



# DEV-MI0430FT-2-V1

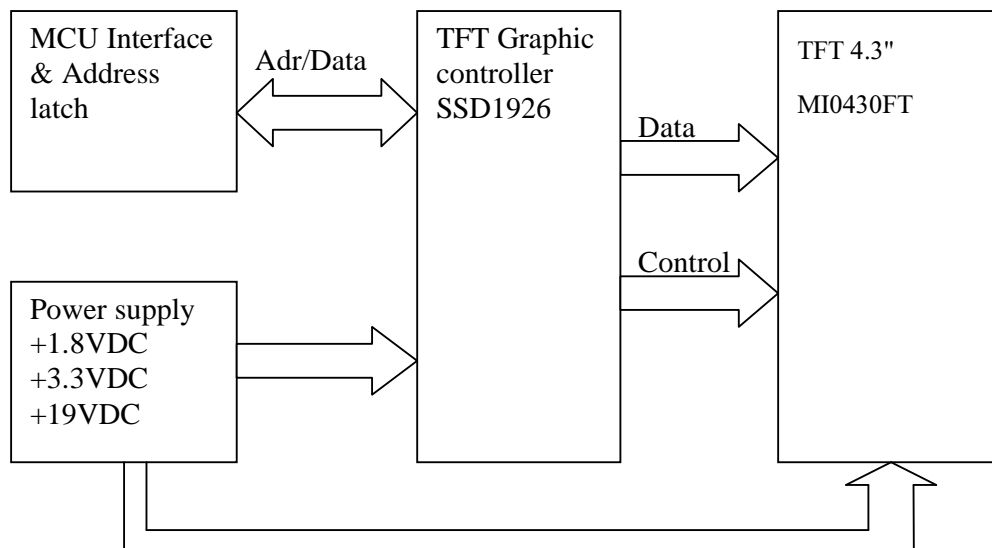
REV1

## คู่มือการใช้งาน DEV-MI0430FT-2-V1

DEV-MI0430FT-2-V1 เป็น TFT module ขนาดของหน้าจอ 4.3" ใช้ SSD1926 เป็น Graphic driver ร่วมกับจอ TFT ของ MIT รุ่น MI0430FT ขนาด 4.3" ความละเอียดของจอเป็นแบบ WQVGA 480x272 pixel

### Feature

- ขนาดจอ 4.3"
- ความละเอียดของจอ 480x272 pixel (WQVGA)
- ความละเอียดของสี 16bpp (65536 สี)
- หน่วยความจำเก็บข้อมูลสำหรับแสดงผล 256KB
- การเชื่อมต่อเป็นแบบขนาน Multiplex ขาข้อมูลกับขา Address สามารถเชื่อมต่อได้ทั้ง MCU 8bit, 16bit และ 32bit
- มี DC/DC converter สำหรับจ่ายไฟ Backlight ภายในบอร์ด
- มี Voltage regulator ภายในบอร์ด ผู้ใช้งานเพียงจ่ายไฟเลี้ยง +5VDC เท่านั้น
- Connector เชื่อมต่อกับ Microcontroller เป็นแบบ pin header 17x2 ระยะ pitch 2.0mm



## รายละเอียดของ Connector ที่เชื่อมต่อกับ Microcontroller

ขา	ชื่อ	I/O/P	รายละเอียด
1	AB17	I	ขา Address 17
2	AB0	I	ขา Address 0
3	HLATCH	I	ขา Latch address 9-16 (active high)
4	LLATCH	I	ขา Latch address 1-15 (active high)
5	GND	P	ขา Ground
6	GND	P	ขา Ground
7	GND	P	ขา Ground
8	+5V	P	ขาไฟเลี้ยง +5VDC
9	+5V	P	ขาไฟเลี้ยง +5VDC
10	+5V	P	ขาไฟเลี้ยง +5VDC
11	ADB1	I/O	ขา Address 2 และ Data 1
12	ADB0	I/O	ขา Address 1 และ Data 0
13	ADB3	I/O	ขา Address 4 และ Data 3
14	ADB2	I/O	ขา Address 3 และ Data 2
15	ADB5	I/O	ขา Address 6 และ Data 5
16	ADB4	I/O	ขา Address 5 และ Data 4
17	ADB7	I/O	ขา Address 8 และ Data 7
18	ADB6	I/O	ขา Address 7 และ Data 6
19	ADB9	I/O	ขา Address 10 และ Data 9
20	ADB8	I/O	ขา Address 9 และ Data 8
21	ADB11	I/O	ขา Address 12 และ Data 11
22	ADB10	I/O	ขา Address 11 และ Data 10
23	ADB13	I/O	ขา Address 14 และ Data 13
24	ADB12	I/O	ขา Address 13 และ Data 12
25	ADB15	I/O	ขา Address 16 และ Data 15
26	ADB14	I/O	ขา Address 15 และ Data 14
27	NC	-	ไม่เชื่อมต่อ
28	#RESET	I	ขา Reset (active low)

29	#WAIT	O	ขา Wait (active low) บอก MCU ว่า Module ยังไม่พร้อมรับข้อมูล
30	#CS	I	ขา Chip select (active low)
31	#RD	I	ขา Read (active low) เป็นขาส่งสัญญาณอ่านข้อมูลจาก module
32	#WE0	I	ขา Write (active low) เป็นส่งสัญญาณเขียนข้อมูลลง module
33	#WE1	I	ขา high data byte เลือก อ่าน/เขียน ข้อมูล high byte
34	MEM/#REG	I	ขา Memory/Register เลือกอ่านเขียนข้อมูลจาก "1" RAM, "0" Register

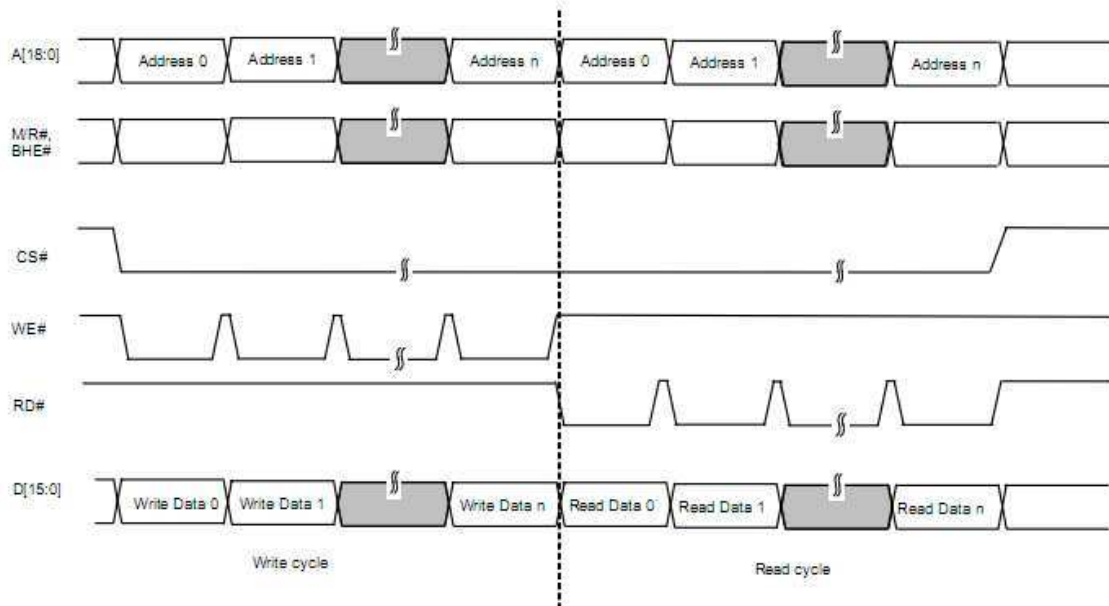
### ตำแหน่งขาสัญญาณของ CN1

1	NC		ไม่มีการเชื่อมต่อ
2	NC		ไม่มีการเชื่อมต่อ
3	YU		ขาสัญญาณ Touch panel (ตัวเลือกเพิ่มเติม)
4	XL		ขาสัญญาณ Touch panel (ตัวเลือกเพิ่มเติม)
5	YD		ขาสัญญาณ Touch panel (ตัวเลือกเพิ่มเติม)
6	XR		ขาสัญญาณ Touch panel (ตัวเลือกเพิ่มเติม)

\*\*\* I = Input, O = Output, P = Power

SSD1926 สามารถตั้งค่าการเชื่อมต่อกับ Microcontroller ได้หลากหลายรูปแบบ แต่ TFT Module ตัวนี้จะใช้รูปแบบการเชื่อมต่อเป็นแบบ Generic #2 addressing Mode และค่า default สำหรับการจัดเรียงข้อมูลจะเป็นแบบ Little endian

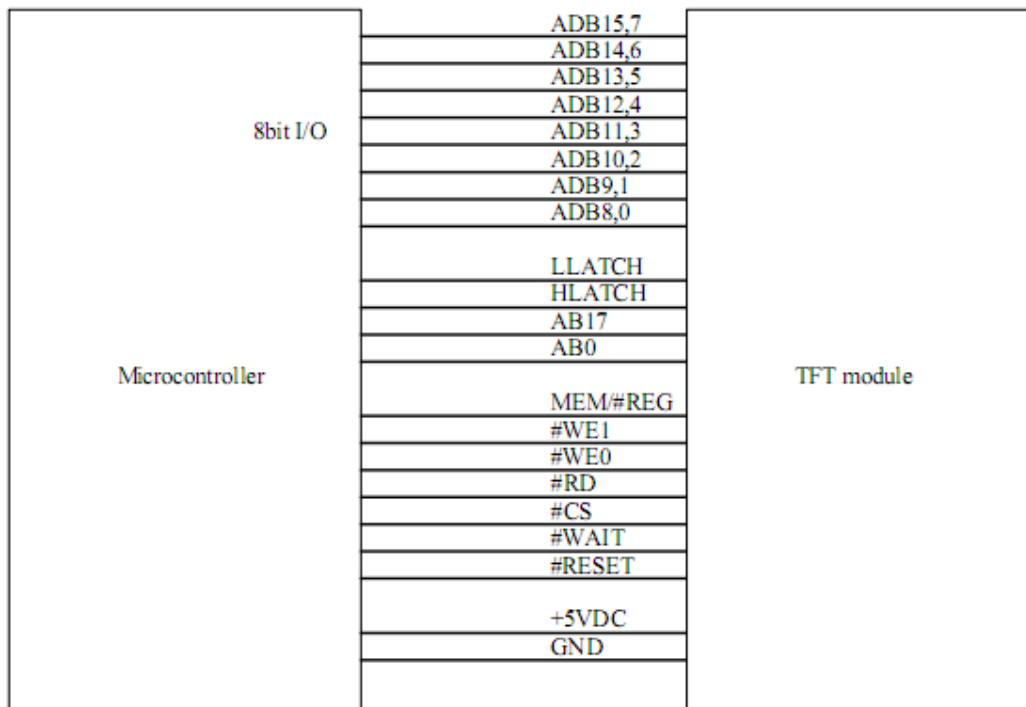
### 7.3.2 Generic #2 addressing Mode



## ตารางแสดงสถานะของขาต่างๆ ตอน อ่าน/เขียน ข้อมูล

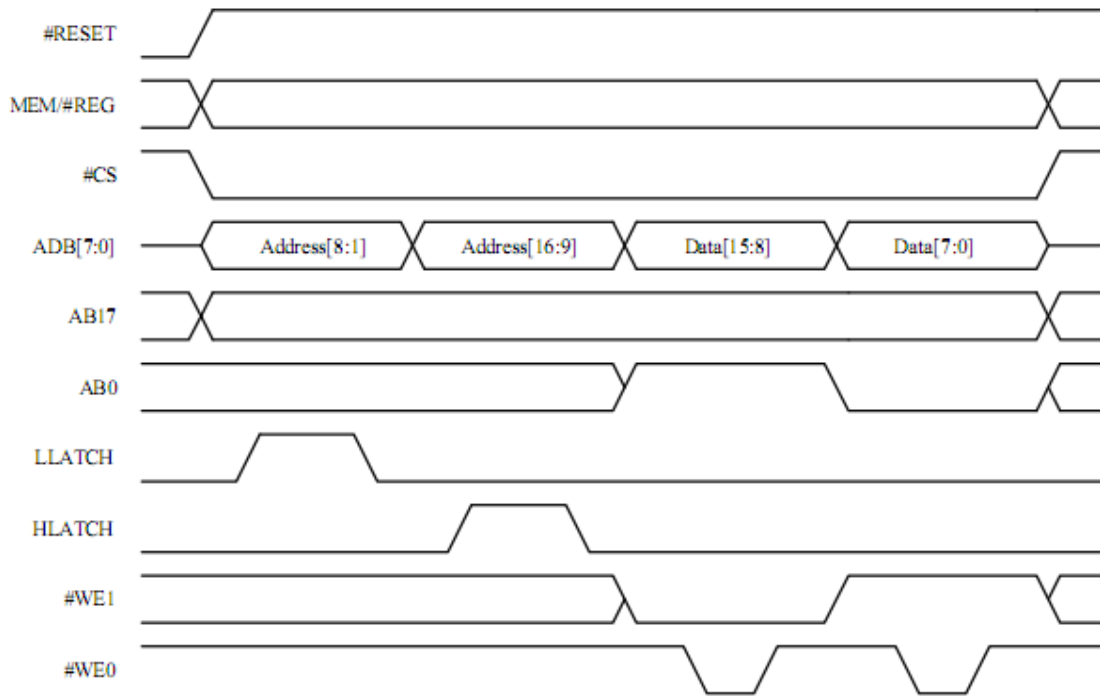
MCU Mode (Endian)	A0	RD/WR#	RD#	WE1#	WE0#	Operation
Generic#1 (Big)	X	0	0	1	1	Word read
	X	0	1	1	1	High byte read 2N
	X	1	0	1	1	Low byte read 2N+1
	X	1	1	0	0	Word write
	X	1	1	0	1	High byte write 2N
	X	1	1	1	0	Low byte write 2N+1
Generic#1 (Little)	X	0	0	1	1	Word read
	X	0	1	1	1	High byte read 2N+1
	X	1	0	1	1	Low byte read 2N
	X	1	1	0	0	Word write
	X	1	1	0	1	High byte write 2N+1
	X	1	1	1	0	Low byte write 2N
Generic#2 (Big)	0	X	0	0	1	Word read
	0	X	0	1	1	High byte read 2N
	1	X	0	0	1	Low byte read 2N+1
	0	X	1	0	0	Word write
	0	X	1	1	0	High byte write 2N
	1	X	1	0	0	Low byte write 2N+1
Generic#2 (Little)	0	X	0	0	1	Word read
	1	X	0	0	1	High byte read 2N+1
	0	X	0	1	1	Low byte read 2N
	0	X	1	0	0	Word write
	1	X	1	0	0	High byte write 2N+1
	0	X	1	1	0	Low byte write 2N

## การเชื่อมต่อกับ microcontroller แบบ 8bit



การเชื่อมต่อกับ microcontroller แบบ 8bit จะช่วยลดเวลาสัญญาณในการเชื่อมต่อกับ TFT module ลงได้ แต่ความเร็วในการเชื่อมต่อก็จะลดลงตามไปด้วย การเชื่อมต่อแบบ 8bit นี้สามารถนำไปเชื่อมต่อกับ PMP (Parallel Master Port) ที่มีอยู่ใน microcontroller ตระกูล PIC24 ได้ และความเร็วในการเชื่อมต่อจะสูงกว่าเรา มาเขียนโปรแกรมติดต่อกอง

### Timing diagram สำหรับการเขียนข้อมูลแบบ 8bit



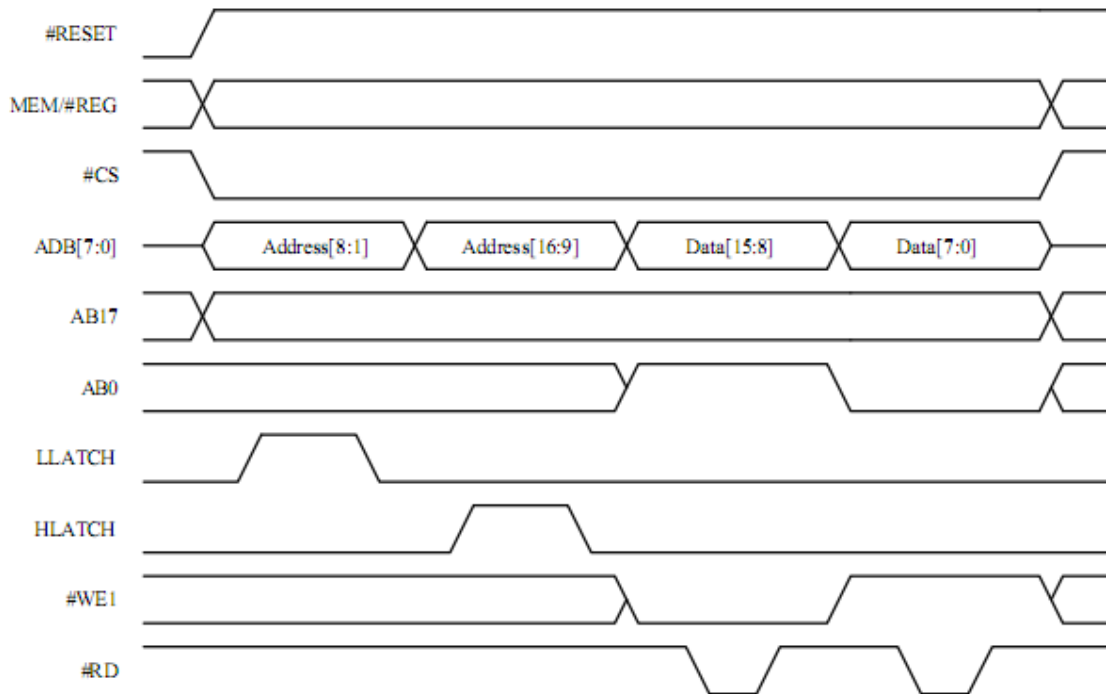
Timing diagram สำหรับเชื่อมต่อกับ microcontroller แบบ 8bit โดยการจัดเรียงข้อมูลเป็นแบบ Little endian

### ขั้นตอนการเขียนข้อมูลไปยัง TFT Module แบบ 8bit

1. กำหนดว่าจะ เขียน ข้อมูล ลงหน่วยความจำ หรือ Register ด้วยขา MEM/#REG "0"=เขียนข้อมูล ลง Register, "1"=เขียนข้อมูลลงหน่วยความจำ
2. ขา #CS="0" เพื่อให้ Module พร้อมรับข้อมูล
3. กำหนดขา Address AB17 ว่าจะเขียนข้อมูลลงตำแหน่งไหน
4. ส่งตำแหน่ง Address[8:1] ไปที่ขา ADB[7:0] หลังจากนั้นสร้างสัญญาณ Latch address ไปที่ขา LLATCH เพื่อ Latch address[8:1] ไปยัง buffer
5. ส่งตำแหน่ง Address[16:9] ไปที่ขา ADB[7:0] หลังจากนั้นสร้างสัญญาณ Latch address ไปที่ขา HLATCH เพื่อ Latch address[16:9] ไปยัง buffer
6. ให้ขา AB0="1" ขา #WE1="0" หลังจากนั้นส่งข้อมูล Data[15:8] ไปที่ขา ADB[7:0]

7. สร้างสัญญาณเขียนข้อมูลไปที่ขา #WE0 เพื่อเขียนข้อมูลไปยัง TFT module
8. ให้ขา AB0="0" ขา #WE1="1" หลังจากนั้นส่งข้อมูล Data[7:0] ไปที่ขา ADB[7:0]
9. สร้างสัญญาณเขียนข้อมูลไปที่ขา #WE0 เพื่อเขียนข้อมูลไปยัง TFT module
10. ขา #CS="1" หลังจากที่เขียนข้อมูลเสร็จแล้ว

#### Timing diagram สำหรับการอ่านข้อมูลแบบ 8bit

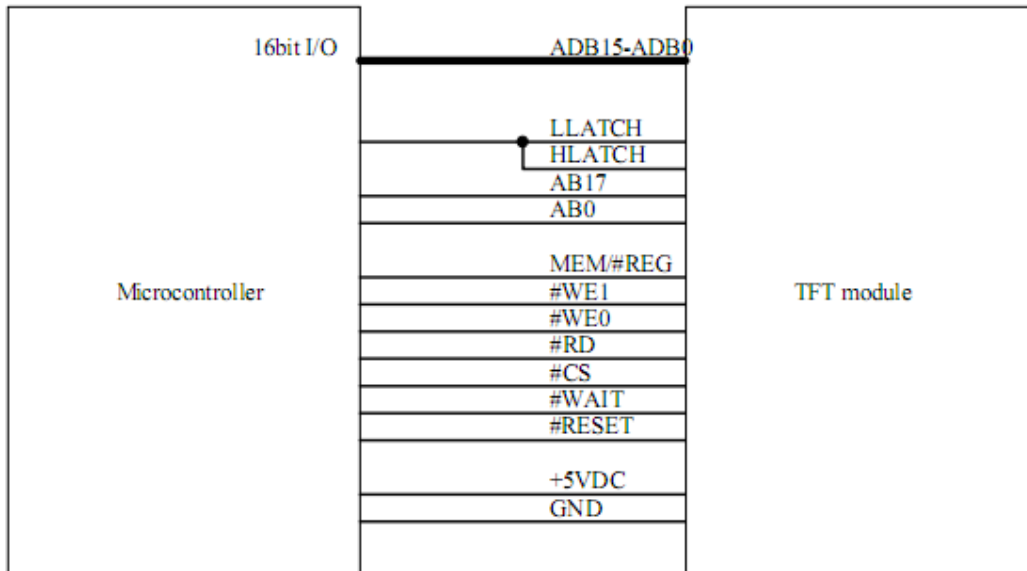


#### ขั้นตอนการอ่านข้อมูลจาก TFT Module แบบ 8bit

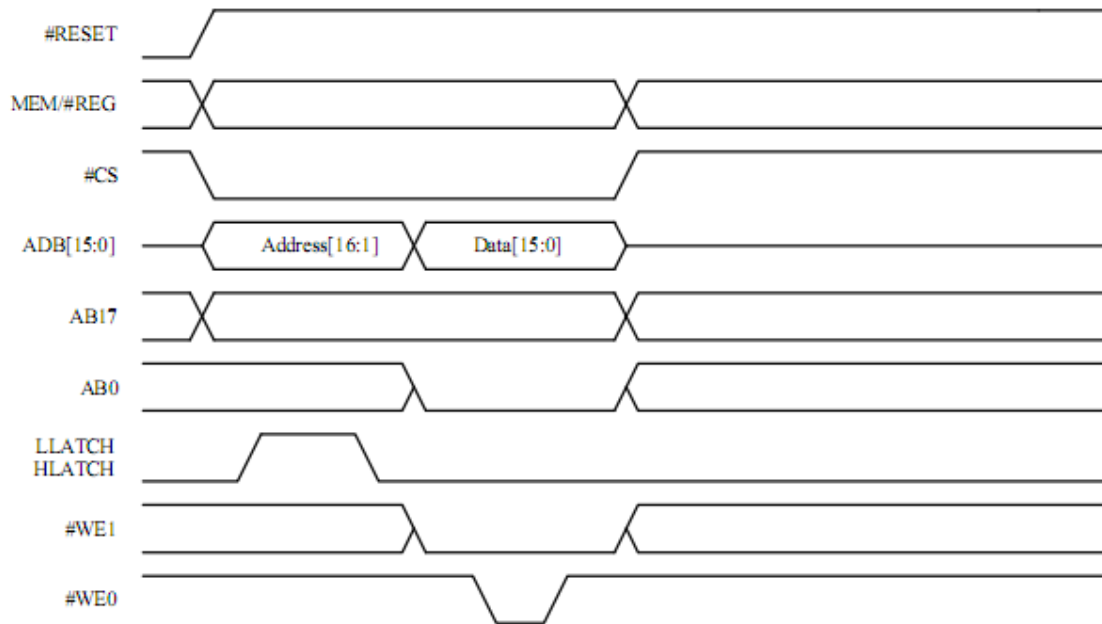
1. กำหนดว่าจะอ่านข้อมูล จากหน่วยความจำ หรือ Register ด้วยขา MEM/#REG "0"=อ่านข้อมูลจาก Register, "1"=อ่านข้อมูลจากหน่วยความจำ
2. ขา #CS="0" เพื่อให้ Module พร้อมรับข้อมูล
3. กำหนดขา Address AB17 ที่จะอ่านข้อมูลจากตำแหน่งไหน
4. ส่งตำแหน่ง Address[8:1] ไปที่ขา ADB[7:0] หลังจากนั้นสร้างสัญญาณ Latch address ไปที่ขา LLATCH เพื่อ Latch address[8:1] ไปยัง buffer
5. ส่งตำแหน่ง Address[16:9] ไปที่ขา ADB[7:0] หลังจากนั้นสร้างสัญญาณ Latch address ไปที่ขา HLATCH เพื่อ Latch address[16:9] ไปยัง buffer
6. ให้ขา AB0="1" ขา #WE1="0"
7. สร้างสัญญาณอ่านข้อมูลไปที่ขา #WE0 เพื่ออ่านข้อมูลจาก TFT module
8. อ่านข้อมูล Data[15:8] จากขา ADB[7:0]

9. ให้ขา AB0="0" ขา #WE1="1"
10. สร้างสัญญาณอ่านข้อมูลไปที่ขา #RD เพื่ออ่านข้อมูลจาก TFT module
11. อ่านข้อมูล Data[7:0] จากขา ADB[7:0]
12. ขา #CS="1" หลังจากทีอ่านข้อมูลเสร็จแล้ว

#### การเชื่อมต่อกับ microcontroller แบบ 16bit



การเชื่อมต่อกับ microcontroller แบบ 16bit เหมาะกับการนำ TFT Module ไปใช้งานร่วมกับ Microcontroller ขนาด 16bit หรือจะนำไปใช้ร่วมกับ PIC32 ที่มี PMP Module แบบ 16bit ซึ่งความเร็วในการรับส่งข้อมูลจะสูงกว่าแบบ 8bit

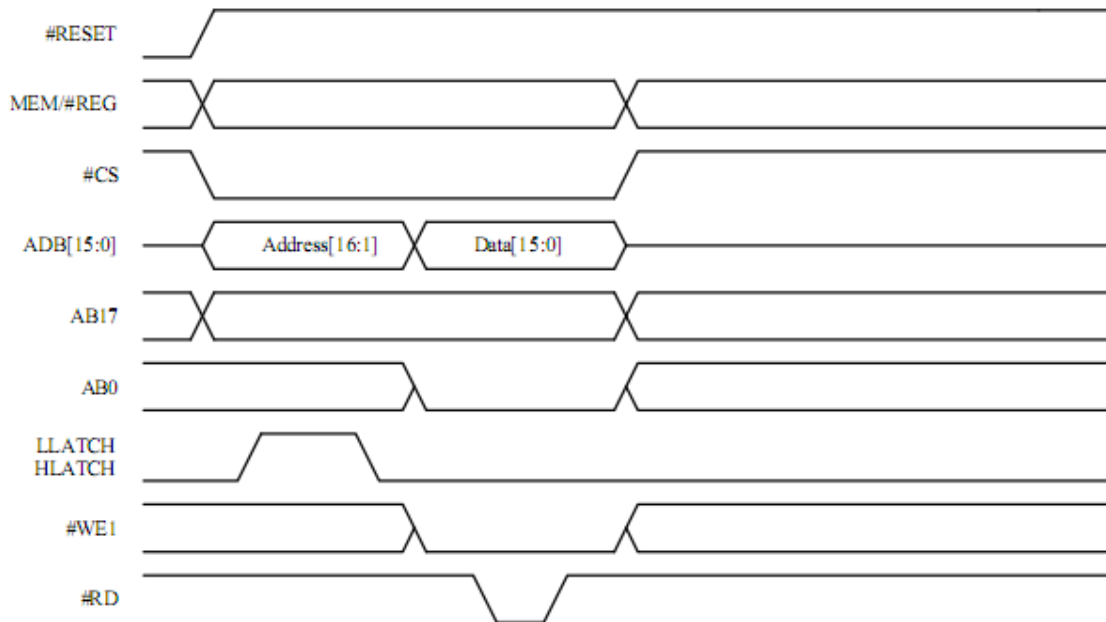
**Timing diagram ของการเขียนข้อมูลแบบ 16bit**

Timing diagram ของการเชื่อมต่อ Microcontroller กับ TFT module แบบ 16bit

**ขั้นตอนการเขียนข้อมูลไปยัง TFT Module แบบ 16bit**

1. กำหนดว่าจะเขียนข้อมูล ลงหน่วยความจำ หรือ Register ด้วยขา MEM/#REG "0"=เขียนข้อมูลลง Register, "1"=เขียนข้อมูลลงหน่วยความจำ
2. ขา #CS="0" เพื่อให้ Module พร้อมรับข้อมูล
3. กำหนดขา Address AB17 ที่จะเขียนข้อมูลลงตำแหน่งไหน
4. ส่งตำแหน่ง Address[16:1] ไปที่ขา ADB[15:0] หลังจากนั้นสร้างสัญญาณ Latch address ไปที่ขา LLATCH กับ HLATCH เพื่อ Latch address[16:1] ไปยัง buffer
6. ให้ขา AB0="0" ขา #WE1="0" หลังจากนั้นส่งข้อมูล Data[15:0] ไปที่ขา ADB[15:0]
7. สร้างสัญญาณเขียนข้อมูลไปที่ขา #WE0 เพื่อเขียนข้อมูล ไปยัง TFT module
8. ขา #CS="1" หลังจากที่เขียนข้อมูลเสร็จแล้ว

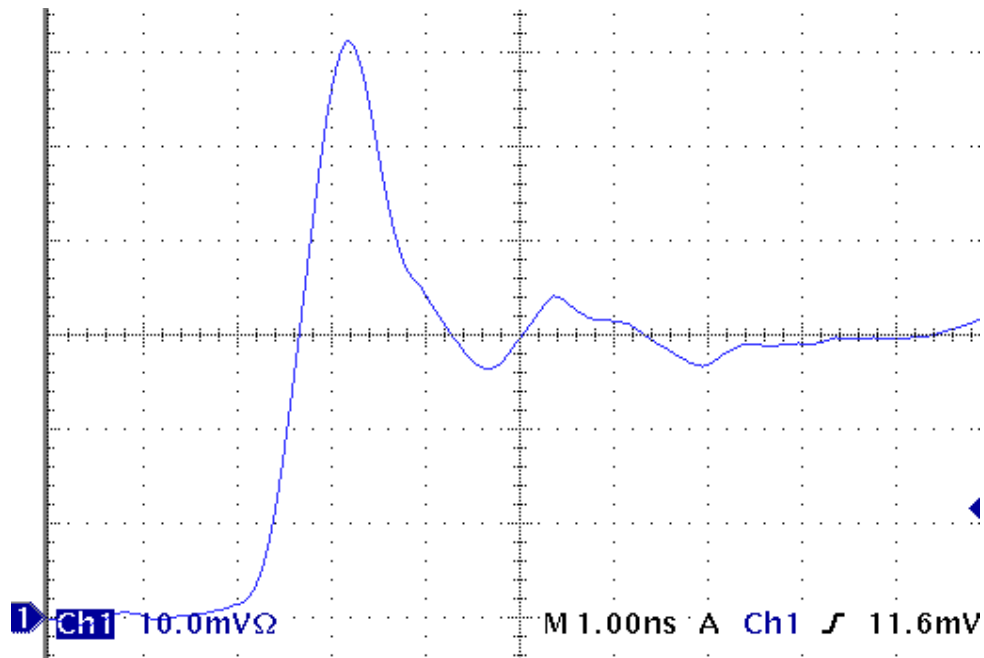


**Timing diagram การอ่านข้อมูลแบบ 16bit****ขั้นตอนการอ่านข้อมูลจาก TFT Module แบบ 16bit**

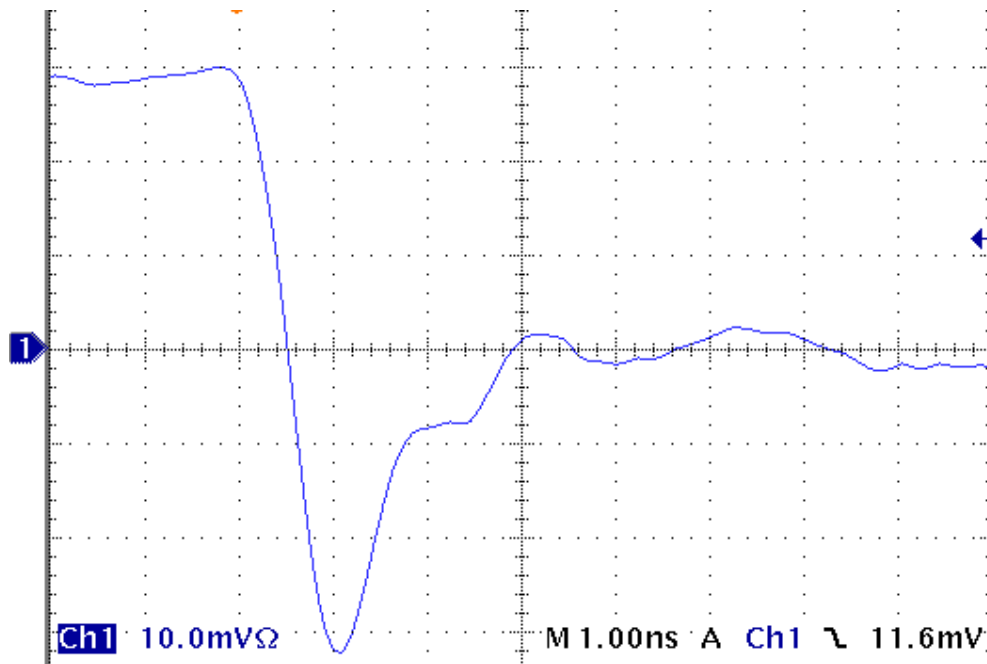
1. กำหนดว่าจะอ่านข้อมูลจากหน่วยความจำ หรือ Register ด้วยขา MEM/#REG เป็น "0"=อ่านข้อมูลจาก Register, ขา MEM/#REG เป็น "1"=อ่านข้อมูลจากหน่วยความจำ
2. ขา #CS="0" เพื่อให้ Module พร้อมรับข้อมูล
3. กำหนดขา Address AB17 ที่จะอ่านข้อมูลลงตำแหน่งไหน
4. ส่งตำแหน่ง Address[16:1] ไปที่ขา ADB[15:0] หลังจากนั้นสร้างสัญญาณ Latch address ไปที่ขา LLATCH กับ HLATCH เพื่อ Latch address[16:1] ไปยัง buffer
6. ให้ขา AB0="0" ขา #WE1="0"
7. สร้างสัญญาณอ่านข้อมูลไปที่ขา #RD เพื่ออ่านข้อมูลไปยัง TFT module
8. ทำการอ่านข้อมูล Data[15:0] จากขา ADB[15:0]
9. ขา #CS="1" หลังจากที่เขียนข้อมูลเสร็จแล้ว

**การป้องกันผลจาก Transmission line effect**

ในวงจรที่มีความถี่สูงๆ จะมอง Trace บน PCB เป็นเหมือนกับ Transmission line ส่วน Transmission line effect ก็คือผลกระทบจาก Impedance ของตัวรับและตัวส่งสัญญาณบน PCB ไม่เท่ากัน ทำให้เกิดความผิดเพี้ยนของสัญญาณขึ้น ความผิดเพี้ยนของสัญญาณนี้มักจะเกิดขึ้นทั้งช่วงขอบขาขึ้น และขอบขาลง ในช่วงขอบขาขึ้นสัญญาณจะเกิด Overshoot ทำให้สัญญาณแกว่งเกินแรงดันไฟเลี้ยงภายในวงจร ส่วนในช่วงขอบขาลง สัญญาณจะเกิด Undershoot ทำให้แรงดันของสัญญาณมีค่าต่ำกว่า 0 Volt ดังรูป



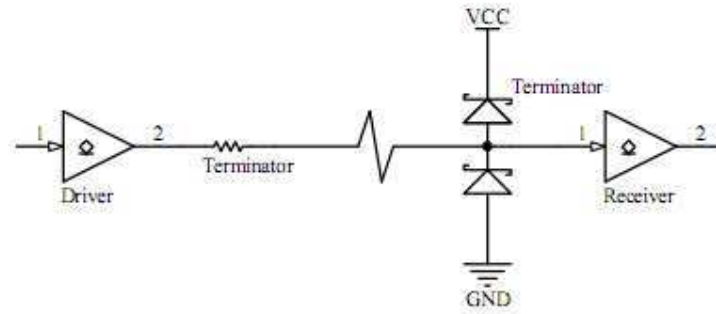
ลักษณะของสัญญาณที่เกิด Overshoot



ลักษณะของสัญญาณที่เกิด Undershoot

การออกแบบ PCB สำหรับวงที่มีสัญญาณขอบขาขึ้น (Rise time) และขอบขาลง (Fall time) ต่ำๆ (สัญญาณลักษณะนี้มักจะถูกสร้างโดยอุปกรณ์ CMOS หรือหน่วยประมวลผล ที่มีความเร็วสูง) จะต้องเดินลายวงจรให้สั้นที่สุดเท่าที่จะทำได้ เพื่อลด Transmission line effect หรือใช้ตัวต้านทาน, Schottky diode มาป้องกัน

บน TFT Module 4.3" มีการต่อ Schottky diode ที่ตัวรับสัญญาณไว้แล้ว เพื่อป้องกันผลจาก ความผิดเพี้ยนของสัญญาณที่จะเกิดขึ้น สำหรับวงจรที่จะมาเชื่อมต่อกับ TFT Module 4.3" ถ้าเกิดความผิดเพี้ยนของสัญญาณขึ้น ให้แก้ไขโดยการต่อตัวต้านทานอนุกรมให้ใกล้ตัวส่งสัญญาณ หรือ Microcontroller ให้มากที่สุดเท่าที่จะทำได้ สำหรับค่าของตัวต้านทานจะต้องทดลองโดยการเปลี่ยนค่าไปเรื่อยๆ ให้ได้รูปสัญญาณที่มีความผิดเพี้ยนน้อยที่สุด



การต่อตัวต้านทานและ Schottky diode เพื่อลดการผิดเพี้ยนของสัญญาณ

อ่านข้อมูลเพิ่มเติมเกี่ยวกับ Optimized Signal Integrity ได้ที่ [www.rickmiller.com/si.htm](http://www.rickmiller.com/si.htm)

**บริษัท อิเลคทรอนิกส์ ซอร์ซ จำกัด**

7/129 อาคารสำนักงานเซ็นทรัลพลาซ่า ชั้น 17 ห้อง 1702

ถนนบรมราชชนนี แขวงอรุณอมรินทร์ เขตบางกอกน้อย กรุงเทพฯ 10700

โทรศัพท์ : 0-2884-9210 โทรสาร : 0-2884-9214

เว็บไซต์ : <http://www.es.co.th>

อีเมลล์ : [info@es.co.th](mailto:info@es.co.th)

