

MS16 & MS26 Test Circuit & PC Interface

G.F Smith

EME Department, QEQM Hospital, St Peters Road, Margate, Kent

December 2005

Version 1.1

Contents

Introduction.....	2
MS16 & MS26 Test Circuit & PC Interface.....	2
Interface Box - Circuit Diagram.....	3
Circuit Description.....	3
Verification of Test Software.....	4
Timing.....	4
Current Measurement.....	5
Appendix A. Programming Example using MS QuickBasic.....	6
Programming Procedure.....	6
Simple BASIC Listing showing how to obtain a single 12 bit sample.....	6
Appendix B. Complete MS16 & MS26 Test Program Listing.....	7
Appendix C. Screenshot.....	10
Future Improvements.....	11

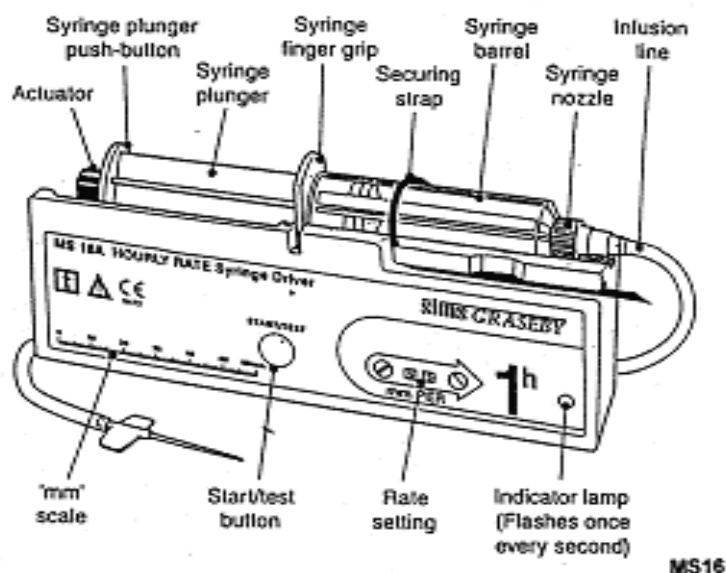
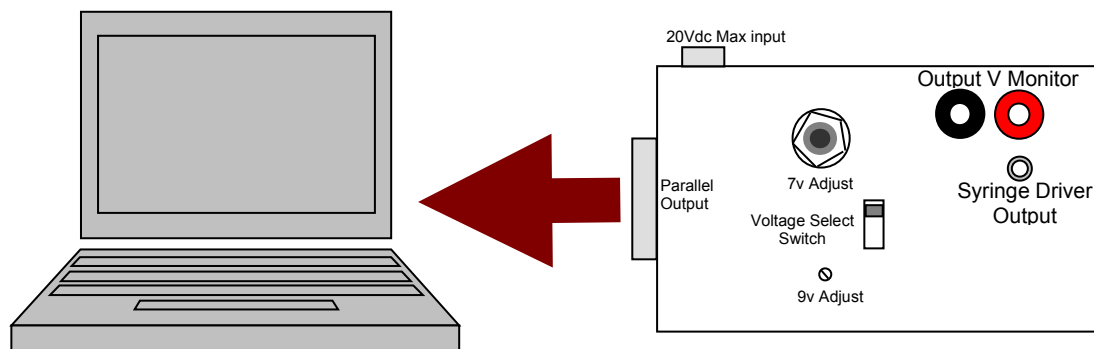
Introduction

This document describes a test box that will reduce the time and labour required to calibrate a Graseby MS16 or MS26 Syringe driver. The test box, in conjunction with some specially developed software, enables the Rate Tests to be conducted with minimal user intervention, and removes the frustration of manually timing a number of motor pulses. The box has a selectable 9V and 7V output. The 9V output can be adjusted by means of a trimming tool through a hole in the top of the box. This is to enable periodic adjustment of the 9V output. The 7V supply is adjustable via a potentiometer mounted on the test box, from less than 5V to 8V. This enables the 7V thrust test and the Low Battery test to be conducted without the use of an additional power supply.

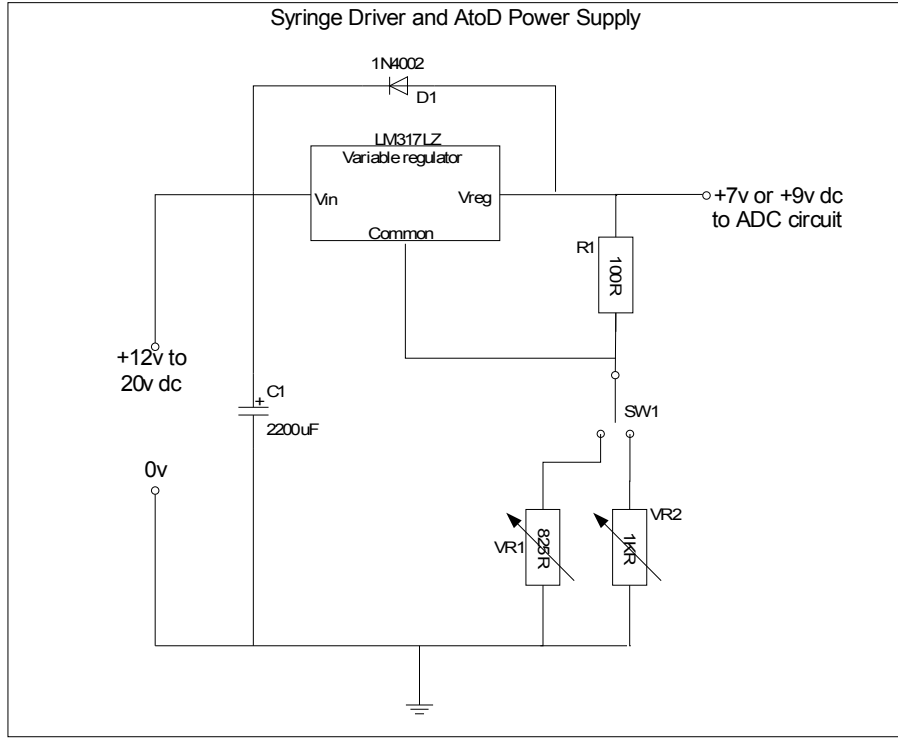
The test box uses an ADC converter to monitor the supply current to the Syringe driver. The Syringe driver is powered by the test box via a dummy battery, connected to the Syringe driver output of the test box.

Using this test box and software saves a considerable amount of time, as the user can get on with something else without the fear of missing a motor pulse.

MS16 & MS26 Test Circuit & PC Interface



Interface Box - Circuit Diagram

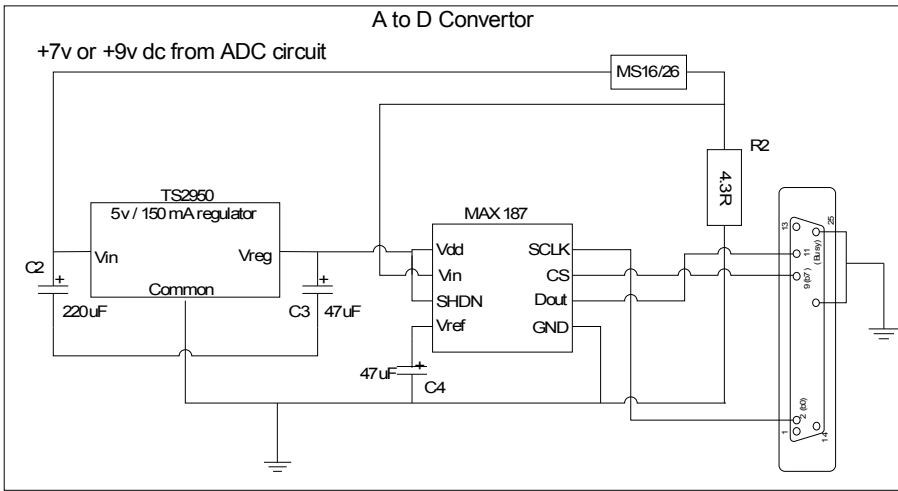


Circuit Description

A 12-20V dc supply is required for the circuit input. The LM317 variable regulator provides a +9V or +7V supply to the Syringe driver. This is to enable the Occlusion test to be conducted at both voltages. VR1 and VR2 can be adjusted to give 7.0V and 9.0V respectively. The voltage supply is selected by SW1.

The LM317 is an adjustable positive voltage regulator, which has built in current limiting. The output voltage is set by the values of R1 and VR1 or VR2. Diode D1 is to protect the regulator from possible reverse voltage in case the output voltage doesn't fall faster than the input voltage.

The TS2950 5V regulator provides the supply for the ADC converter and is derived from the 7V or 9Vdc output of the LM317 regulator.



The mean current measured through the syringe driver was found to be approximately 90mA during a motor pulse. Hence the resistance of the driver can be considered to be:

$$R = 9v/90mA = 100ohms.$$

When the Syringe driver is not operating, its resistance can be considered to be high. In between motor pulses, the voltage across R2 (4.3ohms) will be almost 0v. As soon as the driver operates, it's resistance will drop, causing a voltage to develop across R2.

If the Supply voltage is 9V, the voltage across R2 will be:

$$V = (4.3 / (100+4.3)) * 9 = 371mV$$

Therefore for each operation of the Syringe driver, a voltage in the region of 371mV will develop across R2. This will appear at pin 2 (Vin) of the MAX187 ADC. The ADC is a 12 bit Serial ADC with a built in reference voltage of 4.096V. 12 bits equates to 4095 digital levels. Therefore the resolution of the device is 4.096/4095 = 1.0mV / bit. This means that the ADC is easily capable of measuring input voltages in the region of 381mV.

Only four connections are required to the parallel port on the printer. These are:

MAX187 Pin	Data Direction	Parallel Port Pin	Description
SCLK	←	b0 (Pin 2)	Serial clock input. Clocks data out with rates up to 5MHz.
CS	←	b7 (Pin 9)	Active-low chip select initiates conversions on the falling edge. When CS is high, DOUT is high impedance.
Dout	→	BUSY (Pin 11)	Serial data output. Data changes state at SCLK's falling edge.
GND		GND (Pins 18 to 25)	0v or Common

MAX 187 12bit Serial AtoD Converter Pinout:

1	VDD	Supply voltage, +5V \pm 5%
2	AIN	Sampling analogue input, 0V to VREF range
3	SHDN	Three-level shutdown input. Pulling SHDN low shuts the MAX187/MAX189 down to 10 μ A (max) supply current. Both MAX187 and MAX189 are fully operational with either SHDN high or floating. For the MAX187, pulling SHDN high enables the internal reference, and letting SHDN float disables the internal reference and allows for the use of an external reference.
4	REF	Reference voltage—sets analogue voltage range and functions as a 4.096V output for the MAX187 with enabled internal reference. Bypass with 4.7 μ F if internal reference is used, and with 0.1 μ F if an external reference is applied.
5	GND	Antilog and digital ground
6	DOUT	DOUT Serial data output. Data changes state at SCLK's falling edge.
7	$\overline{\text{CS}}$	Active-low chip select initiates conversions on the falling edge. When CS is high, DOUT is high impedance.
8	SCLK	Serial clock input. Clocks data out with rates up to 5MHz.

Verification of Test Software**Timing****MS16 Syringe Driver**

Set Rate (mm/h)	Computer Displayed time for 5 motor pulses (S)	Manually Measured time for 5 motor pulses (S)	Difference between computer measured and manually measured time (S)
66	32.24	32.11	+0.13
66	32.24	32.49	-0.25
66	31.52	31.93	-0.41
66	31.68	31.85	-0.17
66	31.75	31.71	+0.04
Mean	31.89	32.08	-0.19
	Computer Displayed time for 10 motor pulses (S)	Manually Measured time for 10 motor pulses (S)	Difference between computer measured and manually measured time (S)
99	42.83	42.63	+0.20
99	42.24	42.55	-0.31
99	42.40	42.70	-0.30
99	42.28	42.43	-0.15
99	42.79	42.87	-0.08
Mean	42.51	42.64	-0.13

Set rate to 55mm/h and measure time distance moved over 1 hour

Manually Measured Time when computer alarm sounds	
---	--

MS26 Syringe Driver

Set Rate (mm/h)	Computer Displayed time for 1 motor pulse (S)	Manually Measured time for 1 motor pulse (S)	Difference between manually measured and computer measured time (S)
66	144.5	144.6	-0.1
99	100.6	100.6	0.0

The results shown above indicate that the computer-measured timing is comparable to the timing measured by hand, and shows a maximum difference of 0.2S between the two types of timing measurement.

Current Measurement

Since the test box is only looking for changes in syringe driver current, calibration of the devices current measuring ability is not important. As mentioned previously, the MAX187 ADC is a 12 bit ADC with a built in reference voltage of 4.096V. 12 bits equates to 4096 digital levels. Therefore the resolution of the device is $4.096/4096 = 1\text{mV}$.

To calibrate the device so that the software will display the actual current flowing through the Syringe Driver:

Using a calibrated DMM: **Measure the resistance of a 100ohm resistor (R_{test})**
The Output Voltage should be set to 9.00V

Measured : 99.3ohm

Applied Current (I) = $9.00 / R_{\text{test}} = 9.00 / 99.3 = 0.0906\text{A}$ or 90.6mA .

The software shown in Appendix A stores the output of the ADC in a variable called 'total'. This value is the voltage developed across resistor R2 during a motor pulse.

A calibration factor needs to be calculated in order to convert the value held in 'total' to the actual current measured: Place the 99.3ohm resistor across the voltage monitor terminals of the test box and record the ADC output 'total' as shown below:

'total' = 454, Applied Current = 90.6mA

Cal Factor = $454/90.6 = 4.415$

To get 'total' to read in mA simply divide it by 4.415

Appendix A. Programming Example using MS QuickBasic

Programming Procedure

Initialise AtoD

Set CS HIGH, SCLK LOW

Start a Conversion

Set CS LOW, SCLK LOW

Get 12 bits of Data

Set CS LOW, SCLK HIGH

Set SCLK LOW

Read Dout

Repeat above 11 more times

Simple BASIC Listing showing how to obtain a single 12 bit sample

DIM PortBits(8) AS INTEGER

total = 0

' INTERFACE:

Parallel Port	Data direction	AtoD Interface	Memory Address
Data (b0)	>>>>>>	SCLK	888 (378h) Data register
Data (b7)	>>>>>>	CS (NEG going)	888 (378h) Data register
BUSY (b8)	<<<<<<	Dout	889 (379h) Status Register

BitYouWant% = 8 ' Get serial data from b7 >> (b7 is the eighth bit)

OUT 888, 128 'CS High SCLK Low – Initialise AtoD

OUT 888, 0 'CS Low SCLK Low - Start a conversion on falling edge of CS

' Get the 12 bits of data

FOR n = 0 TO 11

' CLOCK NEXT BIT OF DATA

OUT 888, 1 'CS Low SCLK High – Set SCLK high

OUT 888, 0 'CS Low SCLK Low - and then Low to get next bit of data

' GET NEXT BIT OF DATA

' This will be read in as a byte. But we only want the status of b7 (BUSY line on parallel port)

PortNum% = INP(889)

' Convert the value to Binary

FOR i = 1 TO 8

PortBits%(i) = PortNum% MOD 2

PortNum% = FIX(PortNum% / 2)

NEXT i

' Get the status of b7 from the byte

BitStatus% = PortBits%(BitYouWant%)

' Invert the bit

IF BitStatus% = 0 THEN BitStatus% = 1 ELSE BitStatus% = 0

IF BitStatus% = 1 THEN total = total + 2 ^ (11 - n)

NEXT n

PRINT "AtoD Output="; total

By expanding the listing shown above, it is possible to measure each motor pulse and therefore determine the time between pulses.

Appendix B. Complete MS16 & MS26 Test Program Listing

```
' *****
' MS16 & MS26 SYRINGE DRIVER TEST PROGRAM
' VERSION 1.0
'
DIM PortBits(8) AS INTEGER
elapsed = 0
CLS
SCREEN 12: ' 640x480 - 16 colours
COLOR 15
LINE (170, 5)-(438, 38), 4, BF
LINE (173, 8)-(435, 35), 14, BF
LOCATE 2, 24: PRINT " MS16 & MS26 Test Program V1.0"

readings = 1
threshold = 5
peak = 0
average = 0
total = 0
pulse.count = -1
avg.time = 0
exit.flag = 0
distance.timer = 0
distance.flag = 0

' INTERFACE:
'   Parallel Port  Data direction  AtoD Interface  Address
'   -----
'   b0             >>>           SCLK             888 (378h)
'   b7             >>>           CS (NEG going) 888 (378h)
'   BUSY           <<<           Dout             889 (379h)

BitYouWant% = 8
elapsed = TIMER
distance.timer = TIMER
cal = TIMER
x = 1
OPEN "test.txt" FOR OUTPUT AS #1

' Draw rate test box
LINE (300, 100)-(620, 250), 14, BF
LINE (305, 105)-(615, 245), 0, BF
LINE (300, 150)-(620, 150), 14, B
LINE (300, 185)-(620, 185), 14, B
LINE (300, 210)-(620, 210), 14, B

' Draw threshold box
LINE (1, 345)-(520, 400), 13, BF
LINE (5, 349)-(516, 396), 0, BF
LOCATE 23, 35: PRINT "Press - to decrease threshold"
LOCATE 24, 35: PRINT "Press + to decrease threshold"

' Draw Time Elapsed Box
LINE (165, 40)-(445, 70), 4, BF
LINE (170, 45)-(440, 65), 0, BF

LOCATE 13, 2: PRINT "Time for 1 Pulse = @@@@ S"
LOCATE 15, 2: PRINT "Time for 5 Pulses = @@@@ S"
LOCATE 17, 2: PRINT "Time for 10 Pulses= @@@@ S"
LINE (1, 80)-(293, 300), 10, BF
```

```

LINE (4, 84)-(289, 296), 0, BF
LOCATE 7, 2: PRINT "DRIVER MOTOR PULSE RATE TEST"
LOCATE 28, 1: PRINT "PRESS 'Z' TO TERMINATE PROGRAM"

WHILE exit.flag = 0
    avg.time = TIMER
    cumulative = 0
    WHILE TIMER - avg.time < .5
        a$ = INKEY$
        IF a$ = "+" OR a$ = "=" THEN threshold = threshold + 5
        IF (a$ = "-" OR a$ = "_") AND threshold > 5 THEN threshold = threshold - 5
        IF a$ = "z" OR a$ = "Z" THEN exit.flag = 1
        IF a$ = "r" OR a$ = "R" THEN
            BEEP
            distance.timer = TIMER
            distance.timer = TIMER - 3500
            LINE (305, 212)-(615, 244), 0, BF
        END IF
        IF a$ = "T" OR a$ = "t" THEN
            distance.flag = 1
            distance.timer = TIMER
        END IF
        ' Set CS low
        OUT 888, 128 'CS High SCLK Low -
        OUT 888, 0 'CS Low SCLK Low - Start a conversion

        ' Get bit of data
        total = 0
        FOR n = 0 TO 11
            OUT 888, 1 'CS Low SCLK High
            OUT 888, 0 'CS Low SCLK Low

            PortNum% = INP(889)

            FOR i = 1 TO 8
                PortBits%(i) = PortNum% MOD 2
                PortNum% = FIX(PortNum% / 2)
            NEXT i

            BitStatus% = PortBits%(BitYouWant%)
            IF BitStatus% = 0 THEN BitStatus% = 1 ELSE BitStatus% = 0
            IF BitStatus% = 1 THEN total = total + 2 ^ (11 - n)
        NEXT n
        readings = readings + 1
        cumulative = cumulative + total
        IF total > peak THEN peak = total
        REM Resistance value adjusted to correspond with PSU indicated current
        LOCATE 9, 2: PRINT "Measured Driver Current="; INT(total / 1); "mA"; " "
        LOCATE 23, 2: PRINT "Threshold Current ="; threshold; " mA "
        REM LOCATE 7, 5: PRINT "Peak Current="; peak; " mA "
        PRINT #1, peak
    WEND

    ' Calculate Average of readings
    total = cumulative / readings
    readings = 1

    ' Calculate Elapsed time in Hrs, Mins, Secs
    secs = INT(TIMER - cal)
    hours = INT(secs / 3600)
    mins = (INT(secs / 60)) - (hours * 60)
    secs = INT(secs MOD 60)
    LOCATE 4, 23: PRINT "Time Elapsed: H"; hours; " M"; mins; " S"; secs

```



```

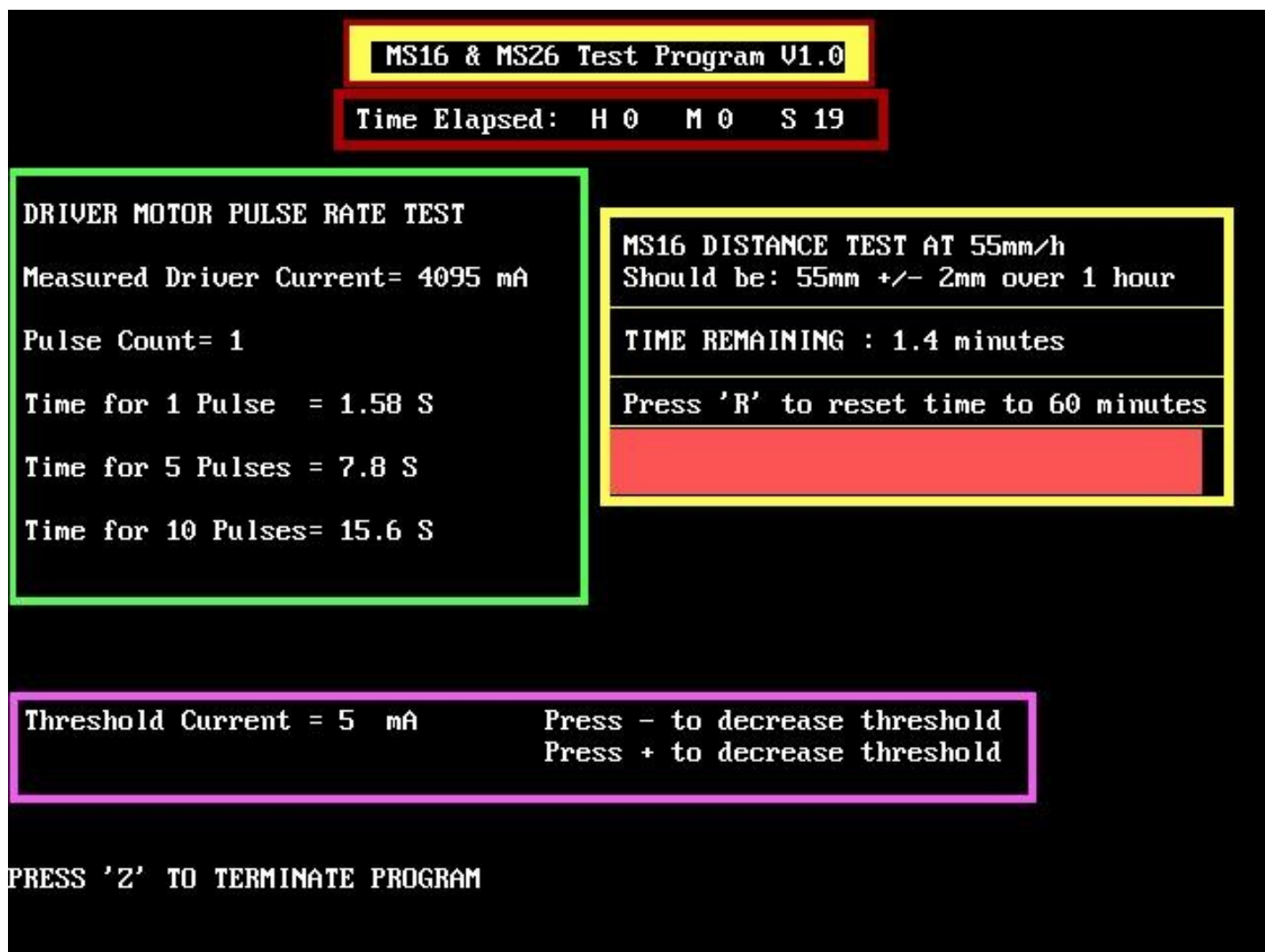
' Has a motor pulse been detected?
IF total > threshold THEN
    capture = TIMER - elapsed
    pulse.count = pulse.count + 1
    IF pulse.count = 0 THEN elapsed = TIMER
    LOCATE 11, 2: PRINT "Pulse Count="; pulse.count
    IF pulse.count = 1 THEN LOCATE 13, 2: PRINT "Time for 1 Pulse ="; INT((capture) * 100) / 100; "S  "
    IF pulse.count = 5 THEN LOCATE 15, 2: PRINT "Time for 5 Pulses ="; INT((capture) * 100) / 100; "S  "
    IF pulse.count = 10 THEN LOCATE 17, 2: PRINT "Time for 10 Pulses="; INT((capture) * 100) / 100; "S  "
    IF pulse.count = 10 THEN
        pulse.count = -1
        elapsed = TIMER
    END IF
    ' Wait for Motor Pulse to finish
    ' 1 SECOND DELAY
    p = TIMER
    WHILE TIMER - p < 1
    WEND
    peak = 0
END IF

' Display 55mm distance timer
LOCATE 8, 40: PRINT "MS16 DISTANCE TEST AT 55mm/h"
LOCATE 9, 40: PRINT "Should be: 55mm +/- 2mm over 1 hour"
IF distance.flag = 1 THEN
    LOCATE 13, 40: PRINT "Press 'R' to reset time to 60 minutes"
    LOCATE 11, 40: PRINT "TIME REMAINING :"; INT(10 * (60 - (TIMER - distance.timer) / 60)) / 10; "minutes  ";
    LOCATE 15, 40: PRINT "TEST IN PROGRESS      "
ELSE
    LOCATE 15, 40: PRINT "PRESS T To START 55mm TEST"

END IF
IF INT(10 * (60 - (TIMER - distance.timer) / 60)) / 10 < 0 THEN
    distance.flag = 0
    LINE (305, 212)-(615, 244), 0, BF
    LOCATE 15, 40: PRINT "REMOVE POWER TO DRIVER !!!!!"
    LOCATE 13, 40: PRINT "PRESS 'ENTER' TO CONTINUE      "
    WHILE INKEY$ = ""
    BEEP
    WEND
    LOCATE 15, 40: PRINT "                "
    LOCATE 13, 40: PRINT "                "
    xpos = 310
    distance.timer = TIMER
ELSE xpos = 305 + ((TIMER - distance.timer) * .085)
    LINE (305, 212)-(xpos, 244), 12, BF
END IF
WEND
CLOSE #1

```

Appendix C. Screenshot



Future Improvements

Measure time between LED flashes.

Incorporate DPM to indicate voltage.

Strain gauge to measure occlusion levels.