ Input Contact หน้าสัมผัส และ Sensor แบบต่างๆ

- +VIN เป็นแรงดัน Output ซึ่งต่อผ่านมาจาก จุดรับแรงดันที่ป้อนให้แหล่งจ่าย POWER ของตัวเครื่อง ทางขั้วต่อ POWER มีค่าระหว่าง 12VDC-24VDC สำหรับต่อออกไปใช้เป็น แหล่งจ่ายให้ Sensor ต่างๆ
- IN (IN0, IN1, IN2) เป็นจุดรับสัญญาณ Input ที่จะป้อนให้กับตัวเครื่อง ซึ่งรับเป็นแรงดัน ได้สูงสุด +24VDC
- COM เป็น Common ของสัญญาณ Input

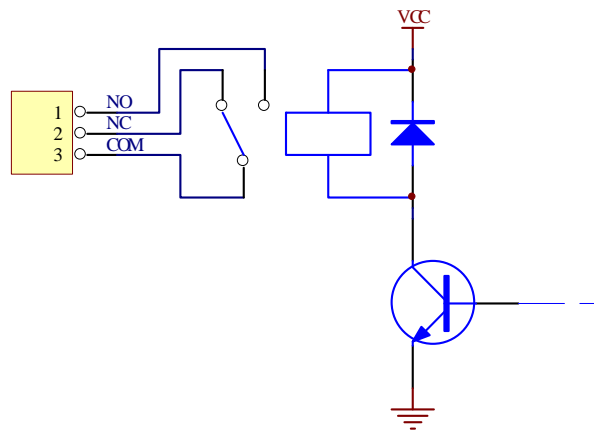


รูปแสดง ขั้วต่อสัญญาณ Input ของ ET-MINI CNT6



รูปแสดง แนวทางการต่อสัญญาณ Input แบบต่างๆให้ ET-MINI CNT6

สัญญาณ Alarm



สัญญาณ Alarm แจ้งเตือน เป็นหน้าสัมผัส รีเลย์ ซึ่งจะทำงานเมื่อมีการกำหนดค่าของ Alarm เป็น 1 แต่ถ้ากำหนดค่า Alarm เป็น 0 จะไม่มีการส่งสัญญาณ Alarm แจ้งเตือนใดๆในระหว่างการทำงาน โดยจะเกิดการ Alarm เมื่อค่าการนับของ Actual Counter มีค่าเท่ากับ หรือมากกว่า Target Counter

Value	Down				Reset	Up			
Target	1000	1000	1000	1000	1000	1000	1000	1000	1000
Actual	-1001	-1000	-999	-1	0	1	999	1000	1001
Diff.	-2001	-2000	-1999	-1001	-1000	-999	-1	0	1
Alarm	OFF	OFF	OFF	OFF	OFF	OFF	OFF	ON	ON

Value	Down				Reset	Up			
Target	0	0	0	0	0	0	0	0	0
Actual	-1001	-1000	-999	-1	0	1	999	1000	1001
Diff.	-1001	-1000	-999	-1	0	1	999	1000	1001
Alarm	OFF	OFF	OFF	OFF	ON	ON	ON	ON	ON

Value	Down				Reset	Up			
Target	-1000	-1000	-1000	-1000	-1000	-1000	-1000	-1000	-1000
Actual	-1001	-1000	-999	-1	0	1	999	1000	1001
Diff.	1	0	-1	-999	-1000	-1001	-1999	-2000	-2001
Alarm	ON	ON	OFF	OFF	OFF	OFF	OFF	OFF	OFF

ตารางแสดง ลักษณะการแสดงผลของ Difference Display และ Alarm

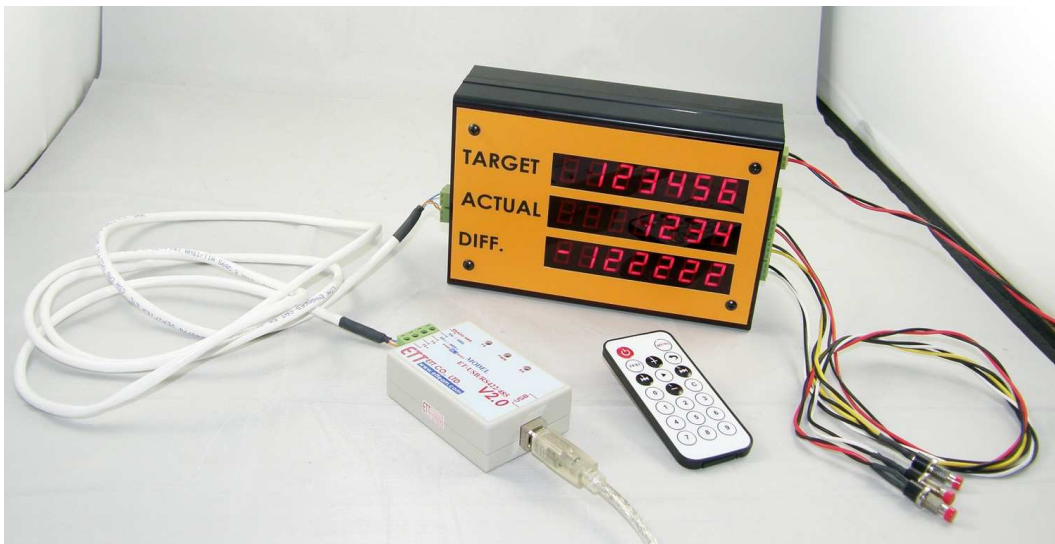
สัญญาณ RS485



ET-MINI CNT6 รองรับการใช้งานแบบเครือข่ายโดยใช้สัญญาณ RS485 (Full Duplex) แบบ 4 เส้น โดยจะมี Dip Switch ID0, ID1, ID2 สำหรับกำหนดรหัส ID Code ของเครื่อง ซึ่งสามารถกำหนดรหัสได้ 8 หมายเลข คือ 0-7

ID2	ID1	ID0	ID Code
OFF	OFF	OFF	0
OFF	OFF	ON	1
OFF	ON	OFF	2
OFF	ON	ON	3
ON	OFF	OFF	4
ON	OFF	ON	5
ON	ON	OFF	6
ON	ON	ON	7

- RS485 ENA ใช้กำหนดการส่งข้อมูลค่าการนับของ Counter ออกไปในเครือข่าย RS485 โดยถ้า Switch[1]:RS485 ENA อยู่ในตำแหน่ง OFF จะเป็นการปิดการส่งค่าการนับของ Counter ออกไปในเครือข่าย RS485 เพื่อป้องกันการชนกันของข้อมูลในบัส ในกรณีที่มีการต่อ ET-MINI CNT6 เป็น Counter ร่วมกันมากกว่า 1 ชุดในระบบบัสเดียวกัน ซึ่งในกรณีนี้จะต้องกำหนดให้อุปกรณ์ที่ทำหน้าที่เป็น Master คอยส่งคำสั่งมาสั่งอ่านค่า Counter ในบัสครั้งละ 1 ตัวตามลำดับเอง แต่ถ้ามีการกำหนด Switch[1]:RS485 ENA อยู่ในตำแหน่ง ON ทุกๆครั้งที่เกิดการนับเกิดขึ้น ET-MINI CNT6 จะส่งข้อมูลค่าการนับออกไปในบัส RS485 ด้วยเสมอทุกครั้ง ซึ่งหากมีการเชื่อมต่อ ET-MINI CNT6 มากกว่า 1 ตัวในบัส RS485 อาจมีการชนกันของข้อมูลทำให้ฝ่ายรับไม่สามารถรับข้อมูลที่ถูกต้องได้ แต่ไม่ว่าจะกำหนดตำแหน่งของ Switch[1] ไว้ตำแหน่งใด ถ้ามีคำสั่งจาก Master มาสั่งอ่านข้อมูล ET-MINI CNT6 จะส่งค่าข้อมูลกลับไปในบัสหลังจากได้รับคำสั่งเสมอ
- RXA(+) และ RXB(-) เป็นคู่สัญญาณด้านรับข้อมูลจากระบบ RS485 บัส เข้ามายัง ET-MINI CNT6
- TXA(+) และ TXB(-) เป็นคู่สัญญาณด้านส่งข้อมูลจาก ET-MINI CNT6 ออกไปยัง RS485 บัส

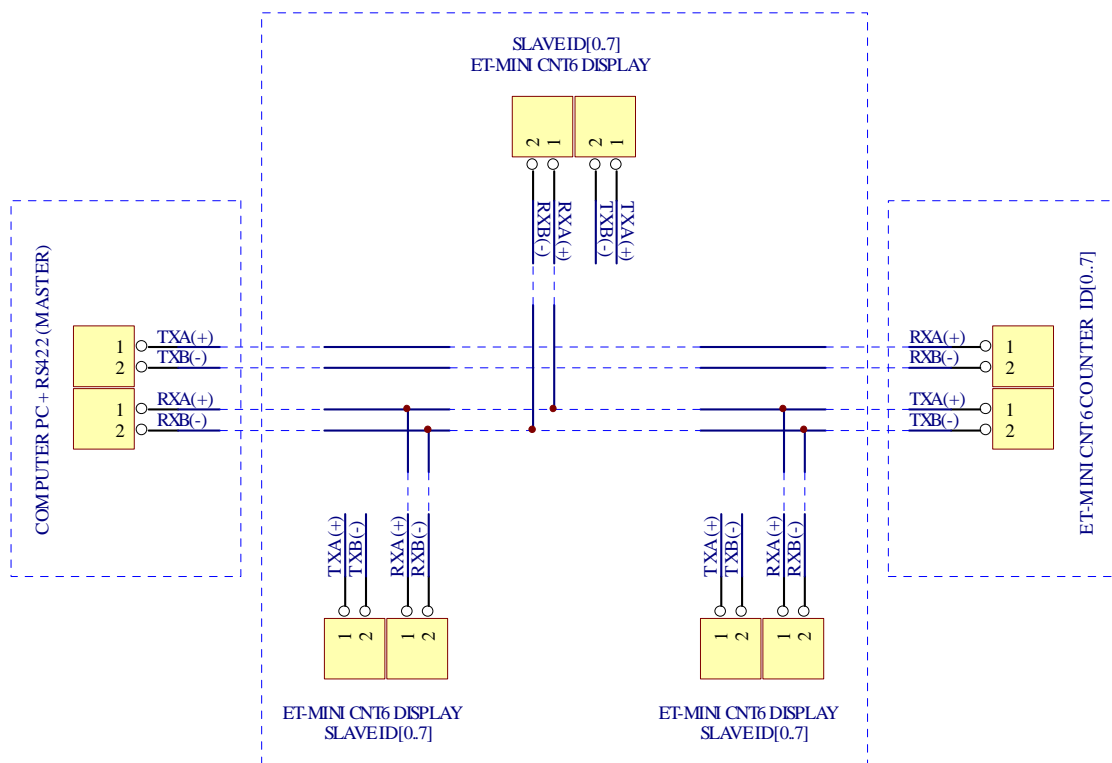


ตัวอย่างการเชื่อมต่อ RS485 ระหว่าง ET-MINI CNT6 กับคอมพิวเตอร์ PC โดยใช้ ET-USB/RS422/485 V2.0

การใช้งาน ET-MINI CNT6 เป็น COUNTER และ SLAVE DISPLAY ร่วมกัน

การเชื่อมต่อแบบนี้จะสามารถใช้ Master สำหรับสั่งตั้งค่าพารามิเตอร์และตรวจสอบค่าการนับต่างๆของ Counter ได้ตลอดเวลาและยังสามารถเพิ่มจำนวน Display สำหรับแสดงค่าการนับในหลายๆจุดโดยใช้สัญญาณการนับเพียงจุดเดียว

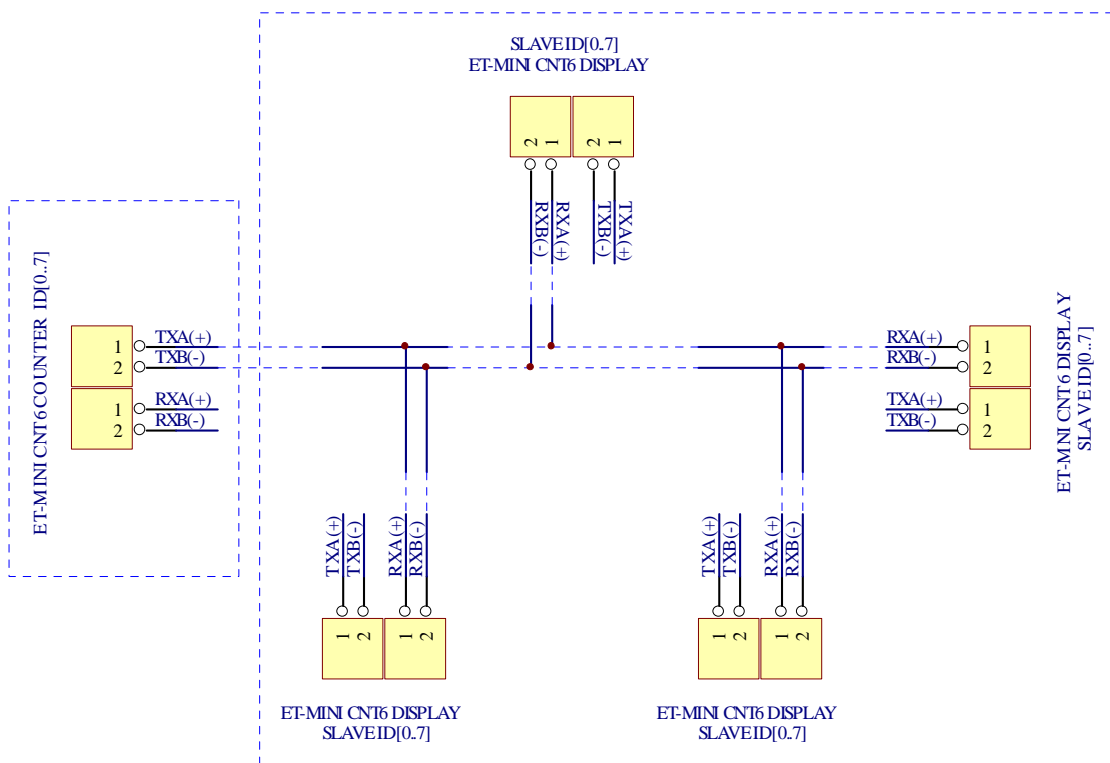
รูปแบบการเชื่อมต่อสัญญาณ จะใช้ TXA(+)และTXB(-) จาก Master ต่อไปยัง RXA(+)และRXB(-) ของ ET-MINI CNT6 ชุดที่จะใช้เป็น Counter และต่อ TXA(+)และTXB(-) จาก ET-MINI CNT6 ชุดที่เป็น Counter กลับไปยัง RXA(+)และRXB(-) ของ Master และ ET-MINI CNT6 ชุดที่ต้องการให้เป็น Slave Display



การต่อ RS485 โดยใช้ PC เป็น Master ร่วมกับ ET-MINI CNT6 COUNTER & DISPLAY

การใช้งาน ET-MINI CNT6 เป็น COUNTER และ SLAVE DISPLAY ร่วมกัน

การเชื่อมต่อแบบนี้จะสามารถใช้ ET-MINI CNT6 ทำหน้าที่เป็น Master Counter ร่วมกับ ET-MINI CNT เป็น Slave Display โดยไม่ต้องใช้คอมพิวเตอร์ PC ต่อร่วมในระบบด้วย โดยในการตั้งค่าพารามิเตอร์ต่างๆของ ET-MINI CNT6 ชุด Master Counter จะกระทำผ่าน IR Remote เป็นหลัก รูปแบบการเชื่อมต่อสัญญาณ จะใช้ TXA(+) และ TXB(-) จาก ET-MINI CNT6 ชุด Master ต่อไปยัง RXA(+) และ RXB(-) ของ ET-MINI CNT6 ชุดที่จะใช้เป็น Slave Display ส่วน TXA(+) และ TXB(-) จาก ET-MINI CNT6 ชุดที่เป็น Slave Display จะต่อกลับไปยัง RXA(+) และ RXB(-) ของชุด Master หรือไม่ได้

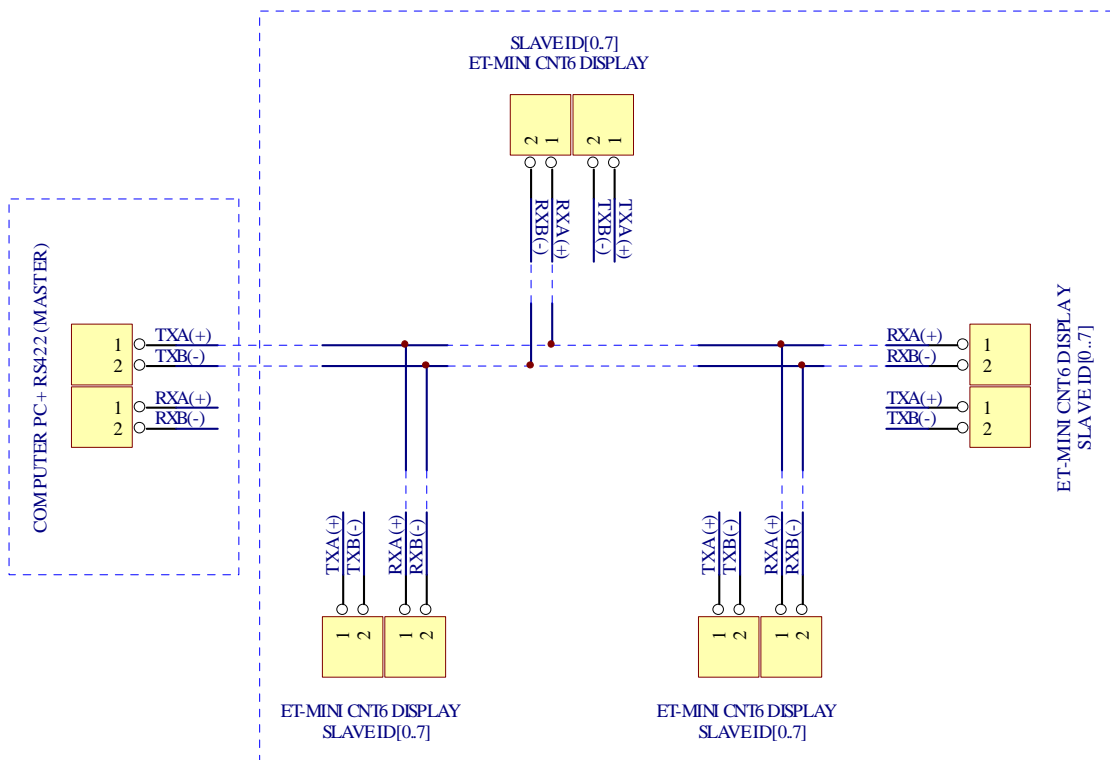


การต่อ RS485 โดยใช้ ET-MINI CNT6 เป็น Master Counter และ Slave Display ร่วมกัน

การใช้งาน ET-MINI CNT6 เป็น Slave Display

ในโหมดนี้จะเป็นการสื่อสารทางเดียว คือใช้ ET-MINI CNT6 เป็นฝ่ายรอรับข้อมูลแล้วนำไปแสดงผลเพียงอย่างเดียวไม่มีการส่งข้อมูลกลับไปยัง Master การเชื่อมต่อสัญญาณ RS485 จะต่อ TXA(+) และ TXB(-) จาก Master ไปเข้ากับ RXA(+) และ RXB(-) ของ ET-MINI CNT6 Slave ตามลำดับเพียงทิศทางใด ส่วน TXA(+) และ TXB(-) จาก ET-MINI CNT6 จะต่อกลับมายัง RXA(+) และ RXB(-) ของ Master หรือไม่ก็ได้ไม่มีผลใดๆ

การสั่งแสดงผล ET-MINI CNT6 สามารถใช้ชุดคำสั่งแสดงผลได้ 3 แบบ ผู้ใช้จะเลือกใช้ชุดคำสั่งรูปแบบใดก็ได้



การต่อ RS485 โดยใช้ PC เป็น Master ร่วมกับ ET-MINI CNT6 เป็น SLAVE DISPLAY

Protocol การรับส่งข้อมูล

ในการสื่อสารกับ ET-MINI CNT6 จะใช้พอร์ตสื่อสารอนุกรม โดยใช้รูปแบบสัญญาณเป็นแบบ RS422/485 แบบ Full Duplex 4 เส้น โดยมีรูปแบบดังนี้

- Baudrate = 19,200 BPS
- Data Bit = 8 Bit
- Parity Bit = None
- Stop Bit = 1

<STX><ID><CMD><Sep1><Packet Type><Sep2><Data..><ETX>

- STX เป็นรหัสเริ่มต้นคำสั่งมีขนาด 1 ไบต์ มีค่าเป็นรหัส ASCII ของ “*” หรือ 2AH
- <ID> เป็นรหัส ID Code ของเครื่อง มีขนาด 1 ไบต์ มีค่าเป็นรหัส ASCII ของรหัส ID เป็นตัวเลข “0” - “7” หรือ 30H-37H
- <CMD> เป็นรหัสคำสั่ง มีขนาด 1 ไบต์ มีค่าเป็นรหัส ASCII ของคำสั่ง
 - “A” หรือ 41H หมายถึง ASCII Command Display
 - “B” หรือ 42H หมายถึง Binary HEX Command Display
 - “C” หรือ 43H หมายถึง Counter Command Display
 - “R” หรือ 52H เป็นคำสั่งอ่านค่า Read Parameter จาก ET-MINI CNT6
 - “W” หรือ 57H เป็นคำสั่งตั้งค่า Write Parameter ให้ ET-MINI CNT6
- <Sep1> เป็นรหัสแยกข้อมูล Separate1 มีขนาด 1 ไบต์ มีค่าเป็นรหัส ASCII ของเครื่องหมาย “.” หรือ 3AH
- <Packet Type> เป็นรหัสเพื่อบ่งบอกชนิดของข้อมูลในคำสั่ง มีขนาด 1 ไบต์ มีค่าเป็นรหัส ASCII ของตัวเลข “0” - “7” หรือ 30H, 37H โดย
 - “?” ใช้ได้กับคำสั่ง “R” เท่านั้น หมายถึงต้องการอ่านค่าพารามิเตอร์ทั้งหมดของ ET-MINI CNT6 เช่น “*0R:?=?”<Cr> ซึ่งหมายถึงการสั่งอ่านค่าพารามิเตอร์ของ ET-MINI CNT6 ชุดที่มีหมายเลข ID Code = “0” ทุกพารามิเตอร์
 - “0” หมายถึง คำสั่งนี้เป็นข้อมูลการอ่านเขียนของ Target Value
 - “1” หมายถึง คำสั่งนี้เป็นข้อมูลการอ่านเขียนของ Actual Value
 - “2” หมายถึง คำสั่งนี้เป็นข้อมูลการอ่านเขียนของ Difference Value

- “3” หมายถึง คำสั่งนี้เป็นข้อมูลการอ่านเขียนของ Reset Value
- “4” หมายถึง คำสั่งนี้เป็นข้อมูลการอ่านเขียนของ Debounce Input0 Value
- “5” หมายถึง คำสั่งนี้เป็นข้อมูลการอ่านเขียนของ Debounce Input1 Value
- “6” หมายถึง คำสั่งนี้เป็นข้อมูลการอ่านเขียนของ Debounce Input2 Value
- “7” หมายถึง คำสั่งนี้เป็นข้อมูลการอ่านเขียนของ Alarm Value
- <Sep2> เป็นรหัสแยกข้อมูล Seperate2 มีขนาด 1ไบต์ มีค่าเป็นรหัส ASCII ของเครื่องหมาย “=” หรือ 3DH
- <Data> เป็นรหัส ASCII สำหรับกำหนดข้อมูลใน Packet มีขนาดไม่แน่นอนขึ้นอยู่กับรูปแบบของ Packet Type
 - “?” จะมีขนาด 1ไบต์ ใช้สำหรับกำหนดรูปแบบคำสั่งสำหรับสั่งอ่านข้อมูล
 - Target Value จะมีขนาดข้อมูล 8 ไบต์ โดยไบต์แรก เป็นเครื่องหมาย “+” หรือ “-” และตามด้วยค่าตัวเลขอีก 7ไบต์
 - Actual Value จะมีขนาดข้อมูล 8 ไบต์ โดยไบต์แรก เป็นเครื่องหมาย “+” หรือ “-” และตามด้วยค่าตัวเลขอีก 7ไบต์
 - Difference Value จะมีขนาดข้อมูล 8 ไบต์ โดยไบต์แรก เป็นเครื่องหมาย “+” หรือ “-” และตามด้วยค่าตัวเลขอีก 7ไบต์
 - Reset Value จะมีขนาดข้อมูล 8 ไบต์ โดยไบต์แรก เป็นเครื่องหมาย “+” หรือ “-” และตามด้วยค่าตัวเลขอีก 7ไบต์
 - Debounce Input0 Value จะมีขนาดข้อมูล 8 ไบต์ โดยไบต์แรก เป็นเครื่องหมาย “+” เสมอ และตามด้วยค่าตัวเลขอีก 7ไบต์ ซึ่งค่า Debounce จะกำหนดย่านข้อมูลได้ระหว่าง 0-999 เท่านั้น ดังนั้นค่าตัวเลข 4หลักแรกต้องกำหนดให้มีค่าเป็น 0 ไว้เสมอ เช่น “+0000**050**”
 - Debounce Input1 Value จะมีขนาดข้อมูล 8 ไบต์ โดยไบต์แรก เป็นเครื่องหมาย “+” เสมอ และตามด้วยค่าตัวเลขอีก 7ไบต์ ซึ่งค่า Debounce จะกำหนดย่านข้อมูลได้ระหว่าง 0-999 เท่านั้น ดังนั้นค่าตัวเลข 4หลักแรกต้องกำหนดให้มีค่าเป็น 0 ไว้เสมอ เช่น “+0000**050**”
 - Debounce Input2 Value จะมีขนาดข้อมูล 8 ไบต์ โดยไบต์แรก เป็นเครื่องหมาย “+” เสมอ และตามด้วยค่าตัวเลขอีก 7ไบต์ ซึ่งค่า Debounce จะกำหนดย่านข้อมูลได้ระหว่าง 0-999 เท่านั้น ดังนั้นค่าตัวเลข 4หลักแรกต้องกำหนดให้มีค่าเป็น 0 ไว้เสมอ เช่น “+0000**050**”

- Alarm Value จะมีขนาดข้อมูล 8 ไบต์ โดยไบต์แรก เป็นเครื่องหมาย “+” เสมอ และตามด้วยค่าตัวเลขอีก 7 ไบต์ ซึ่งค่า Alarm จะกำหนดย่านข้อมูลได้ระหว่าง 0 และ 1 เท่านั้น ดังนั้นค่าตัวเลข 6 หลักแรกต้องกำหนดให้มีค่าเป็น 0 ไร้เสมอ เช่น “+0000001”
- <ETX> เป็นรหัสจบคำสั่ง มีขนาด 1 ไบต์ มีค่าเป็นรหัส ASCII ของ Enter หรือ 13H

ตัวอย่างการใช้คำสั่ง

ส่งคำสั่งต้องการอ่านค่าทุกพารามิเตอร์ให้ ET-MINI CNT6 หมายเลข Device ID0

***0R: ?=?<Cr>**

ต้องการอ่านค่าทุกพารามิเตอร์จาก Device ID0

การตอบกลับจาก ET-MINI CNT6 หมายเลข Device ID0

***0C: 0=+0000100<Cr>** ค่า Target ของ Device ID0 = +100

***0C: 1=+0000000<Cr>** ค่า Actual ของ Device ID0 = 0

***0C: 2=-0000100<Cr>** ค่า Difference ของ Device ID0 = -100

***0C: 3=+0000000<Cr>** ค่า Reset ของ Device ID0 = 0

***0C: 4=+0000050<Cr>** ค่า Debounce Input0 ของ Device ID0 = 50mS

***0C: 5=+0000050<Cr>** ค่า Debounce Input1 ของ Device ID0 = 50mS

***0C: 6=+0000050<Cr>** ค่า Debounce Input2 ของ Device ID0 = 50mS

***0C: 7=+0000000<Cr>** ค่า Alarm ของ Device ID0 = 0(Disable)

ส่งคำสั่งต้องการอ่านค่า Target ให้ ET-MINI CNT6 หมายเลข Device ID0

***0R: 0=?<Cr>**

อ่านค่า Target จาก Device ID0

การตอบกลับจาก ET-MINI CNT6 หมายเลข Device ID0

***0C: 0=+0000100<Cr>**

ค่า Target ของ Device ID0 = +100

ส่งคำสั่งต้องการอ่านค่า Actual ให้ ET-MINI CNT6 หมายเลข Device ID0

***0R: 1=?<Cr>**

อ่านค่า Actual จาก Device ID0

การตอบกลับจาก ET-MINI CNT6 หมายเลข Device ID0

***0C: 1=+0000000<Cr>**

ค่า Actual ของ Device ID0 = 0

ส่งคำสั่งต้องการอ่านค่า Difference ให้ ET-MINI CNT6 หมายเลข Device ID0

***0R: 2=?<Cr>**

อ่านค่า Difference จาก Device ID0

การตอบกลับจาก ET-MINI CNT6 หมายเลข Device ID0

***0C: 2=-0000100<Cr>**

ค่า Difference ของ Device ID0 = -100

ส่งคำสั่งต้องการอ่านค่า Reset ให้ ET-MINI CNT6 หมายเลข Device ID0

***0R:3=?<Cr>**

อ่านค่า Reset จาก Device ID0

การตอบกลับจาก ET-MINI CNT6 หมายเลข Device ID0

***0C:3=-0000100<Cr>**

ค่า Reset ของ Device ID0 = -100

ส่งคำสั่งต้องการอ่านค่า Debounce Input0 ให้ ET-MINI CNT6 หมายเลข Device ID0

***0R:4=?<Cr>**

อ่านค่า Debounce Input0 จาก Device ID0

การตอบกลับจาก ET-MINI CNT6 หมายเลข Device ID0

***0C:4=-0000100<Cr>**

ค่า Debounce Input0 ของ Device0 = -100

ส่งคำสั่งต้องการอ่านค่าทุกพารามิเตอร์ให้ ET-MINI CNT6 หมายเลข Device ID0

***0R:5=?<Cr>**

อ่านค่า Debounce Input1 จาก Device ID0

การตอบกลับจาก ET-MINI CNT6 หมายเลข Device ID0

***0C:5=-0000100<Cr>**

ค่า Debounce Input1 ของ Device0 = -100

ส่งคำสั่งต้องการอ่านค่าทุกพารามิเตอร์ให้ ET-MINI CNT6 หมายเลข Device ID0

***0R:6=?<Cr>**

อ่านค่า Debounce Input2 จาก Device ID0

การตอบกลับจาก ET-MINI CNT6 หมายเลข Device ID0

***0C:6=-0000100<Cr>**

ค่า Debounce Input2 ของ Device0 = -100

ส่งคำสั่งต้องการอ่านค่าทุกพารามิเตอร์ให้ ET-MINI CNT6 หมายเลข Device ID0

***0R:7=?<Cr>**

อ่านค่า Alarm จาก Device ID0

การตอบกลับจาก ET-MINI CNT6 หมายเลข Device ID0

***0C:7=+0000001<Cr>**

ค่า Difference ของ Device0 = -100

ส่งคำสั่งต้องการตั้งค่า Target ให้ ET-MINI CNT6 หมายเลข Device ID0

***0W:0=+0123456<Cr>** ตั้งค่า Target ให้ Device ID0 = +123456

การตอบกลับจาก ET-MINI CNT6 หมายเลข Device ID0

***0C:0=+0123456<Cr>** ค่า Target ของ Device ID0 = +123456

ส่งคำสั่งต้องการตั้งค่า Actual ให้ ET-MINI CNT6 หมายเลข Device ID0

***0W:1=+123456<Cr>** ตั้งค่า Actual ให้ Device ID0 = +123456

การตอบกลับจาก ET-MINI CNT6 หมายเลข Device ID0

***0C:1=+0123456<Cr>** ค่า Actual ของ Device ID0 = +123456

ส่งคำสั่งตั้งค่า Alarm ให้ ET-MINI CNT6 หมายเลข Device ID0

***0W:7=+0000001<Cr>** ตั้งค่า Alarm จาก Device ID0=1(Enable)

การตอบกลับจาก ET-MINI CNT6 หมายเลข Device ID0

***0C:7=+0000001<Cr>** ค่า Alarm ของ Device ID0 = 1(Enable)

การใช้ ET-MINI CNT6 เป็นจอแสดงผลโดยใช้ ASCII Command Display

คำสั่งนี้จะเป็นการสั่งแสดงผลของ 7 Segment ให้แสดงผลตามค่ารหัส ASCII ของตัวเลข ตัวอักษรตามที่ผู้ใช้ต้องการ โดยสามารถสั่งแสดงผล จุดทศนิยม (.) เครื่องหมายลบ (-) ตัวเลข 0-9 และ ตัวอักษร A B C D E F G H I J L M N O P R S T U Y ส่วนรหัสของตัวอักษรอื่นๆเช่น K Q V W X Z และเครื่องหมายอื่นๆที่ไม่สามารถแสดงได้ถ้ามีการส่งรหัสแสดงผลเหล่านี้มาเครื่องจะแสดงเป็น Space แทน โดยคำสั่งนี้จะมีขนาดจำนวนไบต์ ไม่นแน่นอนขึ้นอยู่กับรหัสการแสดงผลในคำสั่ง โดยมีรูปแบบคำสั่งดังนี้

```
<STX><ID><CMD><Sep1><Line><Sep2><Display ASCII Code><ETX>
```

STX	<ID>	<CMD>	<Sep1>	<Line>	<Sep2>	<Display ASCII Code>	<ETX>
*	<ID>	A	:	<Line>	=	ASCII Display Code	<Cr>
2AH	0..7	41H	3AH	0..2	3DH	ASCII Display Code	0DH

- STX เป็นรหัสเริ่มต้นคำสั่งมีขนาด 1ไบต์ มีค่าเป็นรหัส ASCII ของ "*" หรือ 2AH
- <ID> เป็นรหัส ID Code ของเครื่อง มีขนาด 1ไบต์ มีค่าเป็นรหัส ASCII ของรหัส ID เป็นตัวเลข "0" - "7" หรือ 30H-37H
- <CMD> เป็นรหัสคำสั่ง มีขนาด 1ไบต์ มีค่าเป็นรหัส ASCII ของ "A" หรือ 41H
- <Sep1> เป็นรหัสแยกข้อมูล Seperate1 มีขนาด 1ไบต์ มีค่าเป็นรหัส ASCII ของเครื่องหมาย ":" หรือ 3AH
- <Line> เป็นรหัสกำหนดบรรทัดแสดงผล มีขนาด 1ไบต์ มีค่าเป็นรหัส ASCII ของตัวเลข "0" , "1" , "2" หรือ 30H, 31H, 32H โดย
 - "0" หมายถึง สั่งแสดงผลของบรรทัดบนสุด (TARGET)
 - "1" หมายถึง สั่งแสดงผลของบรรทัดกลาง (ACTUAL)
 - "2" หมายถึง สั่งแสดงผลของบรรทัดล่างสุด (DIFF.)
- <Sep2> เป็นรหัสแยกข้อมูล Seperate2 มีขนาด 1ไบต์ มีค่าเป็นรหัส ASCII ของเครื่องหมาย "=" หรือ 3DH
- <Display Binary Code> เป็นรหัส ASCII สำหรับแสดงผล โดยถ้ามีการส่งรหัส ASCII ที่ไม่สามารถแสดงผลได้เครื่องจะแสดงผลเป็น Space แทน และถ้ามีการส่งรหัส ASCII มากเกินจำนวนการแสดงผลของจอ เครื่องจะแสดงผลเพียง 7หลักแรกเท่านั้น
- <ETX> เป็นรหัสจบคำสั่ง มีขนาด 1ไบต์ มีค่าเป็นรหัส ASCII ของ Enter หรือ 13H

ตัวอย่างการใช้ ET-MINI CNT6 เป็นจอแสดงผลโดยใช้ ASCII Command Display

`*0A:0=1234567<Cr>` ค่า TARGET แสดงผลเป็น 1234567

`*0A:1=1234.567<Cr>` ค่า ACTUAL แสดงผลเป็น 1234.567

`*0A:2=-1234.56<Cr>` ค่า DIFF. แสดงผลเป็น -1234.56

`*0A:2=Error<Cr>` ค่า DIFF. แสดงผลเป็น Error

`*0A:2=EMPTY<Cr>` ค่า DIFF. แสดงผลเป็น Empty

`*0A:2=FULL<Cr>` ค่า DIFF. แสดงผลเป็น FULL

`*0A:2=TIME-01<Cr>` ค่า DIFF. แสดงผลเป็น TIME-01

การใช้ ET-MINI CNT6 เป็นจอแสดงผลโดยใช้ Binary HEX Command Display

คำสั่งนี้จะเป็นการสั่งแสดงผลของ 7 Segment ให้แสดงผลเป็นตัวเลขตัวอักษรหรือสัญลักษณ์ใดๆตามการออกแบบและกำหนดของผู้ใช้เอง โดยผู้ใช้จะต้องกำหนดรหัสแสดงผลและส่งข้อมูลสำหรับกำหนดการติดดับของ 7 Segment ได้เอง โดยคำสั่งนี้จะมีขนาด 21 ไบต์ โดยมีรูปแบบคำสั่งดังนี้

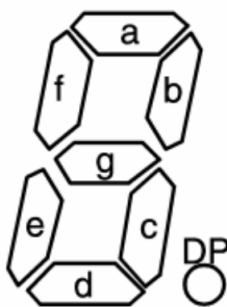
<STX><ID><CMD><Sep1><Line><Sep2><Display Binary HEX Code><ETX>

STX	<ID>	<CMD>	<Sep1>	<Line>	<Sep2>	<14 Byte Display Binary HEX Code>										<ETX>		
*	<ID>	B	:	<Line>	=	d	d	d	d	d	d	d	d	d	d	d	d	<Cr>
2AH	0..7	42H	3AH	0..2	3DH	DG6	DG5	DG4	DG3	DG2	DG1	DG0					0DH	

- STX เป็นรหัสเริ่มต้นคำสั่งมีขนาด 1ไบต์ มีค่าเป็นรหัส ASCII ของ “*” หรือ 2AH
- <ID> เป็นรหัส ID Code ของเครื่อง มีขนาด 1ไบต์ มีค่าเป็นรหัส ASCII ของตัวเลข “0”-“7” หรือ 30H-37H
- <CMD> เป็นรหัสคำสั่ง มีขนาด 1ไบต์ มีค่าเป็นรหัส ASCII ของ “B” หรือ 42H
- <Sep1> เป็นรหัสแยกข้อมูล Seperate1 มีขนาด 1ไบต์ มีค่าเป็นรหัส ASCII ของเครื่องหมาย “:” หรือ 3AH
- <Line> เป็นรหัสกำหนดบรรทัดแสดงผล มีขนาด 1ไบต์ มีค่าเป็นรหัส ASCII ของตัวเลข “0” , “1” , “2” หรือ 30H, 31H, 32H โดย
 - “0” หมายถึง สั่งแสดงผลของบรรทัดบนสุด (TARGET)
 - “1” หมายถึง สั่งแสดงผลของบรรทัดกลาง (ACTUAL)
 - “2” หมายถึง สั่งแสดงผลของบรรทัดล่างสุด (DIFF.)
- <Sep2> เป็นรหัสแยกข้อมูล Seperate2 มีขนาด 1ไบต์ มีค่าเป็นรหัส ASCII ของเครื่องหมาย “=” หรือ 3DH
- <Display Binary HEX Code> เป็นรหัสแสดงผลของ 7 Segment ทั้ง 7หลัก มีขนาด 14 ไบต์ โดยแต่ละหลักจะใช้รหัสกำหนดการแสดงผล 2ไบต์ มีค่าเป็นรหัส ASCII ของ HEX Code สำหรับกำหนดการแสดงผลของ 7 Segment
- <ETX> เป็นรหัสจบคำสั่ง มีขนาด 1ไบต์ มีค่าเป็นรหัส ASCII ของ Enter หรือ 13H

การกำหนดค่าการแสดงผลแบบ Binary HEX Code

การแสดงผลแบบ Binary Code เป็นรูปแบบการแสดงผลที่มีความอ่อนตัวในการใช้งานมากเพราะผู้ใช้สามารถถอดรูปแบบการแสดงผลได้เองตามต้องการ ในการกำหนดการแสดงผลของ 7 Segment ในโหมดนี้ ผู้ใช้ต้องแทนค่าการติดของ Segment แสดงผลด้วยบิตข้อมูลที่เป็น “1” และแทนค่าการดับของ Segment แสดงผลด้วยค่าของบิตข้อมูลที่เป็น “0” โดยจอแสดงผล 7 Segment แต่ละหลักจะมีส่วนแสดงผล 8 ส่วน คือ A B C D E F G และ DP ตามลำดับ โดยการแสดงผลของแต่ละ Segment จะถูกกำหนดผ่านบิตข้อมูลจำนวน 1บิต ดังแสดงความสัมพันธ์ในตาราง



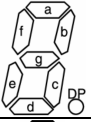
Dp	A	B	C	D	E	F	G
D7	D6	D5	D4	D3	D2	D1	D0

รูปแบบการกำหนดข้อมูลสำหรับ ควบคุมการแสดงผลของ 7 Segment

ตัวอย่างเช่น เราต้องการกำหนดให้ 7 Segment ติดเป็นเลข “8” ซึ่งก็จะต้องกำหนดให้ Segment A-G แสดงผลเพียง Segment ส่วน Segment Dp จะต้องกำหนดให้ดับ ซึ่งจะได้ค่าข้อมูลสำหรับการแสดงผลเครื่องหมายลบ “8” ดังนี้คือ

Dp	A	B	C	D	E	F	G
D7	D6	D5	D4	D3	D2	D1	D0
0	1	1	1	1	1	1	1

จะเห็นว่าค่าของข้อมูลที่ได้คือ “01111111” หรือ 7FH ดังนั้นค่าการแสดงผลจึงต้องกำหนดเป็นรหัส ASCII 2หลัก ของ “7” และ “F” หรือ 37H และ 46H ตามลำดับ

Dp	A	B	C	D	E	F	G	Segment	
D7	D6	D5	D4	D3	D2	D1	D0	HEX Code	
0	1	1	1	1	1	1	0	7E	0
0	0	1	1	0	0	0	0	30	1
0	1	1	0	1	1	0	1	6D	2
0	1	1	1	1	0	0	1	79	3
0	0	1	1	0	0	1	1	33	4
0	1	0	1	1	0	1	1	5B	5
0	1	0	1	1	1	1	1	5F	6
0	1	1	1	0	0	0	0	70	7
0	1	1	1	1	1	1	1	7F	8
0	1	1	1	1	0	1	1	7B	9
0	1	1	1	0	1	1	1	77	A
0	0	0	1	1	1	1	1	1F	B
0	1	0	0	1	1	1	0	4E	C
0	0	1	1	1	1	0	1	3D	D
0	1	0	0	1	1	1	1	4F	E
0	1	0	0	0	1	1	1	47	F

ตัวอย่าง การสร้างรหัสควบคุมการแสดงผลของ 7 Segment ในโหมด Binary Command

ตัวอย่างการใช้ ET-MINI CNT6 เป็นจอแสดงผลโดยใช้ Binary HEX Command Display

`*0B:0=7E306D79335B5F<Cr>` ค่า TARGET แสดงผลเป็น [0][1][2][3][4][5][6]

`*0B:1=30303030303030<Cr>` ค่า ACTUAL แสดงผลเป็น [1][1][1][1][1][1][1]

`*0B:2=01771F4E3D4F47<Cr>` ค่า DIFF. แสดงผลเป็น [-][A][B][C][D][E][F]

การใช้ ET-MINI CNT6 เป็นจอแสดงผลโดยใช้ Counter Command Display

คำสั่งนี้จะเป็นการสั่งแสดงผล 7 Segment ของ ET-MINI CNT6 ให้แสดงผลตามค่าการนับของ Counter เพื่อเป็นการเพิ่มจุดแสดงผลของ Counter ให้แสดงผลได้หลายๆจุดจากการนับของ Counter เพียงจุดเดียว โดยเพียงแต่ทำการกำหนดรหัส ID ของ ET-MINI CNT6 ชุดที่จะให้เป็น Display ให้มีค่า ID ตรงกับ ET-MINI CNT6 ชุดที่ทำหน้าที่เป็น Counter แต่ไม่ต้องเชื่อมต่อสัญญาณการนับให้กับ ET-MINI CNT6 ชุดที่ต้องการใช้เป็นชุดแสดงผล Display และกำหนดการทำงานของ Dip_SW[1] เป็น OFF:RS485 Echo Disable ไว้

<STX><ID><CMD><Sep1><Line><Sep2><Display ASCII Code><ETX>

STX	<ID>	<CMD>	<Sep1>	<Line>	<Sep2>	<Counter Value>	<ETX>
*	<ID>	C	:	<Line>	=	ASCII Counter Value	<Cr>
2AH	0..7	43H	3AH	0..2	3DH	ASCII Counter Value	0DH

- STX เป็นรหัสเริ่มต้นคำสั่ง มีขนาด 1 ไบต์ มีค่าเป็นรหัส ASCII ของ “*” หรือ 2AH
- <ID> เป็นรหัส ID Code ของเครื่อง มีขนาด 1 ไบต์ มีค่าเป็นรหัส ASCII ของตัวเลข “0”-“7” หรือ 30H-37H
- <CMD> เป็นรหัสคำสั่ง มีขนาด 1 ไบต์ มีค่าเป็นรหัส ASCII ของ “C” หรือ 43H
- <Sep1> เป็นรหัสแยกข้อมูล Seperate1 มีขนาด 1 ไบต์ มีค่าเป็นรหัส ASCII ของเครื่องหมาย “:” หรือ 3AH
- <Line> เป็นรหัสกำหนดบรรทัดแสดงผล มีขนาด 1 ไบต์ มีค่าเป็นรหัส ASCII ของตัวเลข “0” , “1” , “2” หรือ 30H, 31H, 32H โดย
 - “0” หมายถึง สั่งแสดงผลของบรรทัดบนสุด (TARGET)
 - “1” หมายถึง สั่งแสดงผลของบรรทัดกลาง (ACTUAL)
 - “2” หมายถึง สั่งแสดงผลของบรรทัดล่างสุด (DIFF.)
- <Sep2> เป็นรหัสแยกข้อมูล Seperate2 มีขนาด 1 ไบต์ มีค่าเป็นรหัส ASCII ของเครื่องหมาย “=” หรือ 3DH
- <ASCII Counter Value> เป็นรหัส ASCII ของค่า Counter Value ที่ต้องการแสดงผลมีขนาด 8 หลัก โดยหลักแรกเป็นรหัส ASCII ของเครื่องหมาย บวก(+) หรือ (-) และตามด้วยรหัส ASCII ของตัวเลข 0-9 จำนวน 7 หลัก
- <ETX> เป็นรหัสจบคำสั่ง มีขนาด 1 ไบต์ มีค่าเป็นรหัส ASCII ของ Enter หรือ 13H

ตัวอย่างการใช้ ET-MINI CNT6 เป็นจอแสดงผลโดยใช้ Counter Command Display

`*0C:0=+0123456<Cr>` ค่า TARGET แสดงผลเป็น 123456

`*0C:1=+0001234<Cr>` ค่า ACTUAL แสดงผลเป็น 1234

`*0C:2=-122222<Cr>` ค่า DIFF. แสดงผลเป็น -122222

ตัวอย่างการใช้งาน

ต้องการนับปริมาณรถในลานจอดรถซึ่งสามารถจอดรถได้ปริมาณ 500คัน สามารถทำได้โดยการกำหนดค่าพารามิเตอร์ของ Counter ไว้ดังนี้

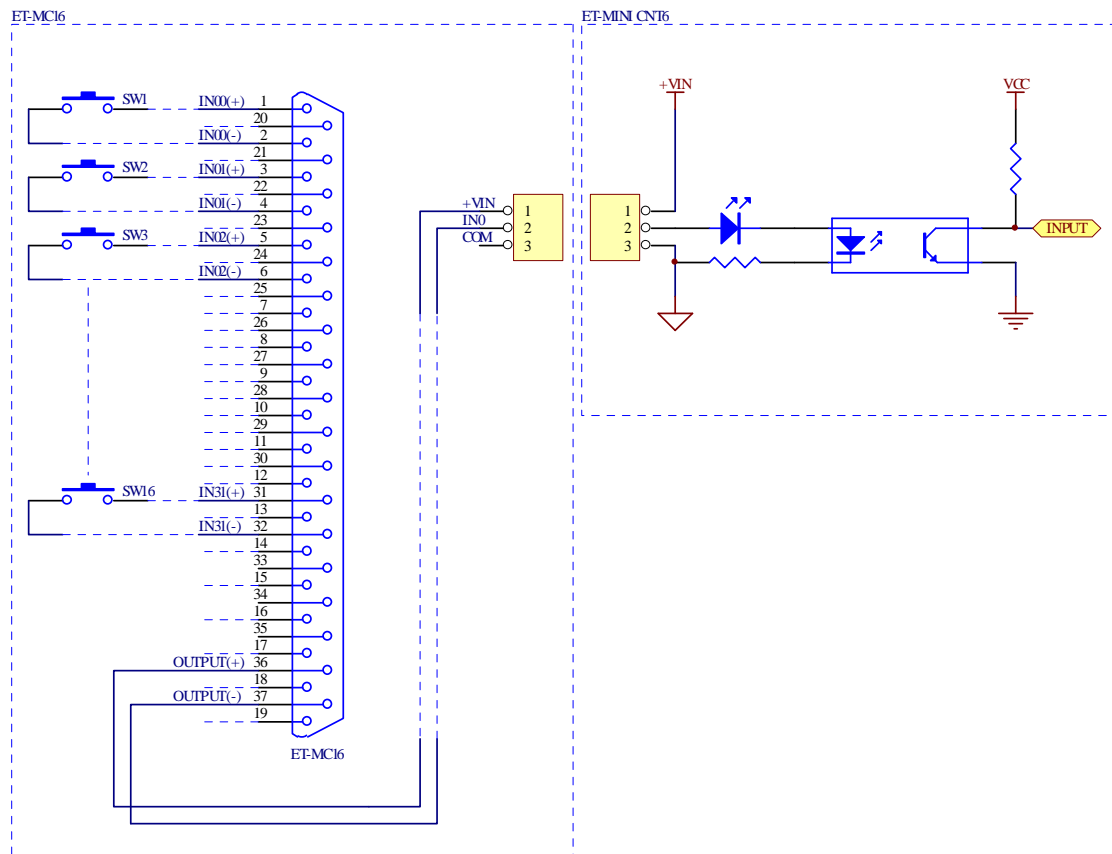
- กำหนดค่า Target = 500 (ปริมาณรถที่จอดได้ทั้งหมด)
- กำหนดค่า Actual = 0 (ปริมาณรถที่เข้าจอดแล้ว)
- กำหนดค่า Preset = 0 (เริ่มต้นนับจาก 0)
- กำหนดค่า Alarm = 1 (Enable Alarm)
- เชื่อมต่อสัญญาณนับรถเข้า กับ Input Up Counter
- เชื่อมต่อสัญญาณนับรถออกกับ Input Down Counter
- เชื่อมต่อสัญญาณรีเซ็ตเข้ากับ Input Reset Counter

เริ่มต้นทำงานเมื่อทำการรีเซ็ตค่าการนับของ ACTUAL จะเริ่มต้นเป็น 0 เพื่อบอกว่ายังไม่มีรถเข้าจอด ส่วนค่า TARGET จะแสดง 500 เพื่อแสดงจำนวนรถที่รับได้ และ DIFF. จะแสดงค่าปริมาณคงเหลือที่รับรถได้เป็น -500 เมื่อมีรถเข้าจอด (Up Input) ค่า ACTUAL จะเพิ่มขึ้นครั้งละ 1 ค่า เพื่อแสดงจำนวนรถที่เข้าจอดแล้ว ส่วน DIFF. จะลดค่าลง 1ค่า เนื่องจากพื้นที่จอดรถลงในขณะเดียวกันเมื่อมีรถออก(Down Input) ค่า ACTUAL จะลดค่าลง 1ค่า ส่วนค่า DIFF. ก็เพิ่มขึ้น 1ค่าเนื่องจากปริมาณพื้นที่จอดเพิ่มขึ้น

สำหรับสัญญาณ Alarm ก็สามารถนำไปประยุกต์ใช้แสดงสถานะ เช่น นำ Contact NC(Normal Close) ไปใช้เปิด-ปิด Tower Lamp สีเขียว และนำ Contact NO(Normal Open) ไปใช้เปิด-ปิดไฟ Tower Lamp สีแดง โดยในกรณีนี้เมื่อค่าการนับของ ACTUAL ยังไม่ถึง 500 Relay Alarm จะยังไม่ทำงาน ไฟ Tower Lamp สีเขียวก็ยังคงติดสว่างเป็นสีเขียวให้ทราบ เพื่อใช้แสดงว่ายังมีพื้นที่ว่างสำหรับจอดรถได้อยู่ แต่เมื่อค่าการนับของ ACTUAL เท่ากับ หรือ มากกว่า 500 จะทำให้สัญญาณ Relay Alarm ทำงานซึ่งจะส่งผลทำให้ Tower Lamp สีแดงติดสว่างเพื่อแสดงให้ทราบว่า พื้นที่จอดรถเต็มแล้ว เป็นต้น

ตัวอย่างการประยุกต์ใช้ในการนับหลาย Input

ในกรณีที่มีสัญญาณการนับจำนวนมากๆแต่ไม่ต้องการใช้ Counter หลายๆชุด เช่น ในสายงานการผลิตที่มีเครื่องจักรสำหรับผลิตสินค้าหลายๆเครื่องร่วมกันผลิตสินค้ารุ่นเดียวกัน ตามจำนวนออเดอร์ที่รับมา ซึ่งต้องการทราบผลรวมของสินค้าที่ผลิตได้จากเครื่องจักรทั้งหมดรวมกันว่าสามารถผลิตได้ปริมาณเท่าใดแล้วและเหลืออีกจำนวนเท่าไร จึงจะครบตามจำนวนที่ต้องการโดยไม่ต้องการทราบว่าเครื่องจักรแต่ละเครื่องผลิตได้เป็นจำนวนกี่ชิ้น ผู้ใช้สามารถนำชุดรวมสัญญาณ ET-MC16 (Multi Counter 16 Channel) ซึ่งรองรับ Input การนับได้ 16ช่อง มาทำหน้าที่รวมสัญญาณการนับซึ่งต่อมาจากเครื่องจักรแต่ละเครื่อง แล้วจึงส่งสัญญาณผลรวมของการนับให้กับ ET-MINI CNT6 เพียงชุดเดียวทำการนับและแสดงผลการนับให้ทราบได้เช่นเดียวกัน



ผังการต่อ ET-MC16 กับ ET-MINI CNT6 เพื่อรวมสัญญาณการนับแบบหลาย Input