Using MEMSIC MXD2020GL in Man Down Application

What is Man down?

Man down is a term used when an operator in the field is disabled/dead. This applies to firemen, policemen, soldiers, miners, nuclear reactor operators and others who are involved with hazardous materials/situations. In all instances, these operators are connected with each other and the central unit by means of radio. But in a situation where an individual is incapacitated, and unable to use the radio to call for help, would be deemed fully functional, where as in reality he/she might be dead/dying/trapped.
In all of these instances, there will be a need in radios and communication units to be able to measure movement and be able to present lack of it, by means of an alert signal.
Thus the man-down functionality in radios enables it to signal the central unit of debilitated operators.

General Operation theory

The purpose of this note is to demonstrate use of a Memsic device in a Man down application in its simplest form. Attributes like number of accelerometers; type of microcontroller, alarm conditions can be modified to be more suitable to a certain type application than others.
In essence, the operation revolves around lack of motion, which asserts the alarm state. The microcontroller samples the accelerometer at regular intervals and compares consecutive samples in an endless loop. If at any point two consecutive values are equal(after filtering for jitter and device min/max margins) the device will assert the mandown state suggesting that a lack of motion has been observed in that sampling period. The sampling frequency, filtering of the jitter can be easily changed in software to make the unit more or less sensitive to movement.

How can it be implemented?

Schematic

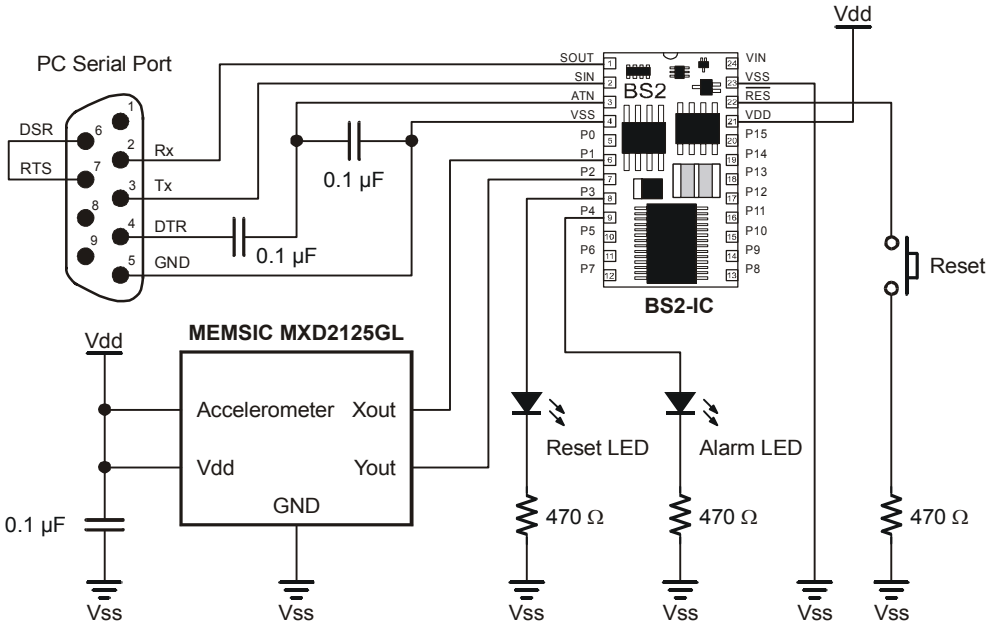

Fig1. Schematic of Demo Mandown circuit showing serial port

Major Components

MXD2020GL.

Memsic is a leading supplier in thermal accelerometers, provides different kinds of outputs, such as pulse width modulated and analog. As the author will discuss later in this note, the microcontroller used in this application is the Basic stamp (BS2 processor) from Parallax. This is a digital only device and therefore the choice of accelerometer inputs were narrowed down to digital as using and A/D could be avoided. The MXD2020GL is a consumer grade accelerometer (operational temperature 0C-70C) which digital outputs. The outputs are Pulse Width Modulated to 50% duty cycle, at 0 g(i.e. no acceleration applied) conditions. At application of +1g on the sensitive axis, the output is shifts to 70% duty cycle and to 30% duty cycle at application of –1g. This true with both its axis. (Please refer to the data sheet for more details about this product)

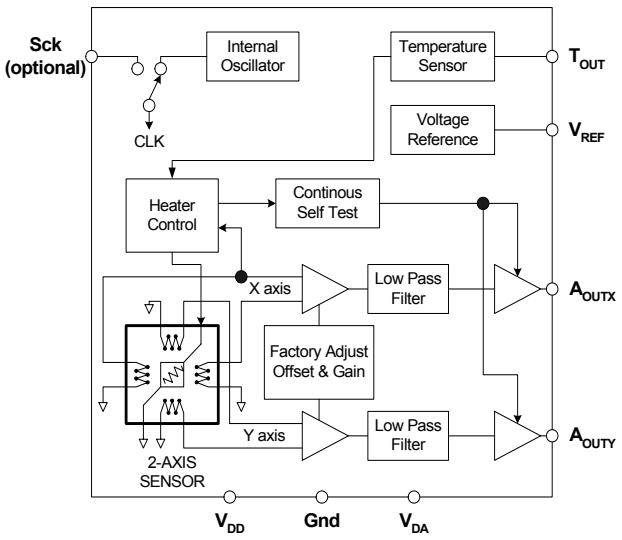

fig2. Functional block diagram of the Memsic Product

Basic Stamp BS2 processor

BASIC Stamps are microcontrollers that are designed for use in a wide array of applications. Many projects that require an embedded system with some level of intelligence can use a BASIC Stamp module as the controller.

Each BASIC Stamp comes with a BASIC Interpreter chip, internal memory (RAM and EEPROM), a 5-volt regulator, a number of general-purpose I/O pins (TTL-level, 0-5 volts), and a set of built-in commands for math and I/O pin operations. BASIC Stamps are capable of running a few thousand instructions per second and are programmed with a customized form of the BASIC programming language, called PBASIC.

Operational Flow chart

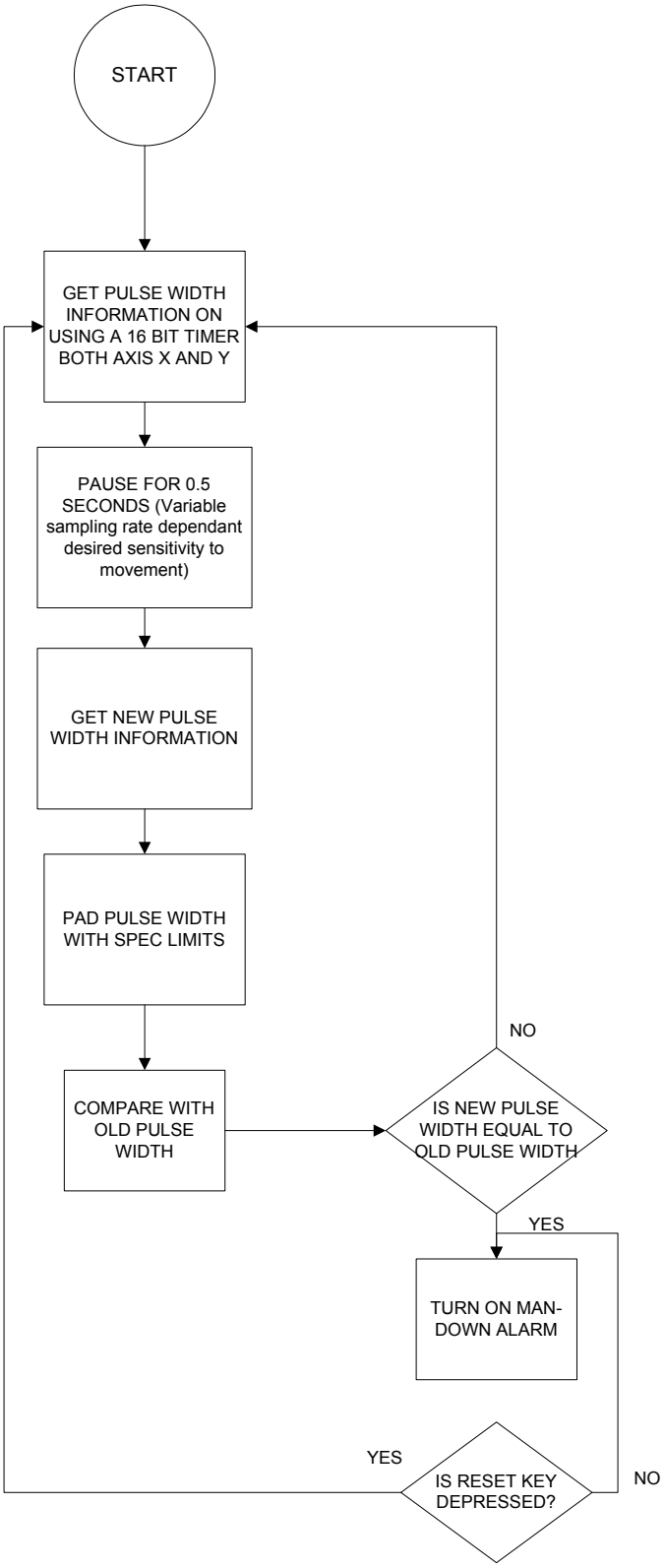

Fig. Flowchart describing general series of steps in a Man down application

SOURCE CODE FOR MAN-DOWN APPLICATION

```
' =================================================================================
'
'   File...... Man-Down Detector.BS2
'   Purpose...
'   Author.... A. Chaturvedi (Memsic), modified by J. Williams (Parallax)
'   E-mail....
'   Started...
'   Updated... 08 JAN 2003
'
'   {$STAMP BS2}
'   {$PBASIC 2.5}
'
' =================================================================================


' -----[ Program Description ]---------------------------------------------
'
' Monitors X and Y inputs from Memsic 2125 and will trigger alarm if no
' movement is detected.  Sample rate, movement thresholds and  "down"
' time threshold are configurable.


' -----[ Revision History ]------------------------------------------------

' 08 JAN 2003   : Converted to PBASIC 2.5 by Jon Williams, Parallax


' -----[ I/O Definitions ]-------------------------------------------------

Xpin            PIN     1                       ' X pulse input
Ypin            PIN     2                       ' Y pulse input
ResetLED        PIN     3                       ' reset LED
AlarmLED        PIN     4                       ' alarm LED


' -----[ Constants ]------------------------------------------------------

SampleDelay     CON     500                     ' 0.5 seconds
DownThreshold   CON     5                       ' 5 x SampleDelay x 2

XLimit          CON     5                       ' x movement minimum
YLimit          CON     5                       ' y movement minimum


' -----[ Variables ]------------------------------------------------------

xTimer1         VAR     Word                    ' first X sample
yTimer1         VAR     Word                    ' first Y sample
xTimer2         VAR     Word                    ' second X sample
yTimer2         VAR     Word                    ' second Y sample
xMove           VAR     Word                    ' x axis movement
yMove           VAR     Word                    ' y axis movement

dnTimer         VAR     Word                    ' "down" timer
idx             VAR     Nib                     ' loop counter


' -----[ EEPROM Data ]----------------------------------------------------


' -----[ Initialization ]-------------------------------------------------

Initialize:
  LOW AlarmLED                                  ' alarm off
  dnTimer = 0                                   ' clear "down" timer

  FOR idx = 1 to 3                              ' show reset
    HIGH ResetLED                               ' - blink green LED 3x
    PAUSE 500
    LOW ResetLED
    PAUSE 500
  NEXT


' -----[ Program Code ]---------------------------------------------------

Main:
```

```
  DO
    GOSUB Get_Data                              ' read inputs
    GOSUB Check_Data                            ' check for movement
  LOOP

  END


' -----[ Subroutines ]------------------------------------------------

' Sample and filter inputs

Get_Data:
  PULSIN Xpin, 1, xTimer1                       ' take first reading
  PULSIN Ypin, 1, yTimer1
  xTimer1 = xTimer1 / 10                        ' filter for noise & temp
  yTimer1 = yTimer1 / 10
  PAUSE SampleDelay
  PULSIN Xpin, 1, xTimer2                       ' take second reading
  PULSIN Ypin, 1, yTimer2
  xTimer2 = xTimer2 / 10                        ' filter for noise & temp
  yTimer2 = yTimer2 / 10
  PAUSE SampleDelay
  RETURN


' Check last sample
' -- update "down" counter if no movement
' -- trigger alarm if threshold exceeded

Check_Data:
  xMove = ABS (xTimer1 - xTimer2)               ' get movement
  yMove = ABS (yTimer1 - yTimer2)

  IF (xMove < XLimit) AND (yMove < YLimit) THEN ' if both...
    dnTimer = dnTimer + 1                       '   update "down" timer
    IF (dnTimer > DownThreshold) THEN Man_Down
  ELSE
    dnTimer = 0                                 ' clear "down" timer
  ENDIF
  RETURN


' Blink Alarm LED
' -- can also be modified to send alarm condition via RF link

Man_Down:
  DO
    TOGGLE AlarmLED                             ' blink alarm LED
    PAUSE 500
  LOOP                                          ' loop until reset
```

Contributions

Jon Williams, Parallax
Rich Allred, Parallax