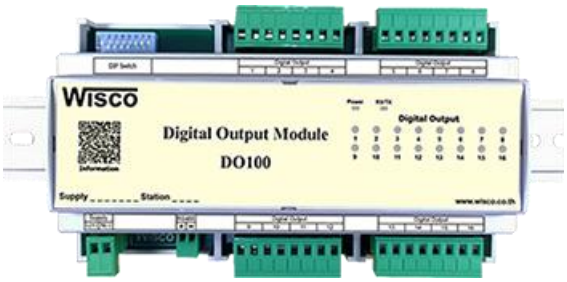
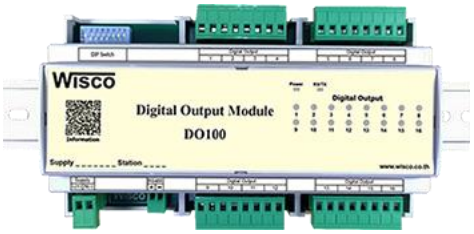
Wisco DO100 Protocol



Wisnu and Supak Co., Ltd 102/111-112 Tessabansongkroh Rd., Ladyao, Jatujak, Bangkok 10900
Tel (662)591-1916, (662)954-3280-1, Fax (662)580-4427, www.wisco.co.th, E-mail info@wisco.co.th

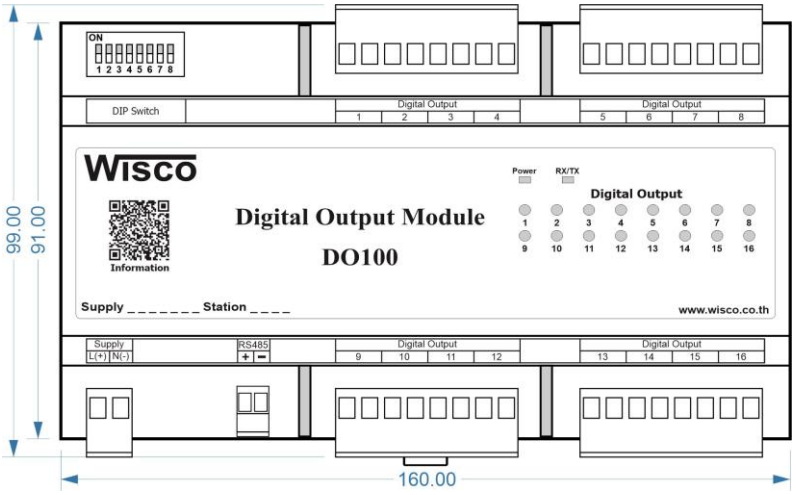
บริษัท วิศณุและสุภัค จำกัด 102/111-112 หมู่บ้านสินพัฒนาธานี ถนนเทศบาลสงเคราะห์ แขวงลาดยาว เขตจตุจักร กรุงเทพฯ 10900
โทร. (02)591-1916, (02)954-3280-1, แฟกซ์ (02)580-4427, www.wisco.co.th, อีเมล์ info@wisco.co.th

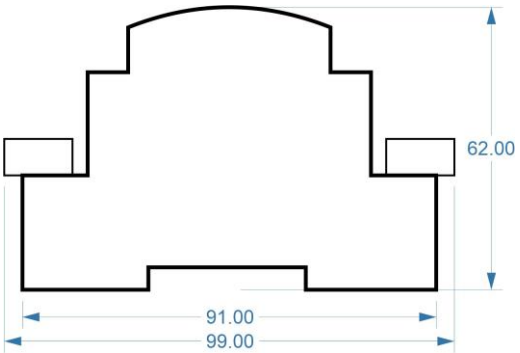
Digital Output Module
DO100



การเชื่อมต่อตัว DO100 สามารถเชื่อมต่อผ่านทาง RS485 โดยจะสามารถเชื่อมต่อกันได้ครั้งละหลายเครื่องโดยสามารถเชื่อมต่อ DO100 ได้ทั้งหมด 32 เครื่องพร้อมกันรวมกับ Computer อีก 1 เครื่อง ซึ่งจะใช้ข้อกำหนด (Protocol) เดียวกันในการติดต่อกับ DO100 โดยมีรายละเอียดดังต่อไปนี้

Dimensions (Unit: mm.)

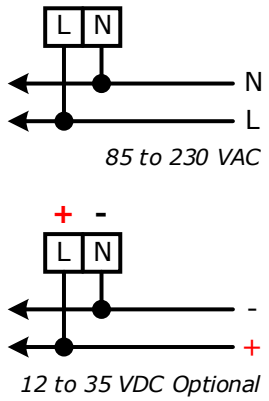




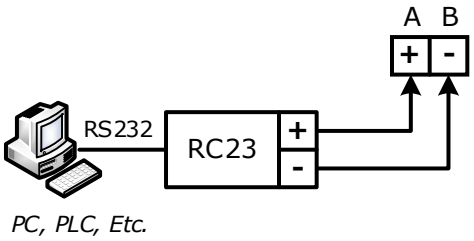
Wisnu and Supak Co., Ltd 102/111-112 Tessabansongkroh Rd., Ladyao, Jatujak, Bangkok 10900
Tel (662)591-1916, (662)954-3280-1, Fax (662)580-4427 – www.wisco.co.th – info@wisco.co.th

Wiring

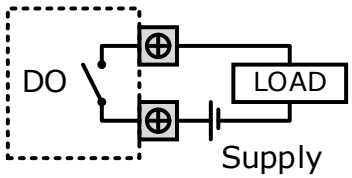
Supply



RS485



Digital Output



การตั้งค่า **Dip Switch**

Dipswitch ที่ใช้สำหรับเลือก Station (ตำแหน่งที่ 1 - 5), Baud Rate (ตำแหน่งที่ 6 - 7), Protocol (ตำแหน่งที่ 8) ดังนี้

1	2	3	4	5	Station
0	0	0	0	0	0 (00h)
1	0	0	0	0	1 (01h)
0	1	0	0	0	2 (02h)
1	1	0	0	0	3 (03h)
0	0	1	0	0	4 (04h)
1	0	1	0	0	5 (05h)
0	1	1	0	0	6 (06h)
1	1	1	0	0	7 (07h)
0	0	0	1	0	8 (08h)
1	0	0	1	0	9 (09h)
0	1	0	1	0	10 (0Ah)

1	2	3	4	5	Station
1	1	0	1	0	11 (0Bh)
0	0	1	1	0	12 (0Ch)
1	0	1	1	0	13 (0Dh)
0	1	1	1	0	14 (0Eh)
1	1	1	1	0	15 (0Fh)
0	0	0	0	1	16 (10h)
1	0	0	0	1	17 (11h)
0	1	0	0	1	18 (12h)
1	1	0	0	1	19 (13h)
0	0	1	0	1	20 (14h)
1	0	1	0	1	21 (15h)

1	2	3	4	5	Station
0	1	1	0	1	22 (16h)
1	1	1	0	1	23 (17h)
0	0	0	1	1	24 (18h)
1	0	0	1	1	25 (19h)
0	1	0	1	1	26 (1Ah)
1	1	0	1	1	27 (1Bh)
0	0	1	1	1	28 (1Ch)
1	0	1	1	1	29 (1Dh)
0	1	1	1	1	30 (1Eh)
1	1	1	1	1	31 (1Fh)

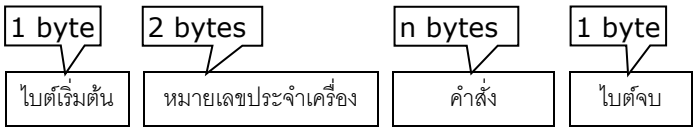
6	7	Baud rate
0	0	4800
1	0	9600
0	1	19200
1	1	57600

8	Protocol
0	MODBUS RTU
1	MODBUS ASCII / WISCO

บริษัท วิศณุและสุภัค จำกัด 102/111-112 หมู่บ้านสินพัฒนาธานี ถนนเทศบาลสงเคราะห์ แขวงลาดยาว เขตจตุจักร กรุงเทพฯ 10900
โทร. (02)591-1916, (02)954-3280-1, แฟกซ์ (02)580-4427 – www.wisco.co.th – info@wisco.co.th

การติดต่อกับโมดูลโดยใช้ Wisco Protocol

ข้อมูลที่ใช้ในการติดต่อกับโมดูล DO100 จะเป็นรหัส ASCII ทั้งหมดและในคำสั่งชุดหนึ่งจะประกอบไปด้วย



ไบต์เริ่มต้น ไบต์แรกที่บอกให้โมดูลรู้ว่าได้เริ่มต้นของชุดคำสั่ง โดยจะใช้อักขระ '#' เป็นตัวเริ่มต้น

หมายเลขประจำเครื่อง หมายเลขที่ใช้อ้างอิงตัวโมดูลสำหรับกรณีที่มีการต่อใช้งานพร้อมกันตั้งแต่ 2 ตัว ขึ้นไป โดยสามารถตั้งได้ที่ DIP Switch บนตัวโมดูล ซึ่งจะมีค่าตั้งแต่ 00h-1Fh และห้ามให้หมายเลขซ้ำกัน

คำสั่ง คำสั่งที่ใช้กับโมดูล สำหรับ DO100 จะมีทั้งหมด 6 คำสั่ง

ไบต์จบ ไบต์สุดท้ายที่บอกให้โมดูลรู้ว่าสิ้นสุดของชุดคำสั่ง โดยจะใช้ [CR] (Carriage Return) ซึ่งเป็นอักขระตัวที่ 13 ในตาราง ASCII เป็นตัวปิดท้าย

Character	#	0	A	W	D	O	1	5	,	0	1	CR
ASCII Code	23H	30H	41H	52H	44H	4FH	31H	35H	2CH	30H	31H	0DH

ตัวอย่างการใช้งานคำสั่งสำหรับ Wisco Protocol

Wisnu and Supak Co., Ltd 102/111-112 Tessabansongkroh Rd., Ladyao, Jatujak, Bangkok 10900
Tel (662)591-1916, (662)954-3280-1, Fax (662)580-4427 – www.wisco.co.th – info@wisco.co.th

รายละเอียดและตัวอย่างของคำสั่ง

(| | = 1 byte, | ... | = n bytes, | CR | = Carriage Return)

1. *คำสั่งที่ใช้อ่านค่า Digital Output*

ขึ้นต้นด้วย 'RDO' และจบด้วย '[CR]' เช่น อ่านค่า DO จากเครื่องหมายเลข 00 จะได้คำสั่งดังนี้ '#00RDO[CR]'

#	0	0	R	D	O	CR

โดยตัวโมดูลจะตอบกลับมาเป็น 'DO>' ตามด้วยค่าที่วัดได้ ทั้ง 16 ช่อง ช่องละ 1 ไบต์ รวม 16 ไบต์ (MSB -> LSB, '0' = OFF, '1' = ON) และจบด้วย '[CR]' ดังตัวอย่างนี้ 'DO>1101010101101010[CR]'

D	O	>	1	1	0	1	0	1	0	1	0	1	1	0	1	0	1	0	CR

2. *คำสั่งที่ใช้อ่านค่า Digital Output (Hexadecimal)*

คล้ายกับข้อ 1 แต่เปลี่ยนเป็นขึ้นต้นด้วย 'RDOH' และจบด้วย '[CR]' เช่น อ่านค่า DO จากเครื่องหมายเลข 04 จะได้คำสั่งดังนี้ '#04RDOH[CR]'

#	0	4	R	D	O	H	CR

โดยตัวโมดูลจะตอบกลับมาเป็น 'DO>' ตามด้วยค่าที่วัดได้ โดยใช้รูปแบบของ bit ทั้งหมด 4 ไบต์ (MSB -> LSB, '0' = OFF, '1' = ON) และจบด้วย '[CR]' ดังตัวอย่างนี้ 'DO>D56A[CR]'

D	O	>	D	5	6	A	CR

บริษัท วิศณุและสุภัค จำกัด 102/111-112 หมู่บ้านสินพัฒนาธานี ถนนเทศบาลสงเคราะห์ แขวงลาดยาว เขตจตุจักร กรุงเทพฯ 10900
โทร. (02)591-1916, (02)954-3280-1, แฟกซ์ (02)580-4427 – www.wisco.co.th – info@wisco.co.th

3. *คำสั่งที่ใช้อ่านค่าจากหน่วยความจำชนิด EEPROM*

ขึ้นต้นด้วย 'REE' ตามด้วยหมายเลขตัว EEPROM ที่จะอ่าน 1 ไบต์ (DO100 จะมี EEPROM เพียงตัวเดียว, คำสั่งจะนับตัว EEPROM โดยเริ่มนับจาก 0) ตามด้วยตำแหน่งเริ่มต้น 4 ไบต์ ตามด้วยจำนวนไบต์ที่จะอ่าน 4 ไบต์ ซึ่งจะต้องไม่เกินความจุของ DO100 คือ 2048 ไบต์ และจบด้วย '[CR]' เช่น อ่านค่า EEPROM จากเครื่องหมายเลข 07 โดยเริ่มจากตำแหน่ง 200H จำนวน 500 ไบต์ (01F4h) จะได้คำสั่งดังนี้ '#07REE0020001F4[CR]'

#	0	7	R	E	E	0	0	2	0	0	0	1	F	4	CR

โดยตัวโมดูลจะตอบกลับมาเป็น 'EE>' ตามด้วยค่าที่อยู่ใน EEPROM เป็นเลขฐาน16 ตามด้วยค่า Checksum อีก 2 ไบต์ (ดูวิธีคำนวณในหัวข้อ *วิธีคิด Checksum สำหรับ Wisco Protocol)* และจบด้วย '[CR]' ดังตัวอย่างนี้ 'EE>0320FF45...A79Dxx[CR]'

E	E	>	0	3	2	0	F	F	4	5	...	A	7	9	D	x	x	CR

4. *คำสั่งที่ใช้เขียนค่า Digital Output*

ขึ้นต้นด้วย 'WDO' ตามด้วยช่องสัญญาณที่จะเขียน คั่นด้วย ',' ตามด้วยค่าที่ต้องการจะเขียนของช่องที่ต้องการจะเขียน (ได้ตั้งแต่ช่องที่ 1-8 เท่านั้น '0' = OFF, '1' = ON) และจบด้วย '[CR]' เช่น เขียนค่า DO ไปที่เครื่องหมายเลข 0A ช่องที่ 1=OFF, 2=ON, 4=OFF จะได้คำสั่งดังนี้ '#0AWDO124,010[CR]'

#	0	A	W	D	O	1	2	4	,	0	1	0	CR

โดยตัวโมดูลจะตอบกลับมาเป็น 'DO>OK' และจบด้วย '[CR]' ดังนี้

D	O	>	O	K	CR

Wisnu and Supak Co., Ltd 102/111-112 Tessabansongkroh Rd., Ladyao, Jatujak, Bangkok 10900
Tel (662)591-1916, (662)954-3280-1, Fax (662)580-4427 – www.wisco.co.th – info@wisco.co.th

5. *คำสั่งที่ใช้เขียนค่า Digital Output (Expand Function)*

เหมือนกับข้อ 4 แต่จะใช้รูปแบบการเขียนเป็น bit รวม 16 ช่อง ทั้งหมดเป็น 4 ไบต์ (MSB -> LSB, '0' = เขียน, '1' = ไม่เขียน) ขึ้นต้นด้วย 'WDOX' ตามด้วยช่องสัญญาณที่จะเขียน คั่นด้วย ',' ตามด้วยค่าที่ต้องการจะเขียนของช่องนั้น ('0' = OFF, '1' = ON) และจบด้วย '[CR]' เช่น เขียนค่า DO ไปที่เครื่องหมายเลข 0D โดยให้ช่องที่ 16, 12, 10, 9, 7, 6, 5, 2 = 1, ช่อง 1 = 0 จะได้คำสั่งดังนี้ '#0DWDOX8B73,8B72[CR]'

#	0	D	W	D	O	X	8	B	7	3	,	8	B	7	2	CR

โดยตัวโมดูลจะตอบกลับมาเป็น 'DO>OK' และจบด้วย '[CR]' ดังนี้

D	O	>	O	K	CR

6. *คำสั่งที่ใช้เขียนค่าไปที่หน่วยความจำชนิด EEPROM*

ขึ้นต้นด้วย 'WEE' ตามด้วยหมายเลขตัว EEPROM ที่จะเขียน 1 ไบต์ (DO100 จะมี EEPROM เพียงตัวเดียว, คำสั่งจะนับตัว EEPROM โดยเริ่มนับจาก 0) ตามด้วยตำแหน่งเริ่มต้น 4 ไบต์ ตามด้วยจำนวนไบต์ที่จะเขียน 2 ไบต์ ตามด้วยข้อมูลที่จะเขียน ตามด้วย Checksum (ดูวิธีคำนวณในหัวข้อ *วิธีคิด Checksum สำหรับ Wisco Protocol)* อีก 2 ไบต์ และจบด้วย '[CR]' เช่น เขียนค่า EEPROM ไปที่เครื่องหมายเลข 13 โดยเริ่มจากตำแหน่ง 100H จำนวน 2 ไบต์ (12 34) จะได้คำสั่งดังตัวอย่างนี้ '#13WEE00100031234B7[CR]' (B7 = checksum)

#	1	0	W	E	E	0	0	1	0	0	0	2	1	2	3	4	B	7	CR

โดยตัวโมดูลจะตอบกลับมาเป็น 'EE>OK' และจบด้วย '[CR]' ดังนี้

E	E	>	O	K	CR

บริษัท วิศณุและสุภัค จำกัด 102/111-112 หมู่บ้านสินพัฒนาธานี ถนนเทศบาลสงเคราะห์ แขวงลาดยาว เขตจตุจักร กรุงเทพฯ 10900
โทร. (02)591-1916, (02)954-3280-1, แฟกซ์ (02)580-4427 – www.wisco.co.th – info@wisco.co.th

วิธีคิด CHECK SUM สำหรับ Wisco Protocol

ใน DO100 จะใช้ CHECK SUM ในการตรวจสอบความถูกต้องของข้อมูลที่ส่งไปสำหรับ Read หรือ Write กับ EEPROM การคิด CHECK SUM นั้นจะใช้การบวกข้อมูลทั้งหมดเข้าด้วยกัน (บวกเฉพาะข้อมูลที่เป็นตัวเลขเท่านั้น) บวกกันครั้งละ 1 ไบต์โดยค่าที่เกิน 1 byte นั้นเราจะตัดทิ้ง จากนั้น นำค่าที่ได้ 1 byte นั้นมาทำ 1's complement และ 2's complement เป็นอันเรียบร้อย

ตัวอย่างเช่น `'# 1A WEE 0 0000 05 11 22 33 44 55 [CR]'

	HEXADECIMAL	BINARY
ไบต์เริ่มต้น	00H	0000 0000
	00H	0000 0000
	05H	0000 0000
	11H	0001 0001
	+ 22H	+ 0010 0010
	33H	0011 0011
	44H	0100 0100
ไบต์สุดท้าย	55H	0101 0101
ผลลัพธ์	104H	1 0000 0100
คิดเฉพาะ 1 byte (8 bit)	04H	0000 0100
ทำ 1's complement (invert)	FBH	1111 1011
ทำ 2' complement	FBH + 1	1111 1011+ 1
ค่า Check sum ที่ได้	FCH	1111 1100

ข้อมูลที่จะส่งจึงเป็น`'# 1A WEE 0 0000 05 11 22 33 44 55 FC [CR]'

Wisnu and Supak Co., Ltd 102/111-112 Tessabansongkroh Rd., Ladyao, Jatujak, Bangkok 10900
Tel (662)591-1916, (662)954-3280-1, Fax (662)580-4427 – www.wisco.co.th – info@wisco.co.th

รหัสตอบกลับมาเมื่อเกิดข้อผิดพลาดในการส่งคำสั่งไปยังตัวโมดูล DO100

ในกรณีที่การส่งคำสั่งไปยังตัวโมดูลนั้น หากชุดคำสั่งนั้นไม่ถูกต้อง ตัวโมดูลจะไม่ทำคำสั่งชุดนั้น และรายงานความผิดพลาดที่เกิดขึ้นกลับมาเป็นรหัสต่างๆ โดยจะขึ้นต้นด้วย 'ERR=' แล้วตามด้วยตัวเลข ตั้งแต่ 1-6 ดังนี้

1 (illegal function)	คำสั่งไม่ถูกต้อง หรือโมดูลไม่รู้จักคำสั่งนี้
2 (illegal data address)	ค่าตำแหน่งเริ่มต้น เกินช่วงตำแหน่งที่กำหนดไว้
3 (illegal data value)	ค่าของข้อมูลที่ใช้ในชุดคำสั่งไม่ถูกต้อง เช่น ค่าของ DO ที่จะอ่าน ไม่ถูกต้อง
4 (invalid data frame)	รูปแบบของชุดคำสั่งไม่ตรงตามข้อกำหนด เช่น เขียนค่า DO โดยไม่มี ',' คั่นระหว่างหมายเลขช่องกับค่าที่จะเขียน
5 (check sum error)	ค่า check sum ไม่ถูกต้อง (อาจเกิดจากความผิดพลาดระหว่างส่งข้อมูล)
6 (invalid number of byte)	จำนวนข้อมูลที่รับมาไม่ครบตามจำนวนที่แจ้งไว้

สรุปคำสั่งที่ใช้กับตัวโมดูล DO100 (Wisco Protocol)

((H) = Heximal Value, (E) = Expand Function, xx = check sum,
[CR] = carriage return)

Function	Command	DO2000 Response
RDO = Read Digital Output	#00RDO[CR]	DO>1101010101101010[CR]
RDOH = Read Digital Output (H)	#03RDOH[CR]	DO>D56A[CR]
REE = Read EEPROM	#07REE0020001F4[CR]	EE>0320FF45...A79Dxx[CR]
WDO = Write Digital Output	#0AWDO124,010[CR]	DO>OK[CR]
WDOX = Write Digital Output (E)	#0DWDOX8B73,8B72[CR]	DO>OK[CR]
WEE = Write EEPROM	#10WEE00100021234B7[CR]	EE>OK[CR]

บริษัท วิศณุและสุภัค จำกัด 102/111-112 หมู่บ้านสินพัฒนาธานี ถนนเทศบาลสงเคราะห์ แขวงลาดยาว เขตจตุจักร กรุงเทพฯ 10900
โทร. (02)591-1916, (02)954-3280-1, แฟกซ์ (02)580-4427 – www.wisco.co.th – info@wisco.co.th

การติดต่อกับโมดูลโดยใช้ MODBUS (ASCII) Protocol

โมดูล DO100 สามารถใช้ Protocol MODBUS ในการติดต่อได้เช่นกัน โดยจะมีรูปแบบของคำสั่งดังต่อไปนี้ (CHAR = Character; 1 CHAR ประกอบไปด้วย 8 Data Bits, 1 Start Bit, และ 1 Stop Bit)

ADDR	FUNCTION	DATA	ERROR CHECK	EOF	READY TO REC RESP
2-CHAR 16-BITS	2-CHAR 16-BITS	N x 4-CHAR N x 16-BITS	2-CHAR 16-BITS	CR	LF

โมดูล DO100 สนับสนุนฟังก์ชันพื้นฐานของ Modbus ทั้งหมด 6 ฟังก์ชัน ดังต่อไปนี้

MODBUS ASCII	Wisco
READ OUTPUT STATUS (CODE 01)	= Read Digital Output
READ OUTPUT REGISTERS (CODE 03)	= Read EEPROM
FORCE SINGLE COIL (CODE 05)	= Write Digital Output
PRESET SINGLE REGISTER (CODE 06)	= Write EEPROM
FORCE MULTIPLE COILS (CODE 15)	= Write Digital Output
PRESET MULTIPLE REGISTERS (CODE 16)	= Write EEPROM

การอ้าง Address บนตัวโมดูลมีดังนี้

Function Code	Reference	Address
01, 05, 15	Digital Output	0xxxx
03, 06, 16	EEPROM	4xxxx

Wisnu and Supak Co., Ltd 102/111-112 Tessabansongkroh Rd., Ladyao, Jatujak, Bangkok 10900
Tel (662)591-1916, (662)954-3280-1, Fax (662)580-4427 – www.wisco.co.th – info@wisco.co.th

Digital Output Table

Name	Address
Digital Output Channel 1	00001
Digital Output Channel 2	00002
Digital Output Channel 3	00003
Digital Output Channel 4	00004
Digital Output Channel 5	00005
Digital Output Channel 6	00006
Digital Output Channel 7	00007
Digital Output Channel 8	00008
Digital Output Channel 9	00009
Digital Output Channel 10	00010
Digital Output Channel 11	00011
Digital Output Channel 12	00012
Digital Output Channel 13	00013
Digital Output Channel 14	00014
Digital Output Channel 15	00015
Digital Output Channel 16	00016

Holding Register Table

Name	Address
Digital Output Mode	40001
Hold Time DO1	40002
Hold Time DO2	40003
Hold Time DO3	40004
Hold Time DO4	40005
Hold Time DO5	40006
Hold Time DO6	40007
Hold Time DO7	40008
Hold Time DO8	40009
Hold Time DO9	40010
Hold Time DO10	40011
Hold Time DO11	40012
Hold Time DO12	40013
Hold Time DO13	40014
Hold Time DO14	40015
Hold Time DO15	40016
Hold Time DO16	40017

รายละเอียดและหน้าที่ของ Holding Register

EEPROM, Digital Output Mode ทำหน้าที่กำหนดการทำงานของ DO ในรูปแบบของข้อมูลที่เป็นบิต (1 bit/channel) ซึ่งค่าจะมีความหมายคือ 0=Latch, 1=Pulse โดยเรียงลำดับดังนี้

MSB	C16	C15	C14	C13	C12	C11	C10	C9	C8	C7	C6	C5	C4	C3	C2	C1	LSB

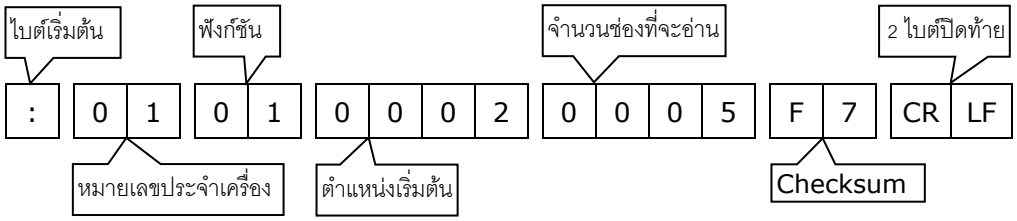
EEPROM, Hold Time DOx ใช้งานร่วมกับ *Digital Output Mode* เพื่อกำหนดความกว้างของ Pulse เมื่อเลือกให้ Digital Output ช่องนั้นทำงานแบบ Pulse แล้ว (1 = 0.1 วินาที) โดยจะกำหนดให้ได้สูงสุด 25.5 วินาที และต่ำสุด 0.1 วินาที

*รายละเอียดที่เหลือของ Modbus สามารถดูได้จาก *'Modbus Reference Guide'* หรือที่ http://www.modbus.org/specs.php

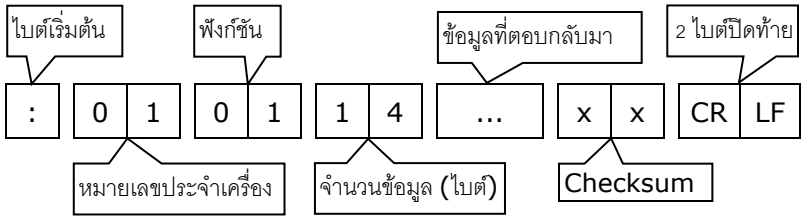
บริษัท วิศณุและสุภัค จำกัด 102/111-112 หมู่บ้านสินพัฒนาธานี ถนนเทศบาลสงเคราะห์ แขวงลาดยาว เขตจตุจักร กรุงเทพฯ 10900
โทร. (02)591-1916, (02)954-3280-1, แฟกซ์ (02)580-4427 – www.wisco.co.th – info@wisco.co.th

ตัวอย่างฟังก์ชั่น **MODBUS(ASCII) PROTOCOL**

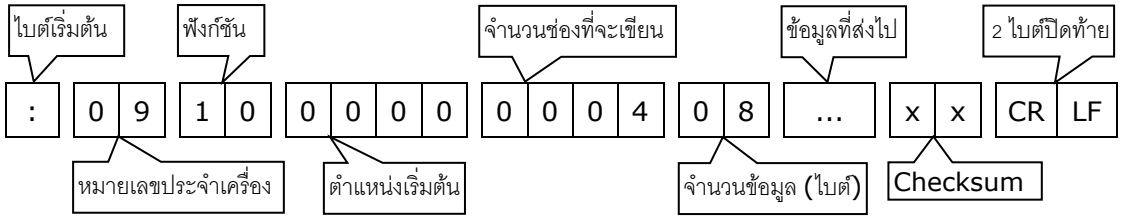
Function Code 01



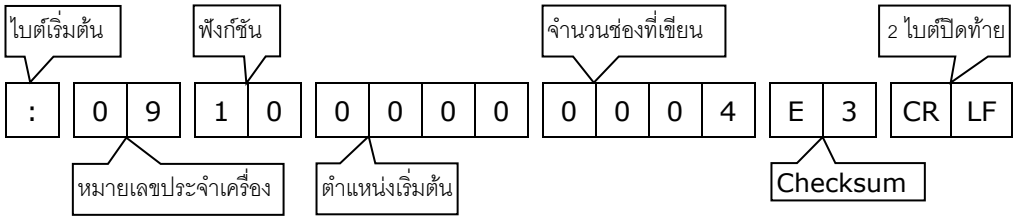
Response



Function Code 16



Response



Wisnu and Supak Co., Ltd 102/111-112 Tessabansongkroh Rd., Ladyao, Jatujak, Bangkok 10900
Tel (662)591-1916, (662)954-3280-1, Fax (662)580-4427 – www.wisco.co.th – info@wisco.co.th

วิธีคิด CHECK SUM สำหรับ MODBUS(ASCII) Protocol

ใน MODBUS Protocol จะใช้ CHECK SUM ในการตรวจสอบความถูกต้องของข้อมูลที่ส่งไปทุกคำสั่ง การคิด CHECK SUM นั้นจะใช้การบวกข้อมูลทั้งหมดเข้าด้วยกัน (บวกเฉพาะข้อมูลที่เป็นตัวเลขเท่านั้น) บวกกันครั้งละ 1 ไบต์โดยค่าที่เกิน 1 byte นั้นเราจะตัดทิ้ง จากนั้น นำค่าที่ได้ 1 byte นั้นมาทำ 1's complement และ 2's complement เป็นอันเรียบร้อย

ตัวอย่างเช่น ':1C 06 0002 01E5 [CR] [LF]'

	HEXADECIMAL	BINARY
ไบต์เริ่มต้น	1CH	0001 1100
	06H	0000 0110
	00H	0000 0000
	+ 02H	+ 0000 0010
	01H	0000 0001
ไบต์สุดท้าย	C5H	1100 0101
ผลลัพธ์	10AH	1 0000 1010
คิดเฉพาะ 1 byte (8 bit)	0AH	0000 1010
ทำ 1's complement (invert)	F5H	1111 0101
ทำ 2' complement	F5H + 1	1111 0101 + 1
ค่า Check sum ที่ได้	F6H	1111 0110

ข้อมูลที่จะส่งจึงเป็น ':1C 06 0002 01E5 F6 [CR] [LF]'

Edit: 04/02/2022

บริษัท วิศณุและสุภัค จำกัด 102/111-112 หมู่บ้านสินพัฒนาธานี ถนนเทศบาลสงเคราะห์ แขวงลาดยาว เขตจตุจักร กรุงเทพฯ 10900
โทร. (02)591-1916, (02)954-3280-1, แฟกซ์ (02)580-4427 – www.wisco.co.th – info@wisco.co.th