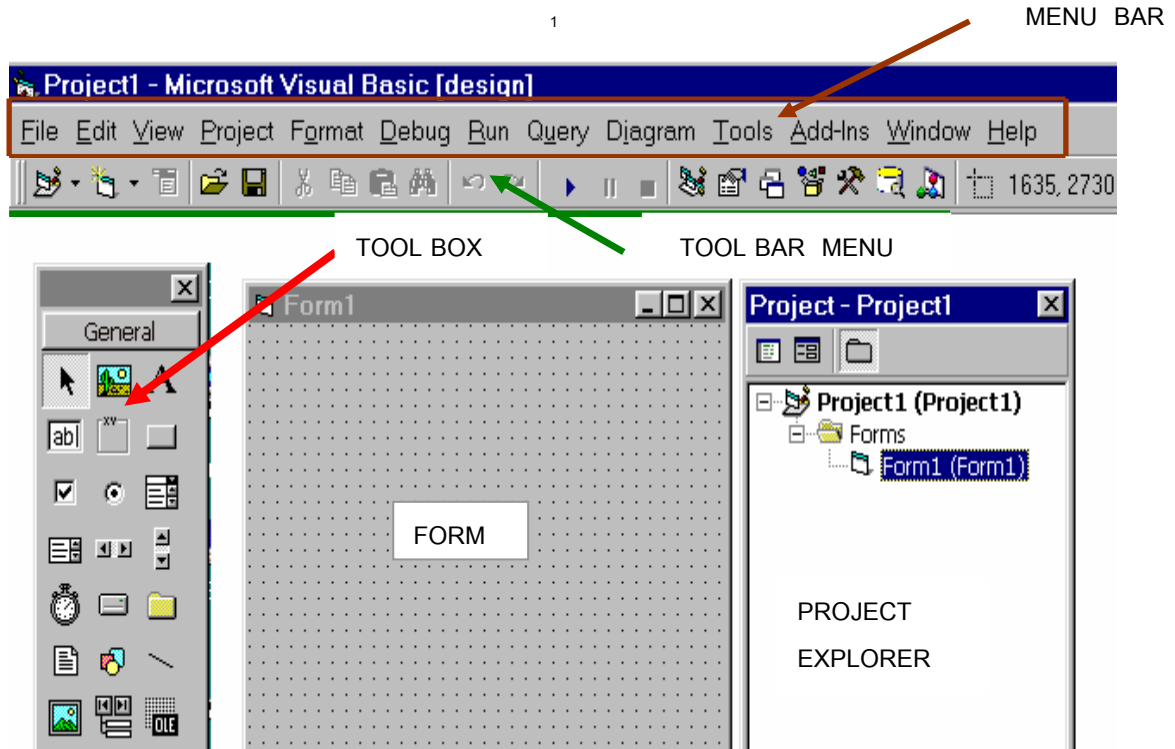


สภาพแวดล้อมของโปรแกรม VISUAL BASIC V6.0

1



รูป 1.1 รูปร่างหน้าต่างของโปรแกรม MICROSOFT VISUAL BASIC V6.0

จากรูป 1.1 แสดงถึงองค์ประกอบพื้นฐานของโปรแกรม MICROSOFT VISUAL BASIC V6.0 เมื่อเราเลือกรูปแบบในการพัฒนาโปรแกรมแบบ STANDARD EXE ซึ่งจะประกอบด้วย

- 1 ส่วนของ tab menu bar และ toolbar menu เหมือนกับโปรแกรมทั่วไป
- 2 ส่วนของกล่องเครื่องมือ (tool box) ซึ่งเป็นเครื่องมือมาตรฐานที่ visual basic สร้างมา
- 3 ส่วนของ form เป็นส่วนที่ใช้สำหรับนำเครื่องมือในกล่องเครื่องมือมาวางเพื่อสร้างภาพในการติดต่อกับผู้ใช้โปรแกรม
- 4 ส่วนของ project explorer เป็นที่ใช้สำหรับดูองค์ประกอบของ project ปัจจุบันว่าประกอบด้วยอะไรบ้างจากรูป 1.1 จะเห็นว่า project ประกอบด้วย form แค่ 1 เดียว

2

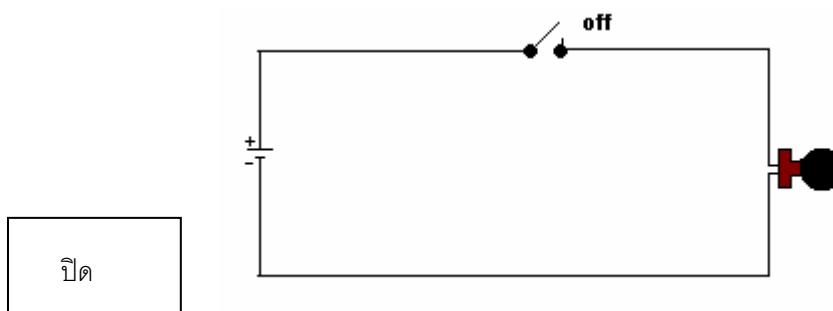
1

ขั้นตอนการพัฒนาโปรแกรม

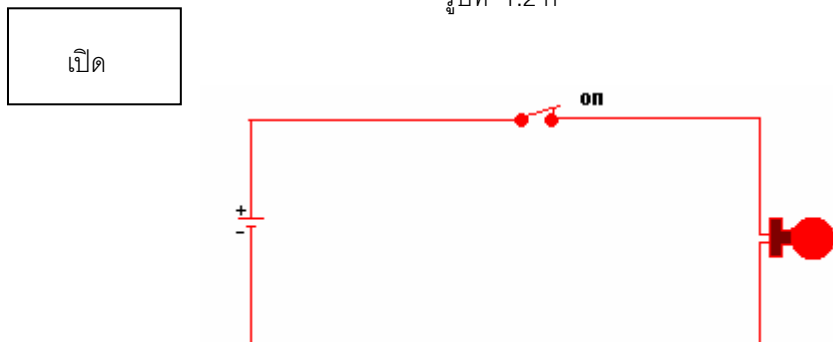
เนื่องจาก visual basic เป็นโปรแกรมที่ใช้รูปแบบการพัฒนาโปรแกรมในรูปแบบ graphic user interface (gui) ส่วนสำคัญก็คือการออกแบบรูปร่างหน้าต่างเพื่อสื่อสารกับผู้ใช้โปรแกรมให้มากที่สุดเพื่อให้โปรแกรมที่สร้างขึ้นใช้งานง่ายเป็นที่ถูกอกถูกใจของผู้ใช้มากที่สุด เพราะฉะนั้นขั้นตอนในการสร้างโปรแกรมจะมีหลัก ๆ อยู่ 2 ส่วนด้วยกันคือ

- 1 ขั้นตอนของการสร้างภาพ
- 2 ขั้นตอนการเขียนโปรแกรมเพื่อให้เกิดงานตามที่ได้คาดหวังเอาไว้

ตัวอย่างที่ 1.1



รูปที่ 1.2 ก



รูปที่ 1.2 ข

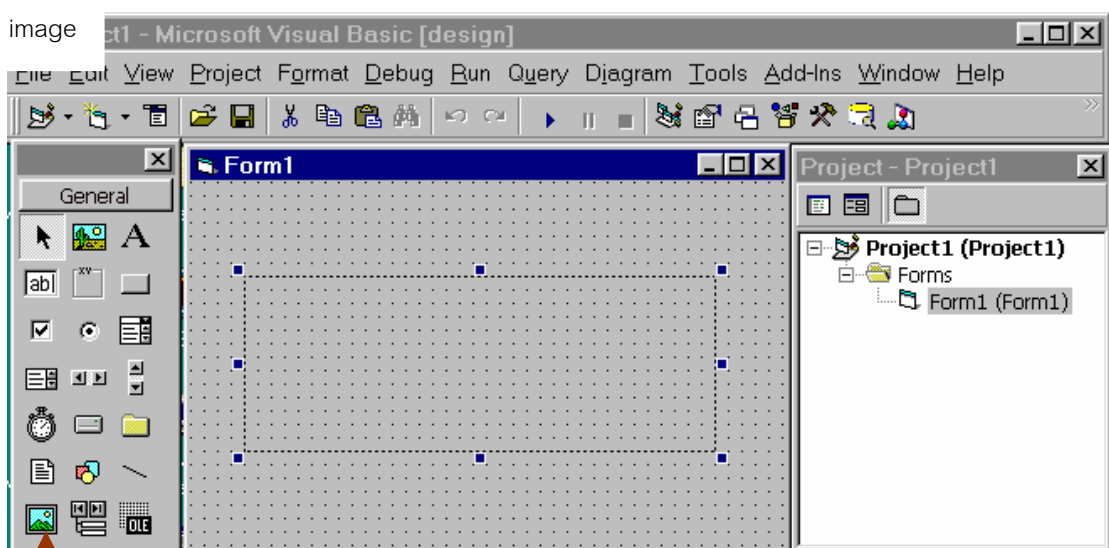
สมมติเราจะสร้าง project ที่มีชื่อว่า basic circuit และเรามีเงื่อนไขของ project ดังต่อไปนี้

- 1 เมื่อเริ่มต้นการทำงานให้หน้าจอคอมพิวเตอร์แสดงรูป 1.2 ก
- 2 เมื่อผู้ใช้คลิกปุ่มเปิดให้หน้าจอคอมพิวเตอร์แสดงรูป 1.2 ข
- 3 และเมื่อผู้ใช้คลิกปุ่มปิดให้หน้าจอคอมพิวเตอร์แสดงรูป 1.2 ก ตามเดิม

ภาพสร้าง project basic circuit



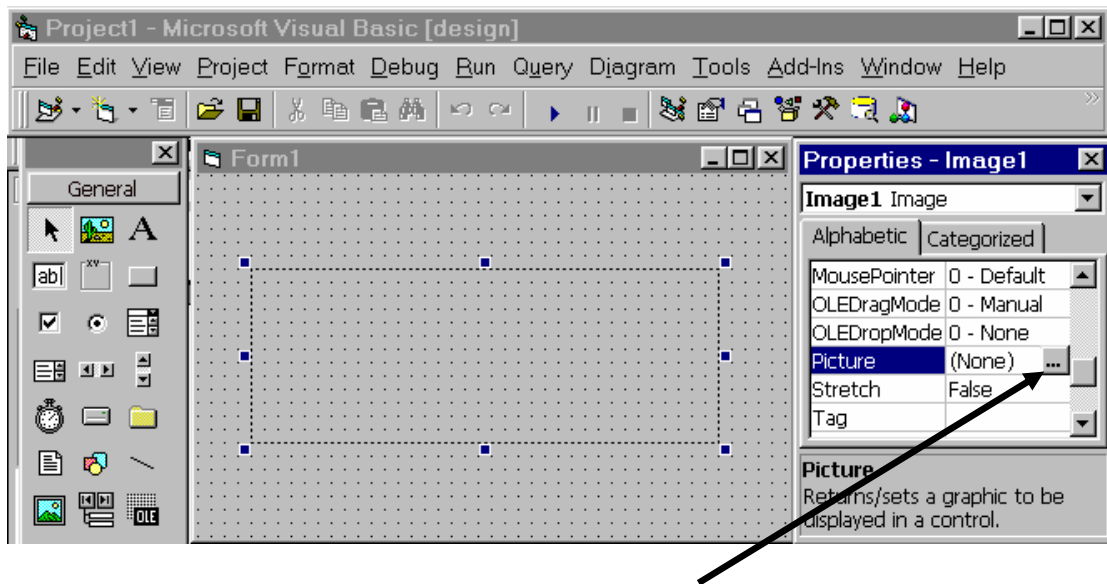
รูป 1.3 เลือกรูปแบบการพัฒนาโปรแกรมแบบ standard exe



รูป 1.4 นำเครื่องมือ image จาก tool box มาวางบน form

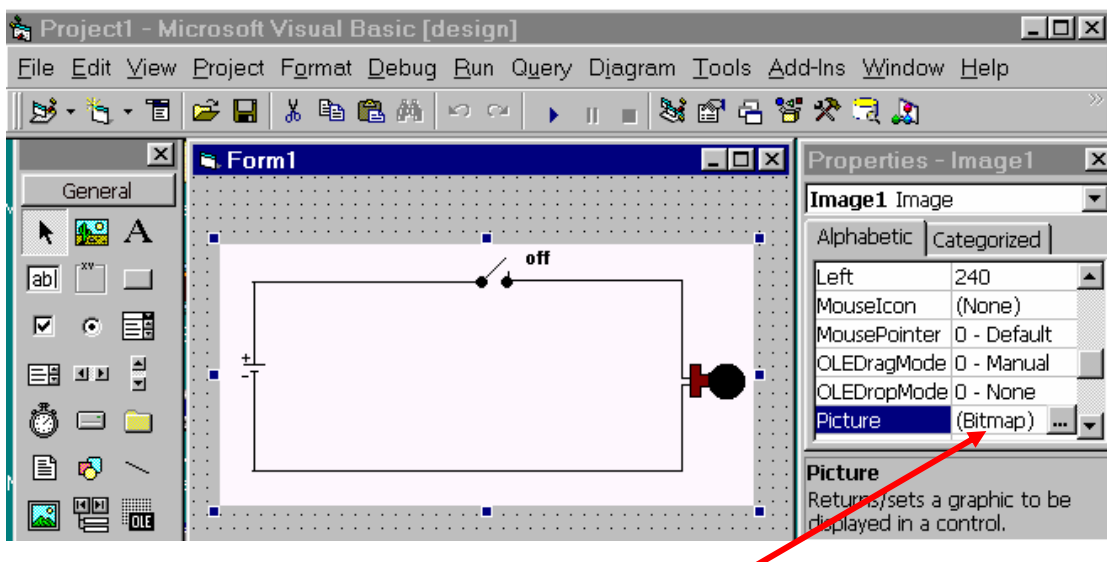
ขั้นตอนการสร้างภาพ

- 1 เลือกเครื่องมือ image เพื่อใช้ในการแสดงรูปวงจรไฟฟ้า
- 2 ใส่คุณสมบัติ (properties) picture ผ่านทาง windows properties โดยการคลิกที่วัตถุ image ให้ active ดังรูป 1.4 ตามด้วย กด keyboard F4



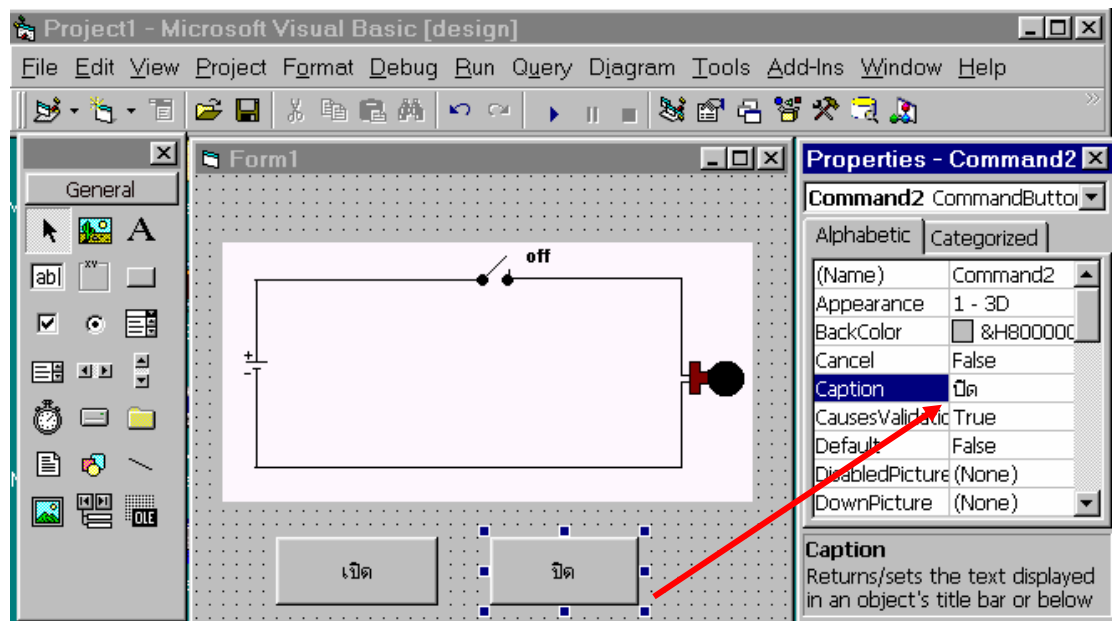
รูป 1.5 แสดงตำแหน่งของ properties picture ของวัตถุ image ใน windows properties

- 3 click ตรงตำแหน่งลูกศรชี้ในภาพ 1.5 เพื่อเลือกภาพให้กับวัตถุ image



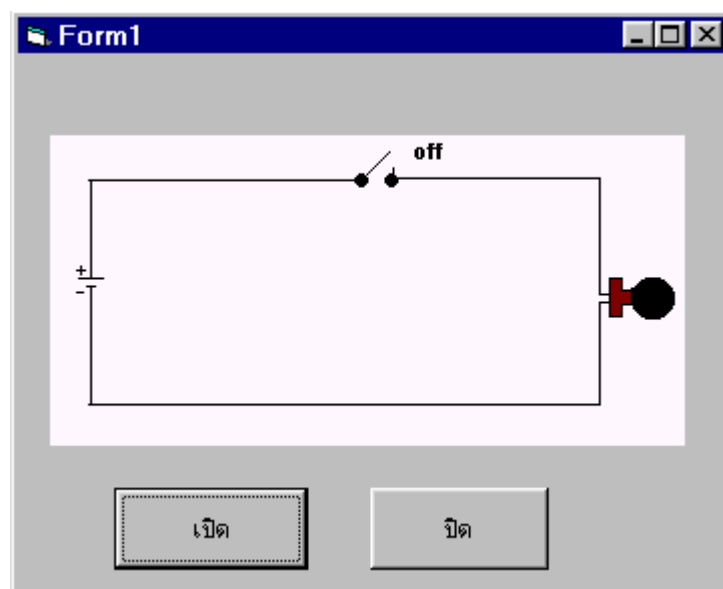
รูป 1.6 properties picture ของวัตถุ image เปลี่ยนตามภาพที่เราต้องการ

- 4 สังเกต ตรงตำแหน่งลูกศรชี้จะบอกชนิดของ file รูปภาพ
- 5 สร้างปุ่มในการควบคุมวงจรไฟฟ้า



รูป 1.7 สร้างปุ่มควบคุมการปิดเปิดวงจรไฟฟ้าโดยใช้ properties caption ของ command button เป็นป้ายบอกชื่อปุ่ม

5 ทดลองรันโปรแกรมโดย กด key F5 หรือ ปุ่ม start

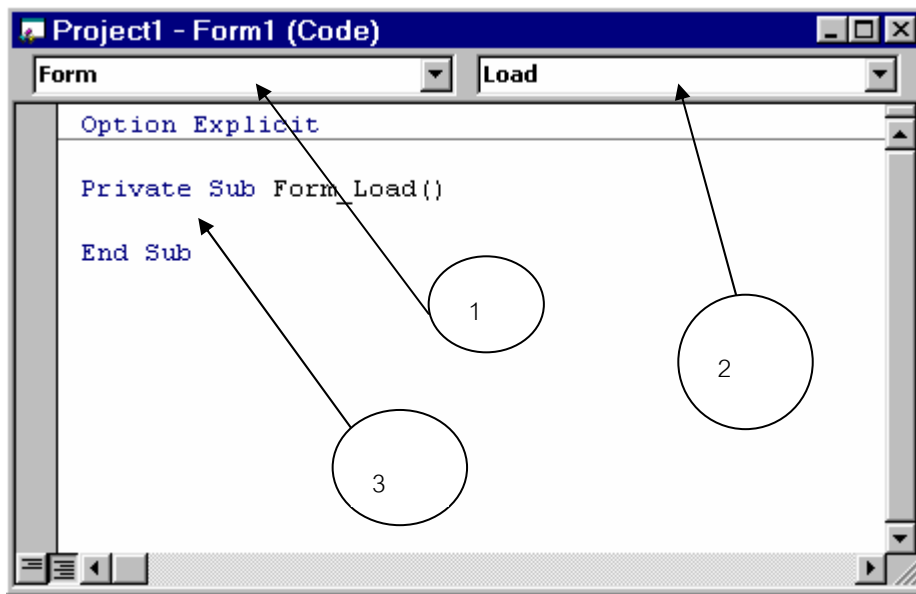


รูป 1.8 แสดงการรันโปรแกรมหลังจากขั้นตอนสร้างภาพเสร็จสมบูรณ์

6 ทดลอง click ที่ปุ่ม เปิด ปิด จะยังไม่มีเปลี่ยนแปลงตามที่เราคาดหวังไว้ตั้งแต่ต้นเนื่องจาก ยังขาดขั้นตอนของการเขียนโปรแกรมกับกับให้กับปุ่ม เปิด และ ปิด ในขณะที่ผู้ใช้ click ปุ่ม

ขั้นตอนการเขียนโปรแกรม

- 1 การเข้าสู่หน้าต่างการเขียนโปรแกรมของ visual basic
double click บน form หรือ วัตถุตัวใดตัวหนึ่งที่วางอยู่บน form



รูป 1.9 แสดงหน้าต่างสำหรับเขียนโปรแกรม

ในส่วน editor ของ visual basic ได้ถูกออกแบบเพื่อให้เหมาะกับการพัฒนาโปรแกรมในรูปแบบเชิงการกำหนดวัตถุเป้าหมาย (object – oriented – programming) อาจจะพูดได้ว่ากระบวนการต่าง ๆ ที่เกิดขึ้นจะถูกกำหนดโดยโปรแกรมย่อยของเหตุการณ์ต่าง ๆ ที่เกิดขึ้นกับ วัตถุนั้น ๆ จากรูป 1.9 จะเห็นว่าการแบ่ง editor ของ visual basic ออกเป็นสามส่วนด้วยกัน คือ

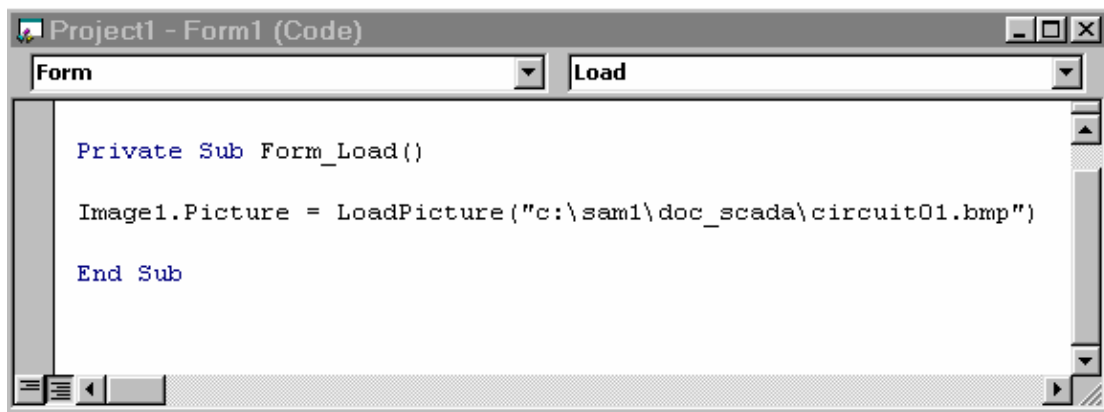
- 1 บอกวัตถุเป้าหมาย
- 2 บอกเหตุการณ์ที่เกิดขึ้นกับวัตถุเป้าหมาย
- 3 โปรแกรมย่อยในเหตุการณ์นั้น ๆ

ซึ่งถ้าเราคู่กันเคยกับระบบปฏิบัติการของ windows เหตุการณ์ที่เกิดขึ้นบ่อย ๆ ได้แก่ เหตุการณ์ click, double click , keypress และ key up เป็นต้น

เพราะฉะนั้นตามเงื่อนไขในตัวอย่าง 1.1 ระบุว่า

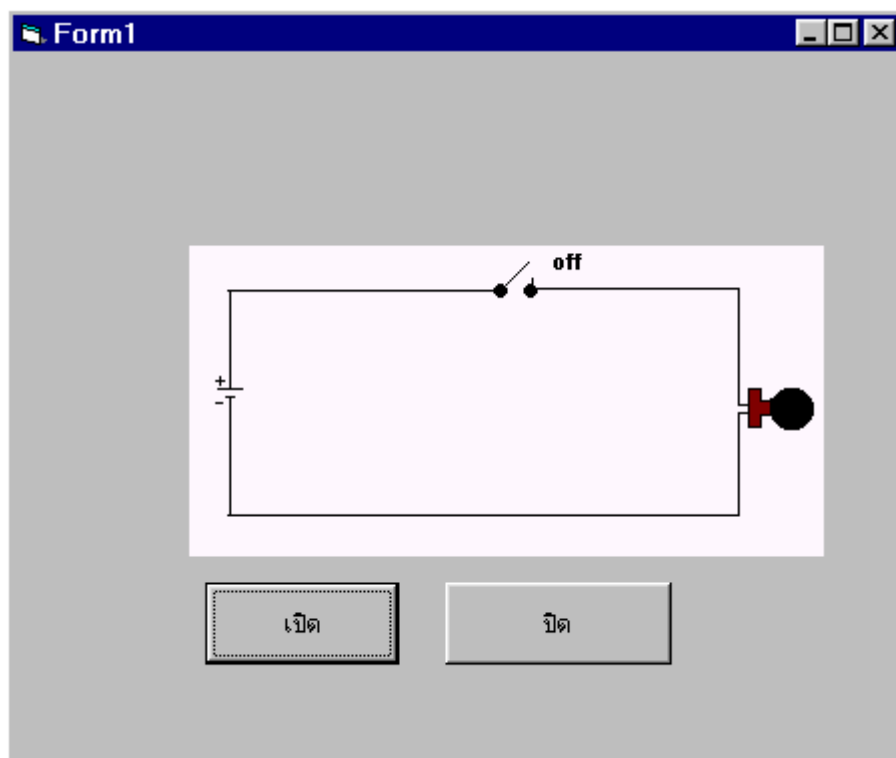
- 1 เมื่อเริ่มต้นการทำงานให้หน้าจอคอมพิวเตอร์แสดงรูป 1.2 ก

การเขียนโปรแกรมเพื่อให้เป็นไปตามเงื่อนไขทำได้ดังนี้



รูป 1.10 โปรแกรมย่อยในวัตถุ form เหตุการณ์การ โหลด

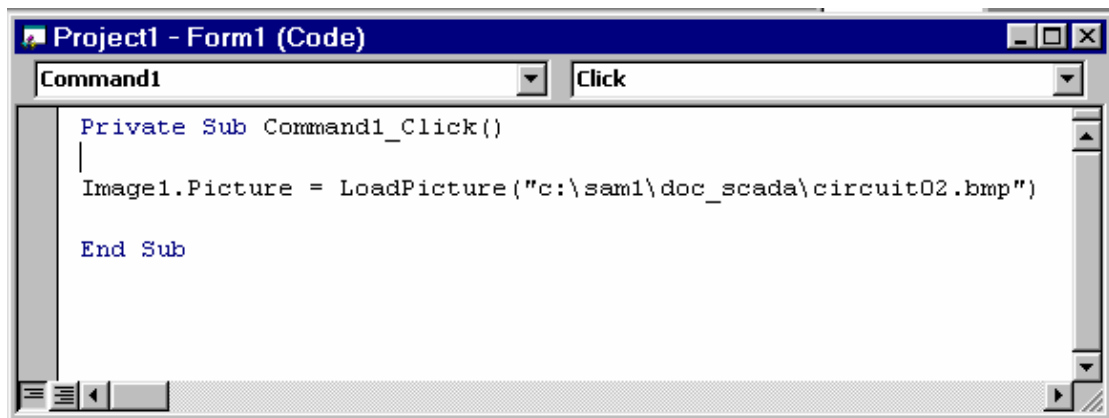
เมื่อ run โปรแกรม จะได้ตามรูป 1.11



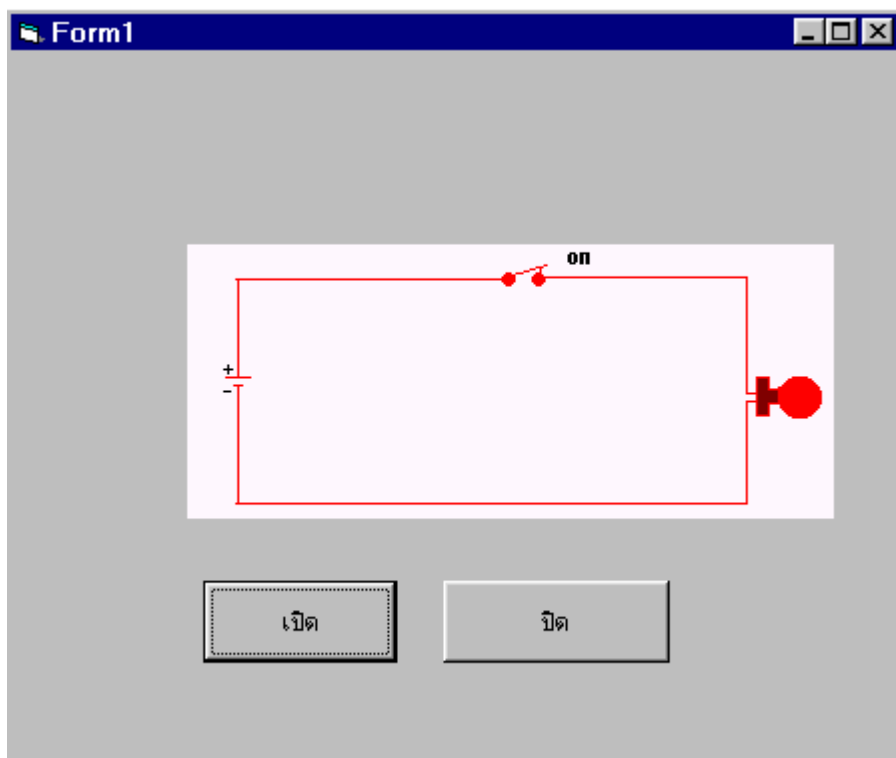
รูป 1.11 โปรแกรมทำงานตามคำสั่ง

Image1.Picture = LoadPicture("c:\sam1\doc_scada\circuit01.bmp")

- 2 เมื่อผู้ใช้คลิกปุ่มเปิดให้หน้าจอคอมพิวเตอร์แสดงรูป 1.2 ข
- เขียนโปรแกรมไว้ในโปรแกรมย่อยดังรูป 1.12

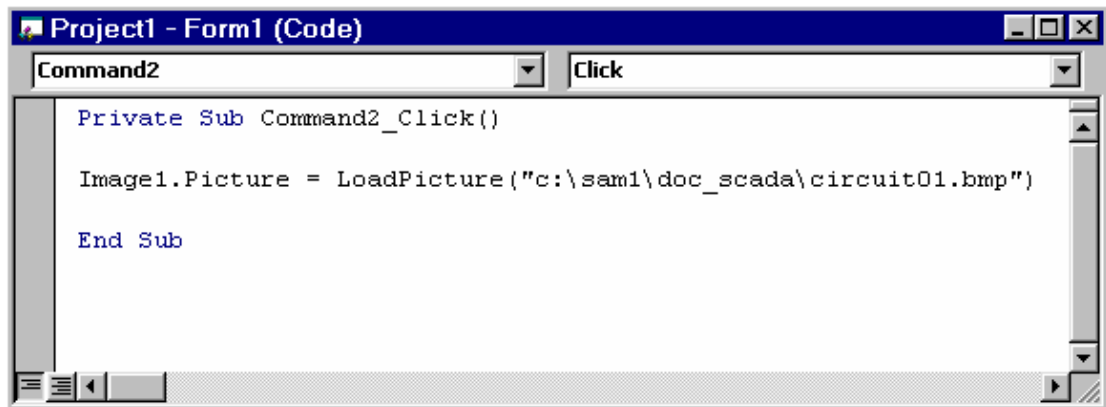


รูป 1.12 โปรแกรมย่อยของวัตถุ command1 (ปุ่มเปิด) เหตุการณ์การ click
เมื่อ run โปรแกรมแล้ว click ปุ่มเปิด จะได้ตามรูป 1.13



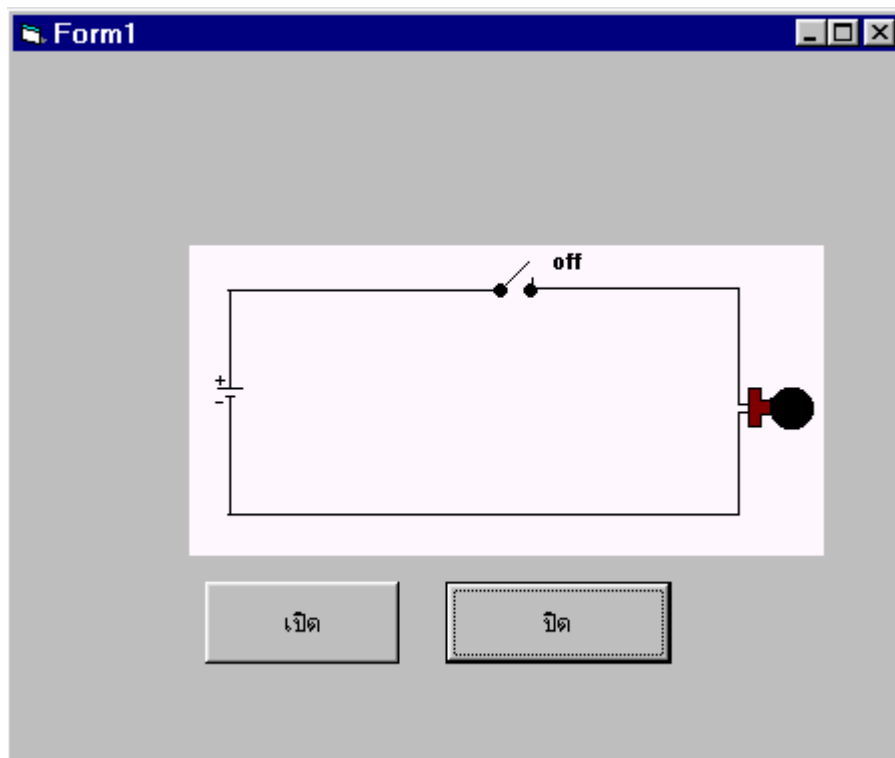
รูป 1.13 เมื่อ run แล้ว click ปุ่ม เปิด

- 2 และเมื่อผู้ใช้คลิกปุ่มปิดให้หน้าจอคอมพิวเตอร์แสดงรูป 1.2 ก ตามเดิม
เขียนโปรแกรมไว้ในโปรแกรมย่อยดังรูป 1.14



รูป1.14 โปรแกรมย่อยที่ปุ่ม ปิด (เหตุการณ์การ click)

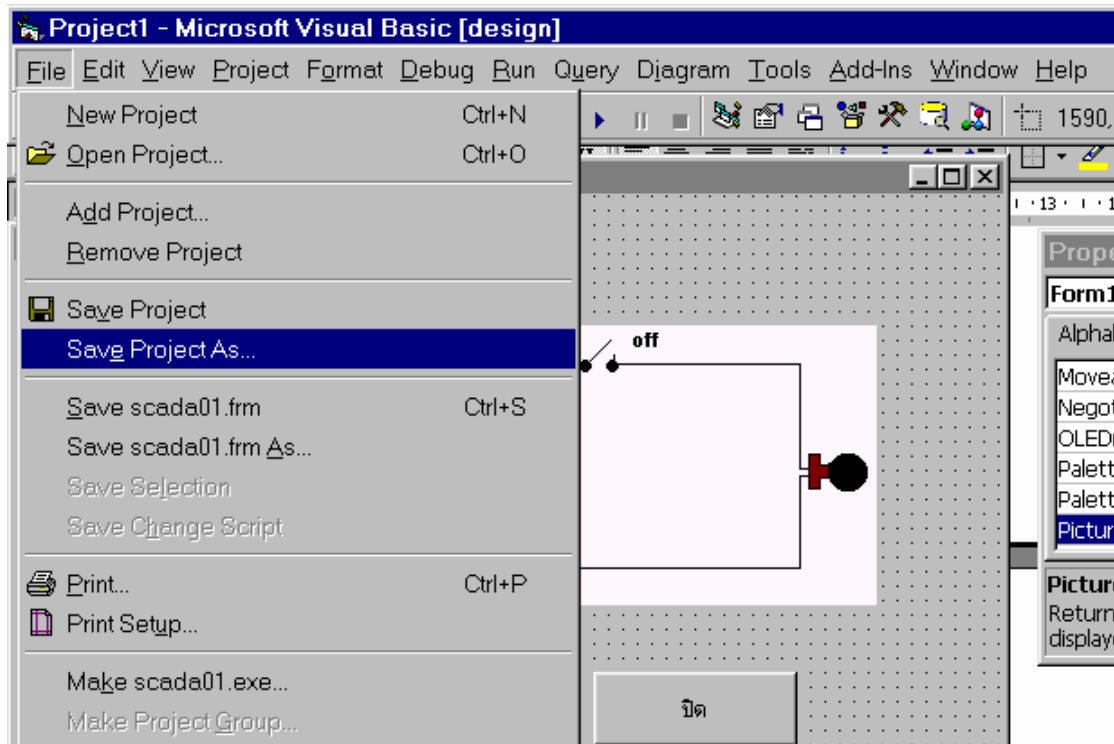
เมื่อ run โปรแกรมแล้ว click ปุ่มเปิด จะได้ตามรูป 1.15



รูป1.15 ขณะrun โปรแกรมแล้วclick ปุ่มปิด

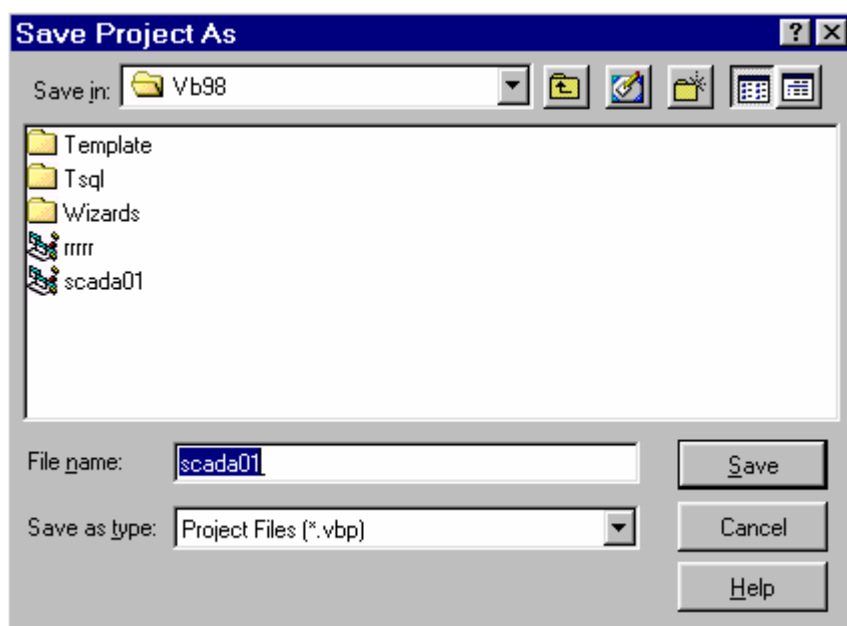
ขั้นตอนการ save project

- 1 เลือกเมนู file → save or save as project



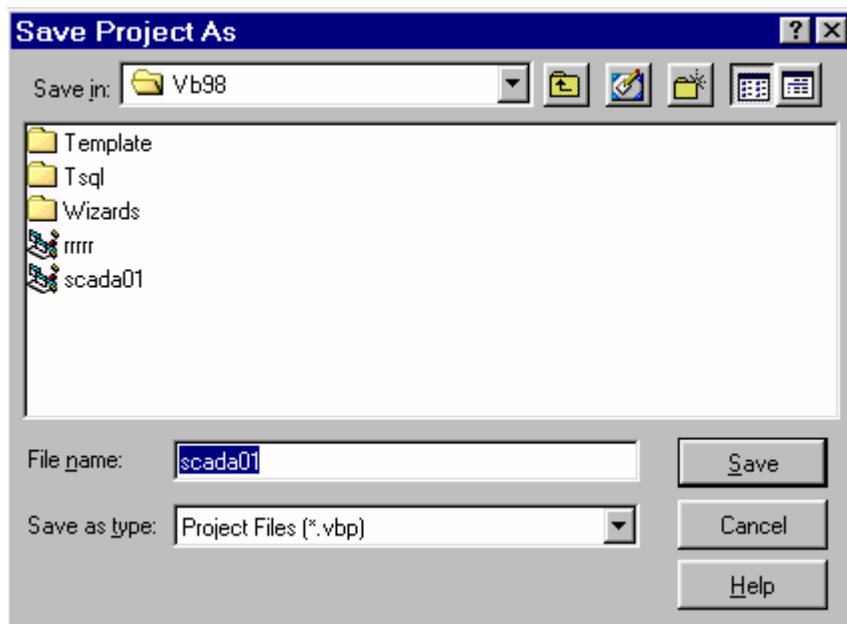
รูป 1.16 การเรียกใช้เมนู save project

- 2 save file นามสกุล .frm



รูป 1.17 save file .frm

3 save file .vbp



รูป1.18 save file .vbp

สรุปสิ่งที่ต้องทำความเข้าใจหลังจากเดินผ่านบทที่ 1

- เข้าใจขั้นตอนการพัฒนาโปรแกรมบนรูปแบบการกำหนดวัตถุเป้าหมาย
- เข้าใจขั้นตอนการสร้าง project visual basic
- เข้าใจการใช้ editor ของ visual basic

2

ตัวกระทำทางคณิตศาสตร์และฟังก์ชันภายในของ visual basic

2.1 ตัวกระทำและตัวเปรียบเทียบทางคณิตศาสตร์

ลำดับ	สัญลักษณ์	การกระทำทางคณิตศาสตร์	ตัวอย่าง	ค่าที่ได้
1	+	บวก	$A=2+3$	$A=5$
2	-	ลบ	$A=3-2$	$A=1$
3	*	คูณ	$A=3*3$	$A=9$
4	/	หาร	$A=6/2$	$A=3$
5	<	น้อยกว่า	$A=3, B=4$ $A < B$	TRUE เป็นจริง
6	<=	น้อยกว่าหรือเท่ากับ	$A=3, B=3$ $A <= B$	TRUE เป็นจริง
7	>	มากกว่า	$A=3, B=4$ $B > A$	TRUE เป็นจริง
8	>=	มากกว่าหรือเท่ากับ	$A=4, B=4$ $A >= B$	TRUE เป็นจริง
9	<>	ไม่เท่ากับ	$A=3, B=4$ $A <> B$	TRUE เป็นจริง
10	^	ยกกำลัง	$A=3$ $A = A^2$	$A=9$

2.2 ฟังก์ชันภายใน visual basic

2.2.1 ฟังก์ชันที่ใช้งานเกี่ยวกับตัวอักษร(string)

ลำดับ	ฟังก์ชัน(Function)	การทำงาน	ตัวอย่างการใช้งาน	ค่าที่ได้
1	LEN	เป็นคำสั่งสำหรับหาจำนวนตัวอักษรในตัวแปร String	$A\$="ELECT"$ $B=LEN(A)$	$B=5$
2	INSTR(K\$,M\$)	คำสั่งสำหรับตรวจสอบตัวอักษรในตัวแปร String ว่ามีอยู่ในตัวแปรนั้นหรือไม่โดยจะเริ่มเช็คจากซ้ายมือของตัวอักษร	$A\$="ELECTRIC"$ $B=INSTR(A$, "C")$	$B=4$
3	STR\$(N)	คำสั่งสำหรับเปลี่ยนค่าตัวเลขเป็นตัวอักษร เมื่อ N เป็นค่าตัวเลขจำนวนเต็ม	$A=123$ $B\$=STR(A)	$B\$="123"$
ลำดับ	ฟังก์ชัน(Function)	การทำงาน	ตัวอย่างการใช้งาน	ค่าที่ได้

4	LEFT\$(S\$,N)	เป็นคำสั่งสำหรับตัดตัวอักษรเริ่มจากด้านซ้ายมาทางขวาจำนวน N ตัวอักษร	A\$="SAMRAENG" B\$=LEFT\$(A\$,3)	B\$="SAM"
5	RIGHT\$(A\$,N)	เป็นคำสั่งสำหรับตัดตัวอักษรเริ่มจากด้านขวามาทางซ้ายจำนวน N ตัวอักษร	A\$="SAMRAENG" B\$=RIGHT\$(A\$,3)	B\$="ENG"
6	LCASE\$(B\$)	เปลี่ยนตัวอักษรให้เป็นตัวพิมพ์เล็ก	A\$="SAMRAENG" B\$=LCASE(A\$)	B\$="samraeng"
7	CHR\$(B)	เปลี่ยน Character Code เป็นตัวอักษร B As long variable	A\$=CHR\$(65)	A\$="A"
8	ASC(B\$)	เปลี่ยนค่า String เป็น Ascii Code	K=ASC("A")	K=65
9	Rtrim\$(b\$)	ใช้ตัดช่องว่างด้านหน้าและด้านหลังตัวอักษร	A\$=" samreang " B\$=Rtrim\$(A\$)	B\$="samreang"
10	STRCOMP(A\$,B\$)	ใช้เปรียบเทียบค่าในตัวแปร A\$ กับ B\$	A\$="SAM" B\$="SAM" K=STRCOMP(A\$,B\$)	K=0 เนื่องจาก A\$=B\$
11	INSTR(N,A\$,B\$)	ใช้หาข้อความ B\$ โดยให้ N เป็นตำแหน่งเริ่มต้นในการค้นหา	A\$="SAMRAENG" K=INSTR(1,A\$,"E")	K=6 E อยู่ตำแหน่งที่ 6
12	SPACE\$	ใส่ช่องว่างระหว่างตัวอักษร	"SAMREANG" SPACE\$(10) "TEMRAM"	SAMRAENG เว้น 10 ตัวอักษร TEMRAM

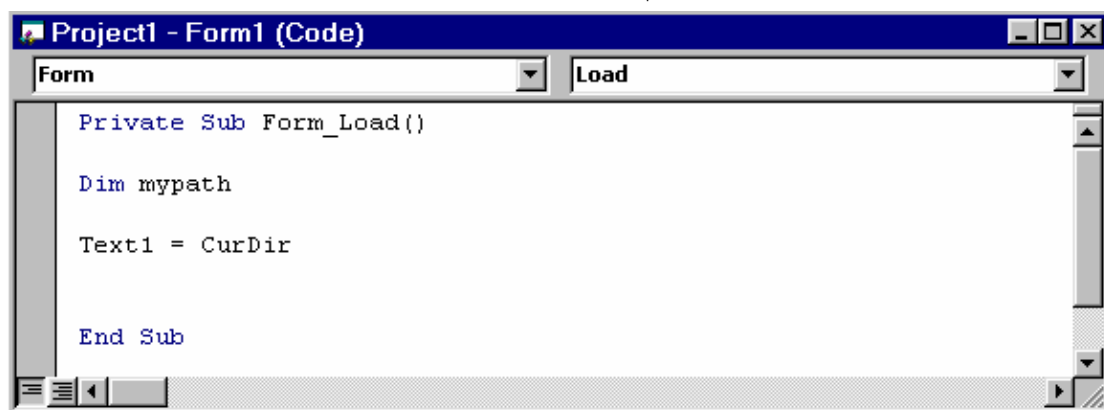
2.2.2 ฟังก์ชันที่ใช้ที่ใช้ในการจัดเก็บข้อมูล

2.2.2.1 CURDIR เป็นฟังก์ชันสำหรับแสดง path ปัจจุบัน

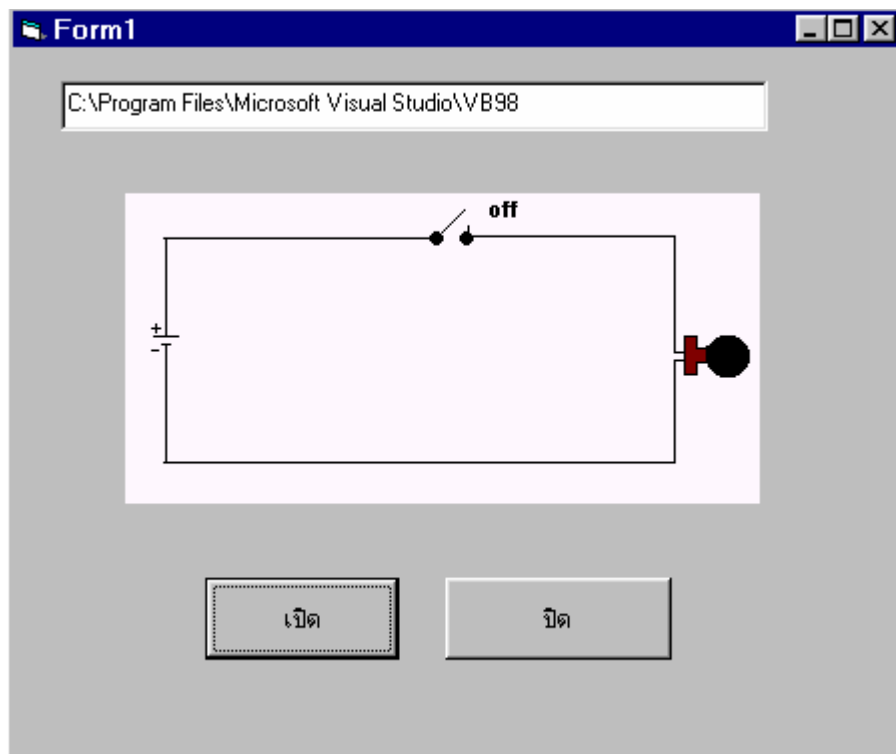
ตัวอย่าง สมเรากำลังทำงานอยู่ที่

path "C:\Program Files\Microsoft Visual Studio\VB98"

และเราจะใช้ text box แสดง path ปัจจุบัน



เมื่อ RUN PROGRAM text1 จะแสดง path ปัจจุบันตามคำสั่ง text1= CurDir



รูป 2.1 เมื่อ run program text1 display current path

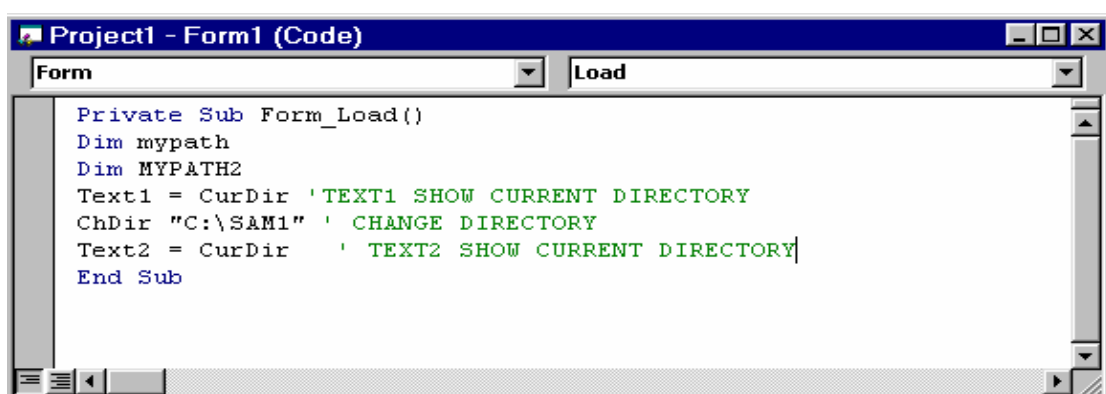
2.2.2.2 CHDIR "new path" เป็นฟังก์ชันสำหรับการเปลี่ยนแปลง path ปัจจุบันไปยัง new path

ตัวอย่าง

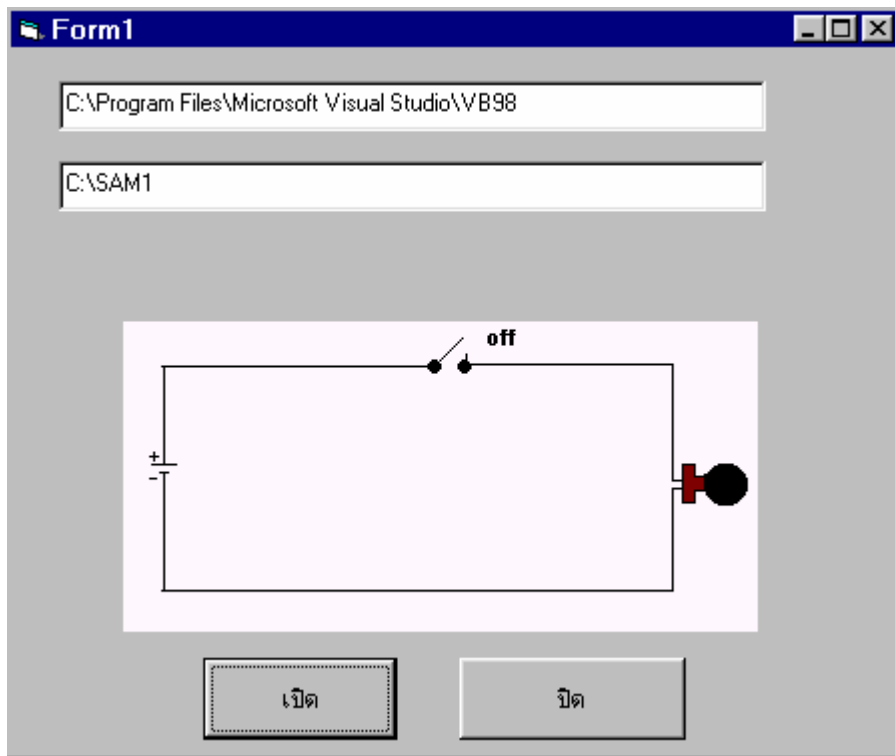
ตัวอย่าง สมเรากำลังทำงานอยู่ที่

path "C:\Program Files\Microsoft Visual Studio\VB98"

แล้วต้องการเปลี่ยนการทำงานไปที่ PATH " C:\SAM1"



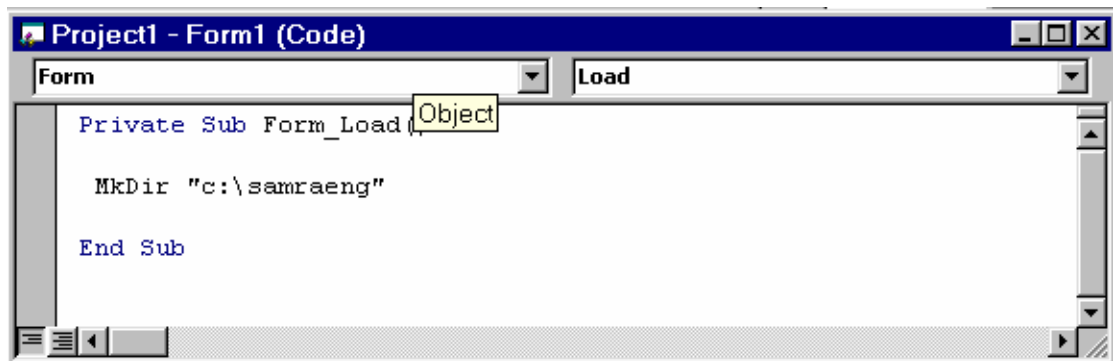
รูป 2.2 แสดงการใช้ ฟังก์ชัน ChDir



รูป 2.3 ผลการรันโปรแกรม ตามรูป 2.2

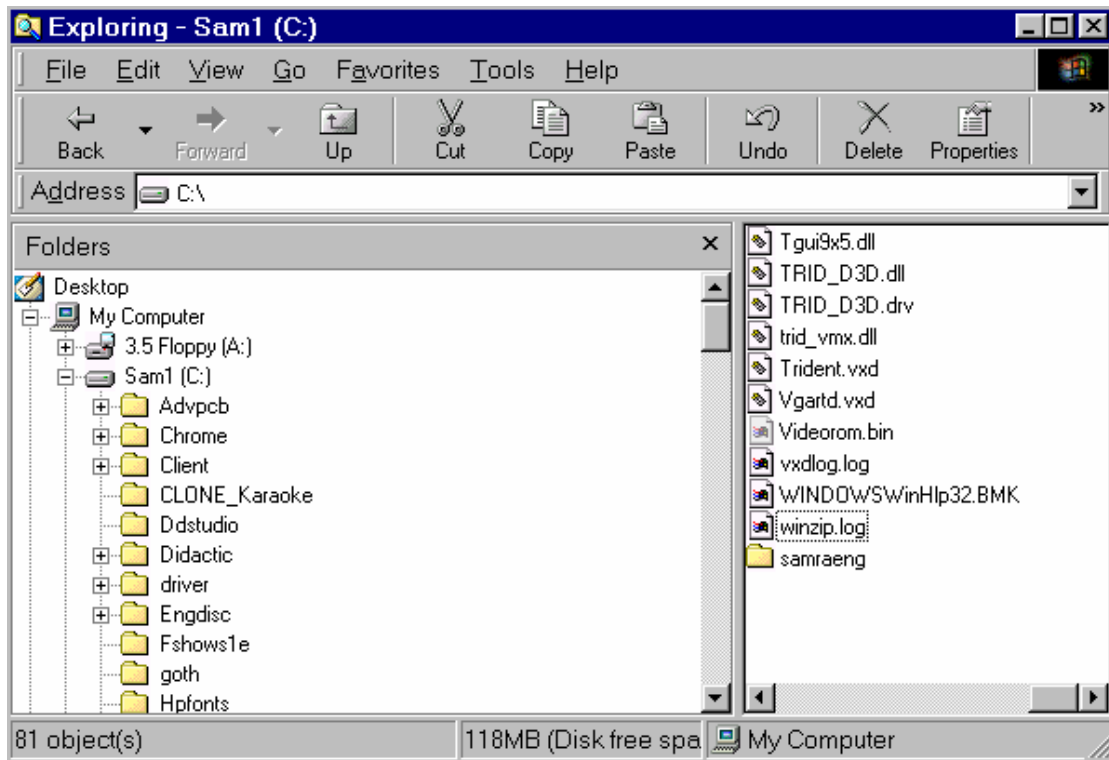
2.2.2.3 Mkdir ใช้สำหรับสร้าง Directory or folder

ตัวอย่าง สมมุติต้องการสร้าง Folder ชื่อ samraeng ใน drive c:



รูป 2.4 การใช้คำสั่งสร้าง directory

หมายเหตุ ชื่อ directory จะต้องไม่ซ้ำกับ directory ที่มีอยู่แล้ว



รูป 2.5 ใช้ windows explorer ดูผลหลังจากการ run program

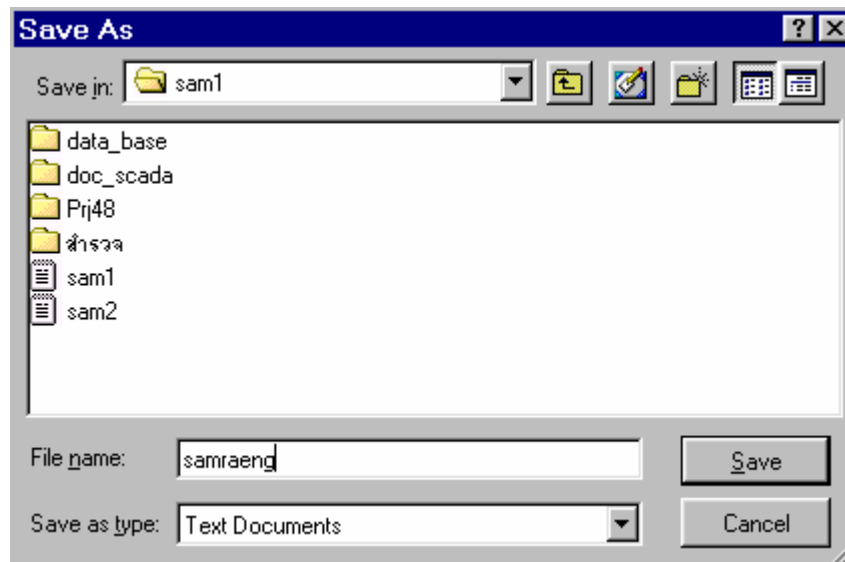
2.2.2.4 FileCopy source file, destination file เป็นฟังก์ชันที่ใช้สำหรับ COPY FILE

ตัวอย่าง

- ใช้ program notepad พิมพ์ข้อความสักหนึ่งข้อความแล้ว SAVE เป็นชื่อใดชื่อหนึ่งในที่นี้ผู้เขียนขอ save เป็นชื่อ samraeng ไว้ที่ path
"c:\sam1\samreang.txt"

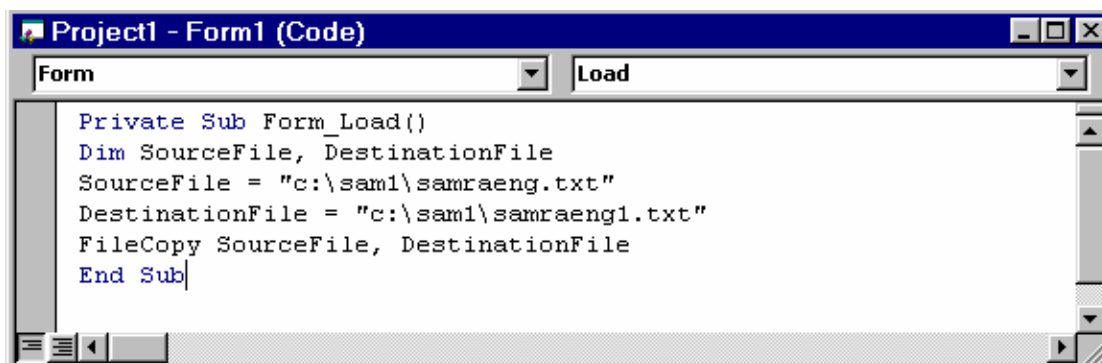


รูปที่ 2.6 ใช้ notepad พิมพ์ข้อความตามใจชอบ



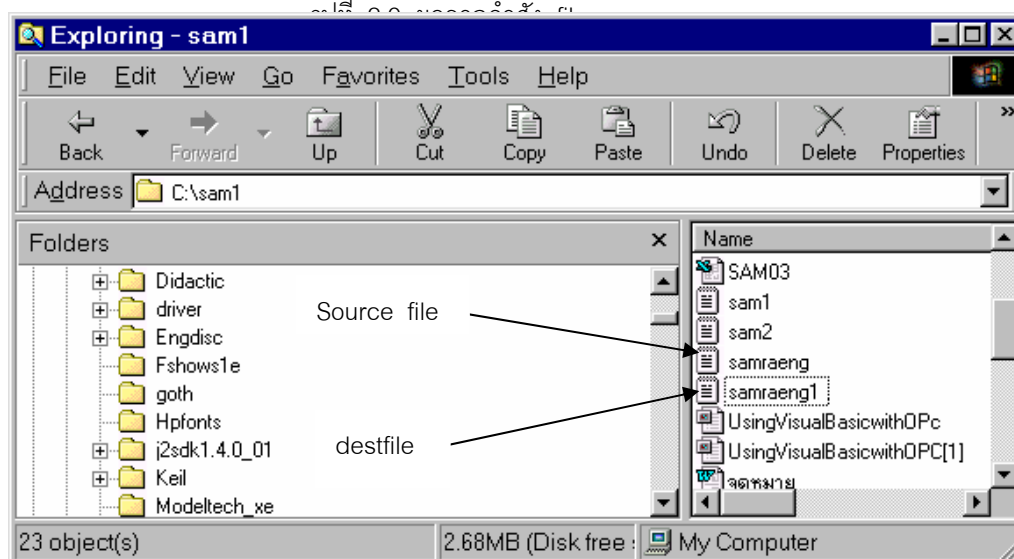
รูปที่ 2.7 save text file

- ทดลองเขียนโปรแกรมโดยใช้คำสั่ง filecopy

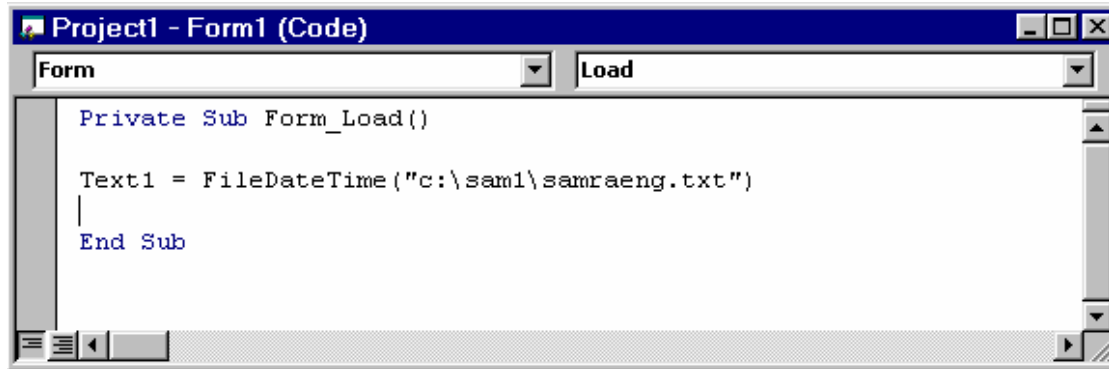


รูปที่ 2.8 ใช้คำสั่ง filecopy samraeng → samraeng1

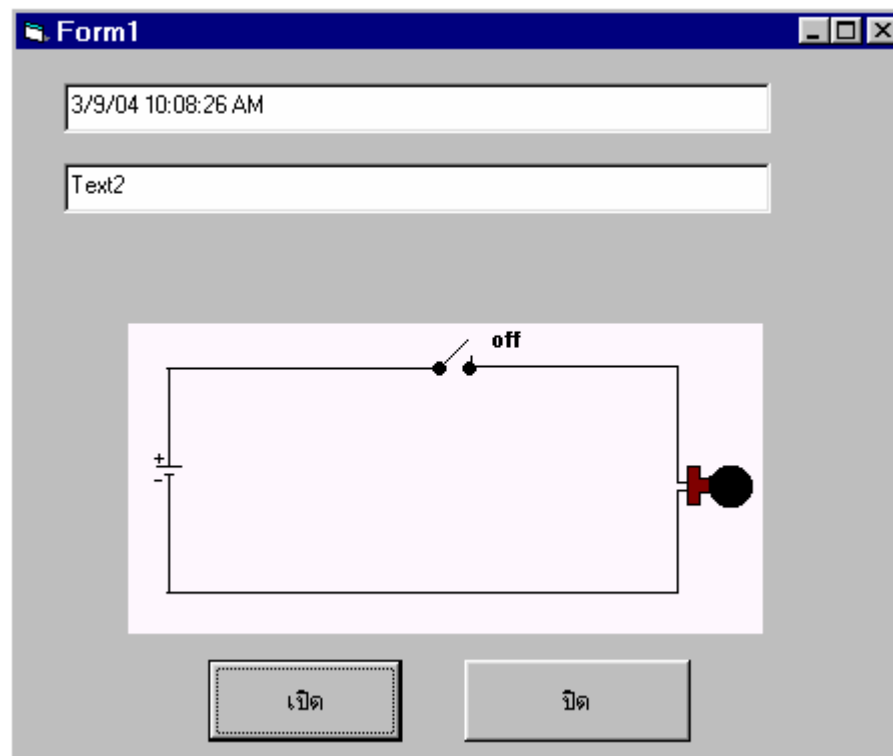
- ใช้ windows explorer ดูผลการทำงานของ program



2.2.2.5 filedatetime เป็นฟังก์ชันใช้บอกวันเวลาการเก็บบันทึก file



รูป 2.10 เขียนโปรแกรมการใช้คำสั่ง filedatetime



รูป 2.11 text1 แสดงวันเวลาการบันทึก file "c:\sam1\samreang.txt"