DESIGN OF A MEMORY CONTROLLED PCM SYSTEM

B. M. TEMKIN and R. E. SHERLIN
Dynatronics Operation
General Dynamics Corporation

Summary Airborne PCM Telemeters are essentially alike in that they multiplex digital and analog data from numerous sources and format it into a serial data stream. Because of radical format differences in data requirements, however, frequent redesign of format generation logic is mandatory. The recent development of small non-volatile Read Only Memories with large bit capacities has suggested a telemeter whose characteristics can be altered merely by replacing a memory. The authors solution to the problem of the "universal encoder" is the design of the MPS-101 whose word format is controlled by a MOS-FET Read Only Memory. The MPS-101 can store six reasonably complex formats in its 6144 bit plug-in memory card. To insure flexibility, a bit rate oscillator and a premodulation filter are included within the interchangeable module. A program plug enables the user to address (select) the desired format, control the word size, select parity and vary the bit rate. Data rates can be clocked up to 750 KB/S, with no effect on system accuracy. This work has resulted in an encoder whose format is totally arbitrary and whose data structuring and major/minor frame synchronization techniques are strictly under "software" control. This work was performed under Sandia Laboratories Contract Number 72-3233.

Introduction PCM encoder design suffers under the precept that different missions mean different formats and channel complements. To meet this requirement, redesign (and re-qualification) is necessary for each specific application. The desired solution to this problem is an encoder which can allow for radical changes without redoing the electronics. Furthermore, the encoder should be capable of storing more than one format for use in adaptive telemetry. Recent advances in integrated circuit technology suggests that such a telemeter can be manufactured whose format is determined by a plug-in Read Only Memory. In addition, current "medium scale integration" allows for more flexible design wherein other encoding parameters can be changed by a simple program plug. The objective of this design effort was to manufacture a PCM encoder whose format is determined by programming a non-volatile Read Only Memory. The software must determine major and minor frame lengths plus synchronization coding so that ground decommutation is not impaired as the number of bits per frame changes. Features that are independent of the format, such as bits per word, bit rate, el alia, should be under the direction of digital or hard-wired inputs to a program plug. In addition, all the analog

circuitry should operate to specification regardless of channel sequence permutations or word and bit rates.

This paper will cover the design highlights in five sections. The first section will be a physical description. The second section will cover general system operation, illuminating the capabilities of the encoder. The next section explains the techniques for organizing the memory, including the sizing tradeoffs . The fourth section will describe the operation of the memory and formatting process. Finally, we will present conclusions and recommendations for usage.

General Description The encoder is packaged in a 48 cubic inch housing designed for a typical airborne environment. The form factor (in inches) is 5 x 3.1 x 3.1. A picture may be seen in Figure 1.

In order to meet the size requirement, but allow for repairability and maintainability, very extensive use of standard integrated circuits was necessary. In some areas, viz. , the analog and digital multiplexers, thin-film assemblies were used. It should be noted that these assemblies were designed for repairability - a necessary feature for large hybrid arrays.

The encoder can be disassembled rapidly to provide access to a customer oriented memory plug-in card. This functionally designed printed circuit has the components which are most likely to be changed. The significant functions located on this card are the:

1) ROM (6144 bit capacity)
2) Bit Rate Oscillator
3) Pre-modulation Filter

The remainder of the circuitry is on conventional double-sided printed circuit cards. Internal wiring and circuitry was designed for “buss” organization of the multiplexers. This allows for future expansion.

Power consumption is less than 12 watts, constant with battery voltage. This is significantly lower than that required for a similar system with plated wire or core memory.

System Operation The system diagram in Figure 2 illustrates the operation of the encoder. The analog channel complement is 64 single ended (0-5.2v) and 16 differential (0-40 mv). The low level signals are multiplexed and amplified by a drift stabilized true differential amplifier and interleaved with the high level channels to the analog-to-digital converter. The resultant binary code is formatted with the 64 parallel digital plus I serial

digital inputs in the format matrix. The sequence of sampled channels is entirely under the command of the contents of the Read Only Memory. Any data channel is randomly addressable by the memory control.

System operation is clocked by either an external or internal bit rate oscillator. If an external clock is present, it automatically overrides the internal oscillator. The bit rate enters a bits/word counter within timing/control. The output, which is the word (or channel) rate , initializes all operation within Memory Control, PCM Control, and the Analog-to-Digital Converter (ADC) . At the start of each new word, the Memory Control is instructed to locate a new multiplexer address. At the next word, this address is routed to the multiplexer address buss and the appropriate channel is selected. Synchronization information is similarly retrieved from memory. These codes are sent directly to the Format Matrix at the proper time.

In general, all parameters which determine the format are a function of Memory Control and the ROM. This includes the frame synchronization code(s), which may be of length independent of PCM word size. All parameters which determine the composition and nature of the PCM word, such as encoded bits per word, parity, bit format, and filtering are determined by the program plug. One area in which the functions cross is that the program plug injects “frame sync complement” , if desired. The decision to have hardware control over this option was determined merely by economical use of the logic elements. As an aside to those who may be unfamiliar with PCM frame synchronization techniques, frame code complement is an older method of maintaining subcommutator “lock” before use of the more modern frame counter (SCID) code became prevalent. The encoder can provide both simultaneously if required.

The Analog-to-Digital Converter, which is essentially a thin-film hybrid, operated admirably under system test. The encoded data was monotonic throughout the bit rate range of 4 KHz to 750 KHz over a temperature change of ±60°C .

A point should be made for the low level amplifier. Although the voltage drift and offset is “clamped” out, the rise time of the amplifier is less than two microseconds for a gain of 130.

Memory Organization The memory should be organized in a manner which provides the optimum software-hardware configuration. If the memory is organized so as to require minimum storage capacity, the required supporting logic could be extensive. If the opposite approach is taken, namely, storing the entire format in memory, the memory size could become impractical. The authors have chosen to select an approach suited for volumetric efficiency. In some cases, the minimum available memory size is more than sufficientto store any format required by the system. In such a case, there is no advantage to reducing the memory capacity by use of ancillary logic. However, if extensive

subcommutation is required, the memory capacity required to repeat store the main frame channel addresses, becomes excessive and supporting logic is attractive.

The memory is organized as 12 bit instructions (or memory words). The standard sizes used here are blocks of 256 x 12 MOS-FET ROM's. The 12 bit word is organized into an 8 bit data field and a 4 bit operation code. The data field contains the following:

1) Multiplexer address (to multiplexer buss)
2) Synchronization coding
3) Memory address modification
4) Housekeeping codes

The op code is used to route the instruction in the data field to the multiplexers or within the memory control for address modification, housekeeping, formatting, or code insertion in the format (such as sync code).

The memory is programmable for super, sub, and sub-sub commutation.

Memory Operation The Memory Control logic in Figure 3 is initiated by a word start pulse which originates from Timing/Control. Once initiated, the internal memory system timing is controlled by an asynchronous 2 MHz clock.

The ROM is addressed by either the Memory Address Counter (MAC) or the output of the Address Modifier. The MAC is advanced each PCM word time, while the Modifier output is utilized only for sub-commutation. At the start of each PCM word time, the memory location addressed by the MAC is read. Each memory location consists of a twelve bit word forwhich four bits are used for an op code and the remaining eight bits for the data field. The op code indicates what type of function is to be performed. For example, the op code may indicate that a high level channel is to be sampled. The data field would indicate which one of the sixty-four channels is to be sampled. The same is true for low level differential and bi-level channels.

Another op code is used to direct the system to route the data field to the output shift register, as in the case of frame sync. An op code is also used to direct the system to sample the Frame Counter contents, as in the case of the Sub Comm ID.

In each of the preceeding examples, a single memory cycle, at the PCM word rate, is used to control the system; i.e., the memory address is always under control of the MAC via the input selector. Also, none of the aforementioned op codes exercise any control over the memory subsystem, but are all routed directly to the PCM control logic.

There are several op codes used to control the memory subsystem. For example, one op code commands the memory to compare its data field to the contents of the Frame Counter by using the Address Modifier. If the comparison is proper, the counter is reset to zero. This occurs at the beginning of a major frame.

Another op code directs the MAC to be preset to a hard-wire address, thus controlling the memory start location for each minor frame. These op codes,which are for memory subsystem control only, do not result in any PCM output.

Whenever one of these op codes is encountered in the operating sequence, the control system will perform as directed, the MAC will advance, and the memory will immediately recycle. The memory will continue cycling, up to a maximum of three times, until an op code which results directly in a PCM output is detected and processed. For example, after the last channel in any minor frame is processed, the MAC is advanced. The resulting memory output is an instruction to test the Frame Counter contents. While the test is being conducted, the MAC is advanced and the memory recycles. The resulting memory output directs the system to preset the MAC, as described previously, and the memory cycles again. This output will be the first eight bits of the frame sync. The memory stops, and the op code, with sync bits, is output to the PCM control logic. The time required for three memory cycles is 7.5 microseconds.

The last type of memory function to be discussed is the generation of sub commands. The design philosophy employed in the system is to minimize memory size by taking advantage of the cyclic nature of sub comms. For example, when four sub comm channels, to be sampled at 1/4 main frame rate, are positioned in the mainframe PCM word slot, the memory contains only the information regarding sample rate and the memory address which contains the first of the four channel addresses. In this routine, memory control logic requires three memory cycles to generate a sub comm channel address for processing by the PCM control logic.

Conclusions This work has produced a PCM telemeter whose format is stored in a plug-in Read Only Memory. The distinctive feature of this encoder is small size, minimal power consumption and adaptability. In use, two applications are readily apparent. In the first, the system designer can maintain a stock of encoders without being committed to such details as formats, sampling rates, and the like. The second suggests an in flight computer link which commands format switching, or frame dwell, as a function of processed data.

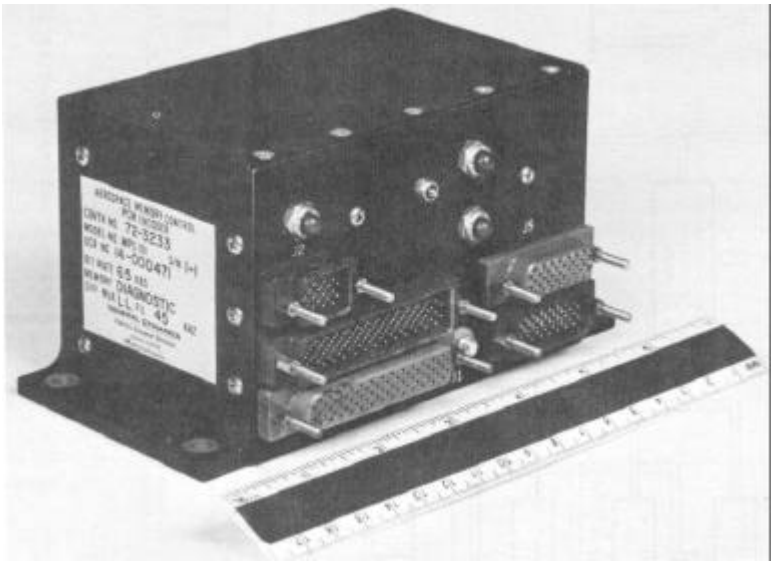

Figure 1. Aerospace Memory Controlled PCM Encoder, MPS-101

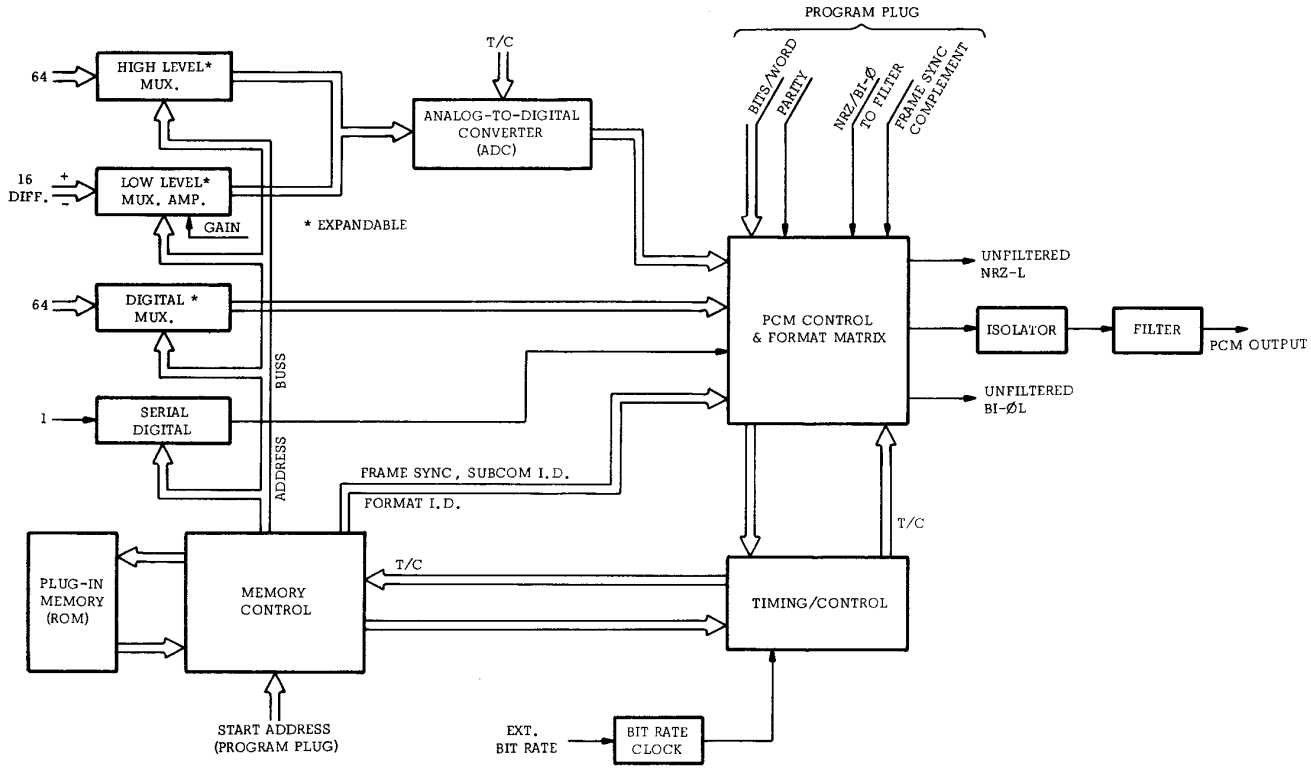

Figure 2. MPS-101 System Diagram

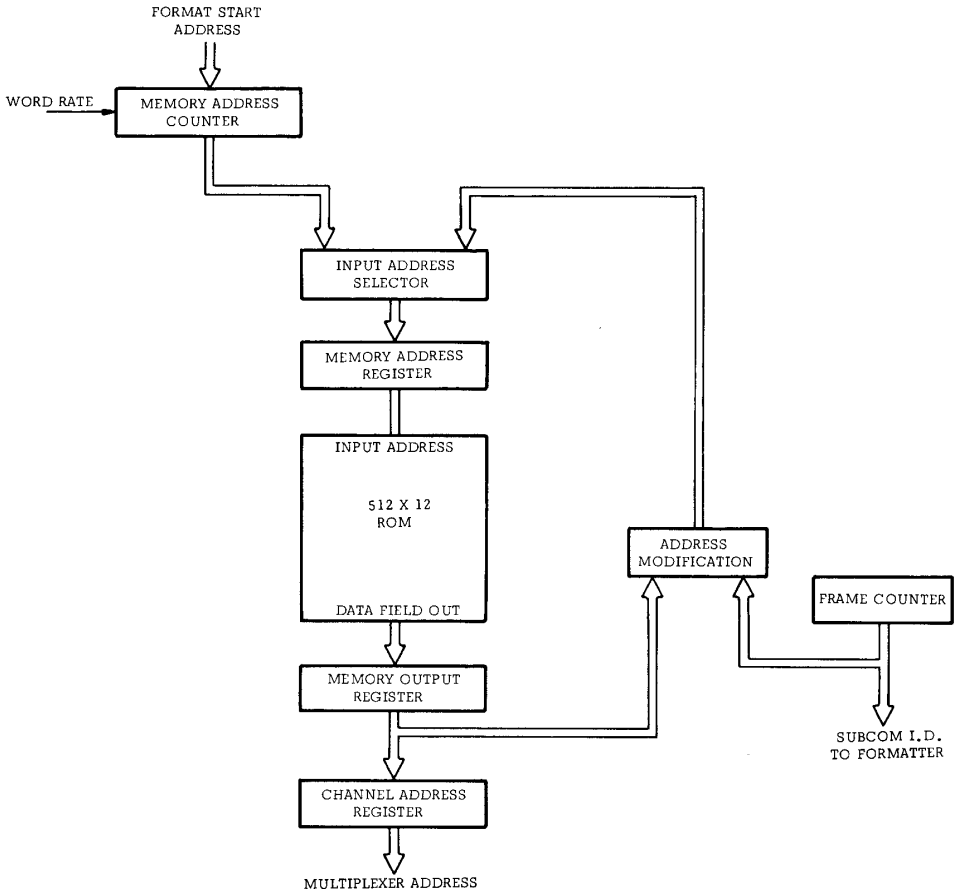

Figure 3. Memory Control