

ชุดคำสั่งของ ET-REMOTE I/O V1.00

สำหรับข้อกำหนดในการสื่อสารนั้น จะใช้รูปแบบข้อมูลแบบของการสื่อสารระหว่าง Master และ Slave ด้วยการสื่อสารแบบอนุกรม ซึ่งมีรูปแบบและข้อกำหนดเป็นดังนี้

- Baud rate สามารถกำหนดได้เอง เช่น 1200,2400,4800,9600,19200,38400,57600 และ115200 BPS
- Data = 8 Bit
- Parity = None
- Stop Bit = 1
- Hardware Flow Control = None

โดยรูปแบบการสื่อสารดังกล่าวข้างต้นนี้ เป็นข้อกำหนดทางด้าน Software เท่านั้น ส่วนข้อกำหนดทางด้านของระบบฮาร์ดแวร์และการเชื่อมต่อสัญญาณที่จะใช้ในการสื่อสารข้อมูลกันนั้น สามารถเลือกกำหนดได้หลายรูปแบบ ไม่ว่าจะเป็นการเชื่อมต่อด้วยสัญญาณแบบ RS232 หรือ RS422/RS485 ซึ่งในส่วนนี้ขอให้ผู้อ่านศึกษาจากรายละเอียดในเรื่องของการเชื่อมต่อสัญญาณเอง โดยในที่นี้จะขอกล่าวถึงเฉพาะข้อกำหนดทางด้าน SOFTWARE เป็นหลักเท่านั้น โดย Monitor Program หรือ BIOS ของบอร์ด ET-REMOTE I/O V1.0 นั้น ทางทีมงาน อีทีที ได้ทำการพัฒนาขึ้นเพื่อให้ผู้ใช้งานสามารถสั่งงานบอร์ด ET-REMOTE I/O V1.0 ได้โดยง่าย ผ่านทางชุดคำสั่งที่ทาง อีทีที ได้กำหนดขึ้น โดยชุดคำสั่งของ Monitor Program ในรุ่นแรกนั้น มีรหัสรุ่นเป็น “ET-REMOTE I/O V1.00” ซึ่งจะประกอบไปด้วยชุดคำสั่งพื้นฐาน สำหรับสั่งควบคุมการทำงานของ Output แบบ Logic และ คำสั่งสำหรับ ตรวจสอบสถานะการทำงานของ Input แบบ Logic ภายในบอร์ด ET-REMOTE I/O V1.0 ได้โดยง่าย ซึ่งชุดคำสั่งทั้งหมดจะมีอยู่ด้วยกันจำนวน 10 ชุดคำสั่ง โดยชุดคำสั่งดังกล่าวนี้ ถูกพัฒนาขึ้นโดยสามารถสนับสนุนการเชื่อมต่อทางฮาร์ดแวร์ทั้งแบบ RS232 และ RS422/RS485 แบบ Multi-Drop ตามข้อกำหนดของ อีทีที โดยผู้ใช้สามารถต่อใช้งานบอร์ด ET-REMOTE I/O ทั้งแบบการใช้งานเพียงบอร์ดเดียว และการใช้งานหลายๆชุดแบบ Network ก็ได้ตามต้องการ โดยลักษณะการทำงานของบอร์ด ET-REMOTE I/O V1.0 ตามข้อกำหนดของ Monitor Program หรือ BIOS รุ่น ET-REMOTE I/O V1.00 นั้น จะแบ่ง โหมด การทำงานของบอร์ดออกเป็น 2 โหมดด้วยกัน คือ

- **Setup Mode** เป็นโหมดการทำงานสำหรับใช้กำหนดค่า Configuration ต่างๆของบอร์ด เพื่อใช้เป็นเงื่อนไขการทำงานในขณะอยู่ใน Run Mode
- **Run Mode** เป็นโหมดการใช้งานปกติของบอร์ด ซึ่งในโหมดนี้ การทำงานของบอร์ดจะเป็นไปตามเงื่อนไขที่ได้กำหนดไว้แล้วใน Configuration ของบอร์ดจาก Setup Mode ซึ่งผู้ใช้สามารถส่งคำสั่งต่างๆทางพอร์ตสื่อสารอนุกรม เข้ามาสั่งงานบอร์ดได้ตามต้องการ

คู่มือการใช้งาน ชุดคำสั่ง ของ BIOS รุ่น “ET-REMOTE I/O V1.00”

การกำหนดค่า Configuration ให้กับบอร์ด

ก่อนที่จะเริ่มต้นใช้งานบอร์ด ET-REMOTE I/O V1.0 นั้น ในอันดับแรกจะต้องมีการกำหนดเงื่อนไขการทำงานให้กับบอร์ดเสียก่อน ซึ่งเงื่อนไขการทำงานต่าง ๆ นั้น จะถูกเก็บเป็นค่า Configuration ของบอร์ด โดยเงื่อนไขต่าง ๆ เหล่านี้จะทำการกำหนดเพียงครั้งเดียวเท่านั้น ซึ่งหลังจากที่ทำการกำหนดค่า Configuration ให้กับบอร์ดเป็นที่เรียบร้อยแล้ว ค่าของ Configuration ต่าง ๆ ของบอร์ดจะถูกเก็บไว้ในหน่วยความจำภายในตัวเครื่องอย่างถาวร และจะยังคงอยู่ตลอดไป ถึงแม้ว่าจะไม่ได้ทำการจ่ายไฟให้กับตัวเครื่องแล้วก็ตาม โดยเมื่อบอร์ดเริ่มต้นทำงานในโหมดของการใช้งานปกติ หรือ Run Mode แต่ครั้งนั้น เงื่อนไขการทำงานต่าง ๆ ของบอร์ดจะเป็นไปตามค่าที่ได้กำหนดไว้ใน Configuration ของบอร์ดทุกประการ ดังนั้น ก่อนการเริ่มต้นใช้งานบอร์ดในครั้งแรกนั้น จึงจำเป็นต้องให้ผู้ใช้งานจะต้องทำการกำหนดค่า Configuration ของบอร์ดให้เรียบร้อยแล้วเสียก่อน และจะต้องจดจำและทราบว่าเงื่อนไขการทำงานต่าง ๆ ที่ได้ทำการกำหนดไว้ใน Configuration ต่าง ๆ นั้น มีค่าเป็นอย่างไร เพื่อที่จะได้สามารถสั่งงานบอร์ดได้อย่างถูกต้องตามเงื่อนไขที่ได้กำหนดไว้อย่างถูกต้อง

โดยในการกำหนดค่า Configuration ของบอร์ดนั้น จะต้องกระทำในขณะที่บอร์ดอยู่ใน Setup Mode ซึ่งทำได้โดยการเลือก Jumper สำหรับกำหนดโหมดการทำงานของบอร์ด (SETUP/RUN) ไว้ในตำแหน่ง SETUP แล้วเริ่มต้นจ่ายไฟให้กับบอร์ด หรือถ้าได้ทำการจ่ายไฟให้กับบอร์ดไปแล้วก็ให้ทำการกดสวิทช์รีเซ็ตเพื่อให้อบอร์ดเริ่มต้นทำงานใน Setup Mode ซึ่งจะสังเกตเห็น LED Run อยู่ในสถานะติดค้างอยู่ตลอดเวลา ซึ่งก็แสดงว่าบอร์ดพร้อมที่จะทำงานใน Setup Mode แล้วนั่นเอง โดยการตรวจสอบและกำหนดค่า Configuration ต่าง ๆ ให้กับบอร์ดนั้น ทางทีมงาน อีทีที ได้ทำการพัฒนาโปรแกรมสำหรับทำหน้าที่ Setup Configuration ไว้ให้ใช้งานด้วยแล้ว โดยหลังจากทำการกำหนดหน้าที่การทำงานของบอร์ดให้ทำงานใน Setup Mode เป็นที่เรียบร้อยแล้ว ผู้ใช้สามารถสั่ง Run โปรแกรม “SETUP_ET485IO.EXE” เพื่อตรวจสอบและบันทึกค่า Configuration ต่าง ๆ ของบอร์ดได้ตามต้องการ

ET-REMOTE I/O (RS485 Control) Setup Configuration Version 1.00 [Update : 20/October/2004]

Port I/O Direction(0=Output,1=Input)

PORT	P0	P1	P2	P3	P4	P5	P6	P7	Hex
PORT P0	OUT	OUT	OUT	OUT	OUT	OUT	OUT	OUT	00
PORT P1	OUT	OUT	OUT	OUT	OUT	OUT	OUT	OUT	00
PORT P2	IN	IN	IN	IN	OUT	OUT	OUT	OUT	F0
PORT P4	IN	IN	IN	IN	IN	IN	IN	IN	FF
PORT P5	IN	IN	IN	IN	IN	IN	IN	IN	FF

Power-on Port Output Default Logic

PORT	P0	P1	P2	P3	P4	P5	P6	P7	Hex
PORT P0	"1"	"1"	"1"	"1"	"1"	"1"	"1"	"1"	FF
PORT P1	"1"	"1"	"1"	"1"	"1"	"1"	"1"	"1"	FF
PORT P2	"1"	"1"	"1"	"1"	"1"	"1"	"1"	"1"	FF
PORT P4	"1"	"1"	"1"	"1"	"1"	"1"	"1"	"1"	FF
PORT P5	"1"	"1"	"1"	"1"	"1"	"1"	"1"	"1"	FF

Setup Communication Port

Com Port: COM3

Connect: Connect OK

BIOS Name: ET-REMOTE I/O

BIOS Version: V1.00

Open Close

Code Setup and Display

Remote Status: Ready

Remote ID Code: 01

Remote Baudrate: 115200

SAVE CONFIG TO DEFAULT

SAVE CONFIG READ CONFIG

รูปแสดง ลักษณะของโปรแกรม “SETUP_ET485IO.EXE”

Port I/O Direction

ใช้สำหรับเลือกกำหนดหน้าที่การทำงานของ I/O ทั้ง 5 พอร์ต (40 Bit I/O) ว่าต้องการให้ I/O ทำหน้าที่เป็น Input หรือ Output โดยสามารถเลือกกำหนดหน้าที่การทำงานของ I/O แต่ละบิต ได้อย่างอิสระทั้ง 40 บิต ซึ่งในการกำหนดค่า Port I/O Direction นั้น สามารถกำหนดได้ 2 แบบ คือ

- การกำหนดค่าด้วย HEX BYTE ซึ่งเป็นการกำหนดค่าเป็นเลข HEX ขนาด 2 หลัก ระหว่าง '00'-'FF' ในช่องรับค่าของแต่ละพอร์ต โดยถ้าตำแหน่งบิตข้อมูลใดมีค่าเป็น “1” จะเป็นการกำหนดให้ บิต I/O ตำแหน่งนั้นๆของพอร์ต ทำหน้าที่เป็น Input และในทำนองเดียวกัน ถ้าตำแหน่งบิตข้อมูลใดมีค่าเป็น “0” จะเป็นการกำหนดให้ บิต I/O ตำแหน่งนั้นๆของพอร์ต ทำหน้าที่เป็น Output
- การกำหนดค่าด้วยปุ่ม Button ซึ่งจะมีด้วยกันทั้งหมด 40 ปุ่ม โดยแต่ละปุ่มจะทำหน้าที่สำหรับการเลือกการทำงานของ I/O แต่ละบิตในพอร์ต โดยตำแหน่งปุ่มที่อยู่ขวาสุดจะเป็นตำแหน่ง Bit0 ของพอร์ต ส่วนปุ่มที่อยู่ทางด้านซ้ายสุดจะเป็นตำแหน่ง Bit7 ของพอร์ต โดยเมื่อทำการคลิกเมาส์ที่ปุ่มเหล่านี้แต่ละครั้ง จะมีข้อความ IN หรือ OUT ปรากฏให้เห็นบนปุ่มนั้นๆทันที ซึ่งผู้ใช้สามารถทำการคลิกเมาส์บนปุ่มแต่ละปุ่มเพื่อกำหนดหน้าที่การทำงานของ I/O ทั้ง 40 Bit ได้ตามต้องการ

Power-on Port Output Default Logic

ใช้สำหรับกำหนดสถานะการทำงานของเริ่มต้นของพอร์ตที่เป็น Output ในขณะทำการจ่ายไฟให้กับบอร์ดครั้งแรก โดยสามารถกำหนดได้อิสระทั้ง 40 Bit ว่าในขณะทำการจ่ายไฟให้กับบอร์ดในครั้งแรกนั้น ต้องการให้สถานะเริ่มต้นของ Output Bit ใดมีค่าเป็น Logic “0” หรือ Logic “1” โดยค่าที่กำหนดนี้จะมีผลต่อการทำงานของบอร์ดเฉพาะ Bit I/O ที่ถูกกำหนดให้ทำหน้าที่เป็น Output Port เท่านั้น ส่วน Bit I/O ที่ถูกกำหนดให้ทำหน้าที่เป็น Input จะไม่มีผลใดๆต่อค่า Power-on Port Output Default Logic เหล่านี้เลย โดยค่าสถานะของ Bit Output จะมีค่าเริ่มต้นเหมือนกับค่า Power-on Port Output Default Logic เฉพาะหลังการจ่ายไฟให้กับบอร์ดครั้งแรกเท่านั้น ซึ่งหลังจากทำการจ่ายไฟให้กับบอร์ดเป็นที่เรียบร้อยแล้ว ถ้าเกิดการ Reset ขึ้น ค่าสถานะของ Output Port หลังการรีเซ็ตจะกลับมามีค่าเหมือนกับค่าในขณะก่อนการรีเซ็ตเสมอ ซึ่งในการกำหนดค่า Power-on Port Output Default Logic นั้น สามารถกำหนดได้ 2 แบบ คือ

- การกำหนดค่าด้วย HEX BYTE ซึ่งเป็นการกำหนดค่าเป็นเลข HEX ขนาด 2 หลัก ระหว่าง '00'-'FF' ในช่องรับค่าของแต่ละพอร์ต ตามค่าที่ต้องการได้ทันที
- การกำหนดค่าด้วยปุ่ม Button ซึ่งจะมีด้วยกันทั้งหมด 40 ปุ่ม โดยแต่ละปุ่มจะทำหน้าที่สำหรับการเลือกการทำงานของ I/O แต่ละบิตในพอร์ต โดยตำแหน่งปุ่มที่อยู่ขวาสุดจะเป็นตำแหน่ง Bit0 ของพอร์ต ส่วนปุ่มที่อยู่ทางด้านซ้ายสุดจะเป็นตำแหน่ง Bit7 ของพอร์ต โดยเมื่อทำการคลิกเมาส์ที่ปุ่มเหล่านี้แต่ละครั้ง จะมีข้อความ “0” หรือ “1” ปรากฏให้เห็นบนปุ่มนั้นๆทันที ซึ่งผู้ใช้สามารถทำการคลิกเมาส์บนปุ่มแต่ละปุ่มเพื่อกำหนดค่าเริ่มต้นของ Output Port ทั้ง 40 Bit ได้ตามต้องการ

คู่มือการใช้งาน ชุดคำสั่ง ของ BIOS รุ่น “ET-REMOTE I/O V1.00”

Remote ID Code

เป็นค่ารหัส ID Code สำหรับใช้ติดต่อสื่อสารกับบอร์ดในขณะที่อยู่ใน Run Mode โดยสามารถกำหนดเป็นค่าเลข HEX ขนาด 2 หลัก ระหว่าง '00'-'FF'

Remote Baud rate

เป็นค่าความเร็วในการสื่อสารอนุกรมที่จะใช้ในการติดต่อสื่อสารกับบอร์ดในขณะที่อยู่ใน Run Mode ซึ่งสามารถเลือกกำหนดได้ 7 ค่า คือ 1200,2400,4800,9600,19200,38400,57600 และ 115200 BPS

SAVE CONFIG TO DEFAULT

เป็นการสั่งบันทึกค่า Configuration เป็นค่ามาตรฐาน โดยคำสั่งนี้จะทำการกำหนดค่า Configuration ให้กับบอร์ดเองโดยอัตโนมัติ โดยมีคุณสมบัติดังนี้

- I/O Bit1..Bit20 จะถูกกำหนดให้ทำหน้าที่เป็น Input
- I/O Bit21..Bit40 จะถูกกำหนดให้ทำหน้าที่เป็น Output
- Power-on Port Output Default Logic ของทุก Output จะถูกกำหนดให้มีค่าเป็น “1”
- Remote ID Code จะถูกกำหนดให้มีค่าเป็น “01”
- Remote Baud rate จะถูกกำหนดให้มีค่าเป็น 9600 BPS

SAVE CONFIG

เป็นการสั่งบันทึกค่า Configuration ให้กับตัวเครื่องตามค่าตัวเลือกที่ผู้ใช้ได้ทำการเลือกไว้ ในหน้าจอของโปรแกรมในขณะนั้น

READ CONFIG

เป็นการสั่งอ่านค่า Configuration เดิมที่เก็บบันทึกไว้ในตัวเครื่อง ออกมาแสดงให้เห็นทางหน้าจอโปรแกรม ในขณะนั้นๆ

คู่มือการใช้งาน ชุดคำสั่ง ของ BIOS รุ่น “ET-REMOTE I/O V1.00”

ระบบ I/O ของบอร์ด ET-REMOTE I/O V1.0

สำหรับระบบ Input / Output ของบอร์ด ET-REMOTE I/O V1.0 ที่สามารถใช้งานร่วมกับชุดคำสั่งต่างๆของ Monitor program รุ่น “ET-REMOTE I/O V1.00” นั้น จะประกอบไปด้วย Input / Output แบบ Logic จำนวนทั้งหมด 40 Bit โดยแต่ละบิต สามารถกำหนดหน้าที่การใช้งานให้เป็นแบบ Input หรือ Output ก็ได้โดยอิสระตามต้องการ แต่ไม่สามารถกำหนดหน้าที่การทำงานของ I/O ให้เป็นแบบ 2 ทิศทางได้ ต้องกำหนดหน้าที่เป็น Input หรือ Output อย่างใดอย่างหนึ่งเท่านั้น โดยการกำหนดหน้าที่การทำงานของพอร์ตนั้น จะกระทำใน Setup Mode

โดยการเข้าถึง Input / Output ด้วยชุดคำสั่งของ Monitor Program รุ่น “ET-REMOTE I/O V1.00” นั้น สามารถอ้างตำแหน่ง Input / Output ได้ทั้งแบบ Bit และ แบบ Byte โดยถ้าต้องการอ้างถึง Input / Output แบบ Bit นั้น จะต้องอ้างหมายเลข Input / Output ด้วยค่า 01 – 40 แต่ถ้าต้องการอ้างตำแหน่งแบบ Byte ค่าของหมายเลข Input / Output จะมีค่าระหว่าง ‘41’-‘45’ โดยมีการจัดลำดับ Input / Output เป็นดังนี้

พอร์ต P0 (41)	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
	P0.7	P0.6	P0.5	P0.4	P0.3	P0.2	P0.1	P0.0
I/O Number	8	7	6	5	4	3	2	1

พอร์ต P1 (42)	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
	P1.7	P1.6	P1.5	P1.4	P1.3	P1.2	P1.1	P1.0
I/O Number	16	15	14	13	12	11	10	9

พอร์ต P2 (43)	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
	P2.7	P2.6	P2.5	P2.4	P2.3	P2.2	P2.1	P2.0
I/O Number	24	23	22	21	20	19	18	17

พอร์ต P4 (44)	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
	P4.7	P4.6	P4.5	P4.4	P4.3	P4.2	P4.1	P4.0
I/O Number	32	31	30	29	28	27	26	25

พอร์ต P5 (45)	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
	P5.7	P5.6	P5.5	P5.4	P5.3	P5.2	P5.1	P5.0
I/O Number	40	39	38	37	36	35	34	33

ตาราง แสดงตำแหน่ง I/O ของบอร์ด ภายใต Monitor Program รุ่น “ET-REMOTE I/O V1.00”

การเชื่อมต่อ Input / Output ให้กับบอร์ด

สำหรับการใช้งานบอร์ด ET-REMOTE I/O V1.0 ภายใต้เงื่อนไขการทำงานและข้อกำหนดของ Program Monitor รุ่น “ET-REMOTE I/O V1.00” นั้น จะสามารถใช้งาน I/O ของบอร์ด ได้ 2 แบบ คือ

- การใช้งานเป็น Input แบบ Logic โดยสามารถเลือกกำหนดหน้าที่การทำงานของ I/O แต่ละบิต ให้ทำหน้าที่เป็น Input ได้ตามต้องการ ซึ่งอาจเลือกกำหนดบิตใดบิตหนึ่ง หรือทั้งหมดก็ได้
- การใช้งานเป็น Output แบบ Logic โดยสามารถเลือกกำหนดหน้าที่การทำงานของ I/O แต่ละบิต ให้ทำหน้าที่เป็น Output ได้ตามต้องการ ซึ่งอาจเลือกกำหนดบิตใดบิตหนึ่ง หรือทั้งหมดก็ได้

สำหรับในกรณีที่ต้องการเชื่อมต่อระบบ Input / Output ของบอร์ดกับสัญญาณที่ไม่ใช่ TTL Logic นั้น ก็สามารถทำได้ โดยสามารถจัดหาชุด Driver Input / Output แบบต่างๆ มาเชื่อมต่อกับขั้วต่อสัญญาณของบอร์ดได้ โดยขอแนะนำให้เลือกใช้ ชุดเชื่อมต่อที่ให้ผลการทำงานเป็นแบบ Logic “0” กล่าวคือ ถ้าเป็นชุด Input ก็ให้เป็นระบบ Input ที่ให้ผลการตรวจจับเป็นสัญญาณ Logic “0” และถ้าเป็นชุด Output ก็ให้เป็นระบบ Output ที่ทำงาน (ON) ด้วยสัญญาณ Logic “0” เพราะเนื่องจากระบบสัญญาณ I/O ของบอร์ด ET-REMOTE I/O นั้น ใช้สัญญาณ I/O จากไมโครคอนโทรลเลอร์ ตระกูล MCS51 ซึ่งในขณะที่เกิดการรีเซ็ตอยู่ คุณสมบัติของพอร์ต I/O ของ CPU จะมีสถานะเป็น Logic “1” เสมอ ซึ่งถ้านำระบบ Output ที่ทำงานด้วยสัญญาณ Logic “1” มาเชื่อมต่อกับระบบ I/O ของบอร์ดแล้ว อาจทำให้อุปกรณ์ Output นั้นๆ ทำงานในขณะที่ CPU เกิดการรีเซ็ตอยู่ได้ ซึ่งคุณสมบัติของ Output Port จะถูกกำหนดให้มีสถานะตามที่กำหนดไว้ใน Power-on Port Output Default Logic จาก Configuration ของบอร์ด ในช่วงที่พ้นจากสถานะของการรีเซ็ตไปแล้วเท่านั้น แต่ในช่วงขณะกำลังอยู่ในสถานะของการรีเซ็ตอยู่จะไม่สามารถสั่งควบคุมการทำงานของ Output Port ได้ ซึ่งถ้าใช้ระบบ Output ที่ทำงานด้วย Logic “0” จะสามารถป้องกันปัญหาดังกล่าวนี้ได้ ซึ่งทาง อีทีที เองได้ทำการพัฒนาชุด I/O Interface ออกมาสนับสนุนการใช้งานร่วมกับบอร์ด ET-REMOTE I/O ด้วย หลายชุด ตัวอย่างเช่น ชุด ET-OPTO I/O4 ซึ่ง ได้แก่

- ชุด ET-OPTO AC-IN4 เป็นชุด Input ใช้สำหรับตรวจจับแรงดันไฟสลับขนาด 220VAC เพื่อแปลงเป็นสัญญาณ Input แบบ Logic ให้กับบอร์ด ET-REMOTE I/O ได้รับรู้
- ชุด ET-OPTO DC-IN4 เป็นชุด Input ใช้สำหรับตรวจจับแรงดันไฟตรง ขนาด +5V ถึง +24V เพื่อแปลงเป็นสัญญาณ Input แบบ Logic ให้กับบอร์ด ET-REMOTE I/O ได้รับรู้
- ชุด ET-OPTO RELAY4 เป็นชุด Output Relay ใช้สำหรับควบคุมการทำงานของอุปกรณ์ไฟฟ้าต่างๆ โดยสามารถสั่งควบคุมการทำงานจากบอร์ด ET-REMOTE I/O ได้ และให้ผลการทำงานเป็นแบบหน้าสัมผัส ซึ่งสามารถนำไปใช้แทนสวิตช์สำหรับเปิดปิดการทำงานของอุปกรณ์ไฟฟ้าต่างๆได้
- ชุด ET-OPTO DC-OUT4 เป็นชุด DC Output ใช้สำหรับควบคุมการทำงานของอุปกรณ์ไฟฟ้าต่างๆ ที่ทำงานด้วยแรงดันไฟตรง โดยสามารถสั่งควบคุมการทำงานจากบอร์ด ET-REMOTE I/O ได้
- ชุด ET-OPTO AC-OUT4 เป็นชุด AC Output ขนาด 220VAC ใช้สำหรับควบคุมการทำงานของอุปกรณ์ไฟฟ้าต่างๆที่ทำงานด้วยไฟสลับ 220VAC ได้

คู่มือการใช้งาน ชุดคำสั่ง ของ BIOS รุ่น “ET-REMOTE I/O V1.00”

การใช้งานบอร์ด ET-REMOTE I/O

ในการใช้งานบอร์ดนั้น จะต้องกำหนดโหมดการทำงานของบอร์ดให้อยู่ใน Run Mode เสียก่อน โดยการเลือก Jumper สำหรับกำหนดโหมดการทำงานของบอร์ด (Setup/Run) ไว้ทางด้านตำแหน่ง Run แล้วจึงเริ่มต้นจ่ายไฟให้กับบอร์ด หรือถ้าทำการจ่ายไฟให้กับบอร์ดไปก่อนแล้วก็ให้กดสวิทช์รีเซ็ตเพื่อให้บอร์ดเริ่มต้นทำงานใน Run Mode ก็ได้ โดยเมื่อบอร์ดเริ่มต้นเข้าทำงานใน Run Mode นั้น จะสังเกตเห็น LED Run เกิดการกระพริบ ติดและดับเป็นจังหวะอยู่ตลอดเวลา

โดยในการสั่งงานบอร์ด ET-REMOTE I/O นั้น จะต้องทำการส่งคำสั่งผ่านทางพอร์ตสื่อสารอนุกรม มาให้กับบอร์ด ด้วยค่าความเร็ว Baud rate ที่กำหนดไว้ใน Configuration ของบอร์ด ซึ่งในการสั่งงานบอร์ดนั้น จะต้องส่งรหัสคำสั่งไปยังบอร์ด ด้วย ASCII Command ตามข้อกำหนดที่ทาง อีทีที ได้กำหนดไว้ ซึ่งเมื่อบอร์ดได้รับคำสั่งที่ถูกต้องมันจะตอบสนองการทำงานตามข้อกำหนดของคำสั่ง พร้อมกับส่งผลลัพธ์การทำงานกลับออกมาด้วยทุกครั้งเสมอ ซึ่งรูปแบบของชุดคำสั่ง มีรายละเอียดดังต่อไปนี้

รูปแบบคำสั่งสำหรับสั่งงานบอร์ด ET-REMOTE I/O

โดย Format ของคำสั่งที่จะใช้ในการสื่อสารกันในระบบนั้นจะทำการส่งข้อมูลเป็น Frame โดยมีขนาดของข้อมูลเท่ากับ 12 Byte โดยส่งกันด้วยข้อมูลแบบ ASCII จำนวน 11 Byte และจบด้วยรหัส Enter หรือ 0DH(13) อีก 1 Byte ดังรูปแบบต่อไปนี้

เริ่ม	ID Code	Mark1	คำสั่ง	Mark2	I/O Number	Mark3	ข้อมูล	จบ
*	ASCII HEX	:	ASCII	,	ASCII DEC	=	ASCII HEX	Enter
2AH	'00'..'FF'	3AH	'0'..'9'	2CH	'00'..'99'	3DH	'00'..'FF'	0DH

ตารางแสดง โครงสร้างของชุดคำสั่งใน Monitor Program รุ่น “ET-REMOTE I/O V1.00”

- รหัสเริ่มต้นของ Frame ข้อมูล มีขนาด 1 Byte ซึ่งจะต้องกำหนดให้เป็นรหัส ASCII ของตัวอักษร ของเครื่องหมาย “*” หรือ 2AH เท่านั้น
- รหัส ID Code มีขนาด 2 หลัก ใช้สำหรับระบุ ID Code ของอุปกรณ์ Slave ที่ Master ต้องการจะสื่อสารด้วยในขณะนั้น ซึ่งจะต้องกำหนดเป็นรหัส ASCII ของตัวเลข HEX ซึ่งมีค่า '0'..'F' จำนวน 2 หลัก ระหว่าง '00'..'FF' ตามค่ารหัส ID Code จริงที่ได้กำหนดไว้ใน Configuration ของบอร์ด
- รหัสเริ่มต้นคำสั่ง มีขนาด 1 Byte ซึ่งจะต้องกำหนดให้เป็นรหัส ASCII ของตัวอักษรเครื่องหมาย “:” หรือ 3AH เท่านั้น

คู่มือการใช้งาน ชุดคำสั่ง ของ BIOS รุ่น “ET-REMOTE I/O V1.00”

- รหัสคำสั่ง มีขนาด 1 Byte ซึ่งต้องกำหนดเป็นรหัส ASCII ของตัวเลข '0'..'9' ซึ่งมีดังนี้
 - '0' ใช้สำหรับสั่งอ่านค่า BIOS Code ของบอร์ด
 - '1' ใช้สำหรับสั่งอ่านค่า สถานะของ Slave
 - '2' ใช้สำหรับสั่งอ่านค่า Input ของบอร์ด
 - '3' ใช้สำหรับสั่งอ่านค่า Output ของบอร์ด
 - '4' ใช้สำหรับสั่งเขียนค่า Output ให้กับบอร์ด
 - '5' ใช้สำหรับสั่งอ่านค่า Direction ของ P0 จากบอร์ด
 - '6' ใช้สำหรับสั่งอ่านค่า Direction ของ P1 จากบอร์ด
 - '7' ใช้สำหรับสั่งอ่านค่า Direction ของ P2 จากบอร์ด
 - '8' ใช้สำหรับสั่งอ่านค่า Direction ของ P4 จากบอร์ด
 - '9' ใช้สำหรับสั่งอ่านค่า Direction ของ P5 จากบอร์ด
- รหัสเริ่มต้น I/O Number มีขนาด 1 Byte ซึ่งจะต้องกำหนดให้เป็นรหัส ASCII ของตัวอักษรเครื่องหมาย “,” หรือ 2CH เท่านั้น
- รหัส I/O Number มีขนาด 2 Byte ซึ่งจะต้องกำหนดให้เป็นรหัส ASCII ของตัวเลข '0'-'9' จำนวน 2 หลัก โดยกำหนดได้เป็น '01'-'45' หรือ '99' แต่ถ้าเป็นการสั่งงานของคำสั่งที่ไม่เกี่ยวกับการสั่งงาน I/O รหัสนี้จะมีค่าเป็น '00' เสมอ
 - '00' ใช้สำหรับกรณีที่การทำงานของคำสั่งไม่เกี่ยวกับการสั่งงาน I/O เช่น อ่าน BIOS Code
 - '01'-'40' ใช้สำหรับกรณีที่ต้องการสั่งอ่านข้อมูลหรือเขียนข้อมูลแบบ Bit
 - '41' ใช้สำหรับกรณีที่ต้องการสั่งอ่านหรือเขียนข้อมูลแบบ Byte กับ P0
 - '42' ใช้สำหรับกรณีที่ต้องการสั่งอ่านหรือเขียนข้อมูลแบบ Byte กับ P1
 - '43' ใช้สำหรับกรณีที่ต้องการสั่งอ่านหรือเขียนข้อมูลแบบ Byte กับ P2
 - '44' ใช้สำหรับกรณีที่ต้องการสั่งอ่านหรือเขียนข้อมูลแบบ Byte กับ P4
 - '45' ใช้สำหรับกรณีที่ต้องการสั่งอ่านหรือเขียนข้อมูลแบบ Byte กับ P5
 - '99' ใช้สำหรับกรณีที่ต้องการสั่งอ่านหรือเขียนข้อมูลกับทุก I/O
- รหัสเริ่มต้นข้อมูล มีขนาด 1 Byte ซึ่งต้องกำหนดเป็นรหัส ASCII ของตัวอักษร “=” หรือ 3DH เท่านั้น
- รหัสข้อมูล มีขนาด 2 Byte ใช้สำหรับกำหนดข้อมูลที่จะเขียนไปให้กับ I/O ซึ่งถ้าเป็นคำสั่งประเภทอ่านค่าจะต้องกำหนดค่าข้อมูลเป็น '00' เสมอ
- รหัสจบ มีขนาด 1 Byte ต้องกำหนดเป็นรหัส Enter หรือ 0DH(13) เท่านั้น

คู่มือการใช้งาน ชุดคำสั่ง ของ BIOS รุ่น “ET-REMOTE I/O V1.00”

รูปแบบคำสั่งตอบรับจากบอร์ด ET-REMOTE I/O

สำหรับรูปแบบของชุดข้อมูลหรือผลลัพธ์ของคำสั่ง ที่บอร์ด ET-REMOTE I/O จะส่งย้อนกลับมาให้กับอุปกรณ์ Master เมื่อได้รับคำสั่งที่ถูกต้องจาก Master และปฏิบัติตามคำสั่งนั้นเป็นที่เรียบร้อยแล้วนั้น จะมีรูปแบบคล้ายกับชุดคำสั่งที่ส่งไปจาก Master เพียงแต่มีขนาดของ Byte ข้อมูลมากกว่า โดย Format ของคำสั่งที่ตอบกลับจาก ET-REMOTE I/O นั้น จะมีขนาดของข้อมูลตั้งแต่ 13 Byte ไปจนถึง 30 Byte ดังรูปแบบต่อไปนี้

เริ่ม	ID Code	Mark1	รหัสตอบกลับ		Mark2	I/O Number	Mark3	ข้อมูล	จบ
*	ASCII HEX	:	E	ASCII	,	ASCII DEC	=	ASCII HEX	Enter
2AH	'00'..'FF'	3AH	45H	'0'..'9'	2CH	'00'..'99'	3DH	'00'..'FF'	0DH

ตารางแสดง โครงสร้างของข้อมูลตอบกลับจากบอร์ด ET-REMOTE I/O

- รหัสเริ่มต้นของ Frame ข้อมูล มีขนาด 1 Byte ซึ่งเป็นรหัส ASCII ของตัวอักษร ของเครื่องหมาย “*” โดยมีค่าเป็น 2AH
- รหัส ID Code มีขนาด 2 หลัก ใช้สำหรับระบุ ID Code ของอุปกรณ์ Slave ที่เป็นผู้ส่งข้อมูลในขณะนั้น ซึ่งจะเป็นรหัส ASCII ของตัวเลข HEX ซึ่งมีค่า '0'..'F' จำนวน 2 หลักระหว่าง '00'..'FF' ตามค่ารหัส ID Code จริงที่ได้กำหนดไว้ใน Configuration ของบอร์ด
- รหัสเริ่มต้นคำสั่ง มีขนาด 1 Byte ซึ่งจะเป็นรหัส ASCII ของตัวอักษรเครื่องหมาย “:” โดยมีค่าเป็น 3AH
- รหัสตอบรับ มีขนาด 1 Byte ซึ่งจะเป็นรหัส ASCII ของตัวอักษร “E” มีค่าเป็น 45H
- รหัสคำสั่งที่ทำการตอบรับ มีขนาด 1 Byte ซึ่งเป็นรหัส ASCII ของตัวเลข '0'..'9' ซึ่งมีดังนี้
 - '0' ใช้สำหรับแสดงการตอบรับ คำสั่งอ่านค่า BIOS Code ของบอร์ด
 - '1' ใช้สำหรับแสดงการตอบรับ คำสั่งอ่านค่า สถานะของ Slave
 - '2' ใช้สำหรับแสดงการตอบรับ คำสั่งอ่านค่า Input ของบอร์ด
 - '3' ใช้สำหรับแสดงการตอบรับ คำสั่งอ่านค่า Output ของบอร์ด
 - '4' ใช้สำหรับแสดงการตอบรับ คำสั่งเขียนค่า Output ให้กับบอร์ด
 - '5' ใช้สำหรับแสดงการตอบรับ คำสั่งอ่านค่า Direction ของ P0 จากบอร์ด
 - '6' ใช้สำหรับแสดงการตอบรับ คำสั่งอ่านค่า Direction ของ P1 จากบอร์ด
 - '7' ใช้สำหรับแสดงการตอบรับ คำสั่งอ่านค่า Direction ของ P2 จากบอร์ด
 - '8' ใช้สำหรับแสดงการตอบรับ คำสั่งอ่านค่า Direction ของ P4 จากบอร์ด
 - '9' ใช้สำหรับแสดงการตอบรับ คำสั่งอ่านค่า Direction ของ P5 จากบอร์ด
- รหัสเริ่มต้น I/O Number มีขนาด 1 Byte ซึ่งจะเป็นรหัส ASCII ของตัวอักษรเครื่องหมาย “,” มีค่า 2CH

คู่มือการใช้งาน ชุดคำสั่ง ของ BIOS รุ่น “ET-REMOTE I/O V1.00”

- รหัส I/O Number มีขนาด 2 Byte ซึ่งจะเป็นรหัส ASCII ของตัวเลข '0'-'9' จำนวน 2 หลัก โดยอาจมีค่าเป็น '01'-'45' หรือ '99' แต่ถ้เป็นการตอบรับการทำงานของคำสั่งที่ไม่เกี่ยวกับการสั่งงาน I/O รหัสนี้จะมีค่าเป็น '00' เสมอ
 - '00' ใช้สำหรับกรณีตอบรับ คำสั่งที่ไม่เกี่ยวกับการสั่งงาน I/O เช่น อ่าน BIOS Code
 - '01'-'40' ใช้สำหรับกรณีตอบรับ คำสั่งอ่านข้อมูลหรือเขียนข้อมูลแบบ Bit
 - '41' ใช้สำหรับกรณีตอบรับ คำสั่งอ่านหรือเขียนข้อมูลแบบ Byte กับ P0
 - '42' ใช้สำหรับกรณีตอบรับ คำสั่งอ่านหรือเขียนข้อมูลแบบ Byte กับ P1
 - '43' ใช้สำหรับกรณีตอบรับ คำสั่งอ่านหรือเขียนข้อมูลแบบ Byte กับ P2
 - '44' ใช้สำหรับกรณีตอบรับ คำสั่งอ่านหรือเขียนข้อมูลแบบ Byte กับ P4
 - '45' ใช้สำหรับกรณีตอบรับ คำสั่งอ่านหรือเขียนข้อมูลแบบ Byte กับ P5
 - '99' ใช้สำหรับกรณีตอบรับ คำสั่งอ่านหรือเขียนข้อมูลกับทุก I/O
- รหัสเริ่มต้นข้อมูล มีขนาด 1 Byte ซึ่งเป็นรหัส ASCII ของตัวอักษร “=” มีค่า 3DH
- รหัสข้อมูล มีขนาด 2 Byte ไปจนถึง 19 Byte ใช้แสดงผลลัพธ์หรือค่าของข้อมูลที่สั่งอ่านจาก Master
- รหัสจบ มีขนาด 1 Byte ซึ่งเป็นรหัส Enter มีค่าเป็น 0DH(13)

คำสั่งในโหมดทำงานปรกติ (Run Mode)

ในโหมดนี้จะเป็นโหมดการใช้งานปรกติของ ET-REMOTE I/O ซึ่งสามารถที่จะสั่ง อ่านค่า สถานะของ Input หรือ กำหนดค่าการทำงานให้กับ Output ต่างๆได้ตามต้องการ ซึ่งการกำหนดให้ ET-REMOTE I/O ทำงานใน Run โหมดนั้น สามารถทำได้โดยการเลือก Jumper สำหรับเลือกโหมดการทำงานไว้ทางด้าน Run แล้ว Reset บอร์ด โดยจะสังเกตเห็น LED แสดงสถานะการทำงาน กระพริบ เป็นจังหวะอยู่ตลอดเวลา ซึ่งก่อนการใช้งานบอร์ด ET-REMOTE I/O นั้น จะต้องมีการกำหนดค่า Configuration ต่างๆให้กับบอร์ดเสียก่อน ซึ่งได้แก่

- หมายเลข ID Code ของบอร์ด ที่จะใช้ในการสื่อสารใน Run Mode
- ค่าความเร็ว Baud rate ที่จะใช้ในการสื่อสารใน Run Mode
- หน้าที่การทำงานของพอร์ตต่างๆ (Port Direction)
- สถานะการทำงานของ Output Port ขณะจ่ายไฟให้กับบอร์ดครั้งแรก (Power-ON Port Output Default)

ซึ่งหลังจากที่บอร์ด ET-REMOTE I/O V1.0 เริ่มต้นทำงานภายใต้ Run Mode นั้น มันจะใช้เงื่อนไขต่างๆที่กำหนดไว้ใน Configuration เป็นเงื่อนไขในการทำงาน โดยรายละเอียดการ Setup ค่า Configuration ต่างๆ นั้น ขอให้ศึกษารายละเอียดจากหัวข้อการ Setup ค่า Configuration ของบอร์ด

1. การสั่งอ่านรหัส BIOS Code ของ Slave

คำสั่งนี้จะใช้สำหรับสั่งอ่านรหัส BIOS Code ของ Slave โดยถ้ารูปแบบของคำสั่งถูกต้อง Slave จะตอบรับการทำงานของคำสั่ง ด้วยการส่งรหัส BIOS Code กลับออกมา แต่ถ้ารูปแบบคำสั่งไม่ถูกต้องจะไม่มีคำตอบรับใดๆจาก Slave

รูปแบบคำสั่ง = `*XX:0,00=00'+Enter`
การตอบรับ = `*XX:E0,00=ET-REMOTE I/O V1.00'+Enter`
XX = รหัส ID Code มีค่าระหว่าง '00' ถึง 'FF'

ตัวอย่างที่ 1.1 การสั่งอ่านรหัส BIOS Code ของ Slave-01

<code>*01:0,00=00.</code>	‘สั่งอ่านค่า BIOS Code จาก Slave-01
<code>*01:E0,00=ET-REMOTE I/O V1.00.</code>	‘รหัส BIOS คือ ET-REMOTE I/O V1.00

2. การสั่งตรวจสอบสถานะการทำงานของ Slave

สำหรับคำสั่งนี้จะใช้สำหรับตรวจสอบว่า Slave หมายเลข ID Code ที่อ้างถึงนี้ ต่อใช้งานอยู่ในระบบ หรือไม่ และยังทำงานที่เป็นปรกติอยู่หรือไม่ โดยถ้ารูปแบบของคำสั่งถูกต้อง Slave จะตอบรับการทำงานของคำสั่งด้วยการส่งรหัสสถานะของบอร์ดกลับออกมา แต่ถ้ารูปแบบคำสั่งไม่ถูกต้องจะไม่มีการตอบรับใดๆ

รูปแบบคำสั่ง = ‘*XX:1,00=00’+Enter

การตอบรับ = ‘*XX:E1,00=YY’+Enter

XX = รหัส ID Code มีค่าระหว่าง ‘00’ ถึง ‘FF’

YY = รหัสสถานะของ Slave โดยถ้ามีค่าเป็น ‘00’ หมายถึง ปรกติ (Ready) และถ้ามีค่าเป็น ‘01’ หมายถึง ค่า Configuration ที่บันทึกไว้เกิดความผิดพลาด (Error)

ตัวอย่างที่ 2.1 การสั่งตรวจสอบ Status ของ Slave-01 (Slave Ready)

*01:1,00=00↵	‘สั่งอ่านค่า Status ของ Slave-01
*01:E1,00=00↵	‘สถานะของ Slave-01 ปรกติ

ตัวอย่างที่ 2.2 การสั่งตรวจสอบ Status ของ Slave-01 (Slave Error)

*01:1,00=00↵	‘สั่งอ่านค่า Status ของ Slave-01
*01:E1,00=01↵	‘สถานะของ Slave-01 เกิดการ Error ของค่า Configuration

3. การสั่งอ่านค่าสถานะของ Input ของพอร์ต

สำหรับกลุ่มคำสั่งนี้จะใช้สำหรับสั่งอ่านค่าสถานะ Input ของ Slave โดยสามารถใช้ได้ทั้ง การสั่งอ่านแบบ Bit และการสั่งอ่านแบบ Byte โดยถ้ารูปแบบของคำสั่งถูกต้อง Slave จะตอบรับการทำงานของคำสั่งด้วยการส่งค่าสถานะ Input กลับออกมา แต่ถ้าหากว่าหมายเลข I/O ที่ระบุไปในคำสั่งนั้น ไม่ได้ถูกกำหนดหน้าที่การทำงานให้เป็นแบบ Input ไว้ค่าผลลัพธ์ที่ได้จากคำสั่งจะได้ค่าเป็น “1” เสมอ

รูปแบบคำสั่ง = `*XX:2,YY=00'+Enter`

การตอบรับ = `*XX:E2,YY=ZZ..ZZ'+Enter`

XX = รหัส ID Code มีค่าระหว่าง '00' ถึง 'FF'

YY = หมายเลข I/O ของ Slave ที่ต้องการจะสั่งอ่านค่า Input โดยถ้าเป็นการสั่งอ่านค่าแบบ Bit หมายเลข I/O จะมีค่าอยู่ระหว่าง '01'..'40' แต่ถ้าเป็นการสั่งอ่านแบบ Byte หมายเลข I/O จะมีค่าอยู่ระหว่าง '41'..'45' และถ้าต้องการอ่านค่า Input ทั้งหมด (40 Bit Input) หมายเลข I/O จะมีค่าเป็น '99' ดังนี้

- '01'..'40' หมายถึง หมายเลข Bit ของ Input
- '41' หมายถึง หมายเลข Input ของพอร์ต P0 (Input หมายเลข 1-8)
- '42' หมายถึง หมายเลข Input ของพอร์ต P1 (Input หมายเลข 9-16)
- '43' หมายถึง หมายเลข Input ของพอร์ต P2 (Input หมายเลข 17-24)
- '44' หมายถึง หมายเลข Input ของพอร์ต P4 (Input หมายเลข 25-32)
- '45' หมายถึง หมายเลข Input ของพอร์ต P5 (Input หมายเลข 33-40)
- '99' หมายถึง หมายเลข Input ของทุก Port (P0,P1,P2,P4 และ P5)

ZZ = ค่าสถานะ Input ของ Slave โดยถ้าเป็นการสั่งอ่านค่า Input แบบ Bit (หมายเลข I/O มีค่าระหว่าง '01'..'40') ค่า ZZ จะมีขนาด 2 หลัก คือ '00' หมายถึง Logic “0” และ '01' หมายถึง Logic “1” และถ้าเป็นการสั่งอ่านค่า Input แบบ Byte (หมายเลข I/O มีค่าระหว่าง '41'..'45') ค่า ZZ จะมีขนาด 2 หลัก โดยมีค่าเป็น ASCII ของเลข HEX ขนาด 1 Byte คือ '00'..'FF' และถ้าเป็นการสั่งอ่าน Input ทั้งหมด (หมายเลข I/O มีค่า = '99') ค่า ZZ จะมีขนาดเป็น 10 หลัก โดยมีค่าเป็น ASCII ของเลข HEX ขนาด 1 Byte คือ '00'..'FF' จำนวน 5 ชุด เรียงลำดับต่อเนื่องกัน โดย Byte แรกเป็นค่าสถานะ Input ของพอร์ต P0 และ Byte สุดท้ายเป็นค่าสถานะ Input ของพอร์ต P5 ตามลำดับ

คู่มือการใช้งาน ชุดคำสั่ง ของ BIOS รุ่น “ET-REMOTE I/O V1.00”

ตัวอย่างที่ 3.1 การสั่งอ่านค่าสถานะ Input ของบิต 5 จาก Slave-01

*01:2,05=00↵	‘สั่งอ่านค่า Input บิต 5 (P0.4) จาก Slave-01
*01:E2,05=01↵	‘ค่าสถานะ Input บิต 5 (P0.4) ของ Slave-01 มีค่าเป็น Logic “1”

ตัวอย่างที่ 3.2 การสั่งอ่านค่าสถานะ Input ของบิต 18 จาก Slave-01

*01:2,18=00↵	‘สั่งอ่านค่า Input บิต 18 (P2.1) จาก Slave-01
*01:E2,18=00↵	‘ค่าสถานะ Input บิต 18 (P2.1) ของ Slave-01 มีค่าเป็น Logic “0”

ตัวอย่างที่ 3.3 การสั่งอ่านค่าสถานะ Input ของพอร์ต P4 จาก Slave-01

*01:2,44=00↵	‘สั่งอ่านค่า Input แบบ Byte ของ P4 จาก Slave-01
*01:E2,44=3F↵	‘ค่าสถานะ Input ของ P4 จาก Slave-01 มีค่าเป็น 3FH (00111111)

ตัวอย่างที่ 3.4 การสั่งอ่านค่าสถานะ Input ของทุกพอร์ต จาก Slave-01

*01:2,99=00↵	‘สั่งอ่านค่า Input แบบ Byte ของทุกบิต จาก Slave-01
*01:E2,99=0001020304↵	‘ค่าสถานะ Input ของ P0 จาก Slave-01 มีค่าเป็น 00H
	‘ค่าสถานะ Input ของ P1 จาก Slave-01 มีค่าเป็น 01H
	‘ค่าสถานะ Input ของ P2 จาก Slave-01 มีค่าเป็น 02H
	‘ค่าสถานะ Input ของ P4 จาก Slave-01 มีค่าเป็น 03H
	‘ค่าสถานะ Input ของ P5 จาก Slave-01 มีค่าเป็น 04H

4. การส่งอ่านค่าสถานะของ Output

สำหรับกลุ่มคำสั่งนี้จะใช้สำหรับส่งอ่านค่าสถานะ Output ของ Slave โดยสามารถใช้ได้ทั้ง การส่งอ่านแบบ Bit และการส่งอ่านแบบ Byte โดยถ้ารูปแบบของคำสั่งถูกต้อง Slave จะตอบรับการทำงานของคำสั่งด้วยการส่งค่าสถานะ Output กลับออกมา แต่ถ้าหากว่าหมายเลข I/O ที่ระบุไปในคำสั่งนั้น ไม่ได้ถูกกำหนดหน้าที่การทำงานให้เป็นแบบ Output ไว้ค่าผลลัพธ์ที่ได้จากคำสั่งจะได้ค่าเป็น “1” เสมอ

รูปแบบคำสั่ง = `*XX:3,YY=00'+Enter`

การตอบรับ = `*XX:E3,YY=ZZ..ZZ'+Enter`

XX = รหัส ID Code มีค่าระหว่าง '00' ถึง 'FF'

YY = หมายเลข I/O ของ Slave ที่ต้องการจะส่งอ่านค่า Output โดยถ้าเป็นการส่งอ่านค่าแบบ Bit หมายเลข I/O จะมีค่าอยู่ระหว่าง '01'..'40' แต่ถ้าเป็นการส่งอ่านแบบ Byte หมายเลข I/O จะมีค่าอยู่ระหว่าง '41'..'45' และถ้าต้องการอ่านค่า Output ทั้งหมด (40 Bit Output) หมายเลข I/O จะมีค่าเป็น '99' ดังนี้

- '01'..'40' หมายถึง หมายเลข Bit ของ Output
- '41' หมายถึง ค่าหมายเลข Output ของพอร์ต P0 (Output หมายเลข 1-8)
- '42' หมายถึง ค่าหมายเลข Output ของพอร์ต P1 (Output หมายเลข 9-16)
- '43' หมายถึง ค่าหมายเลข Output ของพอร์ต P2 (Output หมายเลข 17-24)
- '44' หมายถึง ค่าหมายเลข Output ของพอร์ต P4 (Output หมายเลข 25-32)
- '45' หมายถึง ค่าหมายเลข Output ของพอร์ต P5 (Output หมายเลข 33-40)
- '99' หมายถึง ค่าหมายเลข Output ของทุก Port (P0,P1,P2,P4 และ P5)

ZZ = ค่าสถานะ Output ของ Slave โดยถ้าเป็นการส่งอ่านค่า Output แบบ Bit (หมายเลข I/O มีค่าระหว่าง '01'..'40') ค่า ZZ จะมีขนาด 2 หลัก คือ '00' หมายถึง Logic “0” และ '01' หมายถึง Logic “1” และถ้าเป็นการส่งอ่านค่า Output แบบ Byte (หมายเลข I/O มีค่าระหว่าง '41'..'45') ค่า ZZ จะมีขนาด 2 หลัก โดยมีค่าเป็น ASCII ของเลข HEX ขนาด 1 Byte คือ '00'..'FF' และถ้าเป็นการส่งอ่าน Output ทั้งหมด (หมายเลข I/O มีค่า = '99') ค่า ZZ จะมีขนาดเป็น 10 หลัก โดยมีค่าเป็น ASCII ของเลข HEX ขนาด 1 Byte คือ '00'..'FF' จำนวน 5 ชุด เรียงลำดับต่อเนื่องกัน โดย Byte แรกเป็นค่าสถานะ Output ของพอร์ต P0 และ Byte สุดท้ายเป็นค่าสถานะ Output ของพอร์ต P5 ตามลำดับ

คู่มือการใช้งาน ชุดคำสั่ง ของ BIOS รุ่น “ET-REMOTE I/O V1.00”

ตัวอย่างที่ 4.1 การสั่งอ่านค่าสถานะ Output ของบิต 5 จาก Slave-01

* <u>01</u> : <u>3</u> , <u>05</u> = <u>00</u> ↵	‘สั่งอ่านค่า Output บิต 5 (P0.4) จาก Slave-01
* <u>01</u> : <u>E3</u> , <u>05</u> = <u>01</u> ↵	‘ค่าสถานะ Output บิต 5 (P0.4) ของ Slave-01 มีค่าเป็น Logic “1”

ตัวอย่างที่ 4.2 การสั่งอ่านค่าสถานะ Output ของบิต 18 จาก Slave-01

* <u>01</u> : <u>3</u> , <u>18</u> = <u>00</u> ↵	‘สั่งอ่านค่า Output บิต 18 (P2.1) จาก Slave-01
* <u>01</u> : <u>E3</u> , <u>18</u> = <u>00</u> ↵	‘ค่าสถานะ Output บิต 18 (P2.1) ของ Slave-01 มีค่าเป็น Logic “0”

ตัวอย่างที่ 4.3 การสั่งอ่านค่าสถานะ Output ของพอร์ต P4 จาก Slave-01

* <u>01</u> : <u>3</u> , <u>44</u> = <u>00</u> ↵	‘สั่งอ่านค่า Output แบบ Byte ของ P4 จาก Slave-01
* <u>01</u> : <u>E3</u> , <u>44</u> = <u>3F</u> ↵	‘ค่าสถานะ Output ของ P4 จาก Slave-01 มีค่าเป็น 3FH (00111111)

ตัวอย่างที่ 4.4 การสั่งอ่านค่าสถานะ Output ของทุกพอร์ต จาก Slave-01

* <u>01</u> : <u>3</u> , <u>99</u> = <u>00</u> ↵	‘สั่งอ่านค่า Output แบบ Byte ของทุกบิต จาก Slave-01
* <u>01</u> : <u>E3</u> , <u>99</u> = <u>0001020304</u> ↵	‘ค่าสถานะ Output ของ P0 จาก Slave-01 มีค่าเป็น 00H
	‘ค่าสถานะ Output ของ P1 จาก Slave-01 มีค่าเป็น 01H
	‘ค่าสถานะ Output ของ P2 จาก Slave-01 มีค่าเป็น 02H
	‘ค่าสถานะ Output ของ P4 จาก Slave-01 มีค่าเป็น 03H
	‘ค่าสถานะ Output ของ P5 จาก Slave-01 มีค่าเป็น 04H

5. การส่งเขียนค่า Output ให้กับ Slave

สำหรับกลุ่มคำสั่งนี้จะใช้สำหรับส่งเขียนค่า Output ไปให้ Slave โดยสามารถใช้ได้ทั้ง การส่งเขียนแบบ Bit และการส่งเขียนแบบ Byte โดยถ้ารูปแบบของคำสั่งถูกต้อง Slave จะตอบรับการทำงานของคำสั่งด้วยการส่งค่าสถานะ Output กลับออกมา แต่ถ้าหากว่าหมายเลข I/O ที่ระบุไปในคำสั่งนั้น ไม่ได้ถูกกำหนดหน้าที่การทำงานให้เป็นแบบ Output ไว้ค่าผลลัพธ์ที่ได้จากคำสั่งจะได้ค่าเป็น “1” เสมอ และ ค่าสถานะของ Output นั้นๆจะไม่เปลี่ยนแปลงตามค่าข้อมูลที่ส่งเขียนไปแต่อย่างใด

รูปแบบคำสั่ง = ‘*XX:4,YY=ZZ’+Enter

การตอบรับ = ‘*XX:E4,YY=ZZ..ZZ’+Enter

XX = รหัส ID Code มีค่าระหว่าง ‘00’ ถึง ‘FF’

YY = หมายเลข I/O ของ Slave ที่ต้องการจะส่งเขียนค่า Output โดยถ้าเป็นการส่งเขียนค่าแบบ Bit หมายเลข I/O จะมีค่าอยู่ระหว่าง ‘01’..’40’ แต่ถ้าเป็นการส่งเขียนค่าแบบ Byte หมายเลข I/O จะมีค่าอยู่ระหว่าง ‘41’..’45’ และถ้าต้องการส่งเขียนค่า Output ทั้งหมด (40 Bit Output) หมายเลข I/O จะมีค่าเป็น ‘99’ ดังนี้

- ‘01’..’40’ หมายถึง หมายเลข Bit ของ Output
- ‘41’ หมายถึง ค่าหมายเลข Output ของพอร์ต P0 (Output หมายเลข 1-8)
- ‘42’ หมายถึง ค่าหมายเลข Output ของพอร์ต P1 (Output หมายเลข 9-16)
- ‘43’ หมายถึง ค่าหมายเลข Output ของพอร์ต P2 (Output หมายเลข 17-24)
- ‘44’ หมายถึง ค่าหมายเลข Output ของพอร์ต P4 (Output หมายเลข 25-32)
- ‘45’ หมายถึง ค่าหมายเลข Output ของพอร์ต P5 (Output หมายเลข 33-40)
- ‘99’ หมายถึง ค่าหมายเลข Output ของทุก Port (P0,P1,P2,P4 และ P5)

ZZ = ค่าสถานะ Output ของ Slave ที่ต้องการจะส่งเขียนค่า โดยถ้าเป็นการส่งเขียนค่า Output แบบ Bit (หมายเลข I/O มีค่าระหว่าง ‘01’..’40’) ค่า ZZ จะมีขนาด 2 หลัก คือ ‘00’ หมายถึง Logic “0” และ ‘01’ หมายถึง Logic “1” และถ้าเป็นการส่งเขียนค่า Output แบบ Byte (หมายเลข I/O มีค่าระหว่าง ‘41’..’45’) ค่า ZZ จะมีขนาด 2 หลัก โดยมีค่าเป็น ASCII ของเลข HEX ขนาด 1 Byte คือ ‘00’..’FF’ และถ้าเป็นการส่งเขียน Output ทั้งหมด (หมายเลข I/O มีค่า = ‘99’) ค่า ZZ จะมีขนาดเป็น 2 หลัก โดยมีค่าเป็น ASCII ของ ‘00’ ซึ่งหมายถึง ให้ทุก Output มีค่าเป็น Logic “0” และ ‘01’ หมายถึงให้ทุก Output มีค่าเป็น Logic “1”

คู่มือการใช้งาน ชุดคำสั่ง ของ BIOS รุ่น “ET-REMOTE I/O V1.00”

ตัวอย่างที่ 5.1 การสั่งเขียนค่า Output ของบิต 5 ให้กับ Slave-01

* <u>01</u> : <u>4</u> , <u>05</u> = <u>00</u> ↵	‘สั่งเขียนค่า Output บิต 5 (P0.4) ให้กับ Slave-01 ด้วย Logic “0”
* <u>01</u> : <u>E4</u> , <u>05</u> = <u>00</u> ↵	‘ค่าสถานะ Output บิต 5 (P0.4) ของ Slave-01 มีค่าเป็น Logic “0”

ตัวอย่างที่ 5.2 การสั่งเขียนค่า Output ของบิต 18 ให้กับ Slave-01

* <u>01</u> : <u>4</u> , <u>18</u> = <u>01</u> ↵	‘สั่งเขียนค่า Output บิต 18 (P2.1) ให้กับ Slave-01 ด้วยค่า Logic “1”
* <u>01</u> : <u>E4</u> , <u>18</u> = <u>01</u> ↵	‘ค่าสถานะ Output บิต 18 (P2.1) ของ Slave-01 มีค่าเป็น Logic “1”

ตัวอย่างที่ 5.3 การสั่งเขียนค่า Output ของพอร์ต P4 ให้กับ Slave-01

* <u>01</u> : <u>4</u> , <u>44</u> = <u>55</u> ↵	‘สั่งเขียนค่า Output แบบ Byte ของ P4 ให้กับ Slave-01 ด้วยค่า 55H
* <u>01</u> : <u>E4</u> , <u>44</u> = <u>55</u> ↵	‘ค่าสถานะ Output ของ P4 จาก Slave-01 มีค่าเป็น 55H (01010101)

ตัวอย่างที่ 5.4 การสั่งเขียนค่า Output ของทุกบิต ให้กับ Slave-01

* <u>01</u> : <u>4</u> , <u>99</u> = <u>00</u> ↵	‘สั่งเขียนค่า Output Logic “0” ให้ Slave-01 ทุกบิต
* <u>01</u> : <u>E4</u> , <u>99</u> = <u>0000000000</u> ↵	‘ค่าสถานะ Output ของ P0 จาก Slave-01 มีค่าเป็น 00H
	‘ค่าสถานะ Output ของ P1 จาก Slave-01 มีค่าเป็น 00H
	‘ค่าสถานะ Output ของ P2 จาก Slave-01 มีค่าเป็น 00H
	‘ค่าสถานะ Output ของ P4 จาก Slave-01 มีค่าเป็น 00H
	‘ค่าสถานะ Output ของ P5 จาก Slave-01 มีค่าเป็น 00H

ตัวอย่างที่ 5.5 การสั่งเขียนค่า Output ของทุกบิต ให้กับ Slave-01

* <u>01</u> : <u>4</u> , <u>99</u> = <u>01</u> ↵	‘สั่งเขียนค่า Output Logic “1” ให้ Slave-01 ทุกบิต
* <u>01</u> : <u>E4</u> , <u>99</u> = <u>FFFFFFFFFF</u> ↵	‘ค่าสถานะ Output ของ P0 จาก Slave-01 มีค่าเป็น FFH
	‘ค่าสถานะ Output ของ P1 จาก Slave-01 มีค่าเป็น FFH
	‘ค่าสถานะ Output ของ P2 จาก Slave-01 มีค่าเป็น FFH
	‘ค่าสถานะ Output ของ P4 จาก Slave-01 มีค่าเป็น FFH
	‘ค่าสถานะ Output ของ P5 จาก Slave-01 มีค่าเป็น FFH

6. การสั่งอ่านค่า Direction ของพอร์ต P0

สำหรับกลุ่มคำสั่งนี้จะใช้สำหรับสั่งอ่านค่าสถานะหน้าที่พอร์ต I/O หรือ Port Direction ของ Slave โดยเป็นการสั่งอ่านค่า Direction แบบ Byte โดยถ้ารูปแบบของคำสั่งถูกต้อง Slave จะตอบรับการทำงานของคำสั่งด้วยการส่งค่าสถานะหน้าที่พอร์ต กลับออกมา

รูปแบบคำสั่ง = `*XX:5,00=00`+Enter

การตอบรับ = `*XX:E5,00=YY`+Enter

XX = รหัส ID Code มีค่าระหว่าง ‘00’ ถึง ‘FF’

YY = หมายเลขค่าสถานะ Direction ของพอร์ต P0 ซึ่งจะมีขนาด 2 หลัก โดยมีค่าเป็น ASCII ของเลข HEX ขนาด 1 Byte คือ ‘00’..’FF’ โดยความหมายของค่า Direction นี้จะแบ่งออกเป็นบิตๆ ตามลำดับของข้อมูลใน HEX BYTE ซึ่ง Bit0 จะเป็นค่า Direction ของ พอร์ต P0 Bit0 (P0.0) ส่วน Bit1 ก็จะเป็นค่า Direction ของพอร์ต P0 Bit1 (P0.1) และ Bit7 ก็จะเป็นค่า Direction ของพอร์ต P0 Bit7 (P0.7) ตามลำดับ โดยถ้าค่าของ Bit ข้อมูลใดมีค่าเป็น “1” จะหมายถึงพอร์ต P0 บิตนั้นมีหน้าที่เป็น Input และถ้าค่าของ Bit ข้อมูลใดมีค่าเป็น “0” จะหมายถึงพอร์ต P0 บิตนั้นๆมีหน้าที่เป็น Output

ตัวอย่างที่ 6.1 การสั่งอ่านค่าสถานะ Direction ของพอร์ต P0 จาก Slave-01

<code>*01:5,00=00↵</code>	‘สั่งอ่านค่า P0 Direction จาก Slave-01
<code>*01:E5,00=00↵</code>	‘ค่าสถานะ Direction ของ P0 จาก Slave-01 มีค่าเป็น 00H ‘พอร์ต P0ทุกบิต (P0.7-P0.0) = Output

ตัวอย่างที่ 6.2 การสั่งอ่านค่าสถานะ Direction ของพอร์ต P0 จาก Slave-01

<code>*01:5,00=00↵</code>	‘สั่งอ่านค่า P0 Direction จาก Slave-01
<code>*01:E5,00=FF↵</code>	‘ค่าสถานะ Direction ของ P0 จาก Slave-01 มีค่าเป็น FFH ‘พอร์ต P0ทุกบิต (P0.7-P0.0) = Input

ตัวอย่างที่ 6.3 การสั่งอ่านค่าสถานะ Direction ของพอร์ต P0 จาก Slave-01

<code>*01:5,00=00↵</code>	‘สั่งอ่านค่า P0 Direction จาก Slave-01
<code>*01:E5,00=0F↵</code>	‘ค่าสถานะ Direction ของ P0 จาก Slave-01 มีค่าเป็น 0FH ‘P0.7-P0.4 = Output ส่วน P0.3-P0.0 = Input

7. การสั่งอ่านค่า Direction ของพอร์ต P1

สำหรับกลุ่มคำสั่งนี้จะใช้สำหรับสั่งอ่านค่าสถานะหน้าที่พอร์ต I/O หรือ Port Direction ของ Slave โดยเป็นการสั่งอ่านค่า Direction แบบ Byte โดยถ้ารูปแบบของคำสั่งถูกต้อง Slave จะตอบรับการทำงานของคำสั่งด้วยการส่งค่าสถานะหน้าที่พอร์ต กลับออกมา

รูปแบบคำสั่ง = `*XX:6,00=00'+Enter`

การตอบรับ = `*XX:E6,00=YY'+Enter`

XX = รหัส ID Code มีค่าระหว่าง '00' ถึง 'FF'

YY = หมายเลขค่าสถานะ Direction ของพอร์ต P1 ซึ่งจะมีขนาด 2 หลัก โดยมีค่าเป็น ASCII ของเลข HEX ขนาด 1 Byte คือ '00'..'FF' โดยความหมายของค่า Direction นี้จะแบ่งออกเป็นบิตๆ ตามลำดับของข้อมูลใน HEX BYTE ซึ่ง Bit0 จะเป็นค่า Direction ของ พอร์ต P1 Bit0 (P1.0) ส่วน Bit1 ก็จะเป็นค่า Direction ของพอร์ต P1 Bit1 (P1.1) และ Bit7 ก็จะเป็นค่า Direction ของพอร์ต P1 Bit7 (P1.7) ตามลำดับ โดยถ้าค่าของ Bit ข้อมูลใดมีค่าเป็น “1” จะหมายถึงพอร์ต P1 บิตนั้นมีหน้าที่เป็น Input และถ้าค่าของ Bit ข้อมูลใดมีค่าเป็น “0” จะหมายถึงพอร์ต P1 บิตนั้นมีหน้าที่เป็น Output

ตัวอย่างที่ 7.1 การสั่งอ่านค่าสถานะ Direction ของพอร์ต P1 จาก Slave-01

<code>*01:6,00=00↵</code>	‘สั่งอ่านค่า P1 Direction จาก Slave-01
<code>*01:E6,00=00↵</code>	‘ค่าสถานะ Direction ของ P1 จาก Slave-01 มีค่าเป็น 00H ‘พอร์ต P1ทุกบิต (P1.7-P1.0) = Output

ตัวอย่างที่ 7.2 การสั่งอ่านค่าสถานะ Direction ของพอร์ต P1 จาก Slave-01

<code>*01:6,00=00↵</code>	‘สั่งอ่านค่า P1 Direction จาก Slave-01
<code>*01:E6,00=FF↵</code>	‘ค่าสถานะ Direction ของ P1 จาก Slave-01 มีค่าเป็น FFH ‘พอร์ต P1ทุกบิต (P1.7-P1.0) = Input

ตัวอย่างที่ 7.3 การสั่งอ่านค่าสถานะ Direction ของพอร์ต P1 จาก Slave-01

<code>*01:6,00=00↵</code>	‘สั่งอ่านค่า P1 Direction จาก Slave-01
<code>*01:E6,00=0F↵</code>	‘ค่าสถานะ Direction ของ P1 จาก Slave-01 มีค่าเป็น 0FH ‘P1.7-P1.4 = Output ส่วน P1.3-P1.0 = Input

8. การสั่งอ่านค่า Direction ของพอร์ต P2

สำหรับกลุ่มคำสั่งนี้จะใช้สำหรับสั่งอ่านค่าสถานะหน้าที่พอร์ต I/O หรือ Port Direction ของ Slave โดยเป็นการสั่งอ่านค่า Direction แบบ Byte โดยถ้ารูปแบบของคำสั่งถูกต้อง Slave จะตอบรับการทำงานของคำสั่งด้วยการส่งค่าสถานะหน้าที่พอร์ต กลับออกมา

รูปแบบคำสั่ง = `*XX:7,00=00'+Enter`

การตอบรับ = `*XX:E7,00=YY'+Enter`

XX = รหัส ID Code มีค่าระหว่าง '00' ถึง 'FF'

YY = หมายเลขค่าสถานะ Direction ของพอร์ต P2 ซึ่งจะมีขนาด 2 หลัก โดยมีค่าเป็น ASCII ของเลข HEX ขนาด 1 Byte คือ '00'..'FF' โดยความหมายของค่า Direction นี้จะแบ่งออกเป็นบิตๆ ตามลำดับของข้อมูลใน HEX BYTE ซึ่ง Bit0 จะเป็นค่า Direction ของ พอร์ต P2 Bit0 (P2.0) ส่วน Bit1 ก็จะเป็นค่า Direction ของพอร์ต P2 Bit1 (P2.1) และ Bit7 ก็จะเป็นค่า Direction ของพอร์ต P2 Bit7 (P2.7) ตามลำดับ โดยถ้าค่าของ Bit ข้อมูลใดมีค่าเป็น “1” จะหมายถึงพอร์ต P2 บิตนั้นมีหน้าที่เป็น Input และถ้าค่าของ Bit ข้อมูลใดมีค่าเป็น “0” จะหมายถึงพอร์ต P2 บิตนั้นๆมีหน้าที่เป็น Output

ตัวอย่างที่ 8.1 การสั่งอ่านค่าสถานะ Direction ของพอร์ต P2 จาก Slave-01

<code>*01:7,00=00↓</code>	‘สั่งอ่านค่า P2 Direction จาก Slave-01
<code>*01:E7,00=00↓</code>	‘ค่าสถานะ Direction ของ P2 จาก Slave-01 มีค่าเป็น 00H ‘พอร์ต P2ทุกบิต (P2.7-P2.0) = Output

ตัวอย่างที่ 8.2 การสั่งอ่านค่าสถานะ Direction ของพอร์ต P2 จาก Slave-01

<code>*01:7,00=00↓</code>	‘สั่งอ่านค่า P2 Direction จาก Slave-01
<code>*01:E7,00=FF↓</code>	‘ค่าสถานะ Direction ของ P2 จาก Slave-01 มีค่าเป็น FFH ‘พอร์ต P2ทุกบิต (P2.7-P2.0) = Input

ตัวอย่างที่ 8.3 การสั่งอ่านค่าสถานะ Direction ของพอร์ต P2 จาก Slave-01

<code>*01:7,00=00↓</code>	‘สั่งอ่านค่า P2 Direction จาก Slave-01
<code>*01:E7,00=0F↓</code>	‘ค่าสถานะ Direction ของ P2 จาก Slave-01 มีค่าเป็น 0FH ‘P2.7-P2.4 = Output ส่วน P2.3-P2.0 = Input

9. การสั่งอ่านค่า Direction ของพอร์ต P4

สำหรับกลุ่มคำสั่งนี้จะใช้สำหรับสั่งอ่านค่าสถานะหน้าที่พอร์ต I/O หรือ Port Direction ของ Slave โดยเป็นการสั่งอ่านค่า Direction แบบ Byte โดยถ้ารูปแบบของคำสั่งถูกต้อง Slave จะตอบรับการทำงานของคำสั่งด้วยการส่งค่าสถานะหน้าที่พอร์ต กลับออกมา

รูปแบบคำสั่ง = `*XX:8,00=00'+Enter`

การตอบรับ = `*XX:E8,00=YY'+Enter`

XX = รหัส ID Code มีค่าระหว่าง '00' ถึง 'FF'

YY = หมายเลขค่าสถานะ Direction ของพอร์ต P4 ซึ่งจะมีขนาด 2 หลัก โดยมีค่าเป็น ASCII ของเลข HEX ขนาด 1 Byte คือ '00'..'FF' โดยความหมายของค่า Direction นี้จะแบ่งออกเป็นบิตๆ ตามลำดับของข้อมูลใน HEX BYTE ซึ่ง Bit0 จะเป็นค่า Direction ของ พอร์ต P4 Bit0 (P4.0) ส่วน Bit1 ก็จะเป็นค่า Direction ของพอร์ต P4 Bit1 (P4.1) และ Bit7 ก็จะเป็นค่า Direction ของพอร์ต P4 Bit7 (P4.7) ตามลำดับ โดยถ้าค่าของ Bit ข้อมูลใดมีค่าเป็น “1” จะหมายถึงพอร์ต P4 บิตนั้นมีหน้าที่เป็น Input และถ้าค่าของ Bit ข้อมูลใดมีค่าเป็น “0” จะหมายถึงพอร์ต P4 บิตนั้นไม่มีหน้าที่เป็น Output

ตัวอย่างที่ 9.1 การสั่งอ่านค่าสถานะ Direction ของพอร์ต P4 จาก Slave-01

<code>*01:8,00=00↵</code>	‘สั่งอ่านค่า P4 Direction จาก Slave-01
<code>*01:E8,00=00↵</code>	‘ค่าสถานะ Direction ของ P4 จาก Slave-01 มีค่าเป็น 00H ‘พอร์ต P0ทุกบิต (P4.7-P4.0) = Output

ตัวอย่างที่ 9.2 การสั่งอ่านค่าสถานะ Direction ของพอร์ต P4 จาก Slave-01

<code>*01:8,00=00↵</code>	‘สั่งอ่านค่า P4 Direction จาก Slave-01
<code>*01:E8,00=FF↵</code>	‘ค่าสถานะ Direction ของ P4 จาก Slave-01 มีค่าเป็น FFH ‘พอร์ต P4ทุกบิต (P4.7-P4.0) = Input

ตัวอย่างที่ 9.3 การสั่งอ่านค่าสถานะ Direction ของพอร์ต P4 จาก Slave-01

<code>*01:8,00=00↵</code>	‘สั่งอ่านค่า P4 Direction จาก Slave-01
<code>*01:E8,00=0F↵</code>	‘ค่าสถานะ Direction ของ P4 จาก Slave-01 มีค่าเป็น 0FH ‘P4.7-P4.4 = Output ส่วน P4.3-P4.0 = Input

10. การสั่งอ่านค่า Direction ของพอร์ต P5

สำหรับกลุ่มคำสั่งนี้จะใช้สำหรับสั่งอ่านค่าสถานะหน้าที่พอร์ต I/O หรือ Port Direction ของ Slave โดยเป็นการสั่งอ่านค่า Direction แบบ Byte โดยถ้ารูปแบบของคำสั่งถูกต้อง Slave จะตอบรับการทำงานของคำสั่งด้วยการส่งค่าสถานะหน้าที่พอร์ต กลับออกมา

รูปแบบคำสั่ง = `*XX:9,00=00'+Enter`

การตอบรับ = `*XX:E9,00=YY'+Enter`

XX = รหัส ID Code มีค่าระหว่าง '00' ถึง 'FF'

YY = หมายเลขค่าสถานะ Direction ของพอร์ต P5 ซึ่งจะมีขนาด 2 หลัก โดยมีค่าเป็น ASCII ของเลข HEX ขนาด 1 Byte คือ '00'..'FF' โดยความหมายของค่า Direction นี้จะแบ่งออกเป็นบิตๆ ตามลำดับของข้อมูลใน HEX BYTE ซึ่ง Bit0 จะเป็นค่า Direction ของ พอร์ต P5 Bit0 (P5.0) ส่วน Bit1 ก็จะเป็นค่า Direction ของพอร์ต P5 Bit1 (P5.1) และ Bit7 ก็จะเป็นค่า Direction ของพอร์ต P5 Bit7 (P5.7) ตามลำดับ โดยถ้าค่าของ Bit ข้อมูลใดมีค่าเป็น “1” จะหมายถึงพอร์ต P5 บิตนั้นมีหน้าที่เป็น Input และถ้าค่าของ Bit ข้อมูลใดมีค่าเป็น “0” จะหมายถึงพอร์ต P5 บิตนั้นมีหน้าที่เป็น Output

ตัวอย่างที่ 10.1 การสั่งอ่านค่าสถานะ Direction ของพอร์ต P5 จาก Slave-01

<code>*01:9,00=00↵</code>	‘สั่งอ่านค่า P5 Direction จาก Slave-01
<code>*01:E9,00=00↵</code>	‘ค่าสถานะ Direction ของ P5 จาก Slave-01 มีค่าเป็น 00H ‘พอร์ต P5ทุกบิต (P5.7-P5.0) = Output

ตัวอย่างที่ 10.2 การสั่งอ่านค่าสถานะ Direction ของพอร์ต P5 จาก Slave-01

<code>*01:9,00=00↵</code>	‘สั่งอ่านค่า P5 Direction จาก Slave-01
<code>*01:E9,00=FF↵</code>	‘ค่าสถานะ Direction ของ P5 จาก Slave-01 มีค่าเป็น FFH ‘พอร์ต P5ทุกบิต (P5.7-P5.0) = Input

ตัวอย่างที่ 10.3 การสั่งอ่านค่าสถานะ Direction ของพอร์ต P5 จาก Slave-01

<code>*01:9,00=00↵</code>	‘สั่งอ่านค่า P5 Direction จาก Slave-01
<code>*01:E9,00=0F↵</code>	‘ค่าสถานะ Direction ของ P5 จาก Slave-01 มีค่าเป็น 0FH ‘P5.7-P5.4 = Output ส่วน P5.3-P5.0 = Input