HARDWARE COMPRESSOR REDUCES COMPUTER LOADING

O.J. “Jud” Strock
Senior Applications Engineer
EMR-Telemetry
Sarasota, Florida

ABSTRACT

A hardware Compressor examines measurement data prior to computer entry, discards redundant or otherwise uninteresting words, and passes the appropriate information with tags to a computer. Continuous rates of 100,000 to 1,000,000 words per second are accepted. Under some conditions, 95% to 98% of the data can be discarded while passing every measurement which is of value in real-time analysis.

INTRODUCTION

In a real-time telemetry-computer system, the computer and associated storage media are generally the limiting factors in data throughput. This condition is brought about by the need to examine every data measurement, even though much of the data is not of immediate importance to a system user. Data examination in the computer under software control is a slow process, and after this time-consuming operation is completed on a given measurement most data is discarded as insignificant to the real-time analyst. If a measurement does not change meaningfully for several milliseconds (even several seconds), the use of the computer to examine each occurrence of the measurement imposes an unfair burden on the computer and contributes to overload and the related consequences. Even when throughput rate is not critical, compression is useful in reducing data storage requirements in a system.

For years, the technique of data compression has been considered by telemetry users, with the hope that some form of word-by-word data examination could take place prior to computer entry in order to discard uninteresting data and allow the computer to spend full time in analysis of important measurements. This is not to imply, of course, that discarded measurements are lost forever; generally a tape recording is made of all data, such that subsequent detailed analysis in relative leisure can take place.

Past proceedings of the ITC and other conferences during the past ten to fifteen years have studied the technique of data compression in great detail, and many studies have been made to determine the algorithms which are most desirable in real-time telemetry data processing. Many of the mnemonics used in this paper are adopted from previous publications and studies. This paper is not a study of the technique of compression, but is the description of a unique piece of hardware which implements the compression technique by high-speed hardware preprocessing.

Responding to the requirements of a telemetry user several years ago, EMR developed a hardware Compressor and integrated it into a system in order to reduce the volume of data being entered into the computer. That Compressor was updated for a second user, and a third, and so on as the years went by. New requirements were incorporated as new systems were developed, and finally, a standard Compressor was defined and put into production, reflecting the composite requirements of several users. This Compressor is described on the following pages.

System Application

A typical system application of the Compressor is shown in Figure 1. Data measurements are input from up to six asynchronous sources (such as PCM, PAM, PDM, FM, or analog data streams) and merged with time data. Each measurement is examined in accordance with a uniquely specified algorithm (or algorithms), and is output to the specified ports. Port 1 is usually a direct-memory access (DMA) computer input; Port 2 can go to the same or another computer, or to some other device such as an "array processor"; Port 3 is a special interface to computer memory or to a data display device.

The dynamic merge capability is especially important in many systems, where independently generated but functionally related data streams are to be entered into a computer.

Multiple output ports prove to be valuable also in many systems applications. The use of a hardware device to pre-sort data allows separate buffer areas in the computer to be allocated to separate data functions. Alternatively, two separate computers can be used with the Compressor pre-sorting data measurements for the computers as it merges, compresses, and distributes data words.

In the typical EMR system, a Digital Equipment Corporation (DEC) PDP-11 or VAX-11 computer is used, and data is entered into memory automatically via DEC's UNIBUS as shown, using a special EMR direct-memory interface channel without the need for program intervention. The Compressor is set up and controlled via UNIBUS; individual data measurements are processed in accordance with instructions stored in core memory in the Compressor.

Description of Hardware Compressor

Measurements are input to this Compressor via Ports 1 through 6, along with IRIG time (one millisecond resolution). When two or more measurements occur simultaneously, or when the momentary input rate exceeds the throughput rate of the algorithm processor, an input FIFO acts as a buffer.

Each measurement is assigned an interim identifier, consisting of the input port number and word number in the data frame (and location in the subframe or sub-subframe, if applicable), plus status bits. The identifier becomes a pointer to the 16-word algorithm processor which is unique to that measurement, and which contains all instructions and constants necessary to examine the data content each time the measurement occurs.

Figure 2 is a simplified flow chart which depicts the handling of measurements, and Figure 3 is a simplified block diagram.

Four major modes are possible under program control:

- Normal compression -- each measurement is examined and passed or rejected according to its algorithm test(s).

- Limited-group compression -- high-priority processing takes place; measurements which are not in the high-priority group are rejected without test.

- Force-one-look operation -- one output is made of each measurement (or of certain selected measurements), whether or not it passes the algorithm test.

- Throughput -- all data is output without test (not shown in Flow Chart).

Data Testing

Each measurement value is tested in accordance with the appropriately selected algorithm or algorithms (see Figure 4 and Table 1). If it fails the test(s), it is rejected; if it passes one (or more), it is output.

If the “status” bit in the algorithm process is set, a status word and tag are generated and output with the measurement. This word tells which algorithms were applied to the word, and why the data passed.

One bit in the algorithm process is an “alarm.” bit. If this is set for a given measurement, an alarm pulse is generated and output whenever the measurement passes the test(s). This can be used to trigger a computer interrupt, or to cause other action in the station.

Each passed measurement is accompanied by a “tag,” preassigned to identify the data to the computer. By selection of the tag content, programmers can make optimum use of this important tool.

Part of the algorithm processor information tells the Compressor where to send each measurement and tag (Port 1, 2, and/or 3). Ports 1 and 2 are conventional 16-bit outputs, while at Port 3 the data and tag are on separate sets of lines. Output formats are shown in Table 2.

Special Capabilities

Incidentally, the Port 3 output can be used with a special computer interface to set up a “most recent measurement” buffer in computer memory. Each tag contains the memory address (or relative address) where the accompanying measurement word will be stored; compressed data can then be accessed at known locations in memory.

A special “all-pass” capability at Port 3 enables the user to bypass the algorithm processor on specified words and output them at Port 3 (data and special “all-pass” tag on separate lines). In this operation, time and status are not output, nor is compiled data such as CSU or MMA. This port can be used with a special interface to operate a Word Selector, whereby each tag specifies the address of a device (analog output, discrete bit output, decimal display, or binary display) to which the accompanying data will be directed.

Time is merged with incoming data every millisecond (or every frame or subframe) in the typical system. Minor time (milliseconds) is input with a unique tag; every second (or subframe, etc.) major time (seconds/minutes and hours/days) is input.

CONCLUSION

In summary, capabilities of the Compressor are:

DATA COMPRESSOR, MERGER, DISTRIBUTOR, **PREPROCESSOR**
(12 algorithm processes - 8 other logical tests)

- Up to 3800 separate input measurements examined by individually-specified algorithms.
- Up to 6 input ports, merged with time port.
- Two output ports for entry into a 16-bit computer interface.
- Third output port for special 32-bit applications.
- High-speed inputs (burst rates of 1,600,000 words/second).
- Built-in data simulation capability for self test.
- Group addition or deletion by single command.
- Force-one-look capability for confidence checks.
- Dynamic updating capability from computer.
- Status word generated for output with data if designated.
- Alarm output when designated words pass.
- Memory load-verify from computer.

Table 1. Algorithm Processes

MNEMONIC	ALGORITHM DESCRIPTION	MICROSECONDS PASS	MICROSECONDS NO PASS
REJ	Reject without test.	---	0.9
THR	Throughput (pass) without test.	2.9	---
BMA or NBM	Bit Match or No Bit Match • Test the designated bits. Pass the word if there is: (a) Bit Match, or (b) No Bit Match. • Constants to be put into memory for each measurement: (a) Bits to be tested, and (b) Pattern to be matched by these bits.	4.6	3.3
ILI or OLI	In Limits or Out of Limits • Test the data value to see if it is within the designated limits. Pass the word if it is: (a) In Limits, or (b) Out of Limits. • Constants to be put into memory for each measurement: (a) Upper Limit, and (b) Lower Limit.	4.6	3.3
BCH	Bit Change • Compare the word with its most recent value on a bit by bit basis. Pass the word if there has been a Bit Change in the bits of interest. • Constant to be put into memory for each measurement: Bits to be examined.	5.4	3.3
ZFN	Delta • Compare the data value with its last output value. Pass the word if the value differs from the last output value by more than a specified delta (change). • Constant to be put into memory for each measurement: Allowable delta.	5.4	3.3
DSL	Delta Slope • Compare the data value with its last output value to determine the sign (direction) of the slope. Pass the word if the sign has changed since the measurement has last passed, and the level has changed by more than a specified delta. • Constant to be put into memory for each measurement: Allowable delta.		
NSE	N-Sequential • Discard "N-1" Sequential occurrences of the measurement, then pass the measurement once. • Constant to be put into memory for each measurement: "N."	5.3	3.6

MNEMONIC	ALGORITHM DESCRIPTION	MICROSECONDS PASS	MICROSECONDS NO PASS
MMA	Maximum-Minimum • For a period of "N" occurrences of this measurement, store the accumulated Minimum and Maximum values. Output these extremes, reset, and repeat. • Constant to be put into memory for each measurement: "N."	7.4	5.8
CSU	Cumulative Sum • For a period of "N" occurrences of this measurement, accumulate the sum of the data values. Output this Cumulative Sum as a double-precision (32-bit) word, reset, and repeat. • Constant to be put into memory for each measurement: "N."	7.4	5.8

Chained algorithms require less than the sums of their individual processing times. Words in a subframe require 0.84 microseconds longer for processing; words in a sub-subframe require an additional 1.68 microseconds.

Table 2. Data Output Formats

Passed data is output as:

- Tag (16 bits)
- Data (16 bits)

Where Status has been designated as an output, it precedes the data as follows:

- Status ID (16 bits)
- Status (16 bits)

In the algorithm tests which involve more than 16 bits of data output (MMA and CSU), the format is:

- Tag (16 bits)
- Maximum value (MMA) or least-significant 16 bits of sum (CSU)
- Tag (16 bits)
- Minimum value (MMA) or most-significant 16 bits of sum (CSU)

When data relating to one of these two algorithms (MMA or CSU) is “forced” out, it does not represent the true conditions which are defined for the algorithm; therefore, a special output is generated:

- Tag
- Number of samples (instead of “N”) followed by the tags and data words (above)

When the “DSL” algorithm (Delta Slope) causes an output, the most-significant bit in the data word is the sign of the slope (O = negative, 1 = positive).

Every millisecond or frame interval (if data has been output within the past millisecond), a minor time word is output:

- Time ID (16 bits)
- Milliseconds (16 bits)

Every second or subframe interval, major time is output:

- Time ID (16 bits)
- Seconds, minutes (16 bits)
- Time ID (16 bits)
- Hours, days, or ID (16 bits)

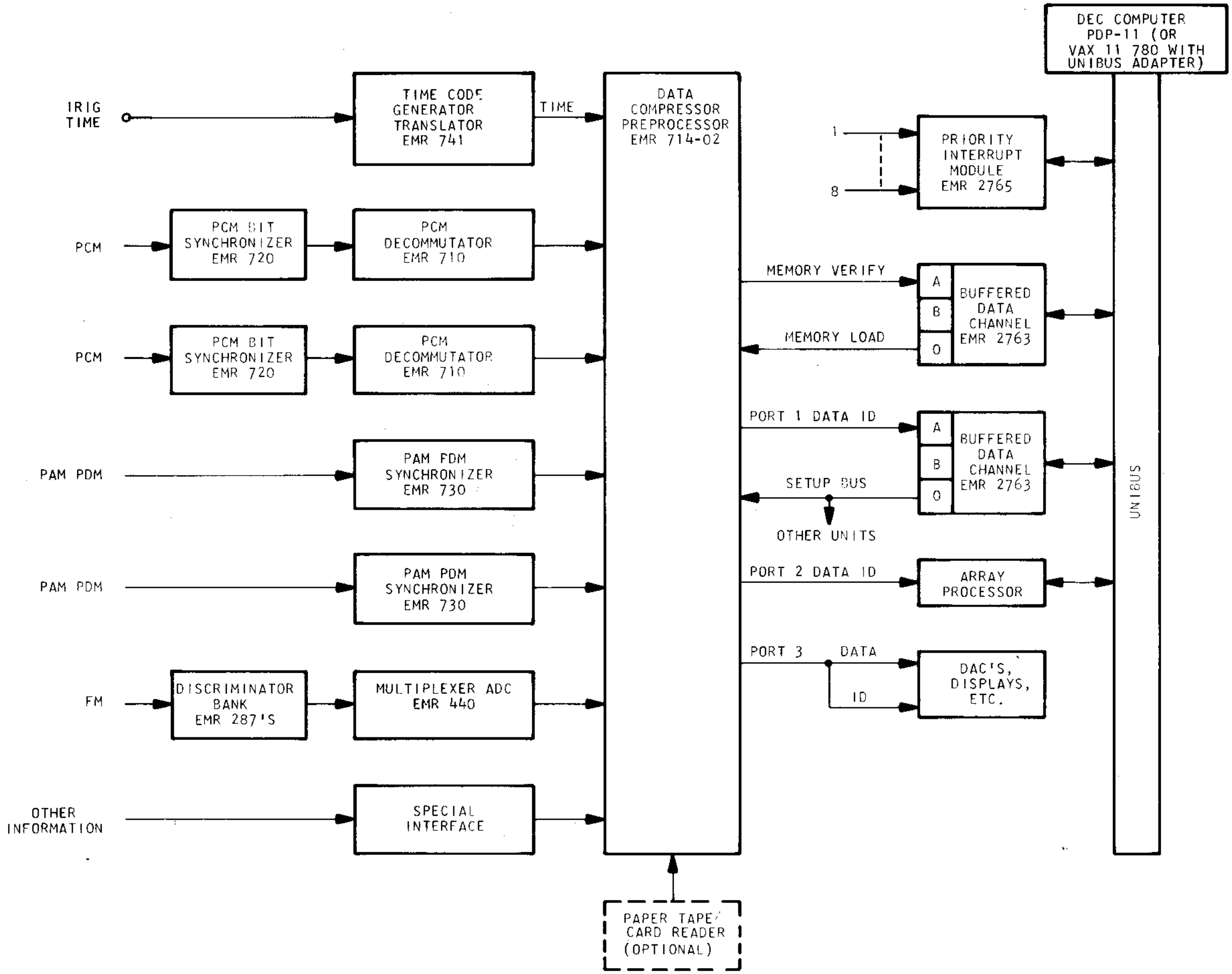

Figure 1. Block Diagram

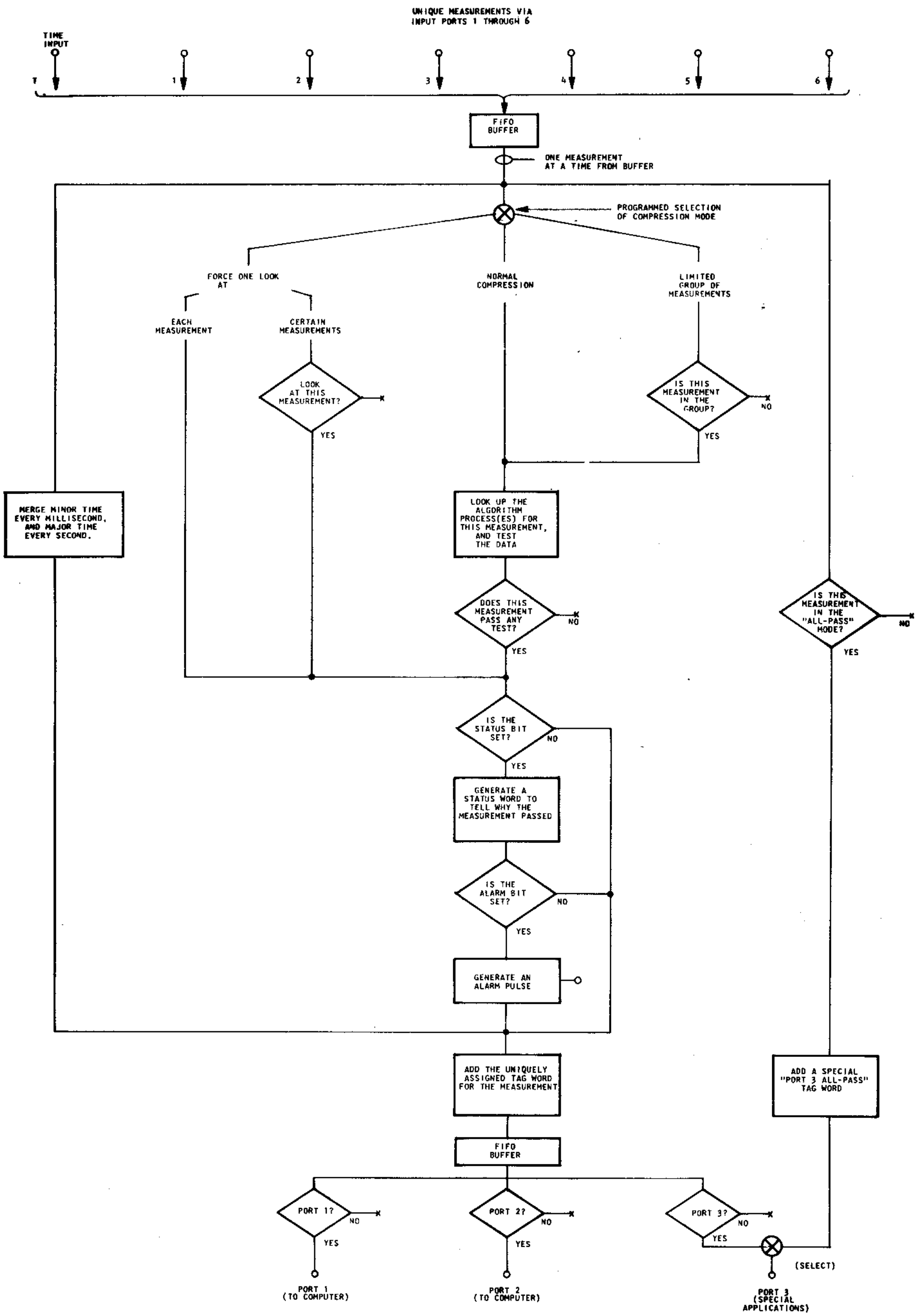

Figure 2. Simplified Flow Chart

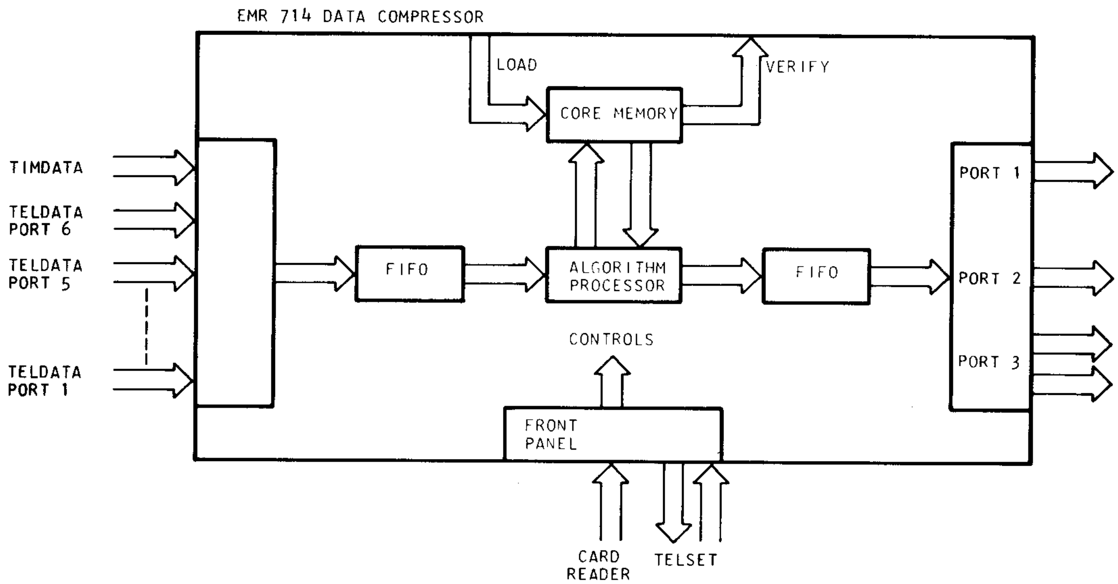

Figure 3. Data Compression Subsystem

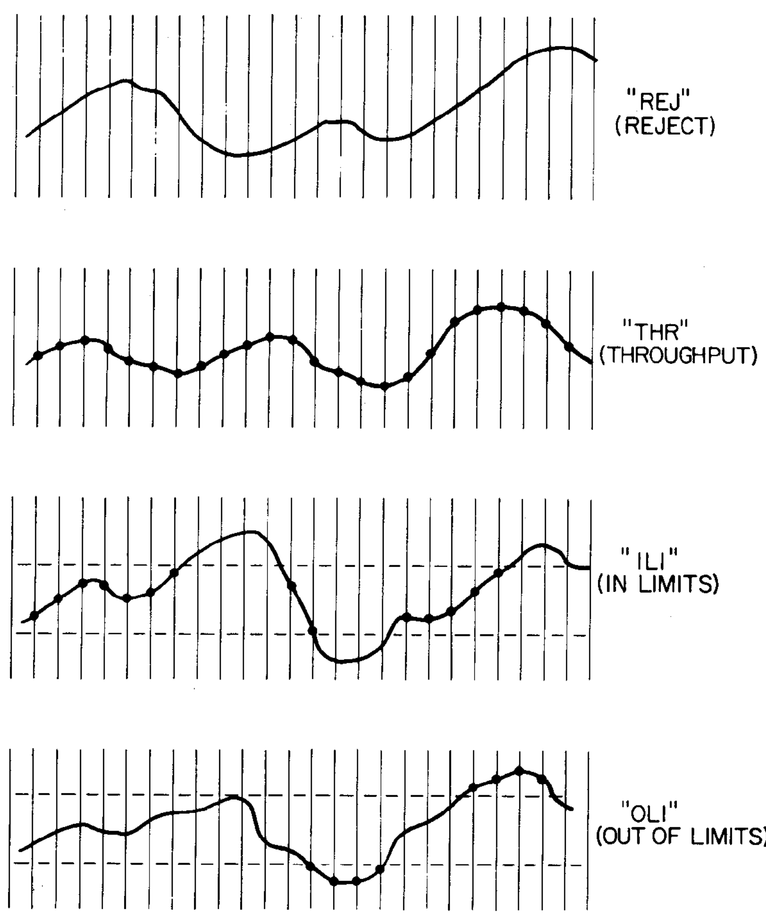

Figure 4A. Algorithms

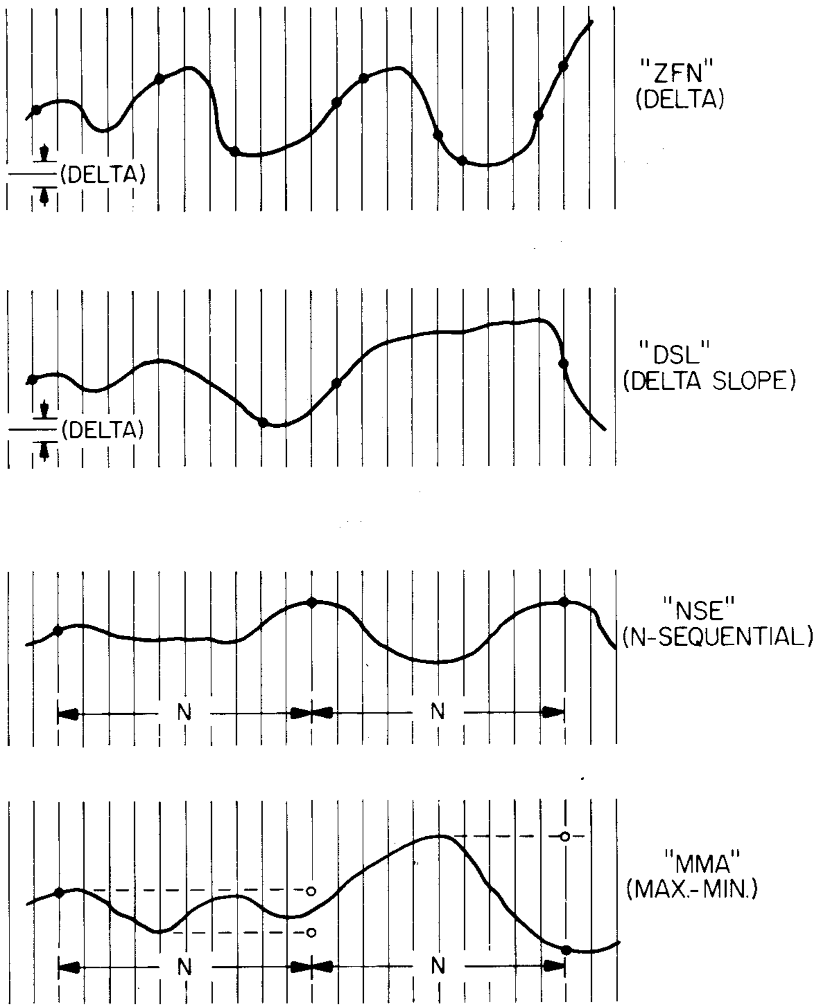

Figure 4B. Algorithms

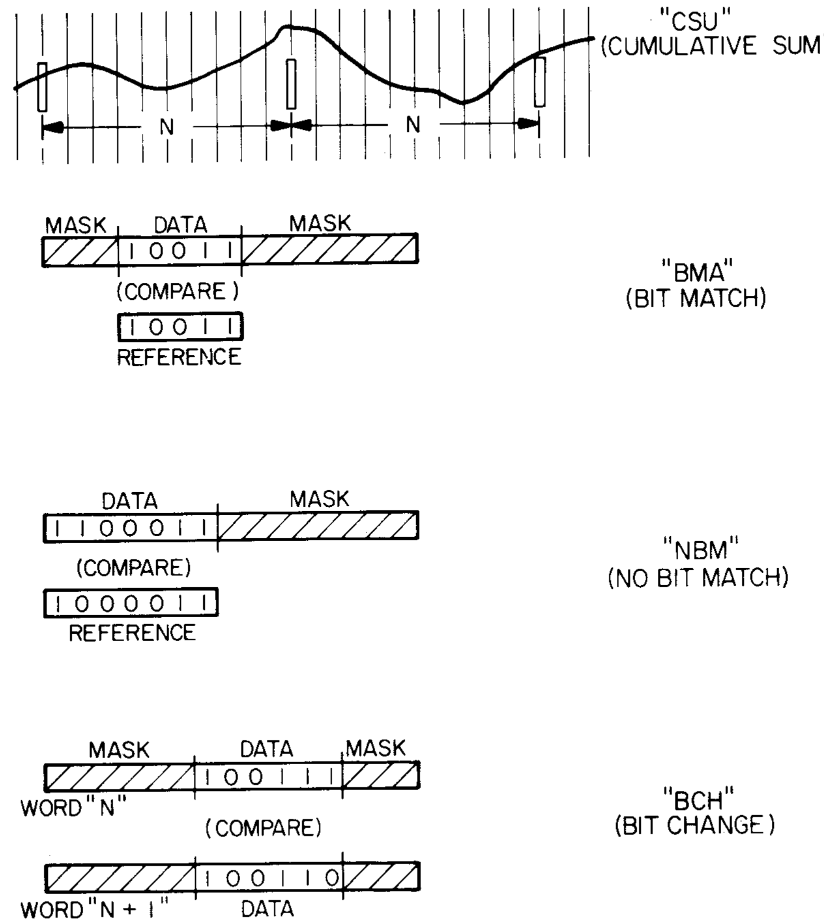

Figure 4C. Algorithms