F. A. Heinz

NO. ASTIR-TM-23-2
REV. NO.
PAGE 1 OF
DATE

DATA MANAGEMENT STUDY

EVENT DETECTION

Prepared by A. Bedford
A. Bedford

Approved by R. Wallace
R. Wallace

TABLE OF CONTENTS

FIGURES

SUMMARY

This report discusses the on-line processing of the data output by Space Scientific Experiments whose prime function is to detect science events which occur in the physical universe. Typically such experiments generate the greater part of the useful information which they observe following an event, and relatively little useful information in the periods between events. The objective of the data processing described in this TM is to allow event information to be displayed in real-time so that the science return from the operational experiments may be maximized.

To complete the analysis of a scientific event, three stages of data processing are normally required:

1. Event detection;
2. Event data reduction; and
3. Event classification.

As the processing progresses through these stages it becomes increasingly specialized. The purpose of this memo is to identify those elements of the processing which are of universal or at least of broad, applicability.

The processing of data channels in which the event is represented by a single bit change, analogous to a contact closure, is considered first. Despite the simplicity of such data channels, several of the most important concepts in event detection can be clarified by studying the processing which is required to handle these channels.

The next case we consider is the detection of simple changes in analog sensor outputs. Processing techniques which permit the science event to be distinguished from the channel noise level are described.

A complete event detector which will detect and analyze simple events using the processing techniques previously considered is then presented. Finally the use of this detector with sensors which generate complex event transient waveforms is considered, and some methods whereby its usefulness may be enhanced by usage in a compatible operational environment are reviewed.

INTRODUCTION

Scientific experiments may be considered in two broad categories, those concerned with the measurement of physical (or other) parameters, and those intended to detect physical events. Parameter-measuring experiments are intended to refine our knowledge of some specific attribute of the real universe, in principle by making "one-point" measurements. Due to the complexity of physical systems, the parameter of interest may be a function of other parameters of the system, e.g. its structure, and the measurement of observable quantities over a period of time is often required to permit the system to be adequately analyzed. An ALSEP experiment in this category is the Lunar Heat Flow Experiment, the HFE.

The significant feature of the parameter-measuring experiments for our present discussion is that the observations can be made on a pre-planned basis. In contrast, event detecting experiments only produce useful data in response to some change in the physical system which is under observation. A typical experiment in this category is the ALSEP Passive Seismometer Experiment, the PSE. Data processing for the event detecting experiments is inherently sophisticated, normally three stages of processing are required.

1. Event detection,
2. Event-data reduction; and
3. Event classification.

As the processing progresses through these stages it becomes increasingly speciallized. The overall processing for each experiment is inevitably special-to-type; the purpose of this memo is to discuss and to identify those elements of the processing which are of universal, or at least of broad, applicability.

The responses of different experiments to single events differ widely in their characteristics. The response may range from a simple bit-change, e.g. analogous to a contact closure, to a complex multiple burst waveform e.g. as is characteristic of seismic events sensed by the PSE. In order to clarify the event-detection concepts which are the subject of this TM, we will discuss first the detection of simple bit-change events, and then the more complex cases in succeeding sections.

Reference: ASTIR/TM-23-1 "Data Management Study", 1974

The simple bit change event introduces the concept of time series of events, and of simultaneity between the response in different downlink channels. It also introduces the concept of a "valid" or "spurious" response, i.e. the significance of noise in the received bit stream.

The next case we consider is the detection of simple changes in analog sensor outputs. This introduces the concept of statistically stationary processes, and their categorization from the sensor output data. Correspondingly, an event is defined as a perturbation of a stationary process.

Finally, we consider complex events of the type sensed by the PSE. These can be regarded as either single complex events or as multiple simple events. Specifically, the problem of defining the end of an event must be considered.

The event detection scheme which evolves from this discussion consists of statistical tests on the data as sampled by the downlink telemetry. The purpose of the tests is described in this TM and suitable test algorithms are presented. The mathematical basis for these algorithms is not considered herein, nor is any attempt made to establish their efficiency. These considerations will be explored in a future TM.

THE BIT-CHANGE EVENT. EVENT TIME SERIES AND VALID RESPONSES.

The simplest type of event to detect in the downlink is a change in a channel which is telemetred by a single status-bit. Despite the simplicity of such data channels, several of the most important concepts in event detection can be clarified by studying the processing which is required to handle these channels. Event detection is a function of the processing, the fact that status telemetry is more commonly used for housekeeping data rather than science data is not relevant.

An event is a change in the data, so that it is specified by:

1. The direction of the change; and
2. the time at which it occurred.

Essentially events occur in a time series, and the object of event detection data processing is to store or to record the individual events to construct the time series. The most direct way to do this is to prepare lists of the direction of the change and the time of occurrence. The time may be saved either as real time, days/hours/minutes/seconds, or as relative time such as a telemetry frame count.

The simplicity of the data is the status-bit channels greatly simplifies the processing which is required to detect the events and to prepare the event list. Comparison of a newly received status-bit with its previous value is a sufficient test, with a list entry being made on a change.

The event data reduction which may be subsequently performed on this list is one or more of the following four types of analysis:

1. Correlation of event times with events detected in other data channels;
2. Auto-correlation of the event series with previous series on the same channel to detect repetitive patterns of events in the data, e.g. for ALSEP data with previous lunar days;
3. Spectral analysis to determine event rates and their frequency distribution; and
4. Determination of spurious events, i.e. downlink data errors masquerading as events.

The choice of the type of data reduction which is performed is dependant on the significance of the events which the data channel can record. This involves considerations which are beyond the scope of this TM, and so event correlations and spectral analysis will not be discussed further.

Establishing that the data channel has exhibited a valid response to an event is of more general significance. A status-bit channel may produce a spurious level due to either a "glitch" in the experiments logic or a "hit" on the downlink. In normal operation, status-bit errors rarely occur, error rates between 1 in 10^4 and 1 in 10^5 are typical. At first glance, it might seem that these infrequent errors could be ignored. This is not the case, normally the actual event rate is also very much less than the bit rate so that the ratio of valid to spurious responses typically is fairly low. The method of testing for a spurious response in a status-bit channel is obvious, if a pair of events is detected in adjacent downlink frames we can assume that the first detected event is a glitch and the second one is the recovery of the channel from the glitch.

This test for spurious responses is implicitly based on some restriction on the time interval between valid events in the physical universe. If the downlink data rate is not substantially higher than the maximum event rate we cannot reliably distinguish between valid and spurious responses. Status-bit errors are due to noise, it is only the difference in distribution of the noise voltage from the familiar analog Gaussian noise which causes us to refer to them as glitches. Hence the effect of noise on a status-bit channel is to limit the maximum event rate which can be reliably handled by the channel. In subsequent sections we will discuss other ways in which noise limits the event detecting capability of analog sensor channels.

Summarizing, the processing which is required to detect events in status-bit channels is simple. If consists of time-tagging all changes observed on the channel and saving lists of the time tagged events for further event-data reduction. One type of reduction is of general applicability, it is the recognition of spurious responses in the channel which generate event-pairs in adjacent downlink frames. Other types of event-data reduction may be implemented as dependent on the significance of the physical events monitored by the data channel.

ANALOG SENSORS. STATIONARY PROCESSES AND NOISE

The majority of scientific experiments use analog sensors to acquire the science data. The analog output voltage is subsequently digitized and downlinked by the telemetry set. In this section we consider event detection, but restrict the discussion to those channels which generate a simple waveform in response to a science event, either a step change of level or a single output pulse. Clearly event detection cannot be achieved by testing any single bit in the downlink; we must be able to define the quiescent data in the channel in the absence of events before we can construct a test to detect events.

A quiescent channel contains data characterized by a constant mean level plus superimposed noise. Whether the noise originates on the science input, or whether it is additive noise from the sensor or associated amplifiers, is not distinguishable in the downlink data channel and does not affect the event detection processing. The telemetry data consists of a stream of numerical values representing the quantized output from the experiment sampled at regular intervals. Event detection is performed by numerical analysis of the data stream. No assumptions are made, nor in fact are any assumptions required, as to the scientific basis of the observed data. Relating the observed data to scientific events is the object of a second, and more speciallized, stage of data reduction.

Strictly, the discussion of analog sensor data processing in this TM is restricted to linear sensors. Special pre-processing of non-linear sensor data would be required prior to the event detection processing discussed herein.

The object of event detection schemes is to automate the first process which a lunar observer performs if presented with a strip-chart record. The observer would rapidly scan the record looking for changes in the data, and would mark the changes to identify the events. Subsequently he would study the events in detail, ignoring the long periods of quiescent or static data between events.

Statistical processing and testing has been extensively developed for many purposes, in particular for quality control of production lines. There is a very close analogy between the detection of a science event and the detection of a deviation of a production process which results in the production of substandard parts. The basic quality control procedure consists of measuring one, or more, parameters of a number of parts, i.e. of a sample. The mean value and standard deviation of each parameter for the sample are then computed. If the process is operating stably, the means and standard deviations of successive samples will only vary within narrow limits. Loss of control of the production will be indicated by either or both the mean and standard deviation of the new sample falling outside the nomal process limits.

For the purpose of event detection, an experiment on a physical system can be regarded as a process which produces one, or more, streams of numerical values. Each stream is analyzed to determine whether a new sample is a continuation of the previous observations, or whether it is significantly different to the previous samples on a statistical basis. In other words, whether the process generating the telemetry data is stationary or whether it is changing due to science events.

There are significant advantages in automating event detection in addition to the operational advantages such as improved operator response time and the possible identification of significant science events in real-time which it provides. Numerical analysis of the telemetry data overcomes the resolution problems which are inherent in every analog display. The scaling of analog displays, both for the Y i.e. parameter value, and the X i.e. time, axes is always a compromise. The compromise is between the sensitivity to display small and large signals, or fast and slow signal changes. If the sensitivity to display small fast changes is increased, large signals overload the display and are lost, and may even damage the display mechanism, and the ability to recognize slow variations of the level is lost. In contrast, numerical analysis inherently possesses a resolution of 1 bit over the full range of the downlink data, and time-wise can compare either adjacent telemetry word values, or the value of words separated by indefinitely long time intervals.

The basic statistical parameters which are used in all event detection tests are:

The mean level;

The standard deviation;

The sample size;

The test limit(s) for variation of the mean level; and

The test limit for variation of the standard deviation.

Different test algorithms can be constructed by choosing different values for the last three parameters listed above. The performance of the different test algorithms will vary in two ways:

1. Their response to events which are barely detectable above the system noise; and

2. The response time between an input event and the detected output printouts.

Science signal inputs which are much larger than the system noise level can be reliably detected by almost any test algorithm. The problem that arises with small signal inputs is that either the test fails to recognize the signal above the noise level, or alternately reports a spurious noise peak as a science event. The analogue to this proglem in quality control testing is buying-off faulty parts as normal production, known as "the consumers risk"; or rejecting normal parts without valid cause, known as "the producers risk". The overall performance of the test, i.e., probability of parts rejection versus true standard deviation of the production process is known as "the operating characteristic" of the test. The statistical operating characteristics of various test algorithms have been extensively studied. They are of significance if we wish to implement an optimal event detector capable of pulling the smallest detectable event "out of the mud". This topic is considered to be too specialized to pursue in this TM, it must be recognized that signals within a few db of the system noise level cannot be detected with 100% reliability. On the other hand, as the signal-to-noise ratio increases, the reliability with which events can be detected rises very rapidly indeed. Event signals +10db above the noise can normally be detected with 99% reliability.

The response time of the test algorithm is principally affected by the number of data words which are used to form a sample, i.e., the sample size. Obviously, we can only obtain one printout from the test for each sample which has been processed, so the use of a very large sample will reduce the sampling rate, perhaps to a value less than the science event rate. This problem is of little significance in normal quality control testing, but is of major importance in devising useful event detection algorithms. Effectively, we must "match" the algorithm to the anticipated science event transient waveform. In order to illustrate the influence of the fundamental requirement for event detection on the choice of the test algorithm we will first discuss the detection of step output responses from the experiment, separated by long time intervals.

Such an event could be described by:

> "Channel CHN changed from level Y_1, S. D S_1 to level Y_2, S. D S_2 at time HR/MM/SECS."

For an event detector to produce meaningful messages of this type, it is necessary for it to have the following capabilities:

1. Evaluation of the mean value and S. D of a sample on-line. This renders the test self-initializing.

2. The ability to determine a representative mean value and S. D from a single sample. Otherwise, the values for Y_2 and S_2 would creep and several printouts produced rather than the single message which is desired.

3. A fast response so that the reported time is a useful approximation of the actual event time.

4. The ability to distinguish between step changes (events) in the data and slow variations (drift) of the mean level.

In general, it is not possible to devise a single test with a fixed sample size and test limits which provides all the capabilities enumerated above. An "intelligent" test algorithm is required which will provide each capability at the necessary time in the operating sequence. A suitable sequence is:

1. Evaluate the mean and S.D of a sample. The sample size will be pre-determined; it is choosen so that it is:

   (a) small enough to be acquired with a high probability of excluding all events,

   (b) large enough to derive representative values from a single sample. With a quantized bit stream, a precision smaller than one-half of the least significant bit is meaningless, even for the S. D. The optimal sample size is a function of the noise bandwidth of the channel which is normally determined by the analog electronics up-stream of the analog to digital convertor. Small sample sizes of the order of 10 to 20 are adequate unless the channel bandwidth has been heavily filtered, i. e., if the data rate is many times greater than the frequency response of the channel.

2. After determining an initial mean and S. D switch the test to a variable sample size test. In such a test, the sum of the squares of the deviation of each data point from the mean is computed. Each data point will, on the average, increase this sum by the variance for a stationary process. This sum is compared with upper and lower limits which are both increased for each new data point. If the sum eventually exceeds the upper limit we have detected an event, if the sum falls below the lower limit, we have positively demonstrated the absence of an event. In either case, the sample is considered to have been completely analyzed, and the acquisition and testing of a new sample is initiated. This automatic termination by disposition of

the previous sample leads to a very fast response, a step change in the data can typically be detected within two or three data points after the step.

3. The event detector would normally run in the variable sample size test mode between events. Immediately after detecting an event, it would return to the fixed sample size mode for one sample to re-initialize the processing; and also printout the event message after it has acquired all the relevant data.

4. To prevent problems due to drift of the mean signal level, another processor loop is required which will update the mean level used in the variable sample size test. This loop can as an option report its activities in detecting and updating the best estimate of the mean level by suitable printouts.

It will be apparent to the reader that the step response event detector described above will correctly and separately process each edge present in the signal provided that these are separated by an interval which exceeds the initializing fixed sample size. "Pulse" event signals are shorter than this initializing period. A pulse event could be described by:

> "Channel CHN pulse, root mean square (RMS) amplitude E, length T at time HR/MM/SECS."

In order to detect and evaluate a pulse event in these terms, we could use the step-response detector to locate the leading edge of the pulse. If now instead of re-initializing the detector with fixed sample procedure, we continue to perform the variable sample size test using the initial pre-event mean and S. D within the test, during the event pulse the test will fail every two or three data samples. Eventually, after the signal level has returned to the pre-event values the variable sample size test will detect this also. By storing the start time for each test sequence, a value for the pulse length T can be determined, as the difference between the start of the first "pass" variable sample test and the detected pulse leading edge time from the first "fail" variable sample test of this sequence. Deriving the RMS pulse amplitude is simply a matter of taking the sum of the squares of the deviations, which is already being computed in the variable sample test, over the entire interval and then calculating the RMS value.

By combining the step-response and the pulse event detectors, we can provide a "universal" detector which will handle in a meaningful way the great majority of events. To achieve the desired performance on this combination of signal waveforms, we must extend the processing which has been described above to include logic to recognize whether a step or a pulse event has been detected. The basis for this type-of-event recognition scheme is:

1. Step event. Re-initialize mean and S. D with fixed sample size test. If the next variable sample test using the new mean produces a pass, then a step has been recognized.
2. Pulse event. Continue variable sample test with old mean and S. D. If a pass is eventually detected, then a pulse has been recognized.
3. It is inconceivable that a transient which exceeds the noise level could be recognized as both a step and a pulse. On the other hand, it is possible that some transients could not be recognized as either. In this case, we have little option but to generate a printout such as:

   "Anomalous transient detected at HH/MM/SECS.
   System re-initialized".

FUNCTIONAL DESCRIPTION OF THE PROPOSED EVENT DETECTOR

The description of the analog sensor event detector which has been developed in the preceding section will now be summarized. The system may be implemented using the elements identified in the Functional Block Diagram, Figure 1. The elements are:

1. Initialization;
2. Edge detector;
3. Drift corrector;
4. Step-event recognition and evaluation;
5. Pulse-event recognition and evaluation;
6. Printout control;
7. Sequence control of all other elements.

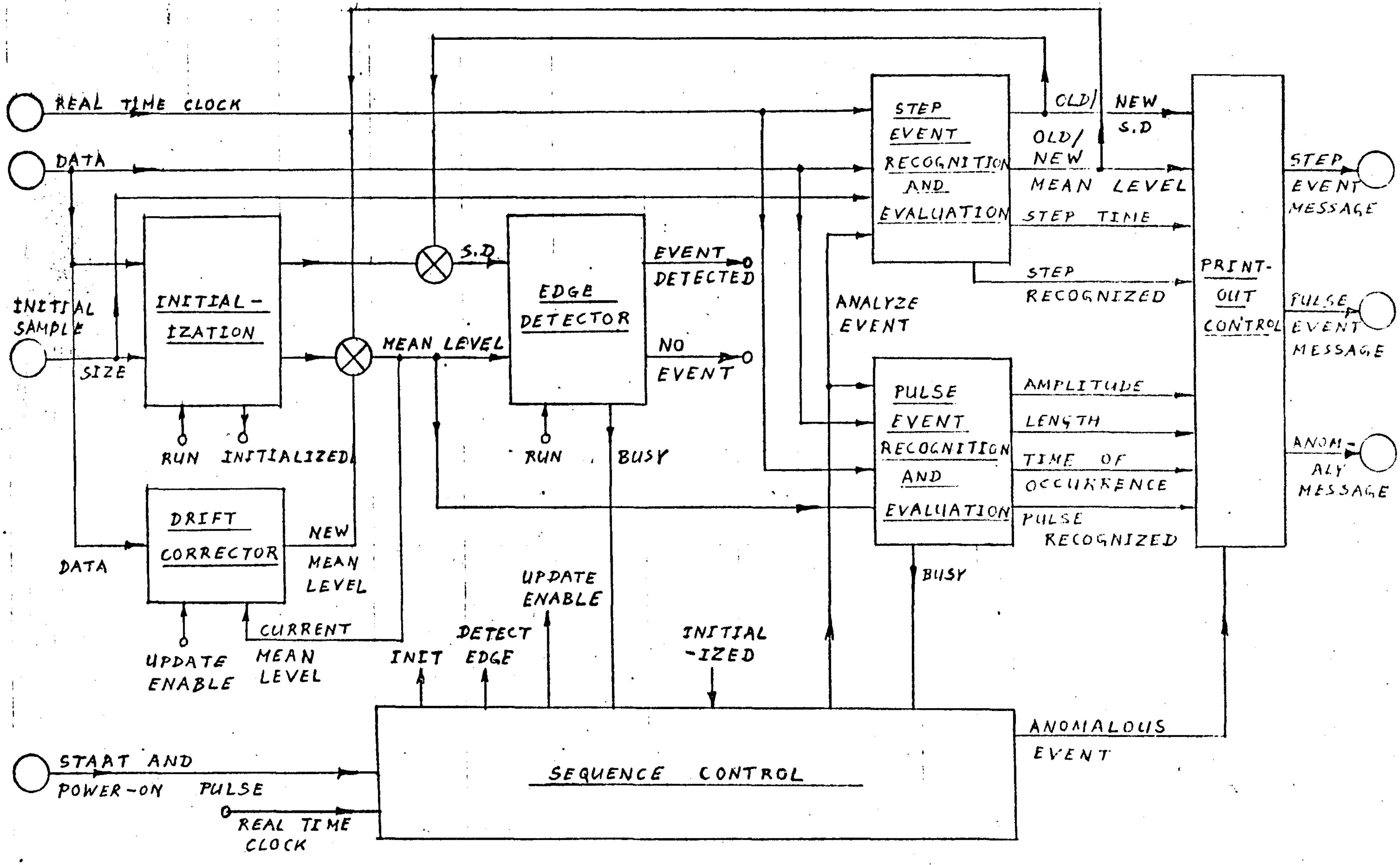

Figure 1. Functional Block Diagram of the Event Detector

Each element should be regarded by the reader as a separate mechanism. These mechanisms could, in principle, be constructed using 54/74 TTL IC's, micro-processors or any other form of logic element. The fact that in most applications the elements will be realized as computer programs may be disregarded in this description. It will also be obvious that there is considerable overlap between the functions of the various elements, and a more sophisticated control element than described herein could reduce the overall complexity. This too should be ignored by the reader, both these topics concern the implementation rather than the function of the event detector.

Similarly, the reader is advised to overlook the limited depth of detail in the description of the statistical testing performed in the various elements. The statistical background of the tests is discussed in many standard textbooks; this is a matter of design which can be resolved as soon as the required functions of each element to achieve event detection have been identified.

The external inputs to the event detector are:

1. The channel data;
2. A real-time clock;
3. A "start" or "power-on" pulse; and
4. The number of data points, i.e., sample size, to be used while initializing the processing.

The inputs to the initialization element are:

1. The channel data;
2. The required initialization sample size; and
3. A run signal from the sequence control.

The initialization element outputs are:

1. A mean value;
2. S.D for the channel; and
3. A status signal to the sequence control, "processing initialized".

The edge detector inputs are:

1. The channel data;
2. A mean value to be used in computing the sum of the squares of the data point deviations;
3. A S. D to be used in computing the test limits; and
4. A run signal from the sequence control.

The edge detector outputs are:

1. A busy signal;
2. A "pass", i. e., no-event, status signal; and
3. A "fail", i. e., event detected status signal.

The drift corrector has three inputs:

1. The channel data;
2. The current mean level; and
3. An "update enable" signal supplied the sequence control following receipt of an edge detector "pass" status signal.

The only output from the drift corrector is a mean level update when required. An optional update printout could be provided.

The inputs to the step-event recognition and evaluation element are:

1. The channel data;
2. The real-time clock;
3. The initialization sample size; and
4. A run signal from the sequence control.

Its outputs are:

1. A new mean level;
2. A new S. D;
3. A busy signal;
4. A "step recognized" status signal; and
5. The step time-of-occurrence.

The inputs to the pulse-event recognition and evaluation element are:

1. The channel data;
2. The current mean level;
3. The real time clock; and
4. The same run signal supplied to the step-event recognition and evaluation element.

Its outputs are:

1. The RMS pulse amplitude;
2. The pulse length;
3. A busy signal;
4. A "pulse recognized" status signal; and
5. The pulse start time.

The printout control element is directly driven from the step-event and pulse-event recognition and control elements. Its only additional input is an anomalous event signal generated by the sequence control if neither a step-event or a pulse-event can be recognized within an appropriate time of the edge detector locating an event start.

All the status signals described above, together with the external "start" pulse and the real-time clock are supplied to the sequence control element. It must decode all these in order to generate the required "run" signals to all the other elements.

The signal processing which is performed in each element in order to generate its outputs has been described in the previous section of this TM. A more detailed description of the required signal processing will be the subject of a further TM.

THE ANALYSIS OF MORE COMPLEX EVENT WAVEFORMS

The event detector which we have just described can provide all the information which is required to describe simple step-events or pulse-events. If the science event results in a more complex transient waveform on the data, additional event descriptor information could be extracted by further processing. There are two basically different classes of additional processing:

1. Processing of the channel transient data generated by the event; and

2. Processing of the event detector output messages.

The purpose and objectives of this additional processing are inevitably more specialized, i.e. more experiment dependant, than the use of the event detector. The objective of the discussion in this section is to describe the conditions under which, and some methods whereby, the usefulness of the basic event detector may be enhanced by usage in a compatible operational environment.

First we note that although the pulse-event recognition circuit was described above for the case where the input transient waveform approximates a rectangular pulse, this element will in fact measure the duration and RMS amplitude of any transient which returns to the baseline regardless of the actual transient waveform. For some experiments this may provide sufficient data for the PI to distinguish between different types of events which produce consistently different waveforms.

Some experiments can produce very complex waveforms such as multiple heavily damped sinusoidal wavetrains; seismometers for example can produce this type of transient. It is often difficult to decide whether such transients are better described as a single long complex transient or as multiple simpler transients. The pulse event detector will provide an arbitrary but consistent definition, as it is searching for the end of the transient in order to determine its duration. Once the signal returns to the baseline for a sufficient period to "pass" the end-of-transient test, the pulse event detector defines all the previous perturbed data as forming a single transient.

In the most complex situations it is necessary to perform "signature analysis" to identify the physical cause of an observed transient. Such specialized processing is beyond the scope of this TM, but the reader should note that the availability of the event recognition times from the event detector greatly simplifies the problem of recalling the transient data for signature analysis.

Processing the event detector output messages to distinguish event types and event patterns is probably of more general applicability than signature analysis. This type of event data reduction has already been considered in this TM in the context of bit change events. Similar considerations apply to the analog sensor events, with the added complexity that the analog sensors may well be detecting different types of events due to different physical phenomena.

Much useful event data reduction could be performed if suitable facilities to sort the event detector output are provided. If the event detector output data is stored in a computer compatible form, for example on disc, then by recalling the output data the following functions could be implemented:

1. Display of any previous event data from one or more experiments;
2. Correlation of events detected in more than one data channel or by more than one experiment;
3. Searches for events of a given type and for repetitive patterns of events.

All these searches could be performed much more easily, and could be performed automatically, if sort routines are provided. By sorting we mean rearranging the event messages so that they are placed in some special order, such as the smallest amplitude event first and then in sequence through to the largest amplitude event last. This provides a simple procedure for bringing all like events together, in the case just consider all events of the same amplitude would appear together in the final event list. Successive sorts by time of occurrence, amplitude, duration, etc. can be used to establish correlations and event patterns. The author strongly recommends that a sort facility should be implemented with, and should be regarded as an integral part of, the event detection facility.