

A SOFTWARE FM DEMODULATOR

Adem GEDİK, Serhat YILMAZ and Hasan DİNÇER

The University of Kocaeli, Electronic and Communication Systems Research and
Application Center (EHSAM), İzmit - Turkey
adem@kou.edu.tr

Abstract - Software FM demodulator is presented which is suitable for analog FM modulator. In the demodulator, zero crossing counter method was applied in computer environment for demodulation of FM signal. Analog FM signal was first converted to TTL level square wave FM signal and then applied the software-based FM demodulator. In software-based FM demodulator, the original signal was obtained by measuring the spacing between zero crossing.

Index Terms : FM Demodulator, Software Radios

1. Introduction

Digital signal processing has three steps: First step is converting analog signals to digital signals (A/D). Second step is digital processing of the converted signals and last step is converting digital signals to analog (D/A). Digital signal processing (DSP) has many advantages. First of all, DSP is done by means of a software. Therefore modifications are done on software instead of hardware. This provides us flexibility in system development.

FM signal demodulation may be done by three methods: Frequency discrimination, phase tracking and zero crossing counters. Zero crossing counter method is suitable for software FM demodulator. DSP approach may also be used for demodulation of FM signals. In this paper, it is aimed to investigate what requirements are needed for FM signal demodulation with the help of computer. In this purpose, a computer program was written for zero crossing counter method. Because digital computer cannot run with analog signal, an analog input / output (I/O) interface card was used. The I/O card consist of an Counter/Timer (C/T), a 12 bit Analog to Digital Converter, a 12 bit Digital to Analog converter. The C/T is a programmable counter/timer chip organized as 3 independent 16-bit counters, which may be clocked at speeds up to 5 MHz. All the modes of operation can be invoked under software control. The ADC is based on the latest technology 12 bit successive approximation analog to digital converter and an 8 channel analog multiplexer. DAC is based on the high

performance 12 bit Digital to Analog Converter. The DAC generates a current output that is converted into a voltage and buffered by two op-amps. Demodulation of FM signal processing is done in real time. Data flow between I/O card and computer is done as fast as possible for recovering amplitude and frequency components of data. Therefore counting intervals must be constant. Computer uses interrupts for this kind of process. Two types of interrupt are exist; hardware and software. The hardware interrupt occurs when electrical signal is applied to the interrupt pin then address of interrupt service routine (ISR) is transferred to the databus then the computer gets address of ISR from databus and run ISR in the address. Generally, ISRs finish "RETI : Return From Interrupt" instruction. When microprocessor operates this instruction pops all data from stack and continues real task. The I/O card can sent a interrupt request to IRQ_n ($n = 1..7$) interrupt request line when complete A/D conversion, pin 0 and pin 3 of C port of 8255 programmable peripheral interface (PPI) are activated, and time interval of three timer finished.

2. Interrupt Vector Table

PCs have 256 possible interrupts numbered from 0 to 255. Each interrupt has an associated interrupt routine to handle the particular condition. To organize the 256 interrupts, the starting addresses of the corresponding interrupt routines are arranged in the interrupt vector table. When an interrupt occurs, the processor automatically retrieves the starting address of the interrupt routine from interrupt vector table. The starting

segment address. Both addresses are 16 bits (2 bytes) wide. So each table entry occupies 4 bytes. The total length of the table is 256 x 4 or 1024 bytes (1K). Because the interrupt vector table is in RAM (Random Access Memory),

any program can change it. However, TSR (Terminate and Stay Resident) programs and device drivers use the table.

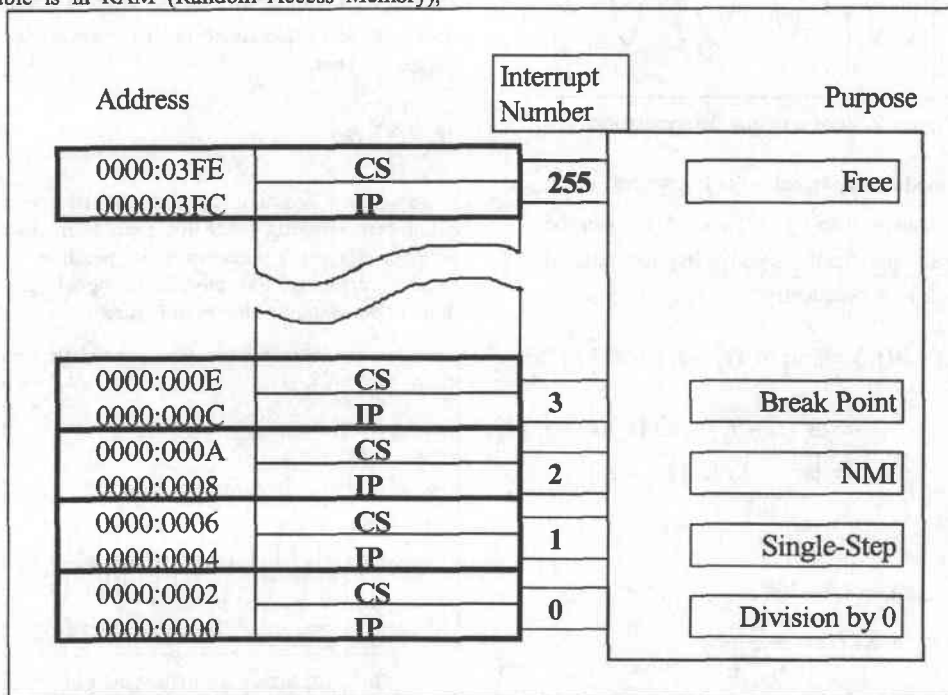


Figure 1. Interrupt vector table

Table 1. Certain interrupts of PC

No	Address	Purpose
00	000 - 003	Processor: Division by zero
01	004 - 007	Processor: Single step
02	008 - 00B	Processor: NMI
03	00C - 00F	Processor: Breakpoint reached
04	010 - 013	Processor: Numeric overflow
05	014 - 017	Hardcopy
06	018 - 01B	Unknown instruction
07	01D - 01F	Reserved
08	020 - 023	IRQ0: Timer (Call 18.2 times/sec.
09	024 - 027	IRQ1: Keyboard
0A	028 - 02B	IRQ2: 2 nd 8259
0B	02C - 02F	IRQ3: Serial port 2
0C	030 - 033	IRQ4: Serial port 1
0D	034 - 037	IRQ5: Hard drive
0E	038 - 03B	IRQ6: Diskette
0F	03C - 03F	IRQ7: Printer
...

2. Zero Crossing Counter Method

FM signal can demodulate by means of count zero crossing. FM signal of number of zero crossing can to take the measurement of momentary frequency in given time interval (Tc). Think about FM signal granted by

$$f_c(t) = A \cos \theta(t) = A \cos \left[\omega_c t + K \int f(t) dt \right]$$

As an example, let $f(t) = at$, $0 \leq t \leq T$, a repetitive ramp. Then $\omega_i = \omega_c t + Kat$,

$$\theta(t) = \omega_c t + \frac{Kat^2}{2}$$

Let t_1 be a zero crossing and $t_2 = t_1 + \Delta t$ be the next zero crossing. This situation is shown in Figure 2. In that case $\theta(t_2) - \theta(t_1) = \pi$. f_c is the carrier frequency. B is bandwidth of $f(t)$ and presume it is much less than f_c .

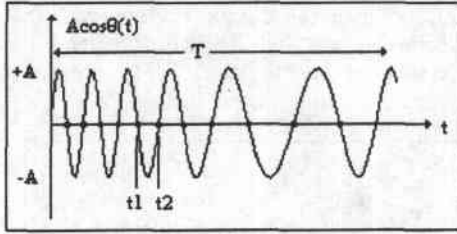


Figure 2. Zero crossing determination.

The modulated signal $f(t)$ changes much more slowly than f_c . Thus $f(t)$ can be assumed practically steady in the interval $(t_2 - t_1)$, consequently

$$\begin{aligned} \theta(t_2) - \theta(t_1) &= \pi = \omega_c(t_2 - t_1) + K \int_{t_1}^{t_2} f(t) dt \\ &= \omega_c(t_2 - t_1) + Kf(t_1)(t_2 - t_1) \\ &= \underbrace{[\omega_c + Kf(t_1)]}_{\frac{d\theta}{dt} = \omega_i} (t_2 - t_1) \end{aligned}$$

From equation 1, we have

$$\begin{aligned} \omega_i &= \omega_c + Kf(t) = \frac{\pi}{t_2 - t_1} \\ f_i &= f_c + \frac{K}{2\pi} f(t) = \frac{1}{2(t_2 - t_1)} \end{aligned}$$

The desired output $f(t)$ can be obtained by measuring the spacing between zero crossing.

If only positive-going zero crossing are considered, at this time the slope of $f_c(t)$ is positive, we get

$$f_i = \frac{1}{t_2 - t_1}$$

In a given time interval, a simple way of measuring the spacing of zero crossing is to actually count the number of zero crossings.

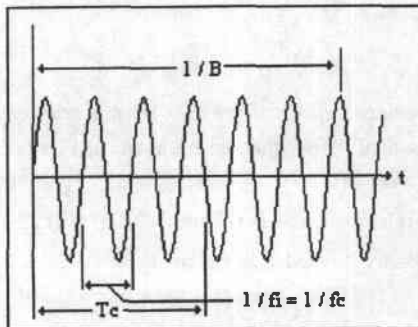


Figure 3. Counting intervals

Consequently, consider a counting interval T_c long enough so that it counts a significant number of zero crossings, yet short enough compared with $1/B$ so that $f(t)$ still does not change too much in this interval (see Figure 3). Then

$$\frac{1}{f_c} T_c \ll \frac{1}{B}$$

Zero crossing counter can make count zero crossing twice for each alternance or ones for each alternance in positive or negative crossing. FM modulated signal time domain presentation shown in Figure 4.

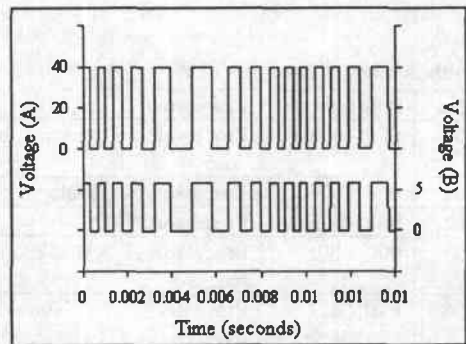
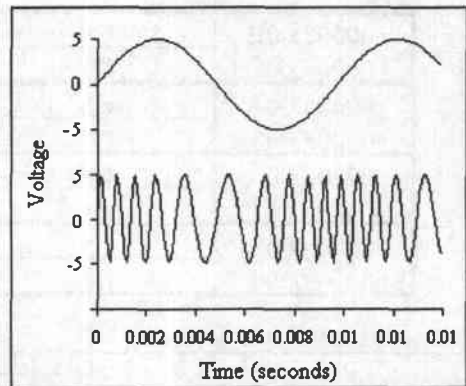


Figure 4. a) Main signal b) FM Modulated Signal c) Limiter output d) TTL Level Transformer Output

3. Applying Zero Crossing Method in Computer Environment

In this paper, GW GFG-813 was used as a FM signal generator. It includes limiter and TTL level transformer. GW GFG-8050 was used for time base generator. Time base generator sends a signal to IRQ4 line in every

generator sends a signal to IRQ4 line in every T_c period for execute ISR routine. Time base generator is connected to counter 0 for adjust time base by means of software. Because every counter have five operation mode. Counter 0 is in mode 2. This mode of operation causes to act as a divide by n counter. So various time interval can be obtained by means of adjust n . Counter 2 is in mode 0. This mode is normal counter mode.

Starting address of program put into interrupt vector table instead of IRQ4 by main program. So, ISR executed by microprocessor when time base generator sent a signal to computer. Because n was selected one.

Flow chart of written program for apply to method in computer can be summarized:

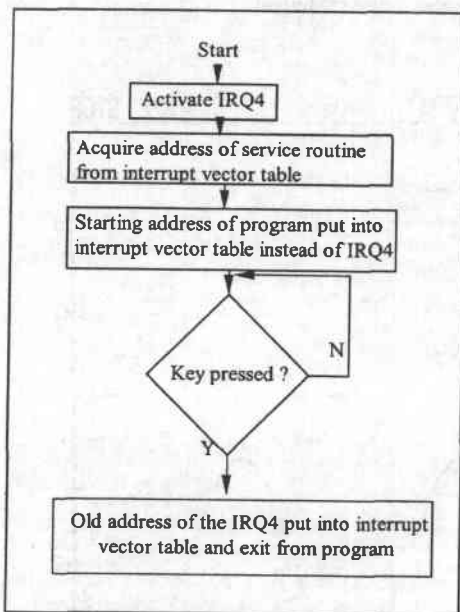


Figure 5. Flow chart of main program

ISR, which its address put into interrupt vector table by means of main program, can be summarized in Figure 6.

In the ISR, 2024 is subtracted from read value for obtain $\pm 5V$ level in the output. The D / A converter is 12 bits. So, 4096, 0 corresponds +5V, -5V respectively. ISR is called every T_c period by means of the signal which is produced by time base generator. Therefore the output voltage is updated every T_c period. Results are shown in Figure 8.

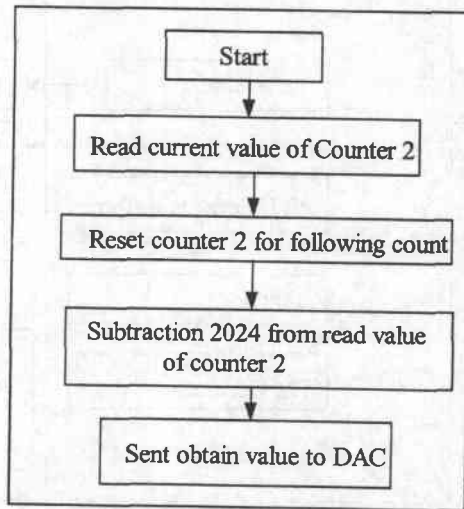


Figure 6. Flow chart of ISR

4. Conclusion

In this paper, zero crossing counter method applied computer environment for demodulation of FM signal. 2.5 MHz \pm 75 kHz signal can be demodulated because counters count speeds not enough on the I/O card. Demodulation was realized for low frequency of main signal. Our study is continuing for high frequency main signal and real FM receiver.

References

- [1] J. A. Wepman, "Analog-to-Digital Converters and Their Applications in Radio Receivers," *IEEE Communications Magazine*, May 1995, pp 39 - 45.
- [2] J. Mitola, "The Software Radio Architecture," *IEEE Communications Magazine*, May 1995, pp 30 - 38.
- [3] Tischer, *PC Intern System Programming*, (U.S.A: Abacus, 1995).
- [4] Schwartz, *Information Transmission, Modulation, and Noise*, (Kogokusha: McGraw-Hill, 1970).

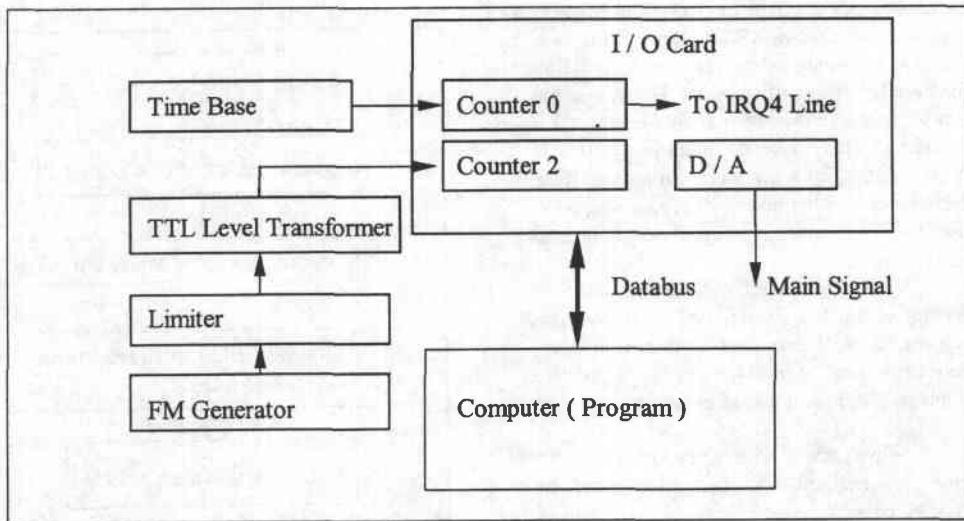


Figure 7. Hardware of Zero Crossing Counter Method

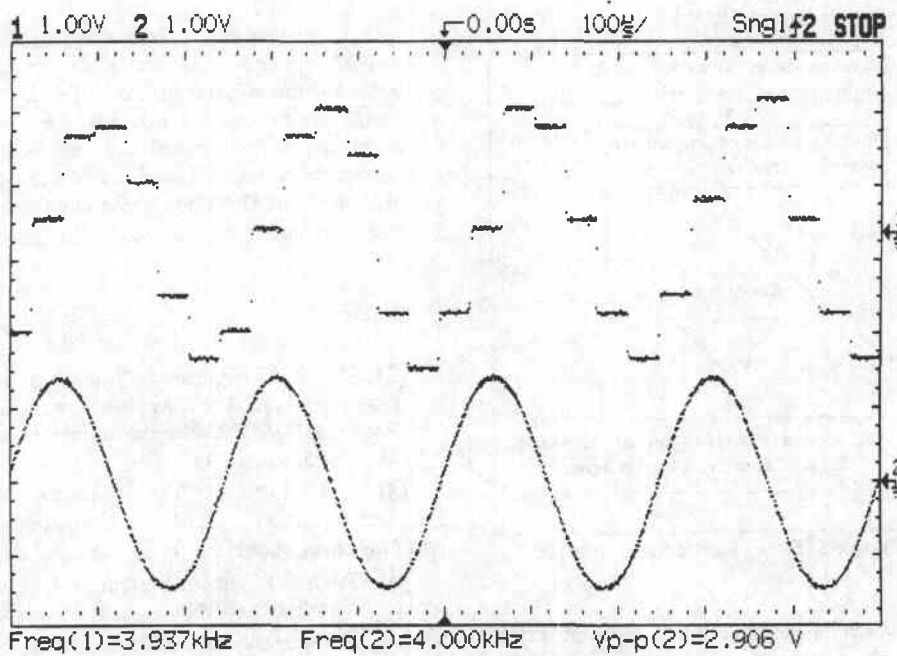


Figure 8. Results a) Original Signal b) Signal that is obtained by using computer