A COMPUTER CONTROLLED PULSE GENERATOR FOR AN ST RADAR SYSTEM

Ghebrebrhan Ogubazghi
Geophysical Observatory
Addis Ababa University

ABSTRACT

A computer controlled pulse genarator for an ST radar system is described. It uses a highly flexible software and a hardware with a small IC count, making the system compact and highly programmable. The parameters of the signals of the pulse generatar are initially entered from the keyboard. The computer then generates one period of the set of signals in a suitable coded form. This data is then stored in the memory of the pulse generator which enables the latter to run in an autonomous manner once it is started.

INTRODUCTION

During the last decade, radars capable of measuring atmospheric parameters such as wind field, turbulence, and stability in the free atmosphere have been developed [1, 2, 3].

These radars can perform atmospheric observations for altitude ranges starting from about 1 kilometer to several kilometers above ground level. Radars that can observe the troposphere (that part of the atmosphere below about 15 kilometers), and the stratosphere (the atmosphere above the troposphere and below about 50 kilometers), are called *ST* (Stratosphere Troposphere) radar systems.

Several of these radars operate in the lower VHF band notably at about 50 MHZ, but there are also radars that operate in the UHF band. Those radars operating in the VHF band use highly directional electrically steerable phased array antennas such as the one described by Balsley and Ecklund [4].

The same antenna is used both for transmitting and receiving. During transmission, the antenna is connected to the transmitter and the receiver is disabled or blanked. On the other hand, during reception, the antenna is connected to the receiver and the transmitter is not excited. The antenna switching is performed with the aid of an electronic switch called the *TR* (for transmit/receive).

The logic signals that control the *TR* switch, the transmitter excitation and the receiver blanking circuit in an *ST* radar system are provided by the radar pulse generator. The pulse generator also provides the sampling signals necessary for processing the received radar signal. The precise phases and durations of these signals and the requirement of changing parameters of these signals make the radar pulse generator a rather complex digital machine.

The purpose of this paper is to describe a computer controlled *ST* radar pulse generator that we have designed and built for an *ST* radar system that may be installed in a site arround Addis Ababa in the near future. The pulser has been tested in an experimental *ST* radar system belonging to the University of Toulon in France, and has worked satisfactorily for more than three years.

As will be shown below, the pulser that has been built needs much fewer components and hence is more compact than a traditional pulser. It is thought that the principles outlined in this paper can also find applications in other areas of instrumentation and control.

PRINCIPLES OF OPERATION

The design of the pulse generator discussed below is based on the principles outlined by Carter *et al* [5]. This work, in turn, was motivated by the work of Woodman *et al* [6], in their design of controllers for the SOUSY radar system in West Germany using a circulating memory scheme.

The present design incorporates a highly flexible software and is organized arround a simple local memory as opposed to a FIFO configuration of the Carter version.

One period of the set of signals that are required to be generated by the pulse generator are shown in Fig. 1(a). These are the receiver blanking *(RB)* pulse, the *TR* pulse, the logic version of the transmitted pulse *(PE)*, a set of sampling pulses *(SM)*, and an end of cycle *(EC)* waveform. The last signal is used for the internal use of the pulser.

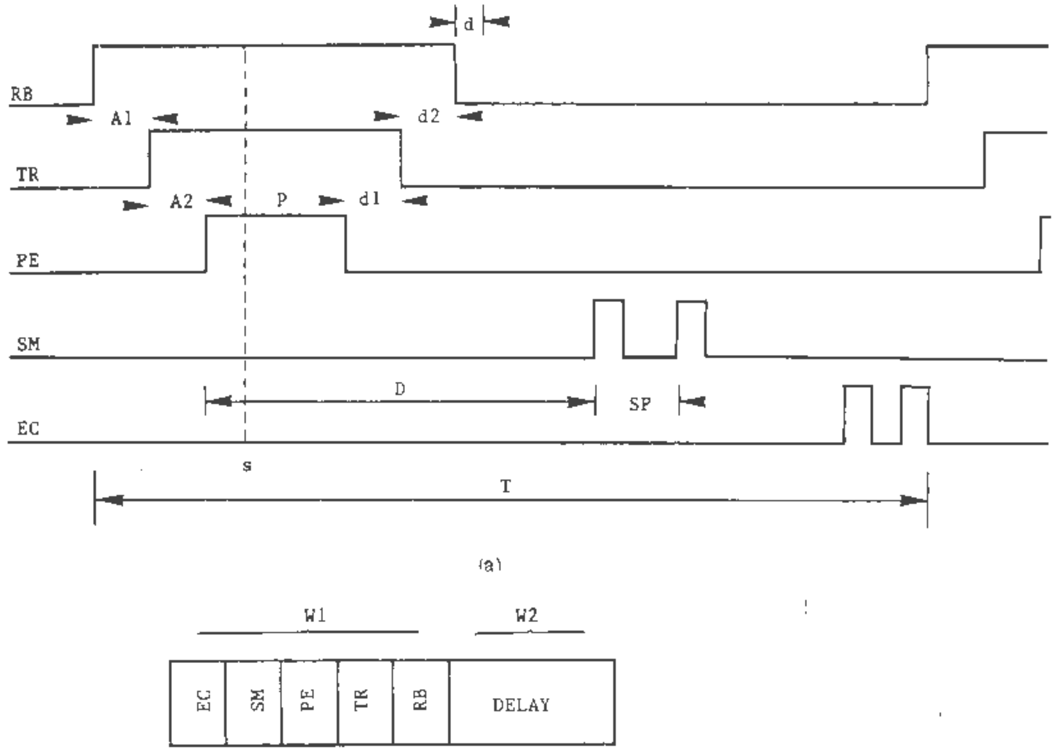

Fig. 1 One period of the signals the pulse generator is required to generate (a), and composition of the memory word (b). For the indicated state "s", $W1 = 00111$ and $W2 = P$; $(P<7)$.

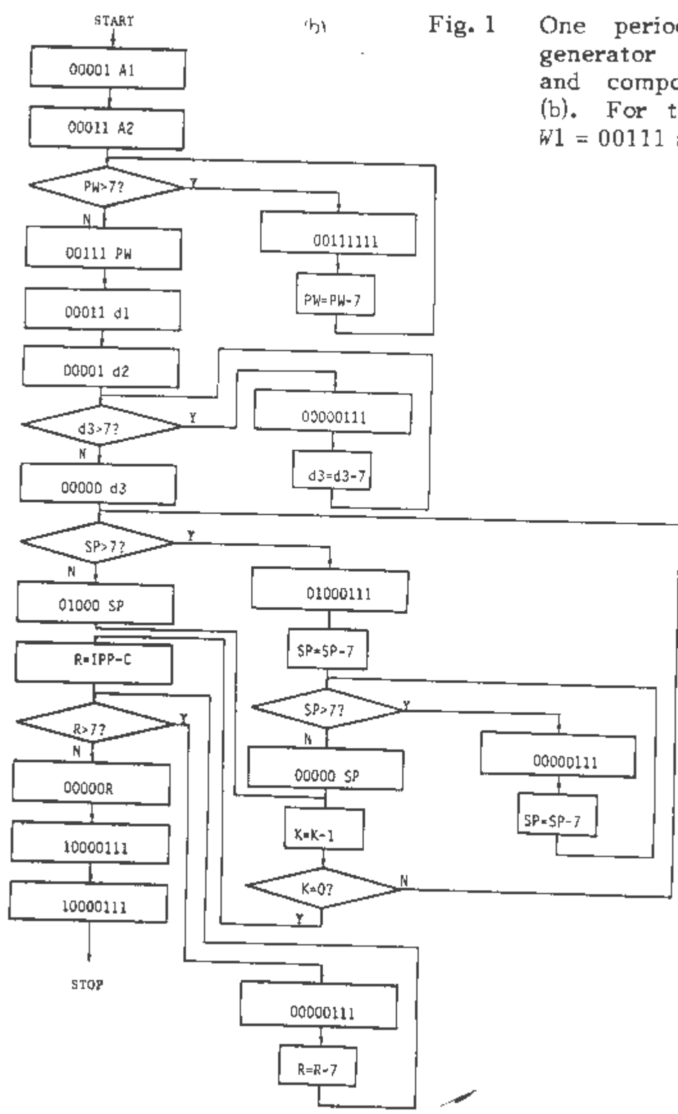

Fig. 2 Flow chart for the generation of the word sequence.

Several parameters of the set of signals are indicated in Fig. 1(a). Most of them are self explanatory. The delay (μ) is the minimum possible duration between the trailing edge of the *PB* pulse and the rising edge of the first *SM* pulse. It is equal to 1μs. *D* must have a value equal or greater than the sum of *P*, *d*1, *d*2, and *d*. *SP* is the spacing between consecutive sampling pulses.

The computer generates a sequence of data words that represent one period of the set of signals shown in Fig. 1(a). And stores this in the memory of the pulse generator which is interfaced to it. Once the data is stored in its memory, the pulse generator is started by the computer and operates freely without external intervention. The data stored in memory is read sequentially in a cyclic manner and upou decoding the required signals are obtained. Details are given in the next section.

The computer used is a Data General machine belonging to the NOVA line computers. Its hardware and software features are described by Data General [7]. A detailed description of interfacing techniques for the NOVA line of computers is also given by Cluley [8].

SYSTEM DESCRIPTION

a. SOFTWARE: The software for the pulser is written in assembly language. It performs its tasks in four stages. First it accepts from the keyboard the signal parameters shown in Fig. 1(a). (In reality, the parameters commonly used are stored in the program and changes are entered). Then the entered parameters are checked for consistency. Next the program generates the set of data corresponding to the entered parameters. Finally, the generated data are loaded to the pulser's memory and the pulser is started.

The program generates the sequence of words in the following manner. One period of the set of signals is considered. The program effectively performs sampling on the set of signals. The sampled state and its duration form a memory word. The memory word is of eight bits and is composed of *W*1 and *W*2 as shown in Fig. 1(b). *W*1 represents the sampled state of the set of signals and is 5 bits long (one bit for each signal). On the other hand, *W*2 represents the duration of the sampled state. It is 3 bits long.

One unit of delay is equivalent to 1 μs. The maximum delay that can be handled in one word is therefore 7 μ s. On this basis, if a given state has a delay greater than 7μs, two or more consecutive words with the same state identifier *W*1, and whose delay sum is equivalent to the delay in question are generated. To optimize memory space, only the last word may pocess a delay different than 7 μs.

As an example, consider the state "8" shown in Fig. 1(a). Here *W*1 is 00111 and the delay is *P*. We will consider two values of *P*.

CASE 1: *P* = 5 μs
In this case, the resulting word is 00111101.

CASE 2: *P* = 16 μs
The value of P in this case is greater than 7μs. Consequently, more than one word are required to represent the given state. The required words are, 00111111, 00111111, and 00111010.

Every state of the set of signals is sampled in this manner consecutively, and the resulting words are arranged similarily. A flow chart for performing this is shown in Fig. 2.

b. HARDWARE: The organization of the hardware is shown in Fig. 3. It is connected to the computer by 8 data lines and 10 coutrol and address lines. The latter group consists of 6 device address lines, 3 control lines (for DOA, DOB and DOC), and a single control line *(CL)*, for starting and stopping the pulser. The associated interface protocols are given in Fig. 4 and instructions of using them are given by Data General [7] and Cluley [8].

A simplified circuit diagram of the hardware is shown in Fig. 5. When the *CL* line is set to zero, the pulser is not running. At this state, the computer can load data to the memory of the pulser.

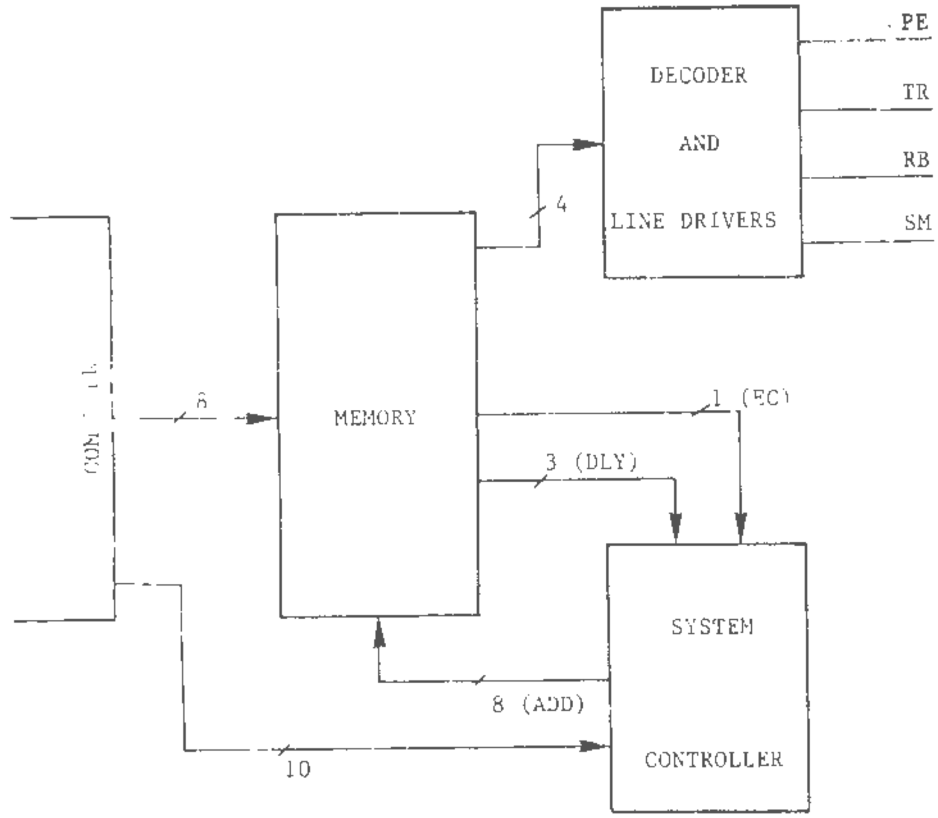

Fig. 3 Organization of the hardware.

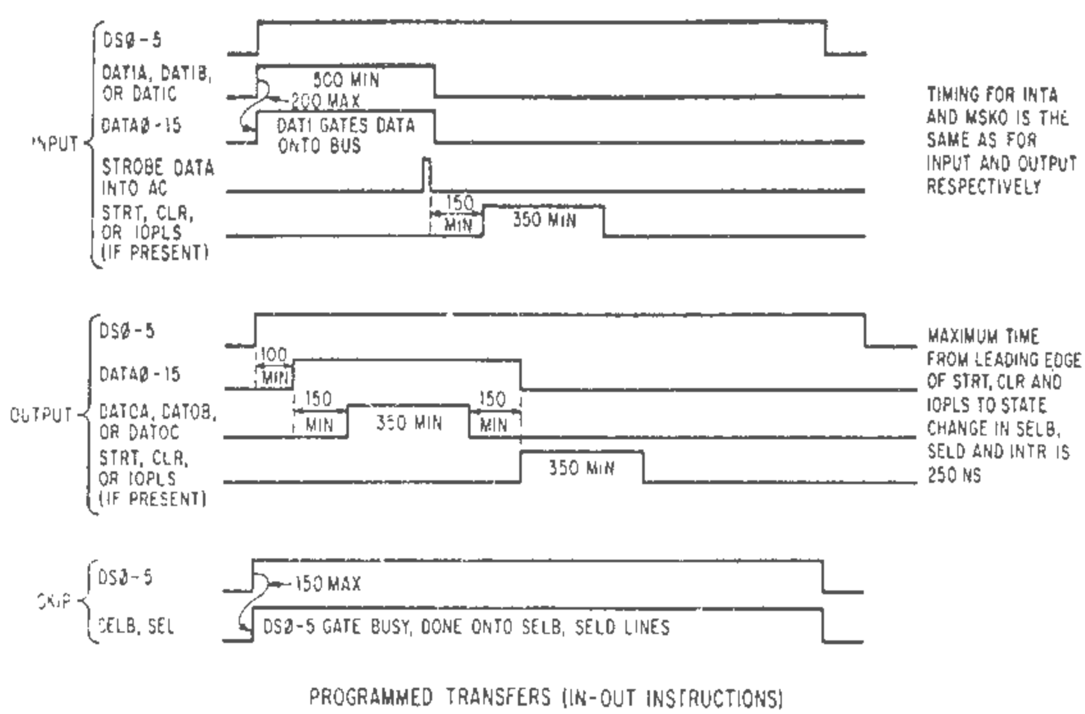

Fig. 4 [After Data General Corp. (1972)]

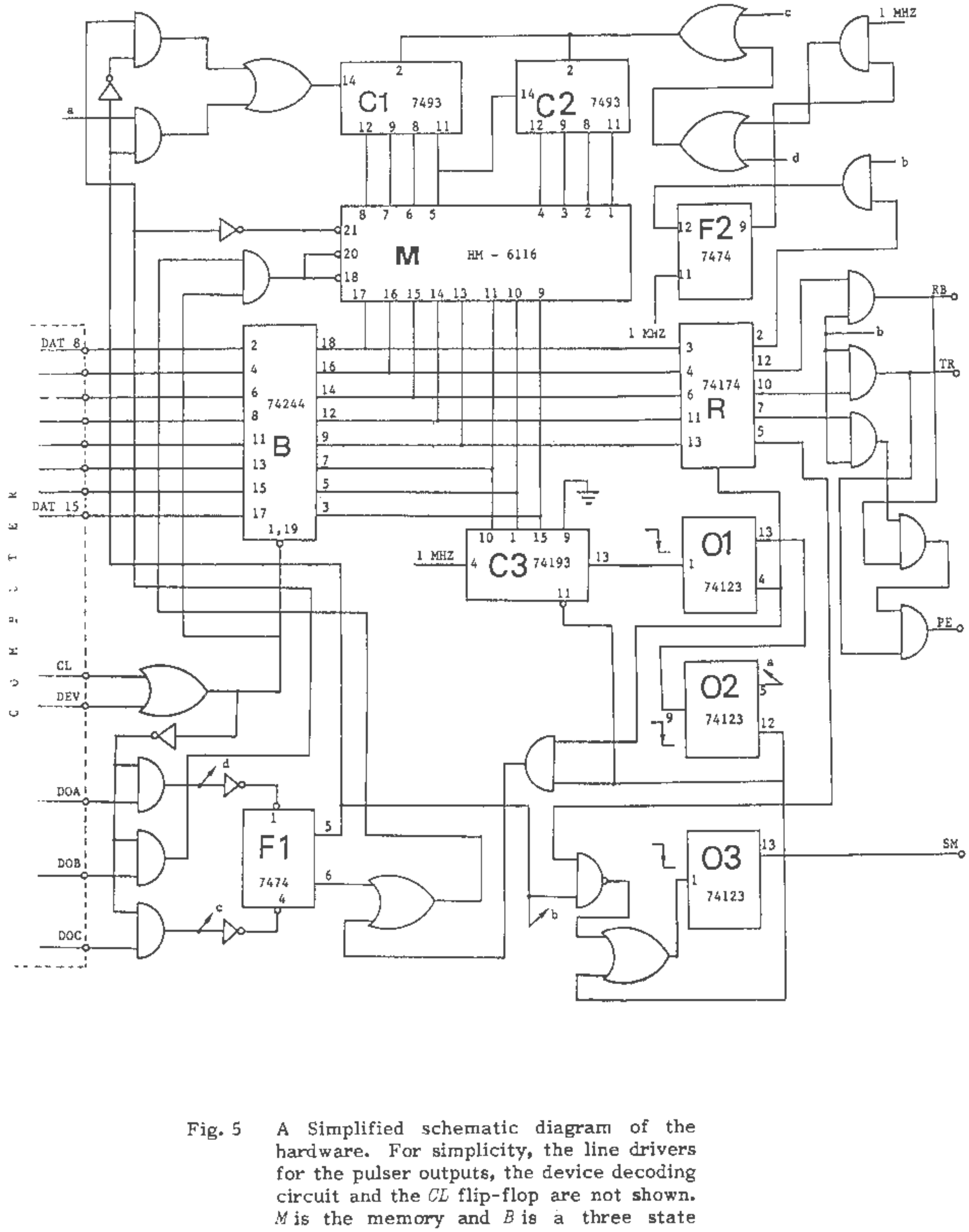

Fig. 5 A Simplified schematic diagram of the hardware. For simplicity, the line drivers for the pulser outputs, the device decoding circuit and the *CL* flip-flop are not shown. *M* is the memory and *B* is a three state buffer.

To load data, the computer uses the DOA instruction to clear flip-flop $F1$ and to set address counters $C1$ and $C2$ to zero. Data are then transferred using the DOB instruction. At the end of data transfer, the computer uses the DOC instruction to set $F1$ and to reset-to-zero the address counters. Finally it sets the CL line to one to start the pulser.

When the pulser is running, down counter $C3$, and oneshots 01 and 02 insure that, $W1$ of the word read from memory is latched to register R at the right moment and that the loaded data remains there for the duration contained in $W2$ of the word. This process continues until the EC pulses are detected, where recycling memory reading is performed with the aid of flip-flop $F2$.

Oneshot 03 shapes the sampling pulses to the widths demanded by the A/D converters of the signal processing system. For system safety, PE is set to ON only during the presence of RB and TR.

ADVANTAGES AND DISADVANTAGES

The pulser under consideration uses a small IC count (about 15). Consequently, it is more compact and less costly than the corresponding conventional pulser which can have an IC count of more than 100. Another advantage offered by the present pulser is that of programability. By developping the radar software, different radar operating modes can be performed automatically [9]. Such operating modes would require human attendance with conventional pulsers. Finally, the present pulser has been found to introduce less internal interference noise, particularily when using small PE and SP [9], as compared to a conventional pulser.

One disadvantage that has been observed is that when tunning the radar system, one must change signal prarameters more often. This can be done easily with conventional pulsers. With the present pulser, the system must be stopped to change the parameters. This disadvantage, however, is not important under normal operating conditions.

ACKNOWLEDGEMENTS

This work was done under a UNDP scholarship grant. I would like to thank Prof. M. Crochet of the University of Toulon for fruitful discussions and for providing the facilities for the work. I also thank Mr. P. Le Louarn and Mr. Y. Ruin (both from the University of Toulon), for software assistances.

REFERENCES

1. Woodman, R. F. and A. Guillen (1974), "Radar Observations of Winds and Turbulence in the Stratosphere and Mesosphere", J. Atmos. Sc., Vol. 31 No. 3, 1974.

2. Gage, K. S., and B. B. Balsley, "Doppler Radar Probing of the Clear Atmosphere", Bull. Am Meteo. Soc. Vol. 59, No. 9, 1978.

3. Green, J. L., S. Gage and T. E. Van Zandt, "Atmospheric Measurements by VHF Pulsed Doppler Radar", IEEE Trans. on Geos. Elect. Vol GE-17, No. 4, 1979.

4. Balsley, B. B., and W. L. Ecklund, "A portable coaxial collinear antenna", IEEE Trans. Antennas Propagat., AP-20, 1972.

5. Carter, D. A., A. E. Ayers, and R. P. Schneider, "A single board preprocessor and pulse generator", Proc. of Workshop on the technical and scientific aspects of MST radar, Puerto Rico, 1985.

6. Woodman, R. F., P. Kugel and J. Rottger "A coherent integrator-decoder preprocessor for the SOUSY-VHF-Radar", Radio Sci., Vol. 15, No. 2, 1980.

7. Data General Corp. "How to use the NOVA computers", Data General Corp, 1972.

8. Cluley, J. C. "Computer interfacing and on-line operation", Crane Russak, N.Y., 1975.

9. Ghebrebrhan Ogubazghi, "Amelioration des Performances d'un Systeme Radar VHF par des Techiniques de Codage et de Pretraitement: Analyse et Mise en Oeuvre", Thesis of Docteur es-Sciences, University of Tuolon, 1987.