

Dual Function Push Button

Introduction:

Sometimes, either to save space, I/O pins or hardware it may be desirable to use a single pushbutton for multiple uses. This routine allows the pushbutton to be used for two functions. A short press will execute one subroutine while a longer press will execute a second subroutine. While this arrangement could be extended to more than two functions that may become a bit unwieldy unless it was an extra long push, say 10 seconds for a reset.

The routine below uses the pushbutton to generate interrupts, one when the switch is operated (input low) and one when the switch is released. The timer is used to measure the length of the switch operation. Less than 2 seconds is a short press while greater than 2 seconds is a long press. A very short operation (<100mS) is considered to be contact bounce and is ignored. A second version of the program is shown below the first one. It uses the STATIC command which was introduced into MMBasic for version 05.04.09. This second version is **NOT** suitable for the Maximate for that reason.

An internal pull up resistor is used but if the switch is mounted away from the board then either an extra 10k pull up resistor should be used and/or a 100nF (0.1uF) capacitor connected from the input pin to 0 volts.

Notable features:

Although this program makes use of the TIMER it does not change its value. The value is recorded at the first operation of the switch and also on the release, a subtraction is then made to determine the elapsed time. Flags are set in the interrupt routine so the main program loop can determine what action to take thus minimising the time spent in the interrupt routine.

Assumptions:

The switch is wired to connect the input pin to zero volts when operated (logic 0).

Acknowledgement:

The original code was by TassyJim on The Back Shed Forum

```
' Dual Function Push Button
' Original code by:
' Tassyjim on The Back Shed Forum
'
DIM integer bt
dim integer SwPin = 14 ' Pin number for the push button input

SETPIN SwPin,INTB,subB,PULLUP
DO
' Main program goes here
```

```
IF bt = 1 THEN sub1
IF bt = 2 THEN sub2
LOOP

SUB subB
  STATIC integer buttontimer
  LOCAL integer b
  IF PIN(SwPin) = 0 THEN ' button down
    buttontimer = TIMER
  ELSE 'button up
    b = TIMER - buttontimer ' this is how long the button was down
    IF b < 100 THEN ' too short, must be contact bounce
      bt = 0'do nothing
    ELSEIF b < 2000 THEN ' short press
      bt = 1 ' call first sub
    ELSE ' longer than ~two seconds
      bt = 2 ' call second sub
    ENDIF
  ENDIF
END SUB

SUB sub1
  PRINT "Switch on for less than 2 seconds"
  ' Funtion 1 code goes here
  bt = 0
END SUB

SUB sub2
  PRINT "Switch on for more than 2 seconds"
  ' Function 2 code goes here
  bt = 0
END SUB
```

Micromite only version:

```
' Dual Function Push Button version 2
' Original code by:
' Tassyjim on The Back Shed Forum
' Uses the STATIC command (added in MMBasic V5.04.09)
,

DIM integer bt
dim integer SwPin = 14 ' Pin number for the push button input

SETPIN SwPin,INTB,subB,PULLUP
DO
  ' Main program goes here
  IF bt = 1 THEN sub1
  IF bt = 2 THEN sub2
LOOP
```

```
SUB subB
  STATIC buttontimer
  LOCAL integer b
  IF PIN(SwPin) = 0 THEN ' button down
    buttontimer = TIMER
  ELSE 'button up
    b = TIMER - buttontimer ' this is how long the button was down
    IF b < 100 THEN ' too short, must be contact bounce
      bt = 0 'do nothing
    ELSEIF b < 2000 THEN ' short press
      bt = 1 ' call first sub
    ELSE ' longer than ~two seconds
      bt = 2 ' call second sub
    ENDIF
  ENDIF
END SUB

SUB sub1
  PRINT "Switch on for less than 2 seconds"
  ' Funtion 1 code goes here
  bt = 0
END SUB

SUB sub2
  PRINT "Switch on for more than 2 seconds"
  ' Function 2 code goes here
  bt = 0
END SUB
```

From:

<http://fruitoftheshed.com/wiki/> - **FotS**

Permanent link:

http://fruitoftheshed.com/wiki/doku.php?id=mmbasic:dual_function_push_button

Last update: **2024/01/19 09:30**

