**Military Technical College**
**Kobry El-kobbah,**
**Cairo, Egypt**

**5ᵗʰ International Conference**
**on Electrical Engineering**
**ICEENG 2006**

# FAULT DETECTION OF THE DATA DISPLAYING SYSTEM IN MARINE CENTERS

Emad* M. Hamad, Mohamed** A. Abd El-Rahman, Mohamed*** E. Nasr
and Usama**** B. Sabbah

**Abstract**

This paper presents a comprehensive analysis of the vessel traffic management system (VTMS). It is shown that displaying system includes the signal state board, with its four driving boards, and station PC. Each driving board feeds one row. We show that, as an example, Row 2 contains six $(7 \times 5 \text{dot matrix})$ modules and each dot has its own flipping coil. In addition, each driving board has 58 Darlington transistors. The address and data lines of the processor Z80180 are used to control the flipping circuit through decoders, addressable latches, buffers and inverters. Interrupt circuit generates signal $\overline{INT}1$ which is used by the processor to switch off all transistors after each dot flipping process. We have built an assembly EPROM program to deal with the two assembly subroutines, activated by $\overline{INT}1$ and $\overline{INT}2$ interrupt signals, to detect faults associated with the flipping circuit transistors. We performed a new visual basic program(paper under publication) in the station PC to handle the detected faults by EPROM program.This work is illustrated practically by tracing measured signals at different points for the example of short and open transistor.

**Introduction**

The subject of the paper is to analyze the displaying data and detect faults of the flipping circuit components in the system applied in Suez Canal. The main objective of the VTMS resides on minimizing accidents to ensure safe passing of the vessels. In this paper, suggested procedures are implemented to determine the faulty transistor or the digital electronic components connected with it.

**Interrupt and Flipping Circuits Analysis**

In the flipping circuit shown in Fig.1, the current passes through T46-D1-coil-T3-R22 when T46, T3 are switched ON and T32 and T17 are switched OFF. The current passes in the opposite direction when T32 and T17 are switched ON and T46, T3 are switched OFF. The

*        Post graduate student, Suez Canal Authority, Egypt.
**       Dr., Air Defense College, Alexandria, Egypt.
***      Professor, Electronics and Communication Department, Tanta University, Egypt.
****     Dr., Director of Planning and Researches Department, Suez Canal Authority, Egypt.

dot will turn to the yellow face in the first case and to black face in the second one. Fig.3 depicts that the current passing through the inductor, rising as a ramp, causes a voltage drop across the resistor R22 of value $1\Omega$. This voltage is normally 4.8V with 380μsec duration.
The interrupt circuit shown in Fig.2 includes two identical wave shaping circuits which are simply RC circuit, two comparators COMP1, COMP2 and two XOR gates. $R_3$-$C_{10}$ and $R_5$-$C_{12}$ represent these two shaping circuits respectively.

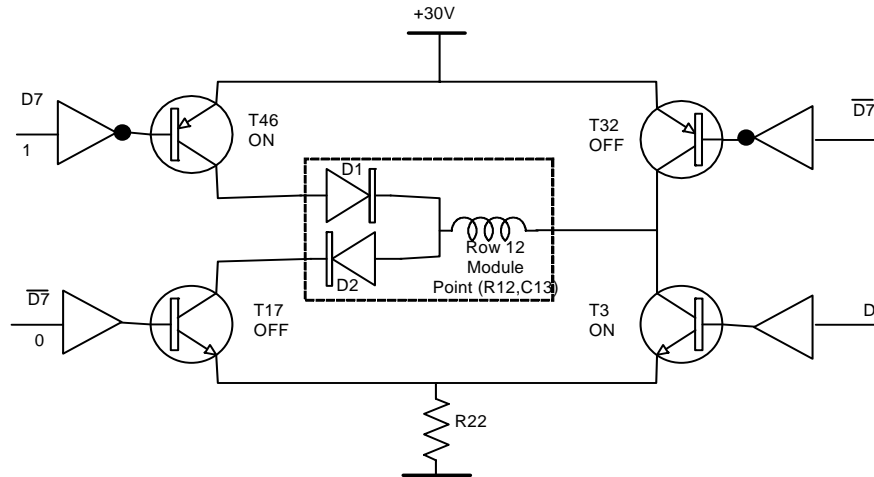

Fig.1. A simplified circuit for flipping single point of (DotMatrix/7 Segments) module

The time constant of these circuits is:
$\tau = R_3 C_{10} = R_5 C_{12} = 100 k\Omega \times 2.2 nF = 220 \mu \sec$.
Therefore the capacitor $C_{10}$ will charge according to the following the equation:
$V_o = V_{inp}\left(1 - e^{-t/\tau}\right)$
Where $V_{inp} = 4.8V$. The output voltage of the shaping circuit across the capacitors $C_{10}$ and $C_{12}$ after $t = 380 \mu \sec$ becomes 3.9V. These capacitors will decay to approximately zero voltage value after $t = 5\tau$ according to $e^{-t/\tau}$ as shown in Fig.4 .This voltage is fed to the inverting terminals of the two comparators COMP1 and COMP2. The reference voltage at the non-inverting terminals of COMP1, COMP2 is 3.4V and 2V respectively. The output waveforms of COMP1 and COMP2 are illustrated in Fig.5 and Fig.6. The output of COMP1 is fed to the pin (9) of the XOR1 and its pin (10) is connected to pin (10) of addressable latch IC21 (74HC259). This allows the processor Z80180 to control the output of the XOR1 which is connected to the $\overline{INT}1$ pin. Similarly, the output of COMP2 is fed to the pin (13) of the XOR2 and its pin (12) is connected to pin (11) of addressable latch IC21 (74HC259). This allows the processor Z80180 to control the output of the XOR2 which is connected to the $\overline{INT2}$ pin.
In the EPROM program of the driving board, the initial settings are as follow:
 i) INT/TRAP control register "ITC" with the address (34H) enables the external interrupts $\overline{INT1}$, $\overline{INT2}$ and disables the external interrupt $\overline{INT0}$. This is done in subroutine (09DEH) which has the following instructions:
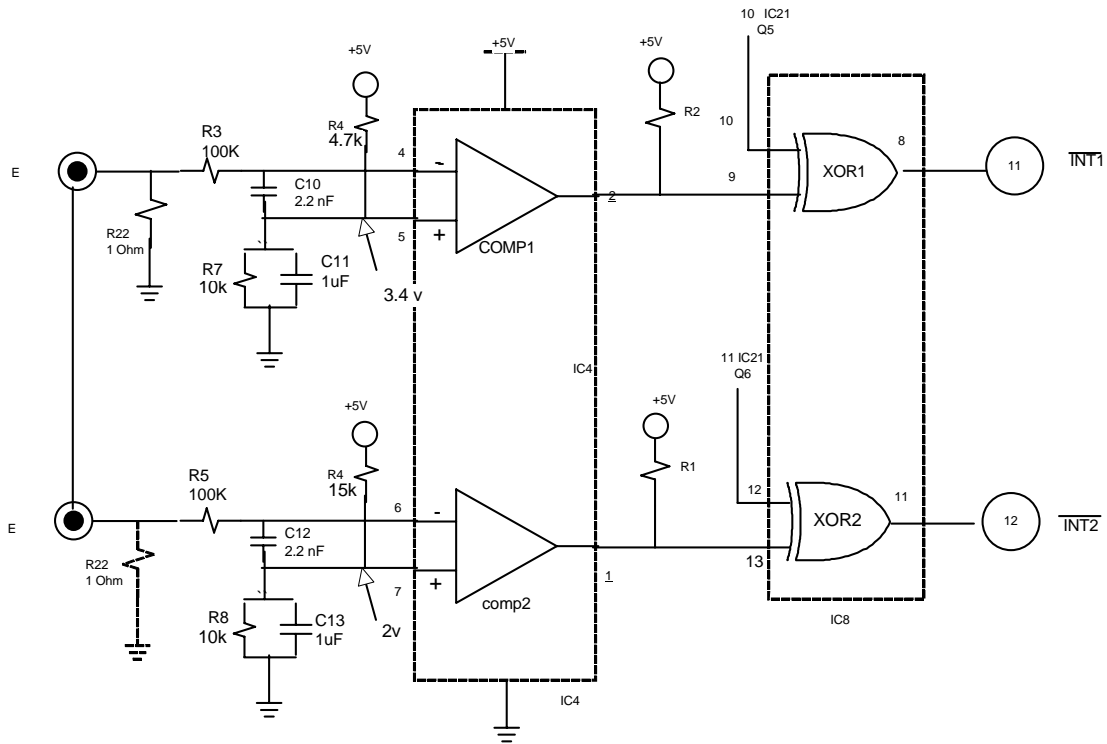    09E8  LD A, (06H)
    09EA  OUT0 (34H), A
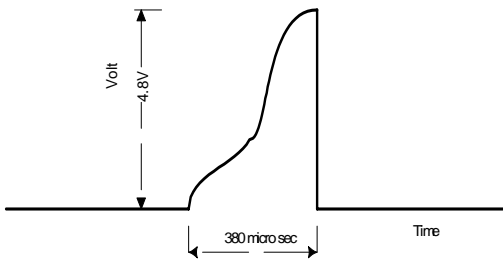
Fig.2. Interrupt Interface circuit



Fig.3. Voltage across R22 " point E and Gnd "
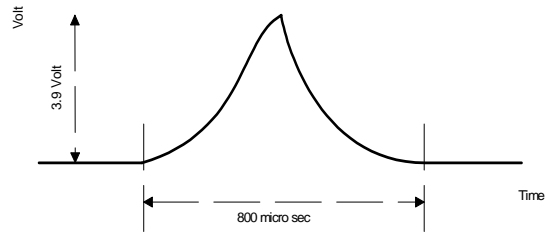


Fig.4. Pins "4" ," 6"  IC4 comparator



Fig.5. Pins "4" ," 2" IC4 comparator
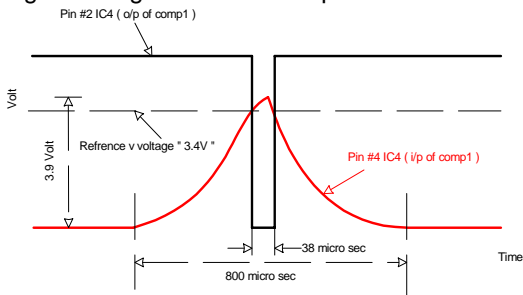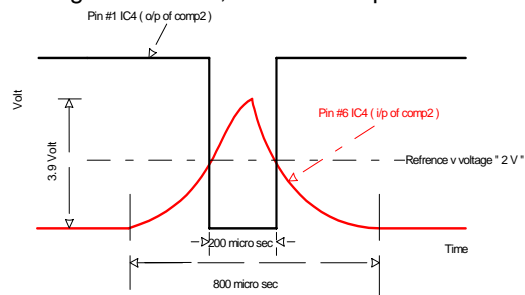


Fig.6. Pins "4" ," 6"  IC4 comparator

| TRAP | UF0 | | | | ITE2 | ITE1 | ITE0 |
|------|-----|---|---|---|------|------|------|
| R/W | R | | | | R/W | R/W | R/W |
| 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 |

Disable external interrupt 0

Enable external interrupt 1

Enable external interrupt 2

ii) Initial adjustment for memory location (9291H) and XOR1 gate:

    09ED      LD A, 00H
    09EF      LD (9291H), A
    09F2      OR    (15H)
    09F4      OUT (0FH), A

These instructions store (00H) in the memory location (9291H) which is used as a flag to the $\overline{INT}1$ subroutine, then OUT (15H) at the address (0FH). This leads to

Data lines

| $D_7$ | $D_6$ | $D_5$ | $D_4$ | $D_3$ | $D_2$ | $D_1$ | $D_0$ |
|------|------|------|------|------|------|------|------|
| 0 | 0 | 0 | 1 | 0 | 1 | 0 | 1 |

Address llines

| $A_7$ | $A_6$ | $A_5$ | $A_4$ | $A_3$ | $A_2$ | $A_1$ | $A_0$ |
|------|------|------|------|------|------|------|------|
| 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |

The address lines A3- A5 are 001 which are connected to the IC11 3-8 decoder (74HC138) causing IC12 and IC14 to be enabled. The address lines A0-A2 are 111, this will select Q7 in IC12-IC14 (74HC259) which nothing connected to it as shown in Fig 7.

The data lines D3- D5 are 010 which are connected to 3-8 decoder (74HC138) to enable IC21. D0- D2 [101] are input to addressable latch IC21 (74HC259) causing D7 (logic 0) to pass through the output Q5. This output (logic 0) is fed to XOR1 pin #10, and XOR1 will work as a buffer connecting the output of COMP1, shown in Figs. 3, to $\overline{INT}1$ pin of the processor.

iii) Initial adjustment for memory location (9292H) and XOR2 gate:

   09F6   LD  A, (00H)
   09F8   LD  (9292H), A
   09FB   OR  (16H)
   09FD   OUT (0FH)

These instructions store (00H) in the memory location (9292H) which is used as a flag to $\overline{INT}2$ subroutine. Then out the byte (16H) at the address (0FH).

Data lines

| D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
|----|----|----|----|----|----|----|----|
| 0 | 0 | 0 | 1 | 0 | 1 | 1 | 0 |

Address lines

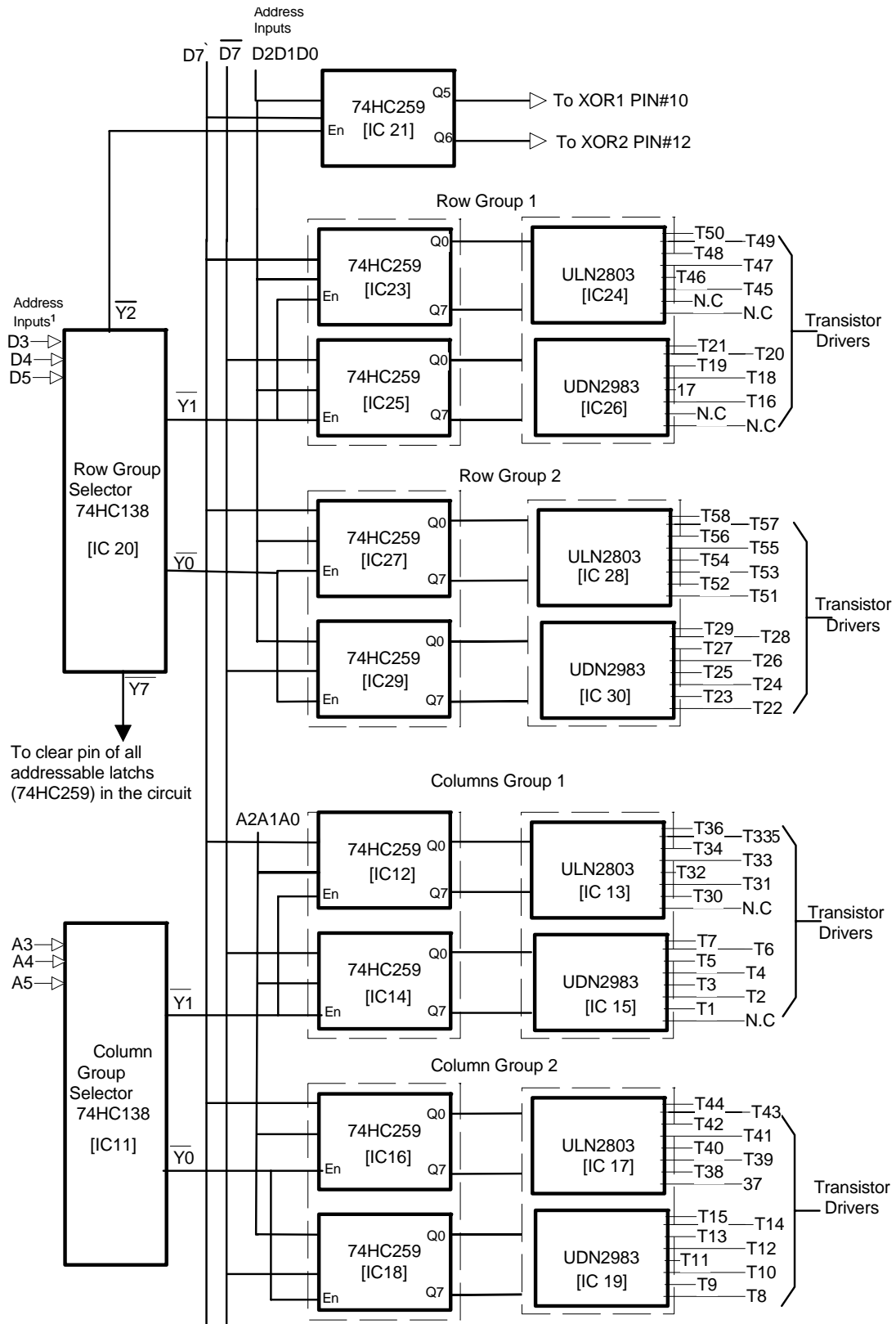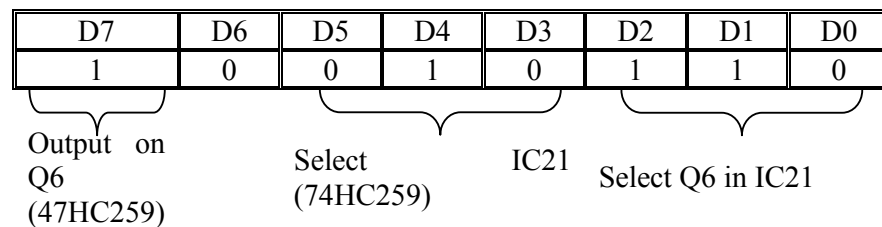| $A_7$ | $A_6$ | $A_5$ | $A_4$ | $A_3$ | $A_2$ | $A_1$ | $A_0$ |
|------|------|------|------|------|------|------|------|
| 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |

Fig.7. Rows and Columns selector

As mentioned before, IC12 & IC14 are enabled, which in conjunction with A0-A2 (111) leads to select nothing as shown in Fig 7..

The data lines D3- D5 are 010 which are connected to 3-8 decoder (74HC138) to enable IC21. D0- D2 [110] are input to addressable latch IC21 (74HC259) causing D7 (logic 0) to pass through the output Q6. This output (logic 0) is fed to XOR2 pin (12), and XOR2 will work as a buffer connecting the output of COMP 2, shown in Fig.3, to $\overline{INT2}$ pin of the processor.
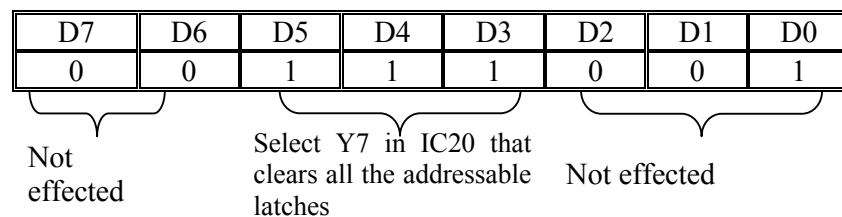
In the proceeding section, we will continue to explain the normal operation for the flipping process of any dot. When the current passing through the coil, the voltage drop across R22 is shaped and compared with the reference values in COMP1 and COMP2 . This leads to the output waveforms in Figs. 5,6 to be appeared at the inputs of XOR1 and XOR2 respectively. We note that the duration of interrupt signal $\overline{INT1}$ (logic 0) is 38 $\mu$sec. Also the duration of interrupt signal $\overline{INT2}$ (logic 0) is 200 $\mu$sec. This shows that the $\overline{INT2}$ occurred first and the processor Z80180 interrupted by $\overline{INT2}$ at location (0BFFH) and begins to execute side A of the flow chart in Fig.8 . It stores logic 1 at bits 1, 4 at memory location (9290H), (80H) at memory location (9292H) to make the program follows side B at the next execution of interrupt 2 subroutine. Then the processor will output (96H) at the address (0FH) causing address lines to select Q7 at IC12 and IC14 which is not connected. The data lines are:

| D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
|----|----|----|----|----|----|----|----|
| 1  | 0  | 0  | 1  | 0  | 1  | 1  | 0  |

Output   on
Q6
(47HC259)

Select        IC21
(74HC259)

Select Q6 in IC21

The input to XOR2 pin (13) is logic 1. Therefore the XOR2 works as an inverter it will invert the low coming from the comp2 to logic 1 which is applied to INT2 pin (12) of the processor Z80180 as shown in Fig.3. After a short time the processor will have low at pin (11) $\overline{INT1}$. Therefore the processor will be interrupted.

The processor starts executing side A of the flow chart related to $\overline{INT1}$ subroutine as shown chart in Fig.9. The first instruction causes the data lines to be as follows

| D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
|----|----|----|----|----|----|----|----|
| 0  | 0  | 1  | 1  | 1  | 0  | 0  | 1  |

Not
effected

Select  Y7  in  IC20  that
clears all the addressable
latches

Not effected

Start

Test if the byte stored in (9292) h is zero

Side B — not zero

Side A — zero

```
0bff 08      ex    af,af'
0c00 d9      exx
0c01 219292  ld    hl,9292h
0c04 7e      ld    a,(hl)
0c05 b7      or    a
0c06 200c    jr    nz,0c14h
0c08 3a9092  ld    a,(9290h)
0c0b f611    or    11h
0c0d 329092  ld    (9290h),a
0c10 3e80    ld    a,80h
0c12 1809    jr    0c1dh
0c14 3a9092  ld    a,(9290h)
0c17 e6ef    and   0efh
0c19 329092  ld    (9290h),a
0c1c af      xor   a
0c1d 77      ld    (hl),a
0c1e f616    or    16h
0c20 d30f    out   (0fh),a
0c22 d9      exx
0c23 08      ex    af,af'
0c24 fb      ei
0c25 c9      ret
```

AND EF with the content of (9290)h , Store the result in (9290)h

In memory location (9290)h,Store "0" in Bit#4

0000  0001

In memory location (9290)h,Store "1" in Bit#0,Bit#4

0001  0001

OR (11)h with the content of (9290)h , Store the result in (9290)h

store (00)h in memory location (9292)h

Next processor inerruption by INT2 ,Execute side A

Next processor inerruption by INT1 ,Execute side B

store (80)h in memory location (9292)h

OUT (16)h on the address (0F)h

Make XOR2 work as a BUFFER

Make XOR2 work as an INVERTER

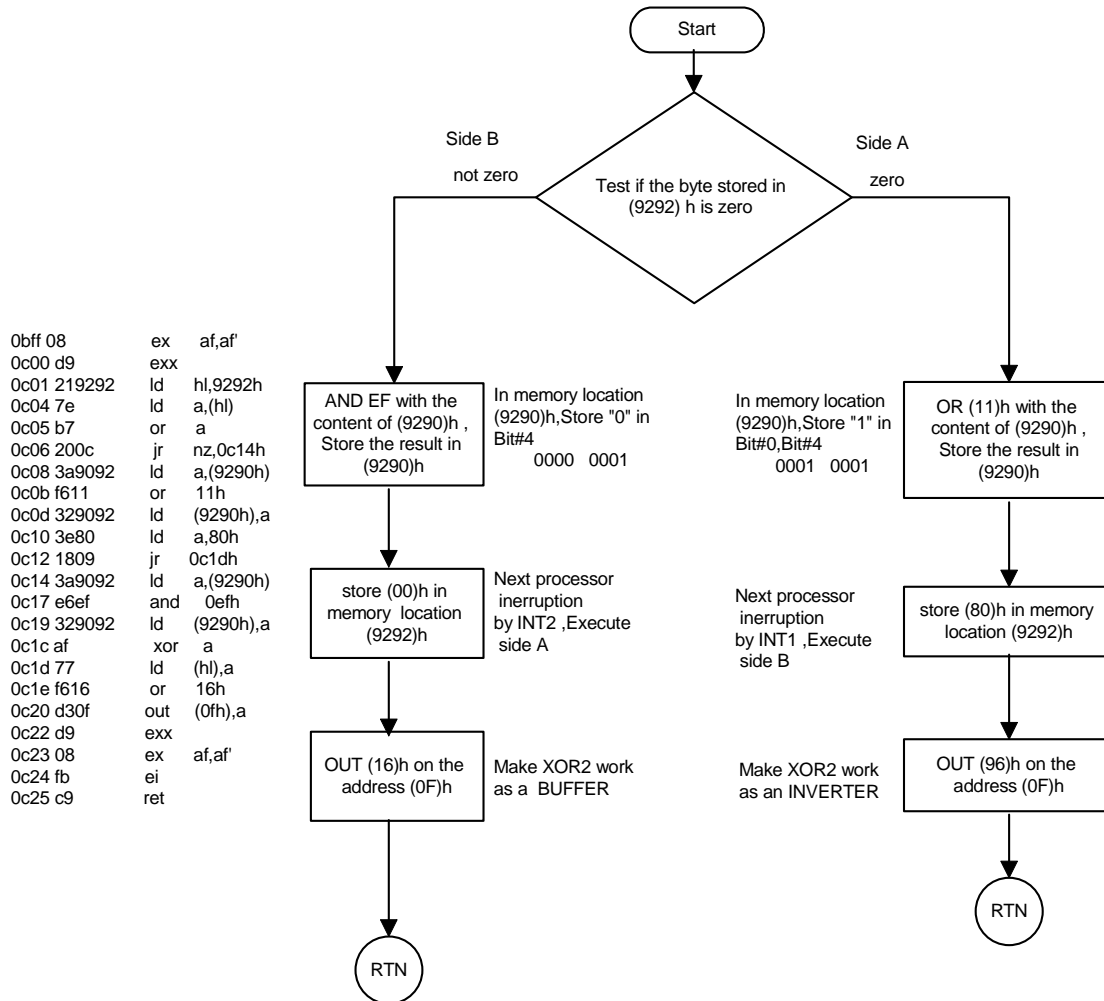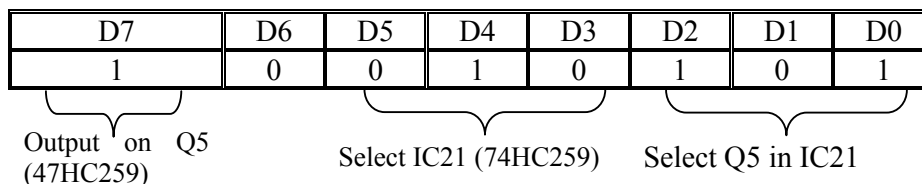OUT (96)h on the address (0F)h

RTN

RTN

Fig.8. Interrupt 2 subroutine flow chart

The processor switches OFF all transistors on the board. It stores logic 1 at bits 2,5 at memory location (9290H), (80H) at memory location (9291H) to make the program follow side B at the next execution of interrupt 1 subroutine. Then the processor will out (95H) at the address (0FH) causing address lines to select Q7 at IC12 and IC14 which is not connected. The data lines are:
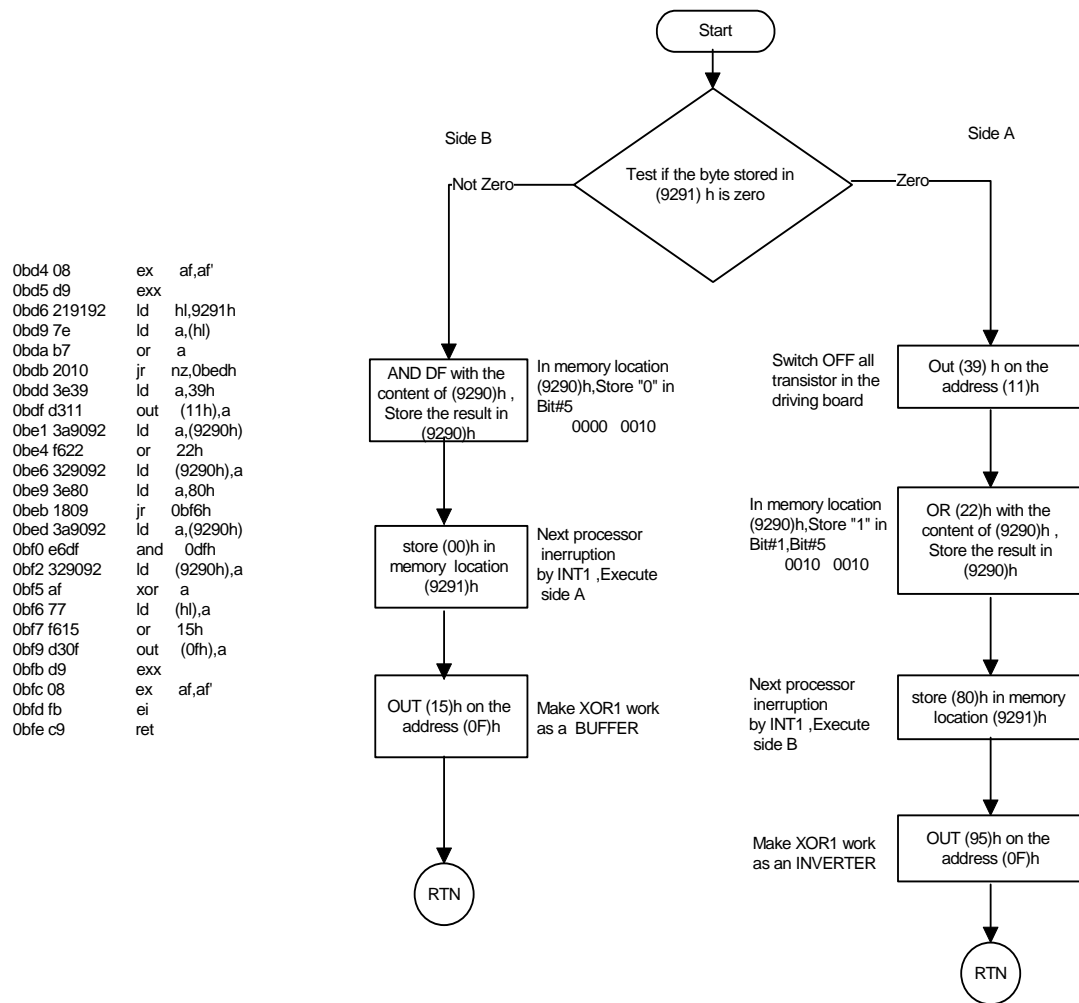
| D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
|----|----|----|----|----|----|----|----|
| 1  | 0  | 0  | 1  | 0  | 1  | 0  | 1  |

Output on Q5 (47HC259)

Select IC21 (74HC259)

Select Q5 in IC21

```
0bd4 08      ex    af,af'
0bd5 d9      exx
0bd6 219192  ld    hl,9291h
0bd9 7e      ld    a,(hl)
0bda b7      or    a
0bdb 2010    jr    nz,0bedh
0bdd 3e39    ld    a,39h
0bdf d311    out   (11h),a
0be1 3a9092  ld    a,(9290h)
0be4 f622    or    22h
0be6 329092  ld    (9290h),a
0be9 3e80    ld    a,80h
0beb 1809    jr    0bf6h
0bed 3a9092  ld    a,(9290h)
0bf0 e6df    and   0dfh
0bf2 329092  ld    (9290h),a
0bf5 af      xor   a
0bf6 77      ld    (hl),a
0bf7 f615    or    15h
0bf9 d30f    out   (0fh),a
0bfb d9      exx
0bfc 08      ex    af,af'
0bfd fb      ei
0bfe c9      ret
```
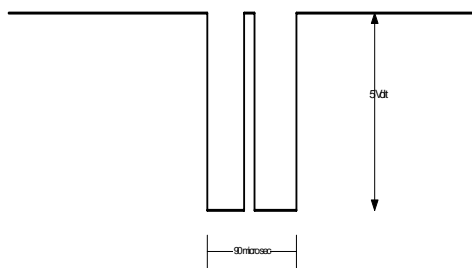
Fig.9. Interrupt 1 subroutine flow chart
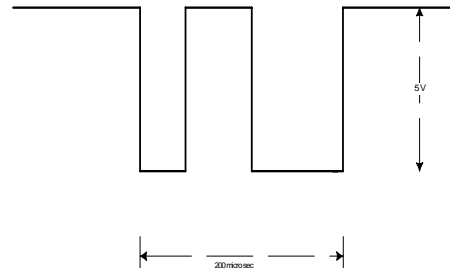
Fig.11. Pin "8" IC8 to processor INT1     Fig.10. Pin "11" IC8 to processor INT2
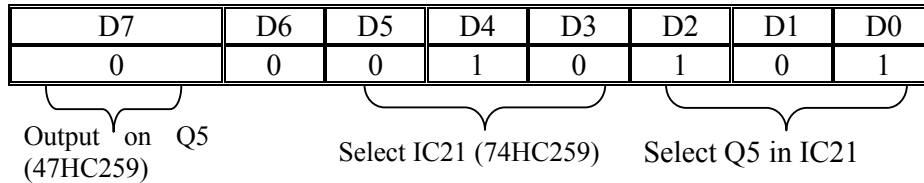
The input to XOR1 pin (10) is logic 1. Therefore the XOR1 works as an inverter it will invert the low coming from the comp1 to logic 1 which is applied to INT1 pin (11) of the processor Z80180 as shown in Fig.3.

As shown in fig.5. the output of the comp1 is logic 0 for about 38μSec before returning to logic 1. Therefore after the low period (38μSec) the output of XOR1, which acts as an

inverter, will be low again as shown in Fig.10. Thus the processor will interrupted again by $\overline{INT}1$ executing side B of the $\overline{INT}1$ subroutine.
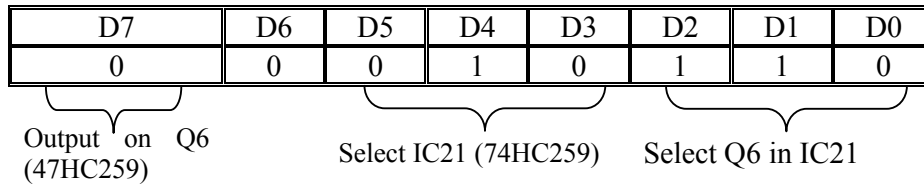
It stores logic 0 at bits 5 at memory location (9290H), (00H) at memory location (9291H) to make the program follows side A at the next execution of interrupt 1 subroutine. Then the processor will out (15H) at the address (0FH) causing address lines to select Q7 at IC12 and IC14 which nothing connected to it as shown in Fig.7. The data lines are:

| D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
|----|----|----|----|----|----|----|----|
| 0  | 0  | 0  | 1  | 0  | 1  | 0  | 1  |

Output on Q5 (47HC259)    Select IC21 (74HC259)    Select Q5 in IC21

The input to XOR1 pin (10) is logic 0. Therefore the XOR1 works as a buffer, it will apply the high coming from the comp1 to INT1 pin (11) of the processor Z80180.

As shown in figure (6) the output of the comp2 is logic 0 for about 200μSec before returning to logic 1. Therefore after the low period (200μSec) the output of XOR2, which acts as an inverter, will be low again as shown in Fig.11. Thus the processor will interrupted again by $\overline{INT}2$ executing side B of the $\overline{INT}2$ subroutine.

It stores logic 0 at bits 4 at memory location (9290H), (00H) at memory location (9292H) to make the program follows side A at the next execution of interrupt 1 subroutine. Then the processor will out (16H) at the address (0FH) causing address lines to select Q7 at IC12 and IC14 which nothing connected to it as shown in Fig.7. The data lines are:

| D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
|----|----|----|----|----|----|----|----|
| 0  | 0  | 0  | 1  | 0  | 1  | 1  | 0  |

Output on Q6 (47HC259)    Select IC21 (74HC259)    Select Q6 in IC21

The input to XOR2 pin (13) is logic 0. Therefore the XOR2 works as a buffer it will apply the high coming from the comp2 to INT2 pin (12) of the processor Z80180.

The output of the XOR1and XOR2 will depend on the two sides A, B of the two flow charts shown in Fig.8 and Fig.9. Table I describes the XOR1 output dependency.

Table I:

| Comparator output | Control input at pin (10) of XOR1 | Output of XOR1 gate |
|-------------------|-----------------------------------|---------------------|
| high | Logic 0 (from initial setting) (so it works as a buffer) | High and no interrupt |
| low | Logic 0 and it woks as a buffer | Low and the processor interrupted and start executing side A |
| low | Logic 1 at the of side A from the subroutine so it works as an inverter | High and no interruption |
| high | Logic 1 and it works as an inverter | Low and the processor interrupted and start executing side B |
| high | Logic 0 at the end of side B and it works as a buffer | High and the processor is not interrupted |

Table II describes the contents of (9290H) after each interrupt.
Table II:

|  | Contents of (9290H) in Binary | Contents of (9290H) in Hex |
|---|---|---|
| After INT2 side A | 0001 0001 | 11H |
| After INT1 side A | 0011 0011 | 33H |
| After INT1side B | 0001 0011 | 13H |
| After INT2 side B | 0000 0011 | 03H |

After this executing sequence for INT1 and INT2 subroutines, the flag byte stored in memory location (9290H) will contain (03H) .Whenever the displaying program flips any required dot. It outs the data and the address required for flipping that dot. After a while it checks the flag byte stored in (9290H). if the displaying program detect that 03|h is stored in 9290|h, it will continue displaying the other dots. If not, the timer programmable reload register will count down ten times (about 5mSec) then the displaying program will switch OFF all transistors in the driver board, then it will continue displaying the other dots.


**Fault Detection**

From the previous section, it is clear that the flag byte stored in (9290H) will give an indication about the flipping process and the execution of the two subroutines due to the two interrupt signals $\overline{INT}1$ and $\overline{INT}2$. In the flipping circuit shown in Fig.1., supposing that T46 is not working (faulty transistor), examining the circuit in the following tow cases
Case 1: T46 is broken (open circuit)
a. If the transistor is open and we want to flip the dot in the direction caused by making T32 & T17 switched ON and T46 & T3 switched OFF. In that case we find that the broken transistor T46 has no effect and the dot will flip without any problem.
b. If we want to flip the dot to the other direction caused by making T46&T3 switched ON and T32&T17 switched OFF, we find that the dot will not flip and no current will pass through the resistor R22. Therefore $\overline{INT}1$ and $\overline{INT}2$ will not work, then flag byte stored in (9290H) will contain (00H).
Case 2:
a. If T46 is short and we want to flip the dot in the direction caused by making T32&T17 OFF and T46&T3 ON. In this case we find that the dot will flip normally. The flag byte stored in (9290H) contains (03H).
b. If T46 is short and we want to flip the dot in the direction caused by making T32&T17 ON and T46&T3 OFF. The 30V will transfer through the two diodes to the collector of T17, which will be switched ON causing a rectangular pulse at R22 as shown in Fig.12. No current will pass through the coil. Therefore no flipping occurred. The voltage drop across R22 is 8V rectangular with 100μSec duration causing the wave shaping circuit to charge until reaching 5.2v then discharging again to zero as shown in Fig.13. Comparing this new signal with 2 V and 3.4v in comp1 and comp2 will cause the $\overline{INT}1$ and $\overline{INT}2$ to be as depicted in fig.14. it is obvious; there are differences between the $\overline{INT}1$ and $\overline{INT}2$ in the case of normal and faulty

transistors. $\overline{INT}1$ Will broaden and overlap in time with $\overline{INT}2$ as depicted in the fig.15. In case of no short circuit the processor will execute INT2 subroutine, this will write (11H) in (9290H), and stay for about 25µSec before executing INT1 subroutine as depicted in fig.15. In case of T46 is short circuit the processor will execute INT2 subroutine and because of the overlapping between the two interrupts it will directly execute INT1 subroutine. This will write (33H) in memory location (9290H). by checking this byte after the first return from interrupt, finding (33H) stored in it means that there is a short circuit transistor, the byte stored in (9290H) is checked twice, after first return from interrupt to check if there is a short circuit transistor and after about 500 µSec to check if there is a broken transistor.

Table III demonstrates the content of the memory location in (9290H) and its indications:

Table III:

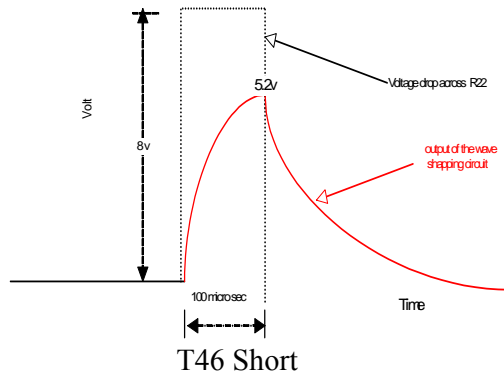| Content of 9290\|h | Indication |
|---|---|
| 03 | No fault |
| 00 | Open circuit (broken transistor) |
| 33 | Short circuit transistor |



T46 Short
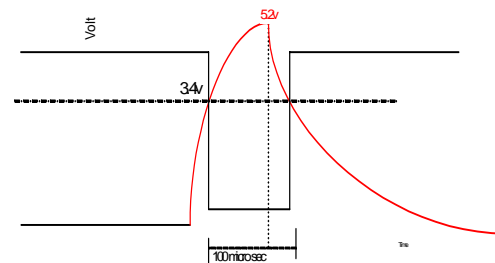Fig.12. Voltage across R22 "point E and GND"
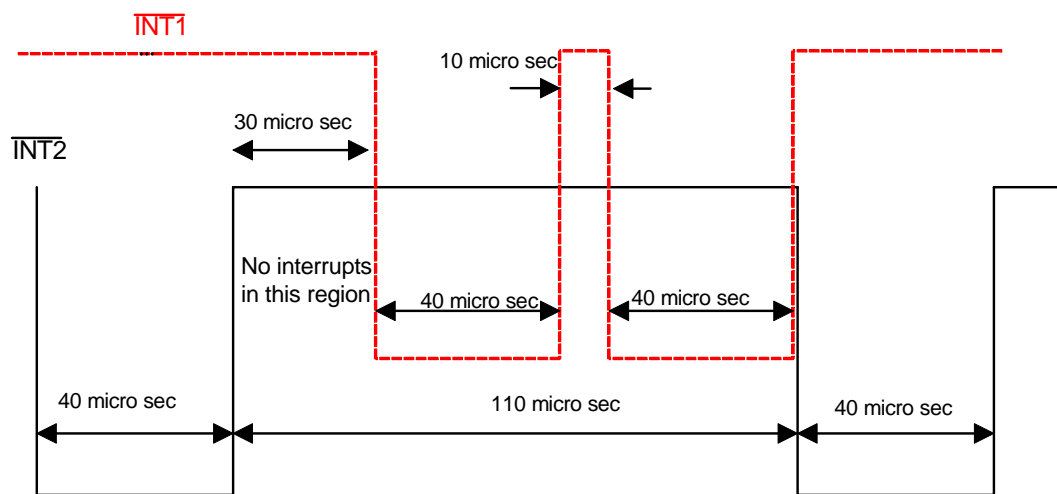
T46 Short
Fig.13. Pins "4","6" IC4 comparator



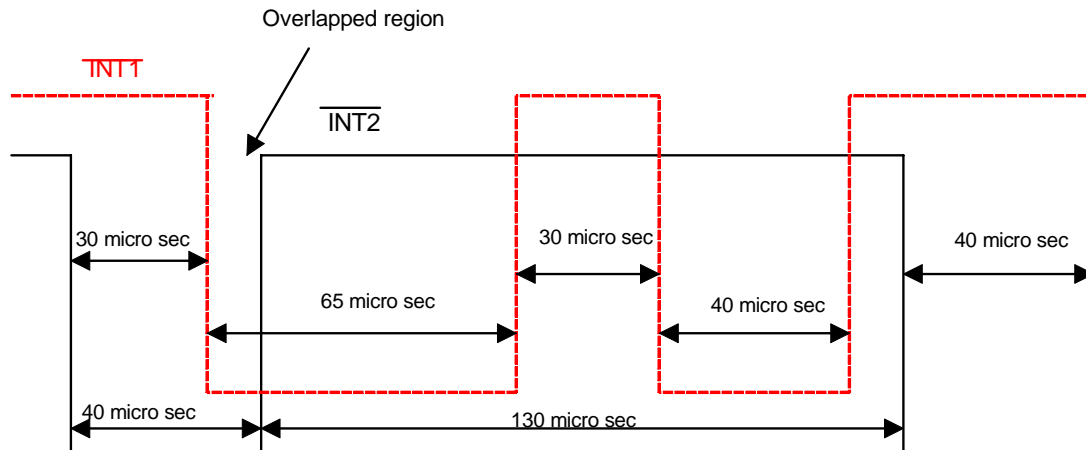Fig.14. Timing diagram for INT1and INT2 in case of No Fault

Fig.15. Timing diagram for INT1and INT2 in case of Short Transistor

**Conclusion**

Fault detection for displayed data in signal state boards is very important. We have analyzed in details the schematic diagram of the SSB driving board. We have studied the existing PC and SSB software programs. We have extracted the hexadecimal program loaded in the driver board EPROM and disassembled to investigate it. We have performed a visual basic program to control the SSB instead of the old one written in Pascal to cope with the difficulties result in executing the Pascal program using high speed computers. The program includes a new added fault tracing section to detect the error in displaying and gives an alarm to the station operator to deal with this error by use another way to tell the right messages to the vessel.

**References**

[1] Ramesh S. Gaonkar ,The Z80 microprocessor: Archtecture, Interfacing, Programming and Design, Prentice Hall, 1993.
[2] Paul Horowitz, Winfield Hill, The art of electronic, Cambridge University press 1983.
[3] C.J.Savant ,Martine S Roden ,Gordon L. Carpenter, Electronic Design , The Benjamin / Cumming Publishing Company Inc.
[4] D.S.Yadav, Arun Kumar Singh, Microcontroller: Features and Applications,  New AGE International (P), Publishers.
[5] R. W. Hanning , Coding and Information Theory, A prentice-Hall, 1980.
[6] Wolfgang W. Gartner, Transistors :priciples, Design and Applications, D. Van Nostrand Inc, 1980.