General Disclaimer

One or more of the Following Statements may affect this Document

- This document has been reproduced from the best copy furnished by the organizational source. It is being released in the interest of making available as much information as possible.

- This document may contain data, which exceeds the sheet parameters. It was furnished in this condition by the organizational source and is the best copy available.

- This document may contain tone-on-tone or color graphs, charts and/or pictures, which have been reproduced in black and white.

- This document is paginated as submitted by the original source.

- Portions of this document are not fully legible due to the historical nature of some of the material. However, it is the best reproduction available from the original submission.

Produced by the NASA Center for Aerospace Information (CASI)
LORAN-C DIGITAL WORD GENERATOR FOR USE WITH
A KIM-1 MICROPROCESSOR SYSTEM

The digital word generator used with Mini-L front end to develop a Loran sensor processor at Ohio University is described.

by

James D. Nickum
Avionics Engineering Center
Department of Electrical Engineering
Ohio University
Athens, Ohio 45701

December 1977

Supported by
National Aeronautics and Space Administration
Langley Research Center
Hampton, Virginia
Grant NGR 36-009-017
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>I</th>
<th>INTRODUCTION</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>II</td>
<td>OVERALL CIRCUIT DESCRIPTION</td>
<td>1</td>
</tr>
<tr>
<td>III</td>
<td>DETAILED CIRCUIT DESCRIPTION</td>
<td>1</td>
</tr>
<tr>
<td>IV</td>
<td>TESTING AND VERIFYING THE INTERFACE</td>
<td>2</td>
</tr>
<tr>
<td>V</td>
<td>SUMMARY</td>
<td>2</td>
</tr>
<tr>
<td>VI</td>
<td>REFERENCES</td>
<td>2</td>
</tr>
<tr>
<td>VII</td>
<td>APPENDICES</td>
<td></td>
</tr>
<tr>
<td>A</td>
<td>Software Listing of Verification Routine</td>
<td>10</td>
</tr>
<tr>
<td>B</td>
<td>Parts List for Interface</td>
<td>12</td>
</tr>
</tbody>
</table>
I. INTRODUCTION

The process of converting analog events to a digital form is a problem whenever information is to be processed by a digital computer. This report will address itself to the problem of translating the time of occurrence of received Loran-C pulses into a time, referenced to a particular period of occurrence.

This digital word generator is designed as part of a Loran-C sensor processor package being developed at Ohio University under the NASA Tri-University Program. The digital information from this word generator is processed in a KIM-1 microprocessor system which is based on the MOS 6502 CPU.

This final system will consist of a complete time difference sensor processor for determining position information using Loran-C charts. The system consists of the KIM-1 microprocessor module, a 4K RAM memory board, a user interface and the Loran-C word generator described in this report.

II. OVERALL CIRCUIT DESCRIPTION

In describing the interface refer to the overall circuit block diagram, Figure 1. The complete word generator is memory-mapped to the CPU. All information transfers under control of the CPU via address and data busses. The interface therefore appears as memory to the CPU.

The counter block is a six-digit BCD counter that is incremented at a $1 \mu s$ rate. The complete counter is reset to zero by the GRI reset logic every 99300 $\mu s$. This corresponds to the GRI rate of the East Coast Loran-C chain. The present counter output is always available to the tristate latches.

Referring to the control logic, a $10 \mu s$ TTL compatible pulse generated by the Mini-L front end, as described by Burhans [1], is the input to the interface. The leading edge of this $10 \mu s$ pulse from the front end also generates an interrupt to the processor which then through memory-mapped addressing reads three bytes of data from the interface.

There are two control flags that are software driven and control the output of the interrupt from the interface and the control of the GRI sync feature of the counters. These flags were provided to allow software flexibility in the sensor processor programs.

III. DETAILED CIRCUIT DESCRIPTION

Referring to Figure 2, a detailed schematic of the interface, devices U1-U5 are BCD counters. They are connected in a count-up scheme and clocked by the buffered 1 MHz system clock from U18a. The counter clear inputs are all tied to the output of U17b-9. The flip flop U17b, the diodes at U15 and the GRI sync enable flag output U20-10 comprise the GRI sync logic. If the GRI sync flag is low, then U17b-9 will be kept low and this will allow the counters to freerun. If the GRI sync flag is high, then as the counter increments to 99300 $\mu s$, the diode decoder will clock U17b-9 high which will clear all the counters. When the MSB of U1 goes low, this resets U17b and the cycle will repeat again.
The control logic is composed of U16a, b; U17a; U18a, b, c; U12b. Referring to the timing diagram of Figure 3, we see that this logic is responsible for the synchronization of the incoming 10μs pulse from Mini-L. It also assures that one and only one interrupt pulse will be generated. In addition, it generates the clock pulse for the tristate latches to clock the data from the counters into the latches. Finally, it provides a flag input from the CPU to enable or disable the Loran interrupts.

The address decoding is provided by U6; U18d, e, f; U12a, c, d. The device that simplifies the decoding is the 74LS138 decoder U6. Devices U18d, e and U12a, c provide a hexadecimal decoding of 3XXX, where the X's are a don't care value. This output along with Φ2 enables the decoder U6 and finally provides unique outputs for 3XX0, 3XX1, 3XX2 used as read only and 3XX3 is used as write only. Referring to Figures 5 and 6, which are the 6502 CPU input and output timing relationships, we can assume the following and allow the decoder U6 to be used as a read and write decoder without read and write steering logic. If we define locations 3XX0-3XX2 for read only and 3XX3 for write only and we use Φ2 as the decoding enable (rising edge clocking edge and low time as enable), we can see that no bus conflict will occur and there is no need to steer the read/write signals. This scheme was used because it simplifies the common logic otherwise required if we were to say read and write on each individual decoder select output.

The control flag logic is simply an 8-bit latch that is enabled by the decoder output and clocked by the rising edge of Φ2.

IV. TESTING AND VERIFYING THE INTERFACE

To test the interface an interrupt process routine was written that would service the interrupt, get the data and display it on the KIM-1 display. Appendix A contains the software listing. Figure 7 is a basic flow chart of the testing routine. This routine is built around some of the display subroutines contained in the KIM-1 monitor[2]. The testing routine allows, along with the Mini-L front end, the observation of waveforms on an oscilloscope to verify the timing and, therefore, proper operation of the word generator. This routine can also be used as a qualitative test of the Mini-L front end, Loran-C word generator and the KIM-1 microprocessor system.

V. SUMMARY

In conclusion, the Loran-C word generator described in this report has been built and is operating on a KIM-1 microprocessor system. This interface is by no means a unique or ultimate solution but it accomplishes present goals in developing Loran-C sensor processor techniques at Ohio University. (See Figures 8 and 9 for photographs of the system).

VI. REFERENCES

Figure 1. Overall Block Diagram.
Figure 3. Timing Diagram.
Two Phase Clock Timing
Figure 4

Timing for Reading Data from Memory or Peripherals
Figure 5
Timing for Writing Data to Memory or Peripherals

Figure 6
START

INITIALIZE IRQ VECTOR

ENABLE INTERRUPTS

DISPLAY

IRQ

READ DATA FROM INTERFACE

STORE TO DISPLAY

Figure 7. Interrupt Flow Chart.
Figure 8. Loran Digital Word Generator Board.

Figure 9. Complete Digital Signal Processor. Loran Word Generator Board on Left, KIM-1 Microprocessor on Right.
VII. APPENDICES

A. Software Listing of Verification Routine.
EXEClUTION BEGINS...

END PASS 1: 0 ERRORS

1 # THIS ROUTINE WILL TEST AND VERIFY THE LORAN C
2 # WORD GENERATOR. THE START ADDRESS IS $1382.
3 # THE PROGRAM WILL RUN UNTIL HALTED. IF THE
4 # HARDWARE IS OPERATING THE DISPLAY WILL
5 # FLICKER THE INPUT DATA. IF ALL SEGMENTS
6 # DO NOT FLICKER THERE IS A DATA TRANSFER
7 # PROBLEM IN HARDWARE.
8 # THE FOLLOWING ARE ADDRESSING AND DATA
9 # DEFINITIONS.
10 *
11 1380 ORG $1380
12 1380 IOS0 EQU 06 INTERRUPT ON SYNC ON
13 1380 IOSX EQU 04 INTERRUPT ON SYNC OFF
14 1380 IXS0 EQU 02 INTERRUPT OFF SYNC ON
15 1380 IXSX EQU 00 INTERRUPT OFF SYNC OFF
16 1380 SCANDS EQU $1F1F
17 1380 CONTL EQU $3003
18 1380 LOR1 EQU $3000
19 1380 LOR2 EQU $3001
20 1380 LOR3 EQU $3002
21 1380 IROL EQU $17FE
22 1380 DISP1 EQU $00F9
23 1380 DISP2 EQU $00FA
24 1380 DISP3 EQU $00FB
25 1380 A0 13 XIRQ ADD LIRQ
26 *
27 "PROGRAM BEGINS HERE"
28 *
29 1382 78 SEI DISABLE INTERRUPTS
30 1383 A9 02 LDA =IXS0 DISABLE INTERFACE INTERRUPTS
31 1385 80 03 30 STA CONTL
32 1388 AD 81 13 LDA XIRQ+1 GET MSB OF INTERRUPT VECTOR
33 138B 8D FF 17 STA IROL+1 SET INTERRUPT VECTOR MSB
34 138E AD 80 13 LDA XIRQ GET LSB INTERRUPT VECTOR
35 1391 BD FF 17 STA IROL SET INTERRUPT VECTOR LSB
36 1394 58 CLI CLEAR CPU INTERRUPT DISABLE
37 1395 A7 06 LDA =IOS0
38 1397 BD 03 30 STA CONTL ENABLE INTERFACE INTERRUPTS
39 139A 20 1F 1F LOOP JSR SCANDS LOOP IN DISPLAY ROUTINE
40 139D 4C 9A 13 JMP LOOP
41 *
42 "INTERUPT SERVICE ROUTINE BEGINS HERE"
43 *
44 13A0 A9 02 LIRQ LDA =IXS0
45 13A2 80 03 30 STA CONTL DISABLE INTERFACE INTERRUPTS
46 13A5 AD 02 30 LDA LOR3 GET LSB OF TIME
47 13A8 85 F9 STA DISP1 SAVE IN DISPLAY
48 13AA AD 01 30 LDA LOR2 GET MSB OF TIME
49 13AD 85 FA STA DISP2 SAVE IN DISPLAY
50 13AF AD 00 30 LDA LOR1 GET MSB OF TIME
51 13B2 27 0F AND #$40 MASK OFF TOP HALF BYTE
52 13B4 85 FB STA DISP3 SAVE IN DISPLAY
53 13B6 A7 06 LDA =IOS0
54 13B8 8D 03 30 STA CONTL ENABLE INTERFACE INTERRUPTS
55 13BB 40 RTI RETURN FROM INTERRUPT
56 END

END PASS 2: 0 ERRORS
B. Parts List for Interface.

<table>
<thead>
<tr>
<th></th>
<th>Parts</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>U1, U2, U3, U4, U5</td>
<td>74LS193</td>
</tr>
<tr>
<td>2</td>
<td>U7, U8, U9, U10, U11</td>
<td>4076</td>
</tr>
<tr>
<td>3</td>
<td>U6</td>
<td>74LS138</td>
</tr>
<tr>
<td>4</td>
<td>U12</td>
<td>74LS00</td>
</tr>
<tr>
<td>5</td>
<td>U18</td>
<td>74LS04</td>
</tr>
<tr>
<td>6</td>
<td>U16, U17</td>
<td>7474</td>
</tr>
<tr>
<td>7</td>
<td>U19, U20</td>
<td>74175</td>
</tr>
<tr>
<td>8</td>
<td>Diodes</td>
<td>IN4148</td>
</tr>
</tbody>
</table>