

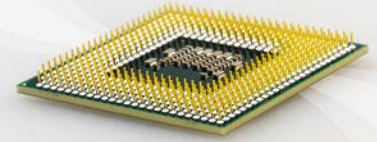
# 微算機應用實習

LCD模組

課程編號：EE4801702

實習課助教：曾子倫

# Outline

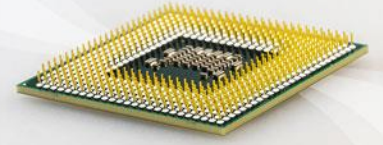


- LCD Module ( 1602A ) 介紹
- 控制指令介紹
- 函式介紹

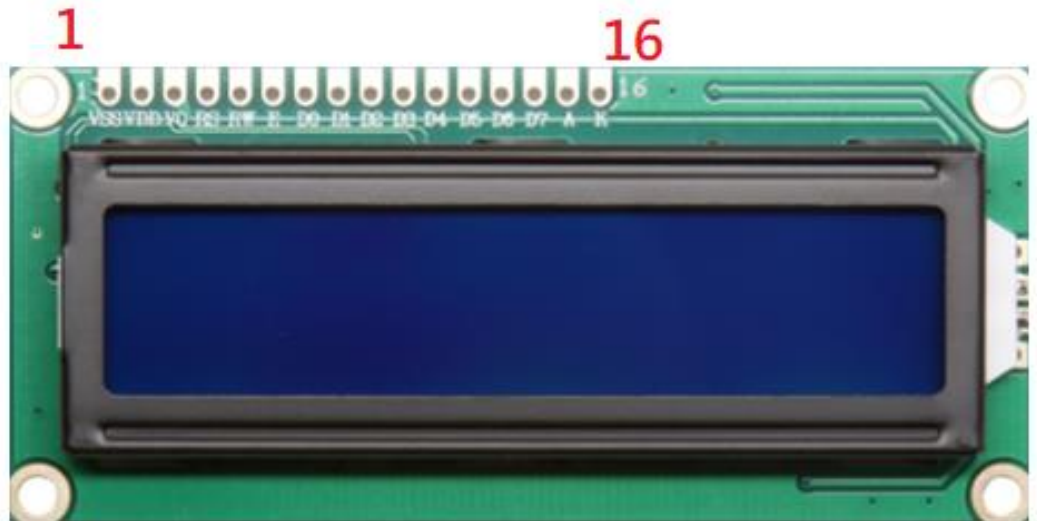


1602A

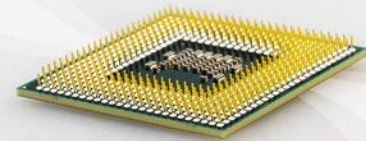
# LCD Module 介紹



- 1602A 是 16 pin 的 LCD Module



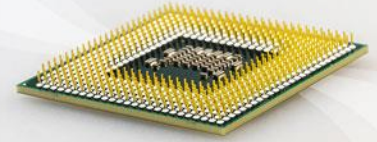
# LCD Module 介紹



- 腳位說明

名稱	腳位	I/O	用途
GND	1	I	接地
VDD	2	I	電源
Vo	3	I	對比(Contrast)調整
RS	4	I	0 : 指令 I: 資料
R/W	5	I	0 : 寫入 I: 讀取
E	6	I	致能(Enable)
DB0~DB7	7~14	I/O	資料(Data)
A	15	I	背光二極體陽極接腳(VDD)
K	16	I	背光二極體陰極接腳(GND)

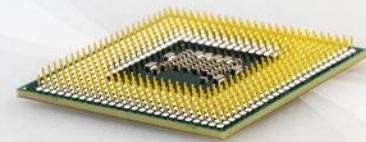
# LCD Module 介紹



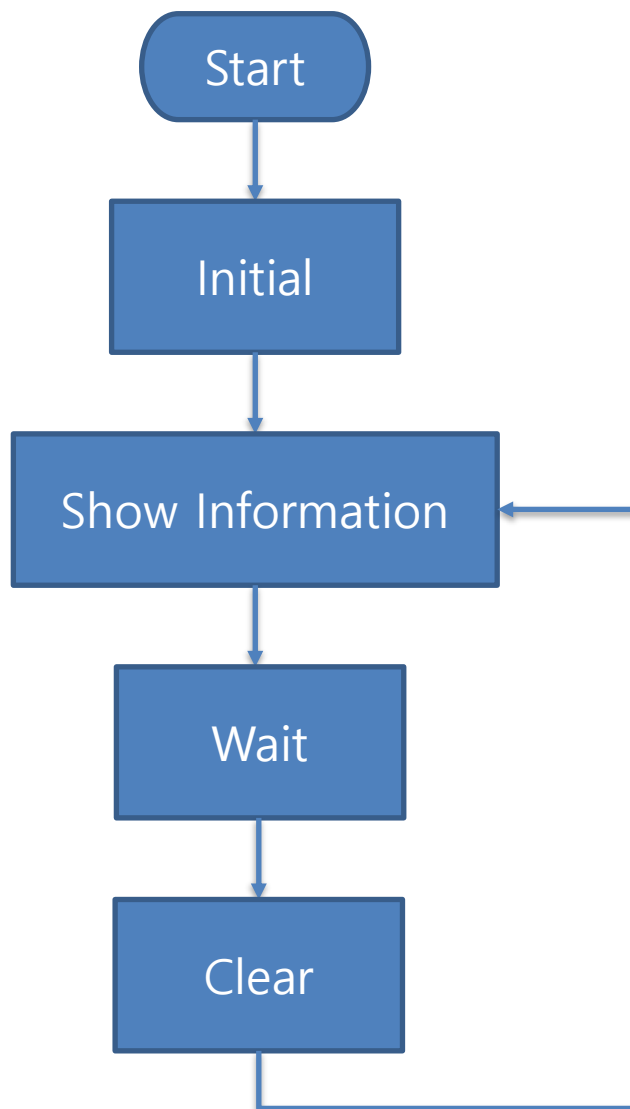
- DDRAM位址與顯示位置關係表

	顯示幕位置	0	1	2	3~14	15
第一列	DDRAM 位址 (0x0~0x27)	0x00	0x01	0x02	0x03~0x0E	0x0F
第二列	DDRAM 位址	0x40	0x41	0x42	0x43~0x4E	0x4F

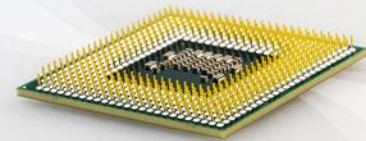
# LCD Module 介紹



- 使用流程



# LCD Module 介紹



- 指令介紹

(1)清除顯示(Clear Display)

**WriteIns(0x01)**

RS	R/W	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
0	0	0	0	0	0	0	0	0	1

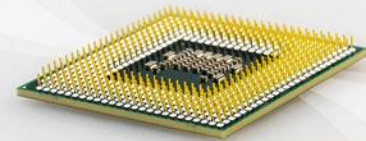
(2)功能設定(Function Set)

**WriteIns(0x38)**

RS	R/W	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
0	0	0	0	1	DL	N	F	-	-

名稱	狀態說明	
DL	0 = 4 位元資料介面	1 = 8 位元資料介面
N	0 = 一列顯示	1 = 2 列顯示
F	0 = 5x8 點字型	1 = 5x10 點字型

# LCD Module 介紹



- 指令介紹

(3) 開啟與關閉顯示(Display On/Off Control)

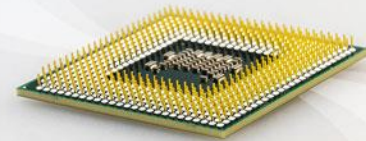
**WriteIns(0x0E)**

RS	R/W	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
0	0	0	0	0	0	1	D	C	B

名稱	狀態說明	
D	0 = 關閉顯示	1 = 開啟顯示
C	0 = 不顯示游標	1 = 顯示游標
B	0 = 游標不閃爍	1 = 游標閃爍



# LCD Module 介紹



- 指令介紹

(4) 進入模式(Entry Mode Set)      **WriteIns(0x06)**

RS	R/W	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
0	0	0	0	0	0	0	1	I/D	S

名稱	狀態說明	
I/D	0 = 每寫入或讀出資料後，游標左移	1 = 每寫入或讀出資料後，游標右移
S	0 = 顯示幕不移動	1 = 顯示幕移動

## ● 指令介紹

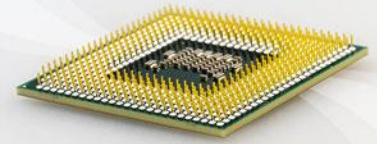
### (5)設定字元顯示記憶體位置(Set DDRAM address)

**WriteIns(0x80)**      回到第一行第一個字元

**WriteIns(0xC0)**      回到第二行第一個字元

RS	R/W	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
0	0	1	Address						

# LCD Module 介紹

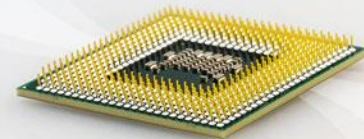


- 指令介紹

(6)讀取忙碌旗標BF與位址計數器AC內容(Read busy flag & address)

RS	R/W	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
0	1	BF	A6	A5	A4	A3	A2	A1	A0

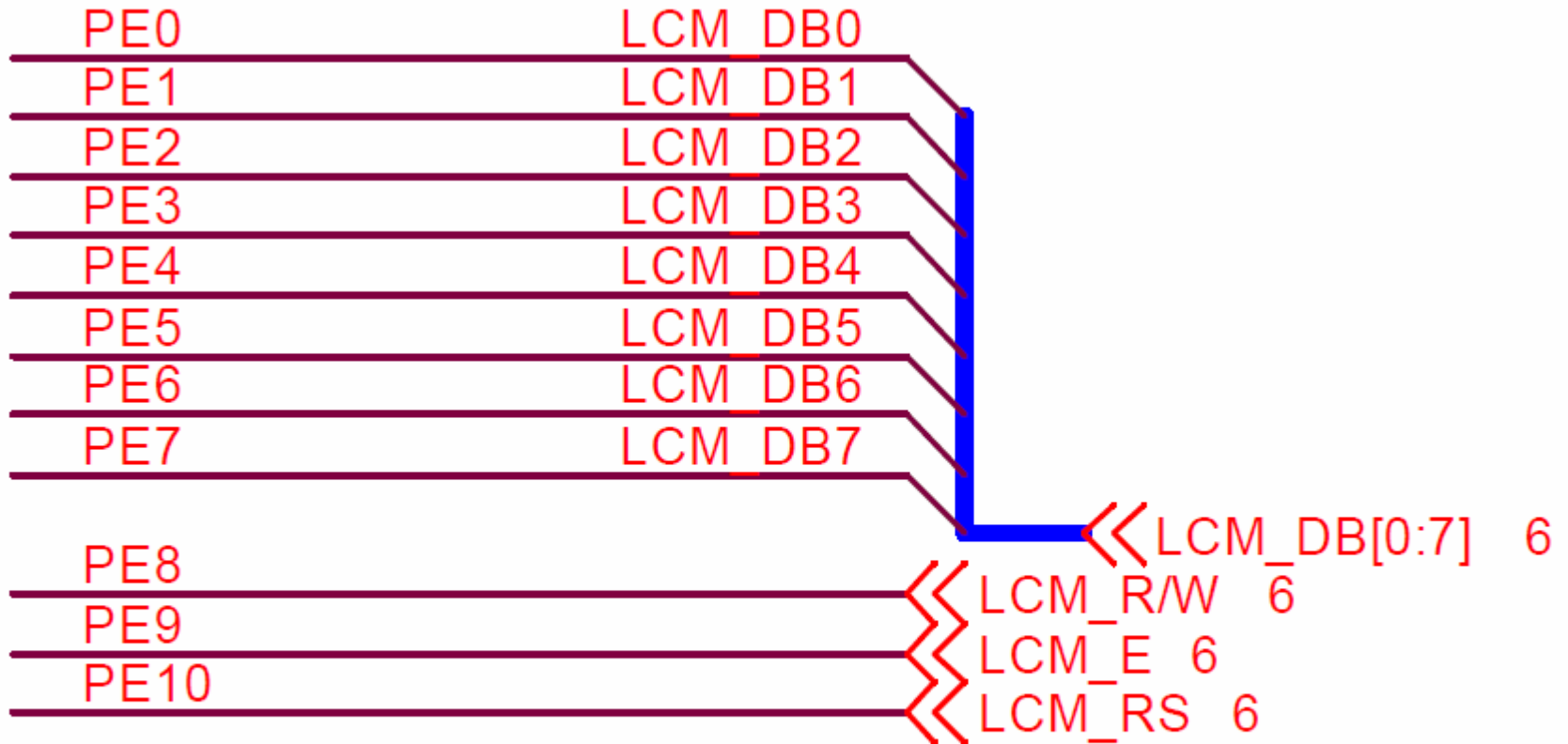
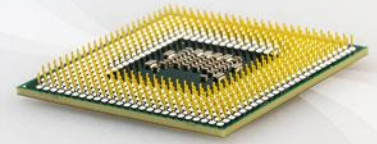
# LCD Module 介紹



- 內建字元表

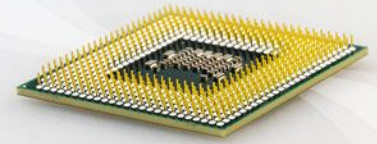
Lower 4 Bits \ Upper 4 Bits	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111	
xxxx0000	CG RAM (1)			0	@	P	`	P				-	タ	ミ	α	p	
xxxx0001	(2)			!	1	A	Q	a	q			。	ア	チ	△	ä	q
xxxx0010	(3)			"	2	B	R	b	r			「	イ	ツ	×	β	θ
xxxx0011	(4)			#	3	C	S	c	s			」	ウ	テ	モ	ε	∞
xxxx0100	(5)			\$	4	D	T	d	t			、	エ	ト	ホ	μ	Ω
xxxx0101	(6)			%	5	E	U	e	u			・	オ	ナ	ユ	σ	Ü
xxxx0110	(7)			&	6	F	V	f	v			ヲ	カ	ニ	ヨ	ρ	Σ
xxxx0111	(8)			'	7	G	W	g	w			フ	キ	ヌ	ラ	g	π
xxxx1000	(1)			<	8	H	X	h	x			イ	ク	ネ	リ	ℓ	×
xxxx1001	(2)			>	9	I	Y	i	y			ウ	ケ	ル	ル	ˆ	γ
xxxx1010	(3)			*	:	J	Z	j	z			エ	コ	ハ	レ	j	〒
xxxx1011	(4)			+	;	K	L	k	l			オ	サ	ヒ	ロ	*	⌘
xxxx1100	(5)			,	<	L	¥	l	l			カ	シ	フ	ワ	φ	⌘
xxxx1101	(6)			-	=	M	J	m	}			ユ	ズ	ヘ	ン	も	÷
xxxx1110	(7)			.	>	N	^	n	†			ヨ	セ	ホ	ˆ	ñ	
xxxx1111	(8)			/	?	O	_	o	€			ッ	リ	マ	°	ö	■

# LCD Module 接線圖



**LCM**

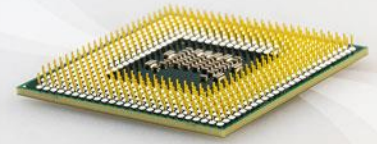
# 函式介紹



- 函式介紹

函式名稱	函式功能
CheckBusy( )	檢查忙碌旗標位元直到忙碌旗標為 0
WriteIns( )	寫入指令
WriteData( )	寫入資料
IntialLCD( )	初始化 LCD Module

# 函式介紹



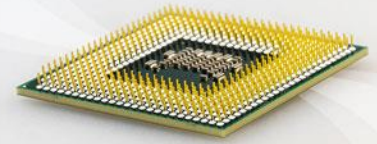
- WriteIns( )

```
void WriteIns(unsigned char instruction)
{
    GPIO_PTE_DIR = 0x0000; //Initialize GPIO_E output
    GPIO_PTE_CFG = 0x0;

    GPIO_PTE_GPIO = (instruction + E ); //Write instruction
    delay1(100);
    GPIO_PTE_GPIO = CleanSet;           //Clean buffer
    delay1(100);

    CheckBusy();
}
```

# 函式介紹



- WriteData( )

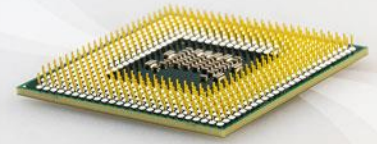
```
void WriteData(unsigned char i)
{
    GPIO_PTE_DIR = 0x0000; //Initialize GPIO_E output
    GPIO_PTE_CFG = 0x0;

    GPIO_PTE_GPIO = (i + RS + E); //Write data
    delay1(100);
    GPIO_PTE_GPIO = CleanSet; //Clean buffer
    delay1(100);

    CheckBusy();
}
```



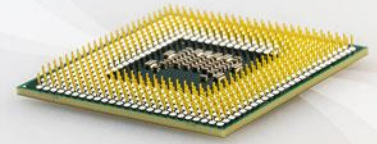
# 函式介紹



- InitialLCD( )

```
void InitialLCD(void)
{
    WriteIns(0x38); //FUNCTION SET
    delay1(300000);
    WriteIns(0x01); // clear buffer
    delay1(300000);
    WriteIns(0x0E); // on display
    delay1(300000);
    WriteIns(0x06); // set input mode
    delay1(300000);
}
```

# 函式介紹



- CheckBusy( )

```
void CheckBusy(void) //BF ,CHECK BUSY FLAG
{
    unsigned short int i=0x80;
    while(i&0x80)
    {
        GPIO_PTE_DIR = 0x0000; //Initialize GPIO_E output
        GPIO_PTE_CFG = 0x0;

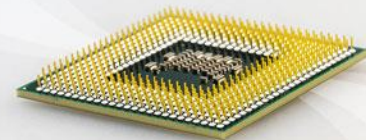
        GPIO_PTE_GPIO = (RW + E);
        delayl(500);

        GPIO_PTE_DIR = 0xFF; //Initialize GPIO_E input
        GPIO_PTE_CFG = 0x0;

        delayl(100);
        i = GPIO_PTE_PADIN; //Read busy flag
        delayl(100);

        GPIO_PTE_GPIO = CleanSet; //Clean buffer
        delayl(1000);
    }
}
```

# Example (顯示字串)



```
int main()
{
    unsigned int i;

    OS_PowerOnDriverInitial();

    InitialLCD();    //Initial LCD Module

    WriteIns(0x80);  //1-LINE DD RAM SET Address
    for(i=0;i<16;i++)
        WriteData(255); //Display

    WriteIns(0xC0);  //2-LINE DD RAM SET Address
    for(i=0;i<16;i++)
        WriteData(255); //Display
    delay1(1000000); //Wait

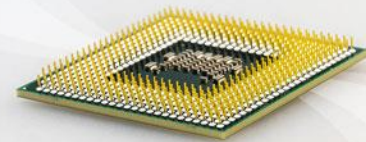
    WriteIns(0x01);  // clear buffer

    Display_1Line(8); //Display 'A','b','C','d','E','f','G','h'
    delay1(1000000); //Wait
    WriteIns(0x01);  // clear buffer

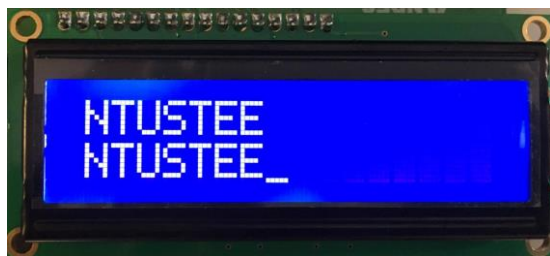
    Display_2Line(5); //Display "NTUST" (Two line)
    delay1(1000000); //Wait
    WriteIns(0x01);  // clear buffer

    return 0;
}
```

# LAB5



- 開機先顯示 "NTUSTEE" 於 LCD Module (兩行)



- 按下 "SW17" 跳至下一頁並透過按鈕輸入學號
- 按下 "SW15" 可切換數字0~9 和字母A~Z (循環顯示)
- 按下 "SW16" ，刪除鍵(刪除前一個字元)
- 按下 "SW17" ，確認鍵(確認目前選擇字元)
- 最後一次確認後請在第二行顯示 "OK"

