



Wisco Remote Display

DP23



Digital Remote Display

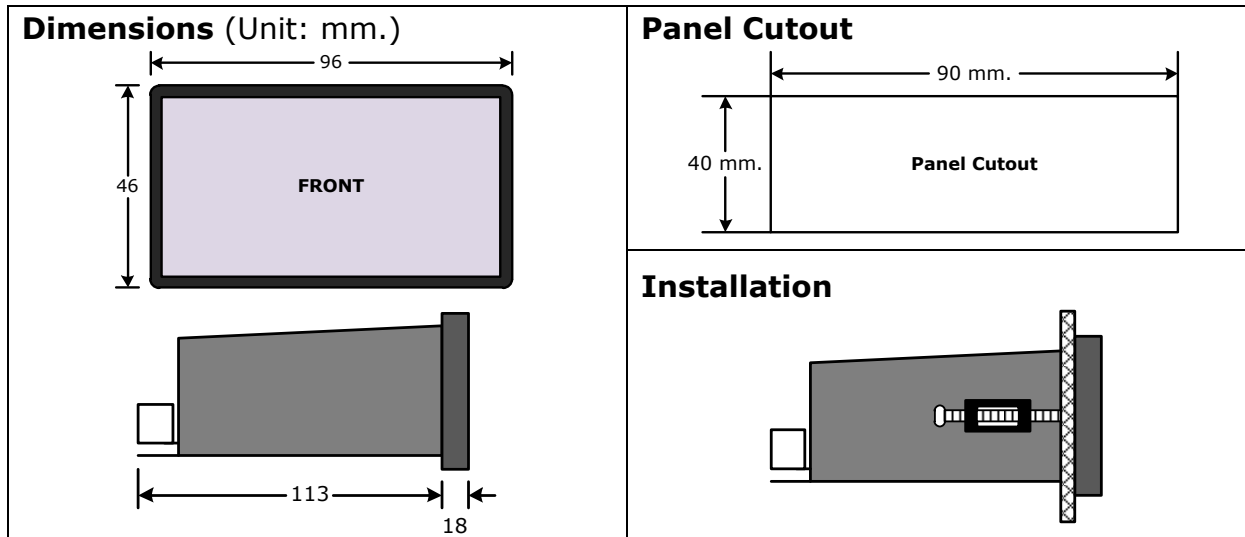
DP23



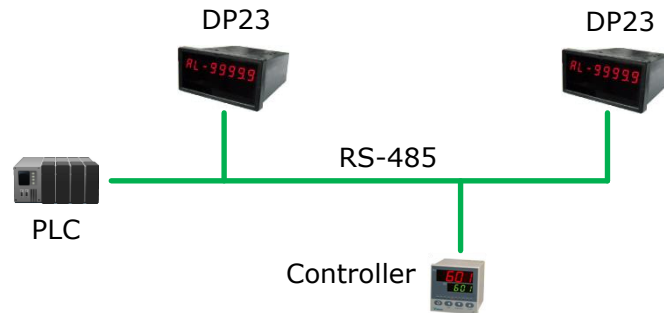
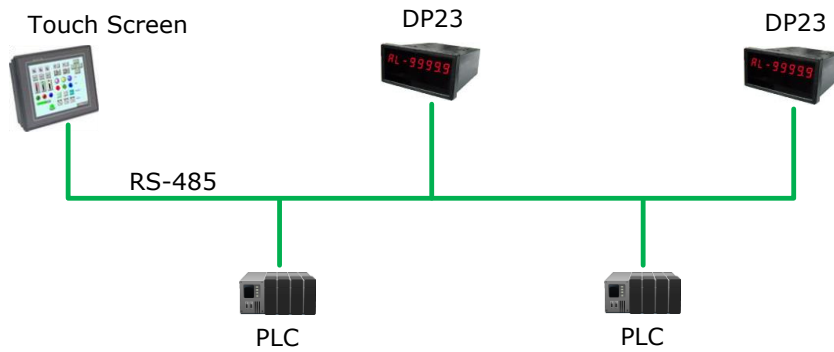
- 7-Segment LED Display
- 8 Digits Display
- Controlled by MODBUS Protocol (RS-485)

Digital Remote Display DP23 เป็นอุปกรณ์แสดงผลด้วย LED 7-segment โดยสั่งให้แสดงผลผ่านทาง Port RS-485 ด้วย MODBUS Protocol

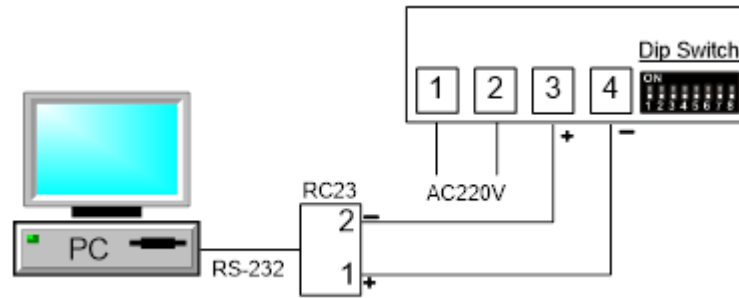
DP23 จะมีหน้าที่เป็น Slave โดยรับคำสั่งจาก Master เช่น PLC, Touch Screen หรือ คอมพิวเตอร์มาแสดงผลตามต้องการได้ถึง 8 Digits



การต่อใช้งาน **DP23**



การต่อใช้งาน DP23



เมื่อดูที่ด้านหลังของ DP23 จะพบ Dipswitch ที่ใช้สำหรับเลือก Station (ตำแหน่งที่ 1 - 5), Baud Rate (ตำแหน่งที่ 6 - 7) และ Protocol (ตำแหน่งที่ 8) มีรายละเอียดดังนี้

ตารางการตั้งค่า Dip Switch

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

การติดต่อกับโมดูลโดยใช้ MODBUS (ASCII) Protocol

โมดูล DP23 สามารถใช้ Protocol MODBUS ในการติดต่อผ่านมาตรฐาน RS-485 (ต่อได้พร้อมกันทั้งหมด 32 ตัว) โดยจะมีรูปแบบของคำสั่งดังต่อไปนี้ (CHAR = Character; 1 CHAR ประกอบไปด้วย 8 Data Bits, 1 Start Bit, 1 Stop Bits, และ 1 (Optional) Parity 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 DP23 สนับสนุนฟังก์ชันพื้นฐานของ Modbus ทั้งหมด 2 ฟังก์ชัน ดังต่อไปนี้

MODBUS Function

- READ HOLDING REGISTER (CODE 03) = Read Register
- PRESET SINGLE REGISTER (CODE 06) = Write Register
- PRESET MULTIPLE REGISTERS (CODE 16) = Write Register

MODBUS Register Table (Hex ดูที่หัวข้อ "ตารางรหัสที่ใช้ในการแสดงข้อความ")

Address	Description
40001	Digit 1
40002	Digit 2
40003	Digit 3
40004	Digit 4
40005	Digit 5
40006	Digit 6
40007	Digit 7
40008	Digit 8

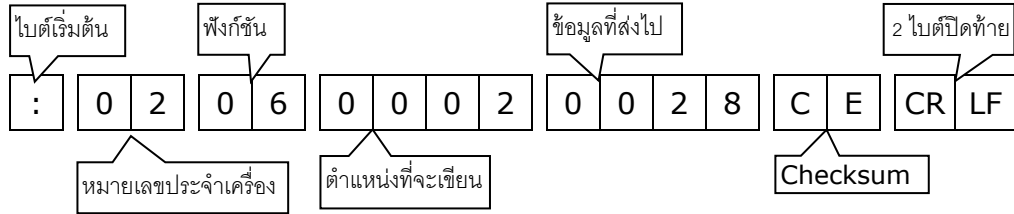


8 7 6 5 4 3 2 1 << Digit แสดงตำแหน่งของ Digit

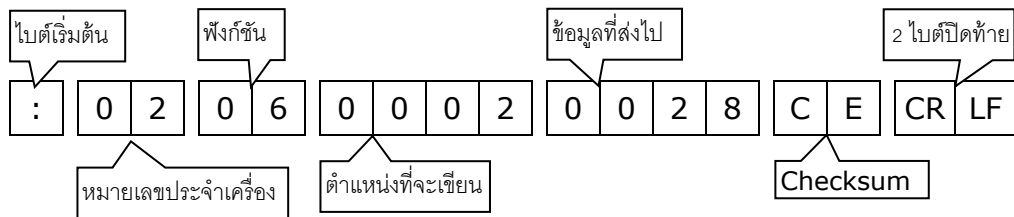
*** รายละเอียดที่เหลือของ Modbus สามารถดูได้จาก 'Modbus Reference Guide' หรือที่ <http://www.modbus.org/specs.php>

ตัวอย่างฟังก์ชัน MODBUS (ASCII) PROTOCOL

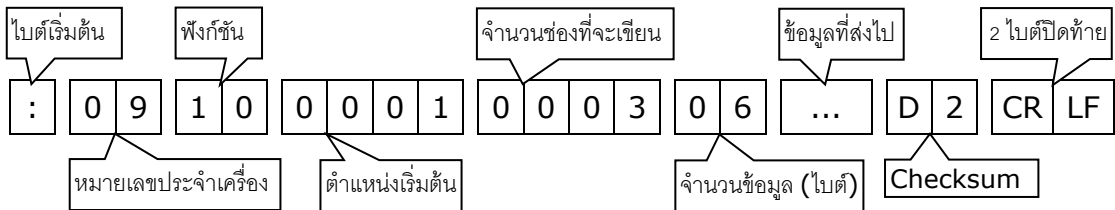
Function Code 06



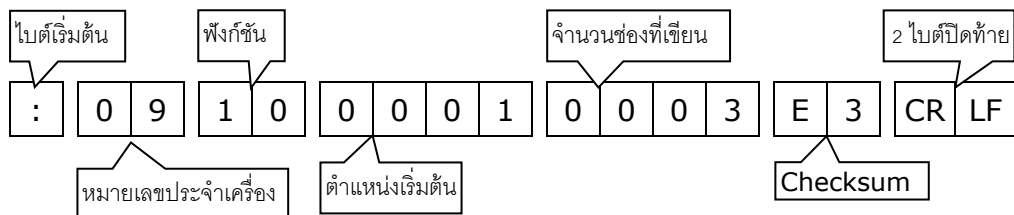
Response



Function Code 16



Response



วิธีคิด CHECK SUM สำหรับ MODBUS (ASCII) Protocol

















ใน DP23 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 `: 0F 06 0001 0028 [CR] [LF]`

	HEXADECIMAL	BINARY
ไบต์เริ่มต้น	0FH	0000 1111
	06H	0000 0110
	00H	0000 0000
	01H	0000 0001
	00H	0000 0000
ไบต์สุดท้าย	28H	0010 1000
ผลลัพธ์	3EH	0011 1110
คิดเฉพาะ 1 byte (8 bit)	3EH	0011 1110
ทำ 1's complement (invert)	C1H	1100 0001
ทำ 2' complement	C1H + 1	1100 0001 + 1
ค่า Check sum ที่ได้	C2H	1100 0010

ข้อมูลที่จะส่งจึงเป็น `: 0F 06 0001 0028 C2 [CR] [LF]`

ตารางรหัสที่ใช้ในการแสดงข้อความ (1/2)

Character	High Byte	Low Byte
	00	00
	00	01
	00	02
	00	03
	00	04
	00	05
	00	06
	00	07
	00	08
	00	09
	00	0A
	00	0B
	00	0C
	00	0D
	00	0E
	00	0F

Character	High Byte	Low Byte
	00	10
	00	11
	00	12
	00	13
	00	14
	00	15
	00	16
	00	17
	00	18
	00	19
	00	1A
	00	1B
	00	1C
	00	1D
	00	1E
	00	1F

ตารางรหัสที่ใช้ในการแสดงข้อความ (2/2)

Character	High Byte	Low Byte
0.	00	20
1.	00	21
2.	00	22
3.	00	23
4.	00	24
5.	00	25
6.	00	26
7.	00	27
8.	00	28
9.	00	29
A.	00	2A
B.	00	2B
C.	00	2C
D.	00	2D
E.	00	2E
F.	00	2F

Character	High Byte	Low Byte
0.	00	30
1.	00	31
2.	00	32
3.	00	33
4.	00	34
5.	00	35
6.	00	36
7.	00	37
8.	00	38
9.	00	39
A.	00	3A
B.	00	3B
C.	00	3C
D.	00	3D
E.	00	3E
F.	00	3F

Edit: 23/09/2021