

The background of the slide is a dark, blue-toned image of a city skyline at night. Overlaid on this is a complex network of white lines connecting various nodes, some of which are represented by Wi-Fi symbols. A silhouette of a person stands in the center, looking out over the city. The overall theme is technology and urban connectivity.

# Image Processing Embedded System

By Mr.Natthapol Nonthasri

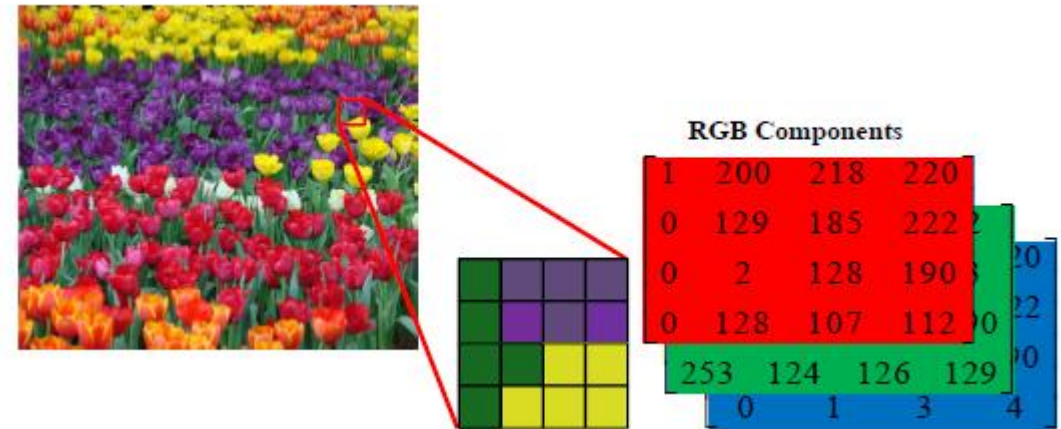
ALLPPT.COM



# Image Processing - Point Operation

## Point Operation

ค่า pixel จะขึ้นอยู่กับเฉพาะค่า pixel เดิมที่อยู่จุดเดียวกัน ไม่ขึ้นอยู่กับค่าที่ตำแหน่งอื่น



3	3	3	3	3
3	3	3	3	3
3	3	3	3	3
3	3	3	3	3
3	3	3	3	3



3	3	3	3	3
3	3	3	3	3
3	3	5	3	3
3	3	3	3	3
3	3	3	3	3

ตัวอย่าง

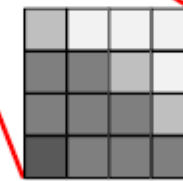
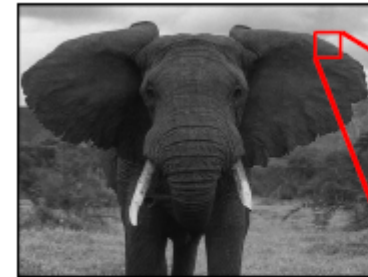
- Intensity (ความเข้ม)
- brightness (ความสว่าง)

# Image Processing - Image Filter

## Image Filter

(การกำจัดสัญญาณรบกวนออกจากภาพ)

อ้างอิงค่า pixel ข้างๆ ด้วยก่อน  
การปรับเปลี่ยนค่าความเข้มแสงใน  
แต่ละ pixel นั้น ๆ



Gray Level

187	214	218	220
128	129	185	222
126	127	128	190
100	124	126	129

3	3	3	3	3
3	3	3	3	3
3	3	3	3	3
3	3	3	3	3
3	3	3	3	3



3	3	3	3	3
3	3	3	3	3
3	3	5	3	3
3	3	3	3	3
3	3	3	3	3

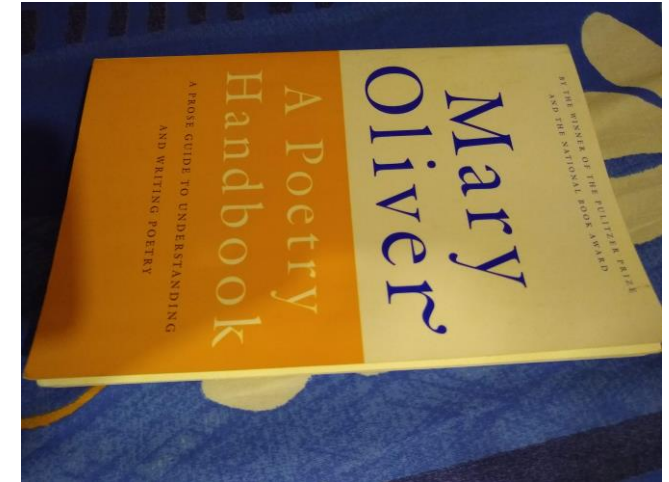
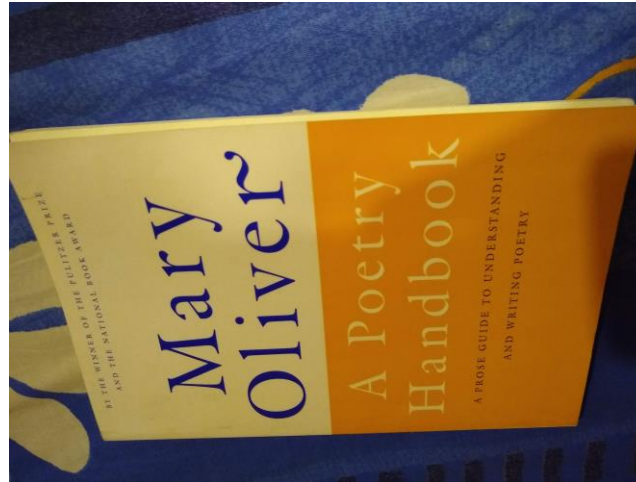
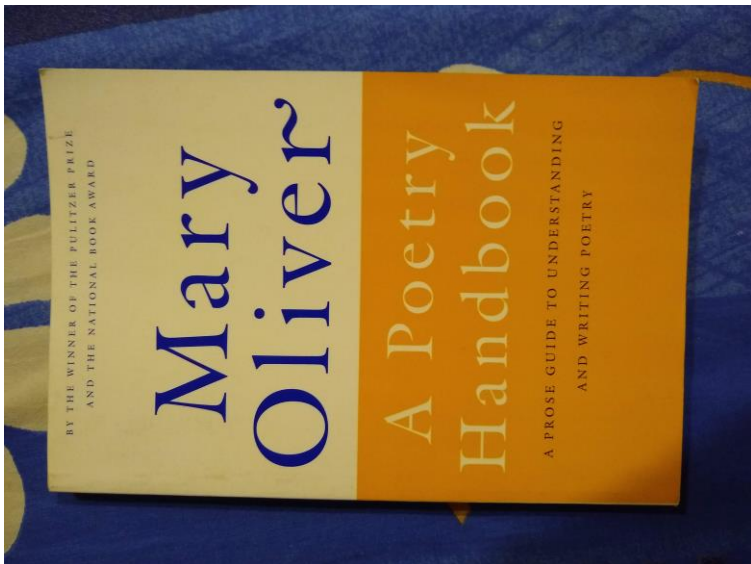
ตัวอย่าง

- การเบลอภาพ
- การหาเส้นขอบ
- การลบจุดขาว/ดำ

# Image Processing - Image Registration

## Image Registration (การซ้อนทับภาพ)

จัดแนวรูปภาพใหม่ โดยใช้รูปภาพ  
จากมุมต่าง ๆ



ภาพที่ 1

ภาพที่ 2

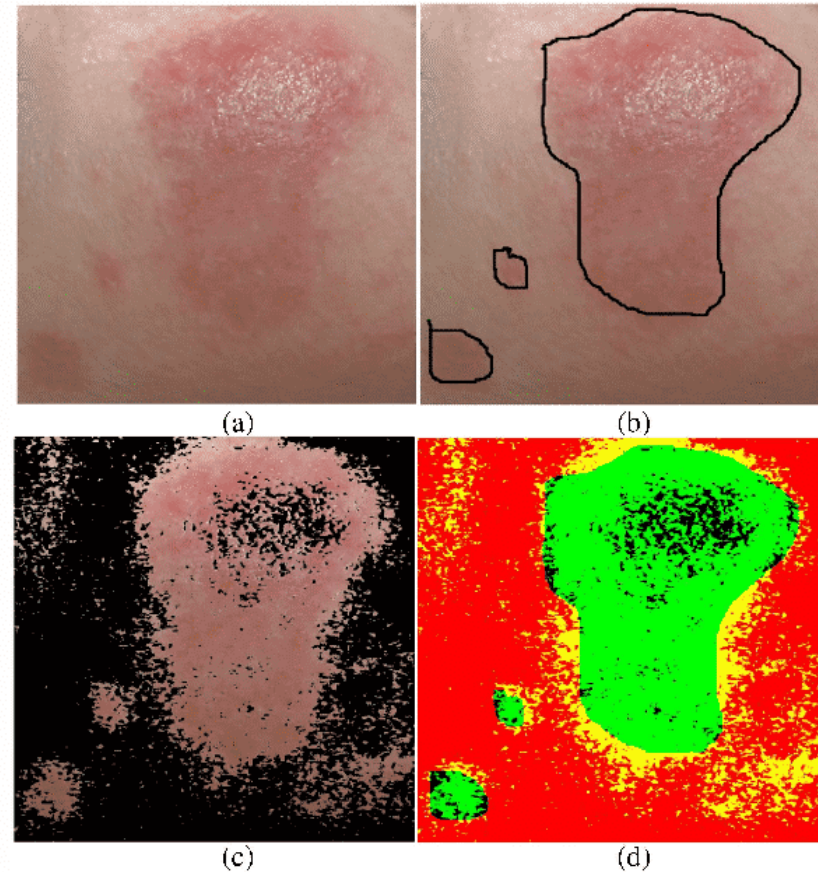
ผลลัพธ์



# Image Processing - Image Segmentation

## Image Segmentation (การแบ่งส่วนภาพ)

จำแนก Pixel หลาย ๆ Pixel ว่า  
Pixel แต่ละจุด คืออะไร จะได้ผล  
ออกมาเป็นแบ่งเป็นพื้นที่สีต่าง ๆ  
ซึ่งแต่ละสีหมายความว่าถึงลักษณะที่  
แตกต่างกัน

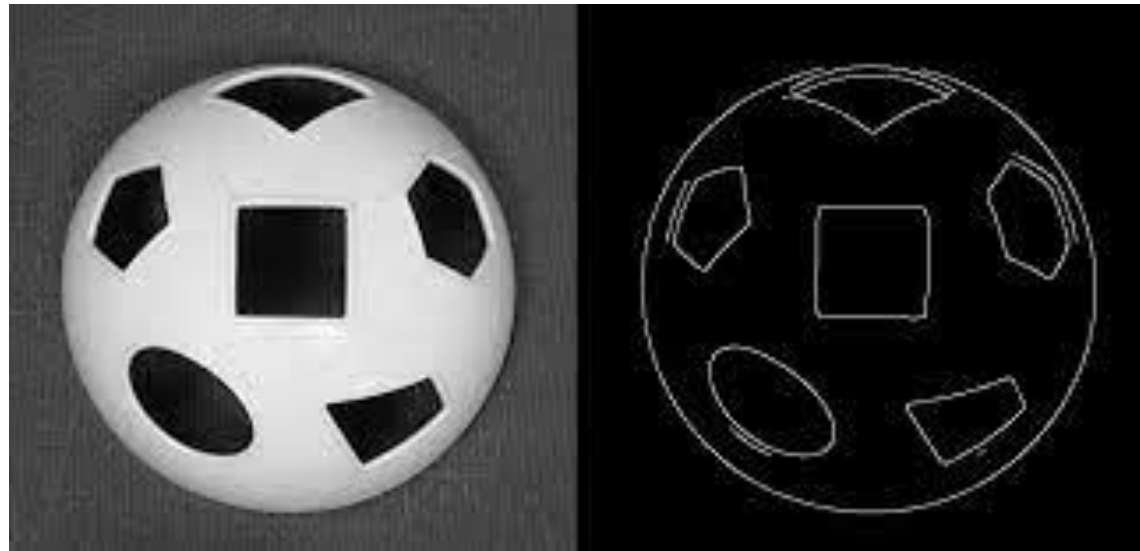


วิเคราะห์ รูปถ่ายผิวหนัง

# Image Processing - Image Edge Detection

## Image Edge Detection (การหาขอบภาพ )

จะใช้หลักการหาความชันของความเข้มสี หรือ intensity เนื่องจากที่ขอบรูปจะเป็นบริเวณมีความแตกต่างของสีมาก ซึ่งหากเราหาความชันของค่า intensity ก็จะได้ความชันมาก แต่หากรูปนั้นไม่มีขอบค่า intensity บริเวณนั้นจะใกล้เคียงกันหรือเท่ากัน ทำให้ไม่มีความชัน

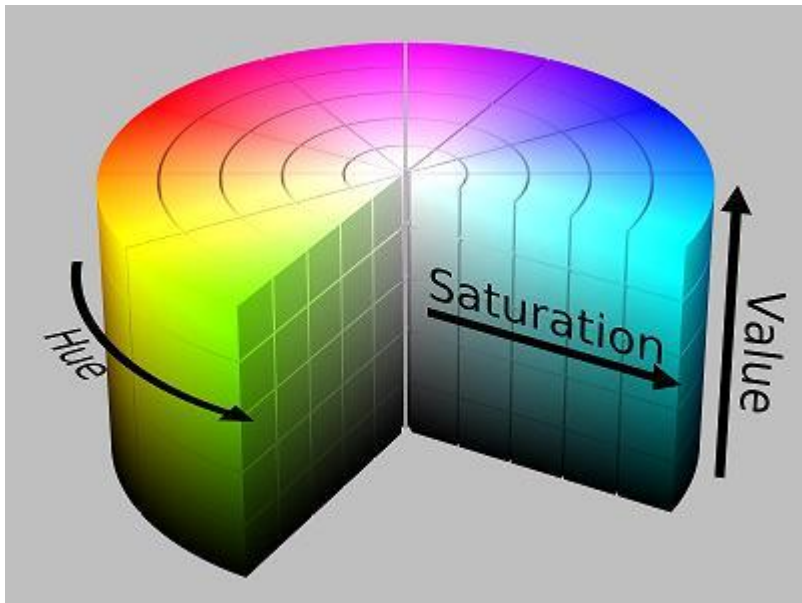




# Image Processing – Color Detection

## Color Detection (การแบ่งส่วนภาพ)

เป็นหนึ่งในวิธีการทำ Partial Segmentation คือ การตรวจจับสี (ที่เราสนใจในภาพ) โดยภาพปกติทั่วไปจะใช้รูปแบบของสีเป็น RGB นั้นหมายความว่าภาพที่เห็นประกอบขึ้นจากการรวมกันของภาพ 3 ภาพ โดยแต่ละภาพจะเก็บค่า Intensity สีแดง สีเขียว และสีน้ำเงิน ไว้ตามลำดับ



## HSV Model

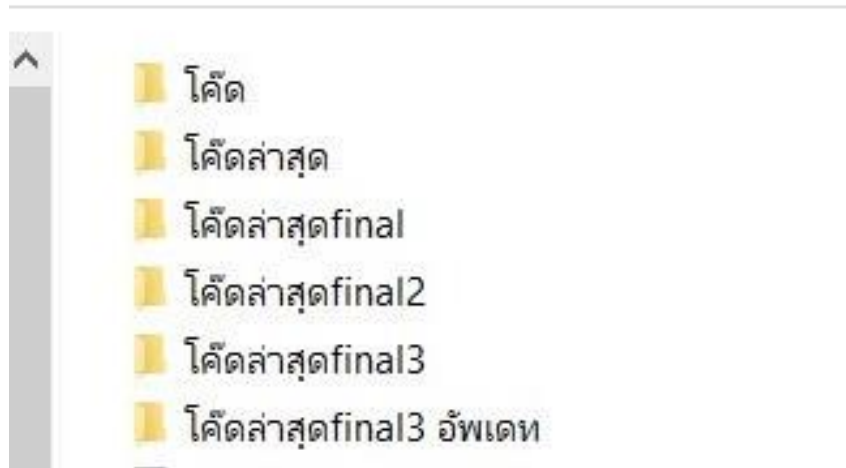
H (Hue)	เป็นค่าสี	0 - 180
S (Saturation)	เป็นค่าบริสุทธิ์ของสี (สีขาวผสมกับ Hue)	0 - 255
V (Value)	เป็นค่าความสว่าง	0 - 255



# GitHub – version control



## Problem 1



git version

## Problem 2

```
# programmer A
x = 5, y = 10
z = x + y
print('z is ' +str(z))
```

```
# programmer B
x = 5, y = 10
z = x + y
print('value of z = ' +str(z))
```



git merge

```
# programmer B
x = 5, y = 10
z = x + y
print('value of z = ' +str(z))
```





# Git และ GitHub คืออะไร?



## **Git**

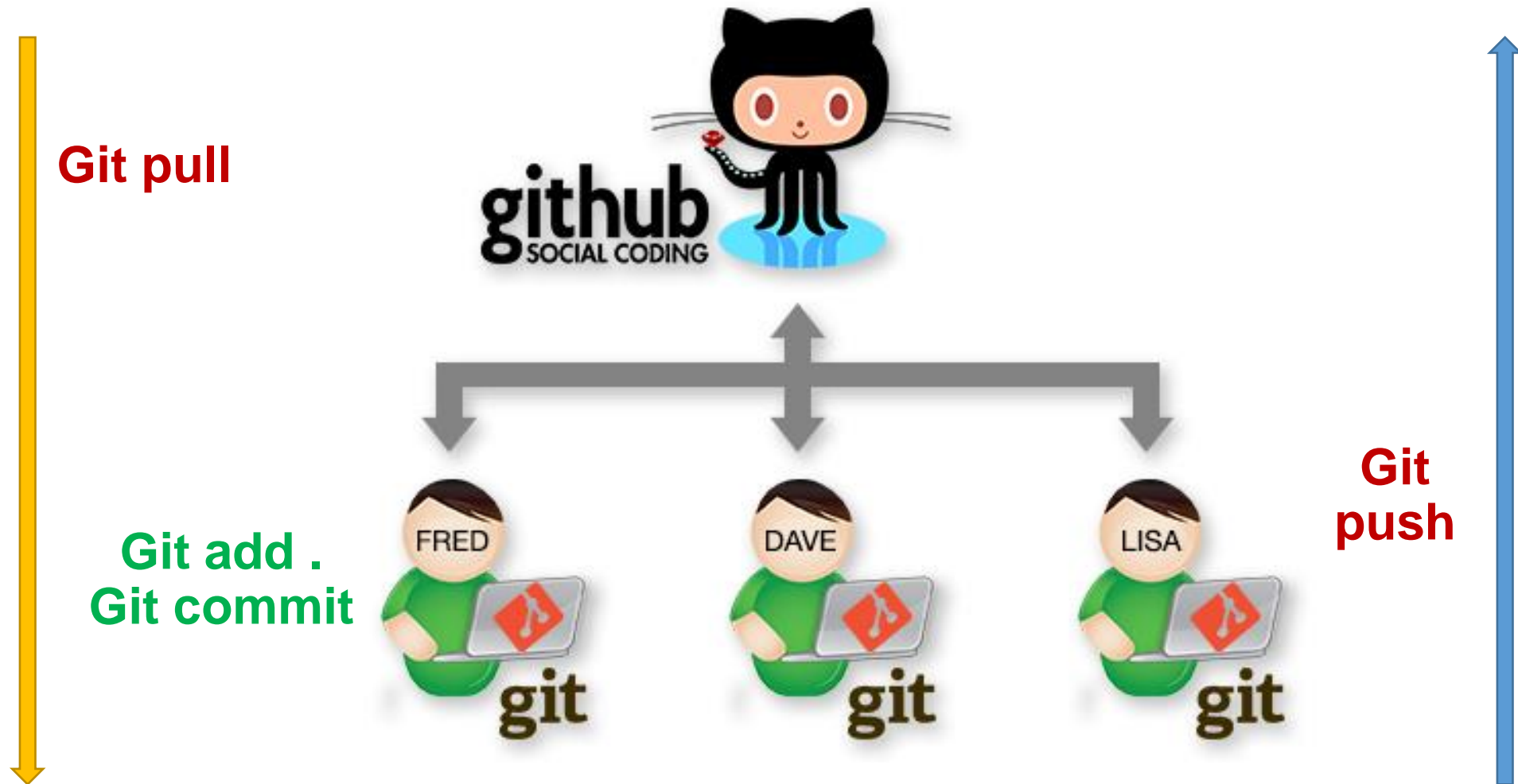
คือ Version control ที่คอยจัดเก็บความเปลี่ยนแปลงของไฟล์ในโปรเจกของเราหรือของทีมเราไม่ว่าเพื่อนคนไหนจะเปลี่ยนแปลง, เพิ่ม, ลบอะไรในโปรเจกเราก็สามารถที่จะรับรู้ได้ทำให้รู้ว่า ‘งานที่กำลังทำอยู่ถึงไหนแล้ว’

## **GitHub**

คือ เว็บไซต์ที่ทำให้เราสามารถใช้ Git ร่วมกับคนอื่นได้ หรือพูดง่าย ๆ ก็คือ Git ที่อยู่บนเว็บไซต์นั้นแหละก็อย่าลืมสมัครก่อนอ่านขั้นตอนการใช้ต่อไปด้วยละ



# Git และ GitHub อยู่ที่ไหน?

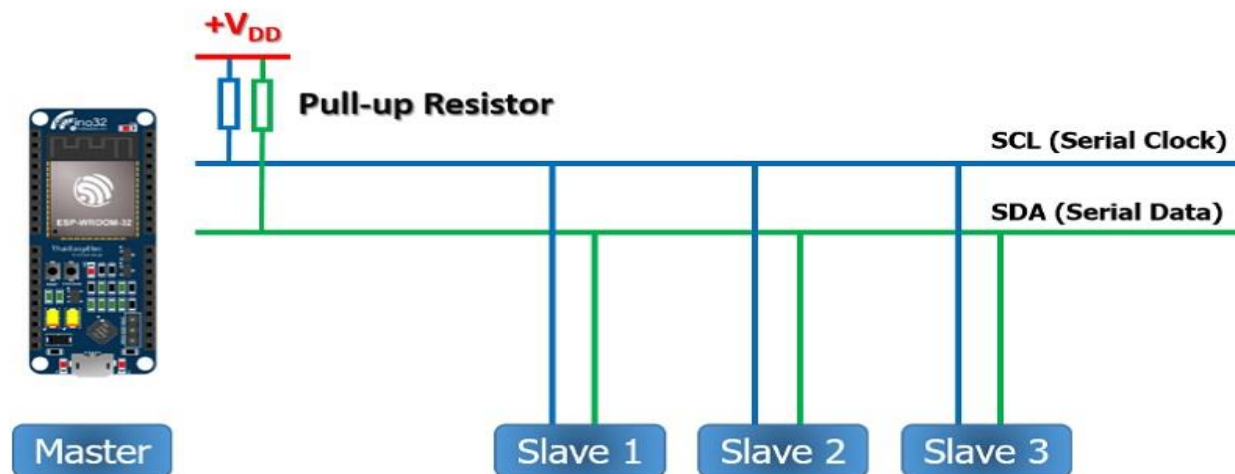


# Arduino Communication – I2C

Inter-Integrated Circuit คือรูปแบบการสื่อสารข้อมูลอย่างหนึ่งที่สร้างขึ้นมาเพื่อสื่อสารข้อมูลความเร็วต่ำ นิยมใช้กับอุปกรณ์จำพวกไมโครโปรเซสเซอร์ ไมโครคอนโทรลเลอร์และอุปกรณ์ต่างๆ ที่เกี่ยวข้อง

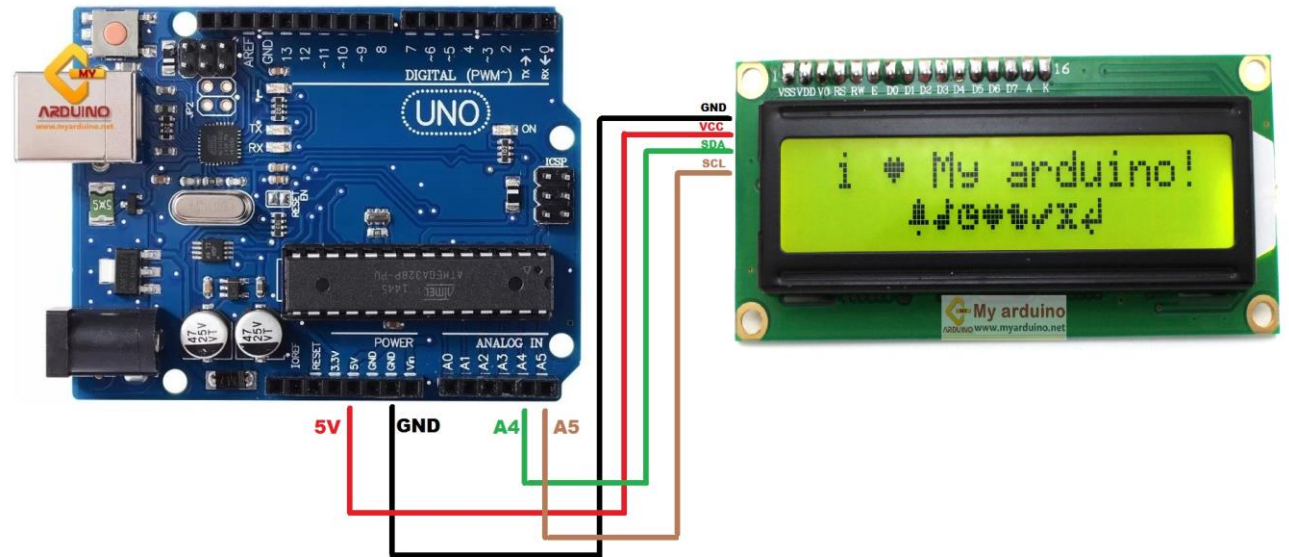
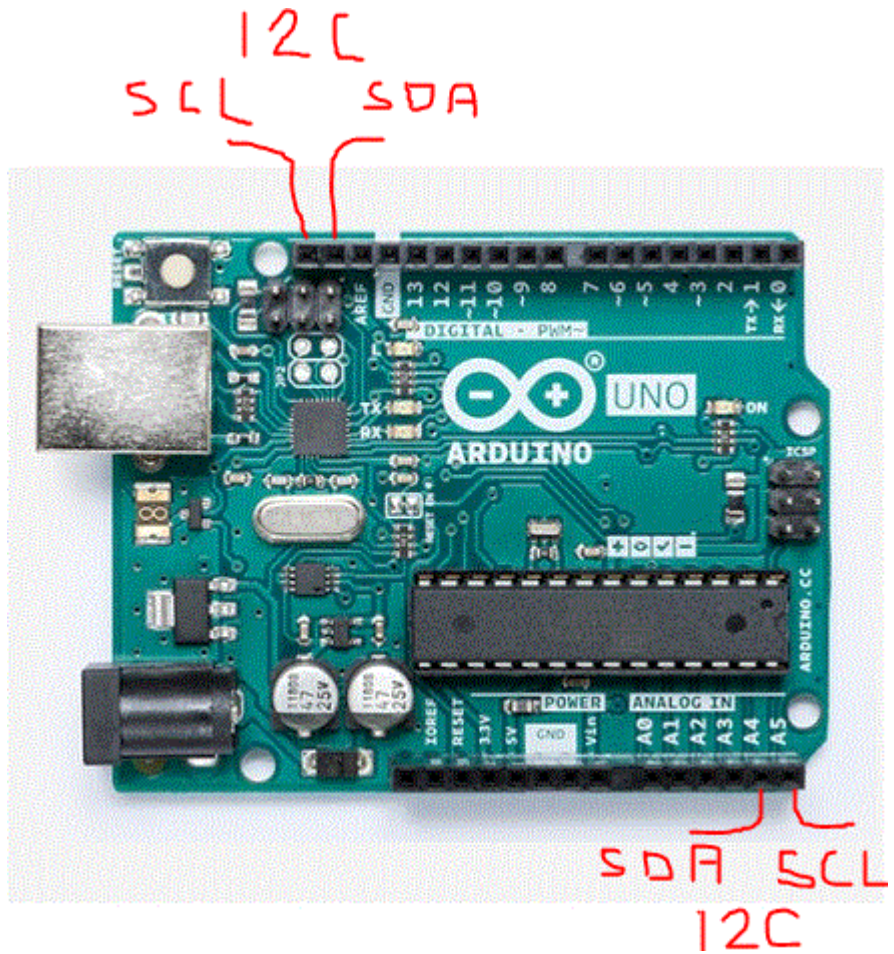
ข้อดี สามารถรับ-ส่งข้อมูลได้หลายอุปกรณ์ในบัสเดียวกัน

- SDA (Serial Data) คือ สายสัญญาณสำหรับรับ-ส่งข้อมูล
- SCL (Serial Clock) คือ สายสัญญาณนาฬิกา ใช้เป็นสำหรับควบคุมการรับ-ส่งข้อมูล





# Arduino Communication – I2C



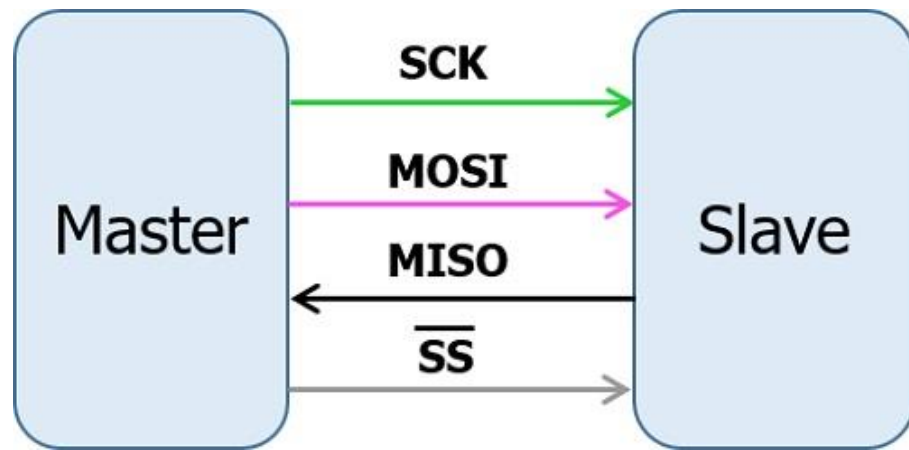
**Arduino Uno R3 + LCD**

# Arduino Communication – SPI

Serial Peripheral Interface คือรูปของแบบการสื่อสารข้อมูลแบบอนุกรมแบบซิงโครนัส รูปแบบหนึ่ง ใช้ในการการสื่อสารระยะใกล้โดยเฉพาะในระบบสมองกลฝังตัว การสื่อสารอนุกรมแบบ SPI จะอาศัยสัญญาณนาฬิกาเป็นตัวกำหนดจังหวะการรับส่งข้อมูล สามารถส่งข้อมูลไปยังปลายทาง และรับข้อมูลจากปลายทางกลับมาในครั้งเดียวกัน (Full Duplex)

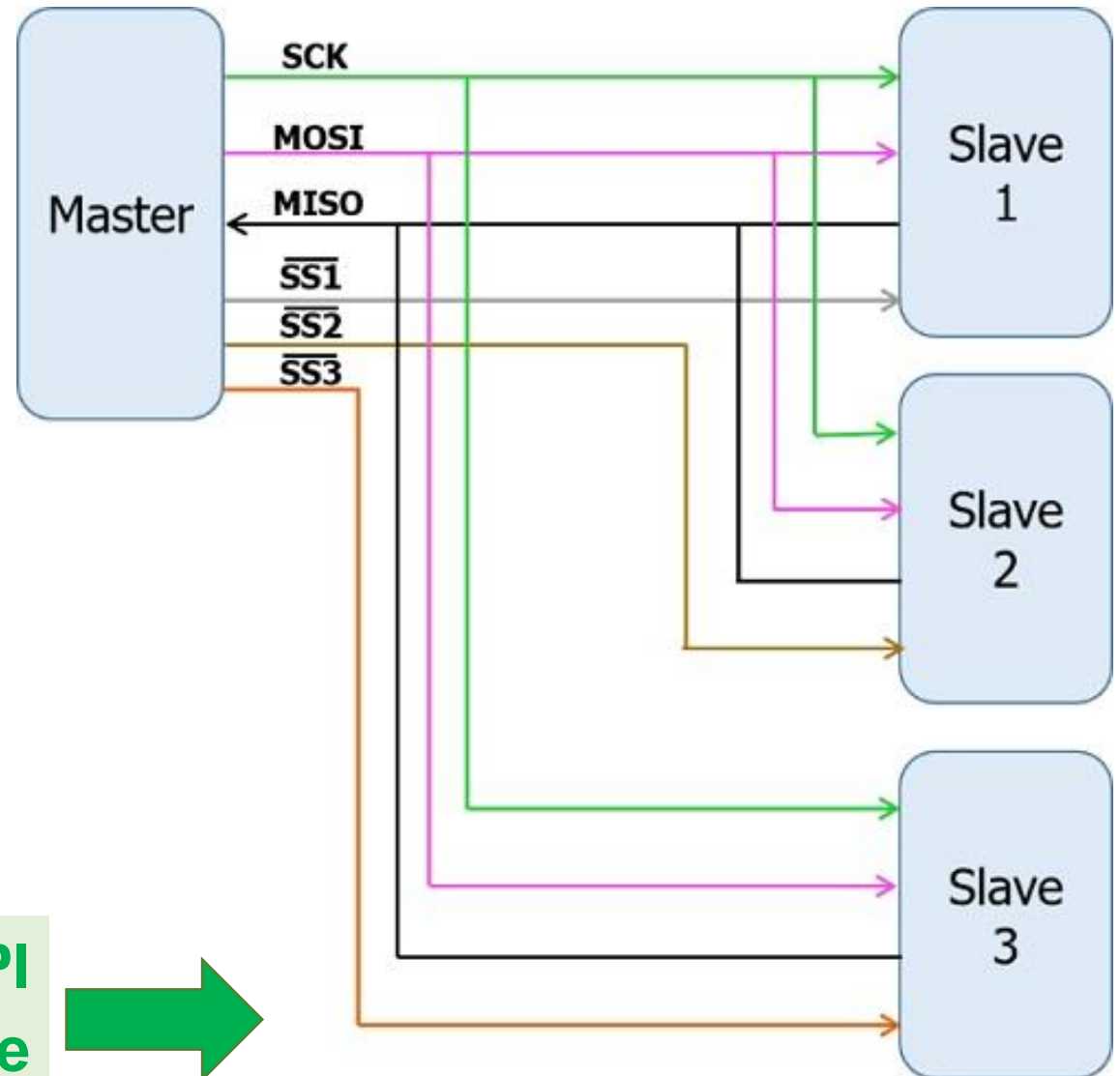
SCK (Clock Data)	ใช้สำหรับส่งสัญญาณนาฬิกาจาก Master ไปยัง Slave
MISO (Master In Slave Out)	ใช้สำหรับรับข้อมูลจาก Slave
MOSI (Master Out Slave In)	ใช้สำหรับส่งข้อมูลจาก Master ไปยัง Slave
SS/CS (Slave Select/Chip Select)	ใช้สำหรับเลือก Slave ที่ต้องการใช้งาน

# Arduino Communication – SPI



การสื่อสารอนุกรมแบบ SPI  
แบบ Slave เดียว

การสื่อสารอนุกรมแบบ SPI  
แบบหลาย Slave



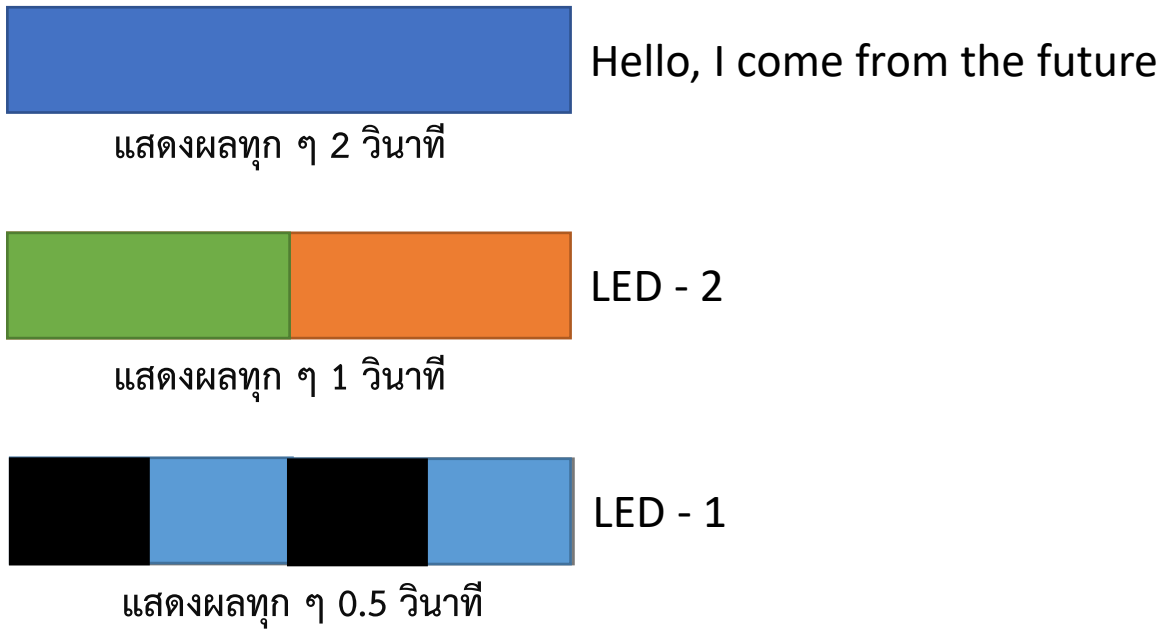


# Difference between SPI and I2C

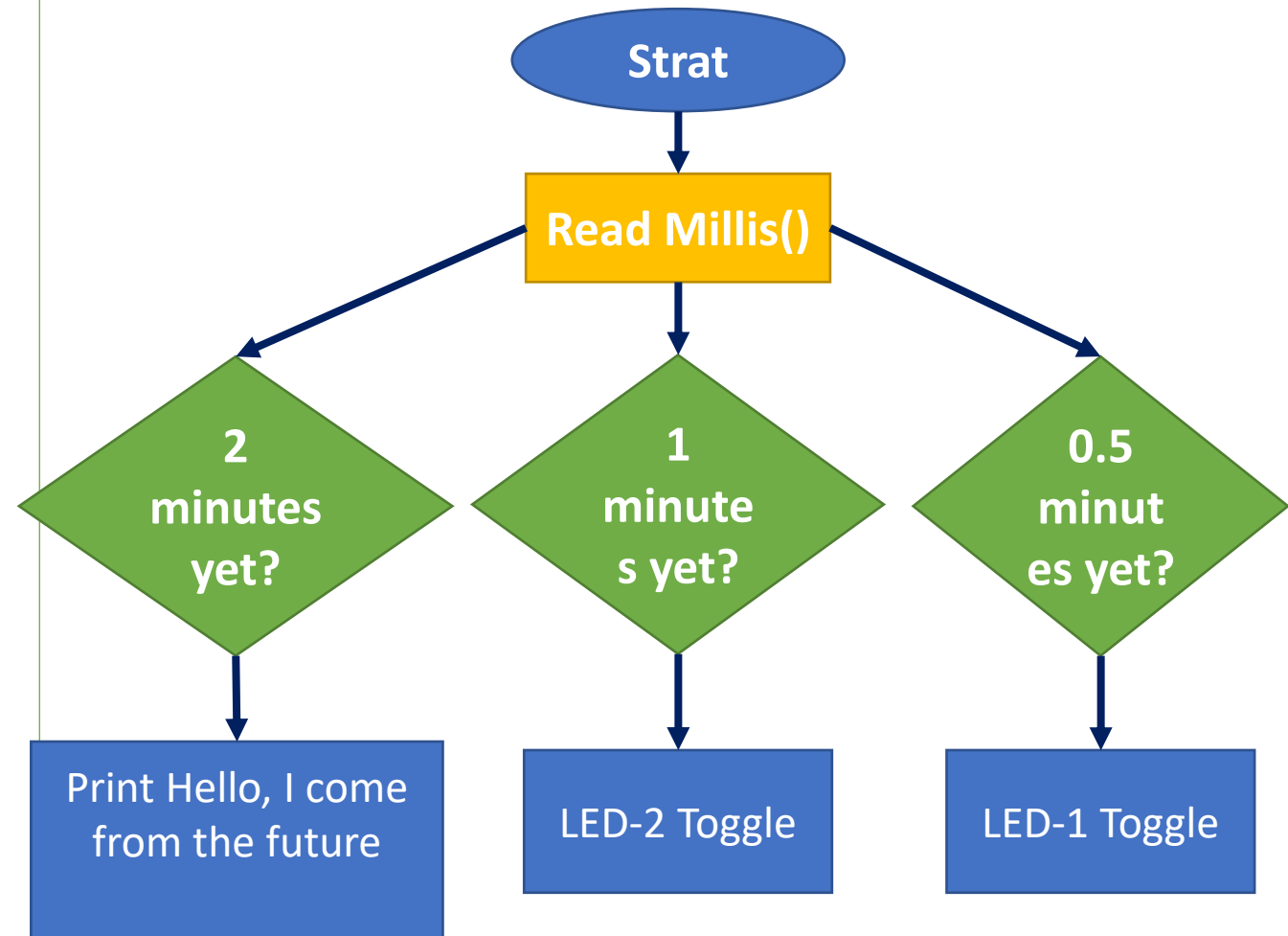
ประเด็น	SPI	I2C
การรับส่งข้อมูลสองทิศทาง	full-duplex	half-duplex
ความเร็วโดยทั่วไป	100kHz หรือ 400kHz	ได้สูงกว่า 10MHz
จำนวนสัญญาณเพื่อเชื่อมต่อ	4 หรือมากกว่า	2 (SCL และ SDA)
การเลือกสื่อสารกับ Slave	ใช้สัญญาณ Slave Select	ระบุ Device Address
การตอบกลับเมื่อได้รับข้อมูล	ไม่มี Acknowledge	ใช้ ACK/NACK bit
ขนาด Data Frame	8 บิต	8 บิต

# -> Lab 1 Arduino Delay() or Millis()

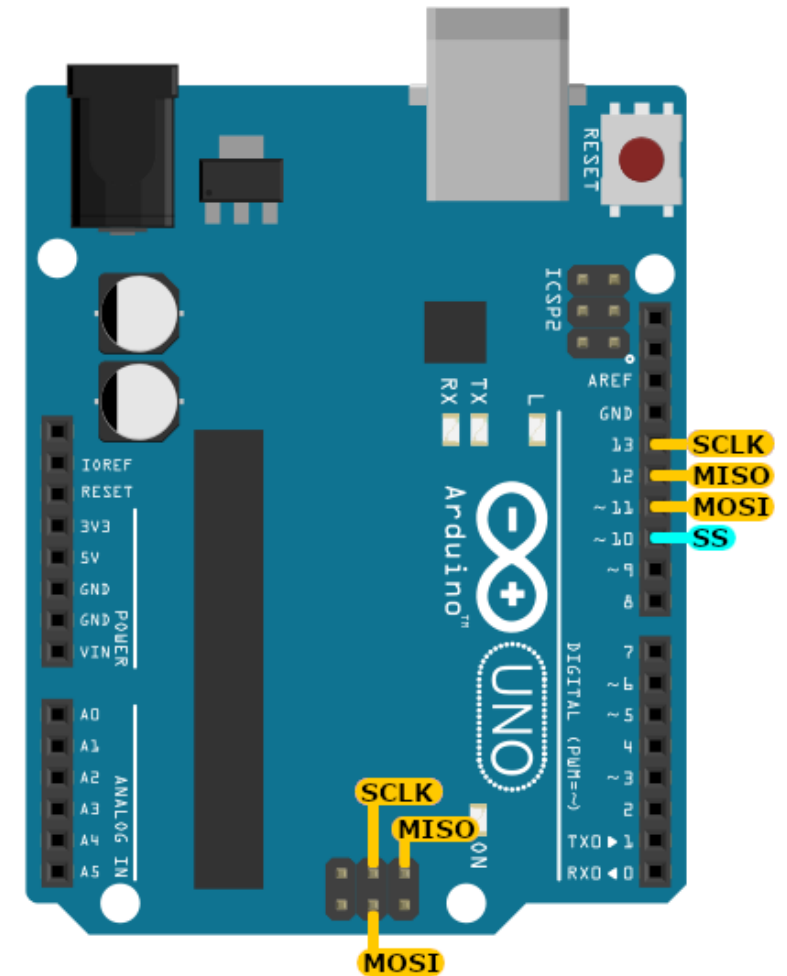
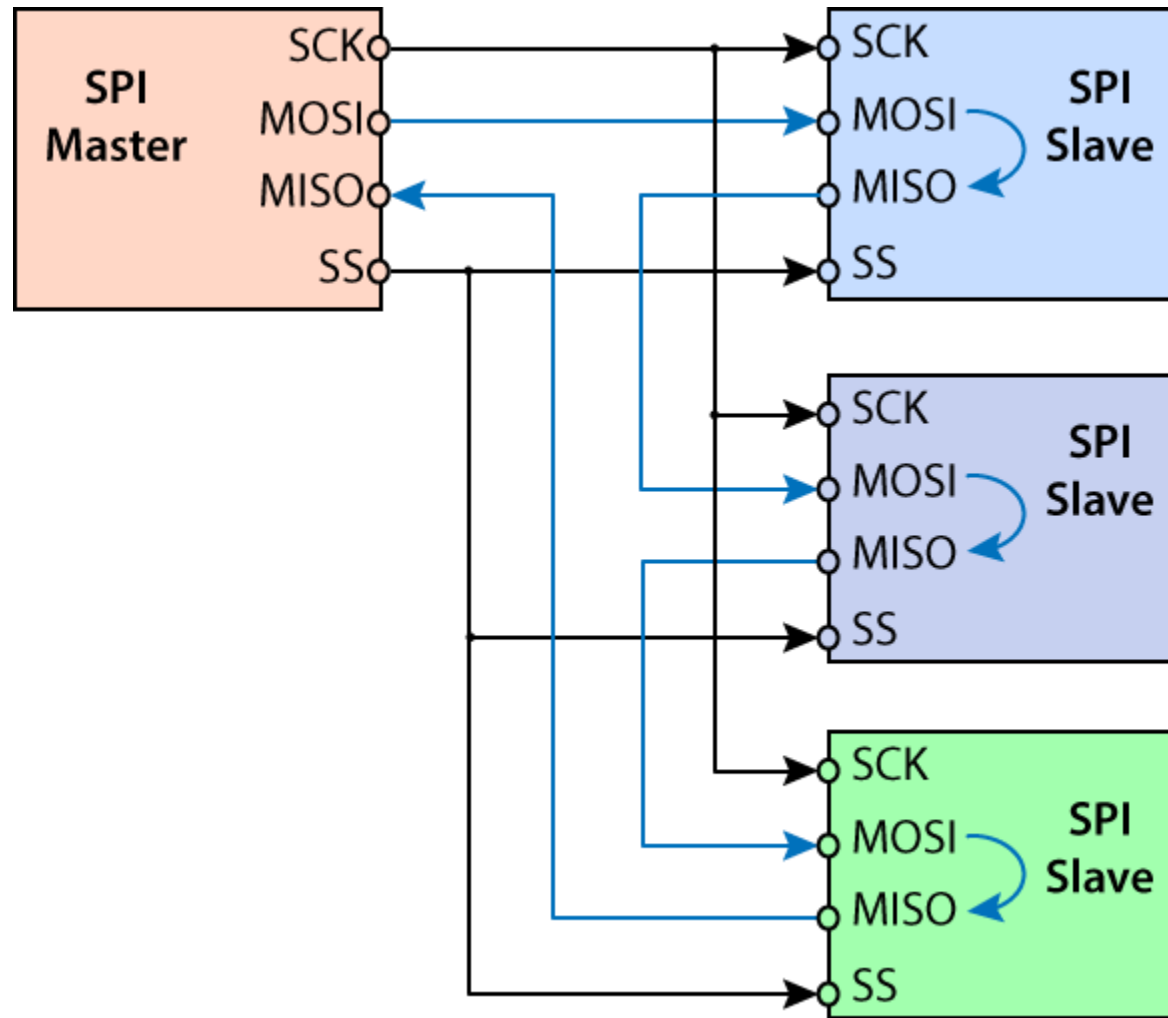
## Delay()



## Millis()

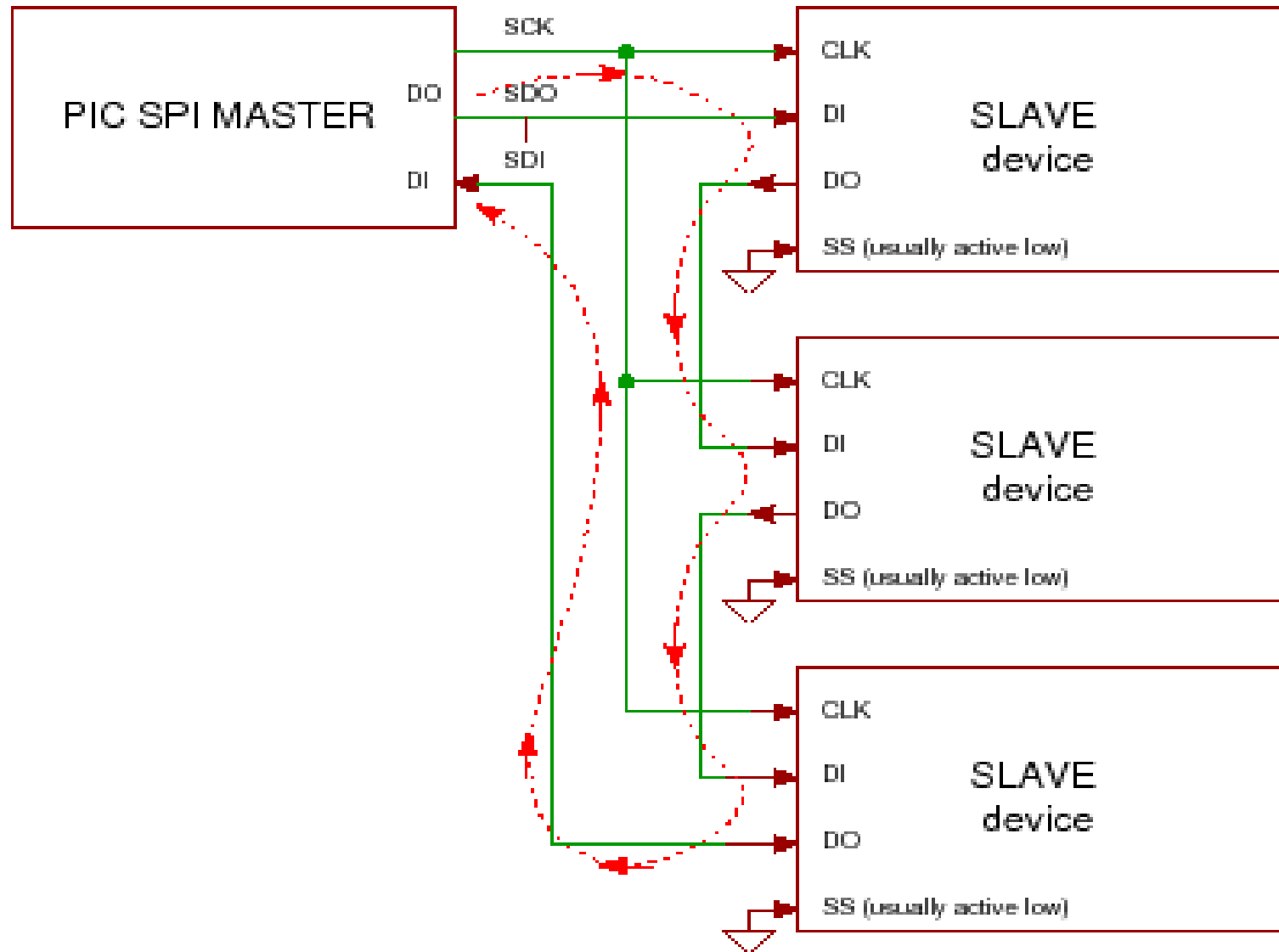


# --> Lab3 Daisy Chain with Shared Buffer





# --> Lab3 Daisy Chain with Shared Buffer



ตารางแสดงรหัส ASCII แทนตัวอักษรภาษาอังกฤษและภาษาไทย

				b7	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1
				b6	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1
				b5	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1
				b4	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
b3	b2	b1	b0																
0	0	0	0					0	@	P	'	p				ฐ	ภ	ะ	เ
0	0	0	1				!	1	A	Q	a	q				ก	ท	ม	ั
0	0	1	0				"	2	B	R	b	r				ข	ฅ	ย	า
0	0	1	1				#	3	C	S	c	s				บ	ณ	ร	ำ
0	1	0	0				\$	4	D	T	d	t				ค	ต	ฤ	ิ
0	1	0	1				%	5	E	U	e	u				ค	ด	ล	ี
0	1	1	0				&	6	F	V	f	v				พ	ถ	ภ	ี
0	1	1	1				'	7	G	W	g	w				ง	ท	ว	ี
1	0	0	0				(	8	H	X	h	x				จ	ช	ศ	ุ
1	0	0	1				)	9	I	Y	i	y				ฉ	น	ษ	ู
1	0	1	0				*	:	J	Z	j	z				ช	บ	ส	ุ
1	0	1	1				+	:	K	[	k	{				ช	ป	ท	ุ
1	1	0	0				,	<	L	\	l					ณ	ผ	พ	ุ
1	1	0	1				-	=	M	]	m	}				ญ	ฝ	อ	
1	1	1	0				.	>	N	^	n	~				ฎ	พ	อ	
1	1	1	1				/	?	O	-	o					ฎ	ฟ	ร	฿



# THANK YOU

Insert the Subtitle of Your Presentation