# ALGORITHMS FOR REAL-TIME DATA ACQUISITION USING A SMALL MICRO-COMPUTER: A CASE STUDY FOR THE RADIO WAVE PROPAGATION STUDIES

A. Sharaf Eldin, M.A. Alhaider and A.A. Ali

College of Engineering, King Saud University
Riyadh 11421, Saudi Arabia

## ABSTRACT

A micro-computer-based real-time data acquisition system has been developed and implemented in order to study the radio wave propagation parameters. The main features of this system are: (i) Menu-driven easy to use system, (ii) Optimization of the use of the limited computer resources, (iii) Different scanning modes and rates are supported, (iv) On-line real-time tracing could be done under user control; and (v) On-line help facility is provided. The developed software is explained in greater details including the designed algorithms for the computation of the maximum possible scanning rate and the optimization of the cassette tape utilization. The same micro-computer is used also for normal data processing functions but not simultaneously with the data acquisition activity. Such processing includes data retrieval, statistical analysis and data abstraction. Several system management functions are also provided to ensure trouble free run of the system. Using this system it is possible to reach reliable scanning rates up to 17 ms. Such fast rate is needed for monitoring and recording fast variations of the received radio signal during fade events.

## 1. INTRODUCTION

The propagation of electromagnetic waves into the atmosphere is affected by refractive fading scattering and absorption by various hydrometers and by sand and dust particles. At frequencies above 10 GHz, the excess attenuation due to rain becomes a critical design para meter. Only short hops of few kilometers length will be able to meet a reasonable reliability objective. Such is the case, however, in regions of moderate and high rainfall. For dry areas such as deserts and arid regions, excess attenuation due to rain is not so severe, even at millimetric wave lengths. Hence, longer hops can be operated with acceptable reliability.

Unfortunately, such arid areas are frequently hit by sand and dust storms lasting for few minutes or several days. The propagation of radio waves into sand/dust storms has recently become the subject of study by many researchers.

A field study, aimed at studying the propagation of millimetric radio waves in arid land is presently run by the authors. The study involves the continuous monitoring and recording of the received signal from five radio links working at millimetric wave length. The radio transmitters are placed at separate locations, while the receivers are all housed at the data receiving and processing site. Along with the signals from radio links, data signal from a meteorological station is also received and recorded. Hence the effect of various meteorological parameters on wave propagation can be deduced.

A complete description of the system set up, radio links, sensors, etc., is found in [1]. In a companion paper [2] a summary of the system's hardware and major features of software is presented. The micro-computer used in this system is the HP9825 of 24K bytes RAM, one cassette tape drive and a small printer. This is used as a controller to the data acquisition system (DAS). In this paper, algorithms for the real-time data acquisition are presented. This includes: deter mination of system parameters and optimization of cassette tape utilization among others. The main philosophy of the developed software is based on the new MRP technique as described in [3]. In this technique, a run profile or a run parameter is simulated and created on a file before the start of the actual measuring run.

## 2. SYSTEM DESIGN

### 2.1 Design Philosophy

An automatic data acquisition system used for the radio propagation experiment must be capable of detecting fast fade events lasting for few seconds. Field experiments showed that the speed of the HP9825 microcomputer sets the limit on the system scanning rate. Thus, in order to maintain a sufficiently fast scanning, only the necessary amount of processing may be done during real-time data acquisition. On the other hand, the data acquisition system has some built-in intelligence, which enables a limited amount of data processing. Such capability is utilized as well to relieve the computer from its burden resulting in a higher system throughput.

Although the actual number of channels that will be connected to the DAS is expected to be around 20, the design allows for 40 channels. Each channel may be scanned in one of the available scanning modes, viz, fast, slow, operator selected, counter and adaptive scanning.

The fast scanning speed is the highest speed the system may allow and depends on the number of scanned channels and the scanning mode for each. The slow scan is arbitarily taken here as one scan per hour and the operator mode is left to the operator choice, within the system speed limit. The counter mode is used for counter driven events e.g. rain gauges. The adaptive scanning allows a scanning rate that vary with the speed of variation of the sensed signal themselves.

Each data acquisition run is considered as a separate one. Preparations of these runs include the determination of the channels to be scanned, mode of scanning, threshold % of readings variation and other reference data. These are done in a preparation run called the "Define Run Parameters" run. When satisfied with the channel configuration these information are saved on a cassette tape which is to be used later on during actual data acquisition. When the start time of data acquisition is due, another programme is run to drive the DAS and store relevant information on the tape cassettes. Later on, processing of collected data is done in separate runs. Since the computer is a single user, single process one, it is not possible to have more than one job simultaneously. Other system management and utility functions could also be run when needed. Figure (1) shows the system flow chart. The main reasons for separation between preparation of run parameters, actual data collection and processing of data is as follows:

i. The "define run parameter" run allows the configuration of the system before the actual data collection takes place. This also allows the study of "what-if" cases.

ii. The data collection run is dedicated to pure data acquisition which results in the fastest possible scanning rates.

iii. Due to memory limitations of the computer it is not efficient to combine the three different functions into a single run.

The overall software structure is a hierarchical tree consisting of activities, programs and functions as shown in Figure (2).

## 2.2 Tape Utilization

Two types of tapes are used in this system: MMW program tapes and MMW data tapes. In order to have a systematic, error free, handling of tapes all tapes are formatted in a standard manner. This standard is called the MMW standard.

**MMW program tape**

This tape contains the programs used in the MMW computer system. In the current version of the system the MMW program has a set of programs which enable the choice of run parameters, data collection, data retrieval, statistical analysis, tape listing, tape identification and system management. The first file on this tape is a tape header file which identifies the tape as being a MMW program tape.

**MMW data tapes**

These tapes contain three different files.

**(i) Tape header:**

This is a one record file which contains the string "MMW nnnc" where:

- nnn is a 3 digit number identifying the tape. Thus the system can house up to 1000 tapes simultaneously.

- c is a case character. It could be:

  o to indicate that the tape contains no other files except the header.

  1 to indicate that the tape contains the run parameters information in addition to header.

  2 to indicate that the tape contains actual data in addition to header and run parameters.

c is maintained by the computer.

The header contains also a "reel" serial number. This number is maintained by the computer for the multi-tape files, i.e., a file spanning over several tapes. The last reel is designated by -1 as a 'reel' serial number.

**(ii) Run parameters file:**

This file contains one record that stores the reference infor mation of the data acquisition run. These informations are:

- Start date of data collection (YYMMDD)
- Start time of data collection (HHMMSS)
- End date of data collection
- End time of data collection
- User name [24 chs]
- Run title [32 chs]
- Channel information table.

A 40x32 character matrix contains all the necessary infor mation regarding the channels supported by the system. Up to 40 channels could be handled. Each one has a 32 bytes vector holding its attributes.

Channel attributes are:

| | | |
|---|---|---|
| 6 | chs. | channel-ID |
| 24 | chs. | channel name |
| 1 | ch. | sampling mode |
| 1 | ch. | status (0=OFF and 1=ON) |

Channel speed and threshold %.
These are among the channel attributes. However, they are separated from other attributes since they are floating point numeric data rather than characters. The speed is in milleseconds, while the threshold % is in absolute value. Scanning speeds will be discussed later. Threshold % is the absolute percentage difference between two values that to be considered by the system as a new value to be recorded.

**(iii) Actual data files:**

These are the files containing the collected data in real-time from DAS. Each file contains 39 records on each tape reel. 19 records are recorded on track o while 20 records are on track 1. Each record is 7400 bytes. The reason for having one record on track 1 more than those on track o is due to the fact that tape header and run parameters files are stored on track o. The data stored in each record is an array of 365x18 bytes where 365 different readings (events) could be stored in one shot (to optimize tape utilization and minimize tape movement to have highest possible scanning rate).

Each entry contains:

- 2 bytes for channel number
- 6 bytes for reading date
- 6 bytes for reading time
- 4 bytes for sensed value (DC volt or counter)

### 2.3 Conversion Tables

The used sensors in this project output a DC or pulsed voltage which is proportional to the actual measured data. Mapping from the DC volt to the real world data is accomplished using conversion tables. Since different types of sensors are being used, several mapping functions (convertion tables) are needed. The current version of the system can house 10 different conversion tables.

The relationship between channels and conversion tables is defined by another map which relates some channels to a particular conversion table. This is not a one-to-one map since more than one channel can be mapped to the same table. Each table can have up to 40 pairs of readings, each pair represents DC volt and the corresponding physical value. Units of physical quantities are also stored to enhance pro duced reports readability.

Conversion tables are stored as a data file on the MMW program tape. This file is a one record file containing a logical array of 10x40 ele ments. This matrix is the map of channel number → conversion table. A zero in an entry denotes no association betwee table number = row number and channel number = column number; while a one denotes the existence of this association. Actual conversion tables are stored in a 3-D array of (10x40x2) elements and units of measured values are stored in a vector of 10 elements each one is 6 bytes long.

# 3. SYSTEM PARAMETERS

## 3.1 Scanning Speeds:

The system is designed to support five scanning modes. These are: Fast [F], Slow [S], Operator Selected [O], Counter [C] and Adaptive [A]. In order to compute the fastest scanning speed for the [F] mode, the following algorithm is used:

**Algorithm A1**

1. 0 → J and 0 → K
2. Compute speed limit "M" (algorithm A2)
3. 0 → I
4. If status of channel I is OFF go to 7.
5. If scanning mode of channel I is F or A then 1+K → K.
6. If scanning mode of channel I is O and requested rate < M, then 1 + K → K.
7. 1 + I → I.
8. If I < 39 go to 4.
9. Fastest scanning rate = 1000K/60.

Hints:

1. First channel is numbered o.
2. The HP3497 DVM in the DAS is programmed to autorange a DC volt measurement using 1 power line cycle (1/60 second) integration at maximum resolution from a selected 3497 channel.

**Algorithm A2**

1. 0 → M
2. 0 → I
3. If channel I status is OFF or scanning mode <> F then go to 5.
4. Speed of channel I → M; go to 7.
5. 1 + I → I
6. If I < 39 go to 3.
7. End

For the counter mode sensors, since rain events are very rare in Riyadh, it is not economic to make continuous scanning for the counters as the radio links. When rain starts only the counters are scanned for a period of time $\Delta t$ to compute the rain intensity. A reasonable value of the integration time $\Delta t$ is between 1 to 5 minutes. The system com putes a "sleeping factor" Q for which the counter is not scanned while other channels are scanned. Q is a factor of the run parameter as follows:

$$Q = \text{int}\left(\frac{60000}{n_1 \ s_1}\right)$$

where $n_1$ = number of fast scanned channels, $s_1$ = fastest scanning rate in m.sec. and thus the scanning cycle will be as follows:

```
Compute the value of Q;
while (not end of run) do;
scan all fast channels Q times;
check counter; if there is a reading scan it and reset
the counter else continue;
end while;
```

A flowchart of the logic of the measuring run is shown in Figure (3).

### 3.2 Optimizing Tape Utilization

Since the only available external storage media is a one cassette tape drive it is mandatory to optimize its utilization in order to minimize the need for frequent change of tapes. Several techniques have been used in order to realize this goal. These include;

(i) Data are kept in memory to the maximum available size of RAM and recorded on tape in batches.

(ii) Only relevant data are kept in memory. A measured data element is considered to be relevant if the percentage of relative deviation from the previous measured data from the same channel is > the prescribed threshold for this particular channel.

(iii) Data are stored on basis of event recording rather than systematic recording of data. In other words, if a channel value is changing faster than others, more entries will be allocated to this channel. No predetermined allocation of space to channels is assumed.

(iv) All data items of the same type [characters] are grouped together in a continuous string to minimize the string overhead which is computed according to:

6 + 2x  where x is the number of strings in the string array.

(v) All integers stored in the system are stored in a short precision format rather than the normal storage of numerics. This saves 6 bytes for each integer without any loss of precision.

(vi) Whenever feasible, floating point numbers are stored in short precision format rather than in the normal storage mode of numerics. This saves 4 bytes for each number. Only six significant digits are saved using this method rather than the standard 13 digits.

(vii) Determination of Data record size: According to the specifications of the cassette tape; the relationship between a file size and its length is given by [4];

$$L = 1.278 + .209 \left\lceil \frac{A}{256} \right\rceil + .0105A \qquad (1)$$

where A is absolute file size in bytes
L is file length in inches

Assuming total tape length is T inches then the number of files/track is:

$$F = \left\lfloor \frac{T}{L} \right\rfloor \qquad (2)$$

where F is the total number of files per track.

Combining (1) & (2), the total storage S is:

$$S = F*A \qquad (3)$$

where S is the total storage capacity in bytes per track.

Since an event requires 18 bytes of storage (as previously given):

2 for channel no.
6 for date of event
6 for time of event
4 for value of event

and since events are grouped into a batch; the problem is to determine the optimum batch size. Assuming the number of events to be grouped is R we get:

$$A = R*18 + 2R + 6$$

i.e.

$$A = 20R + 6 \qquad (4)$$

Substituting (1), (2) and (4) into (3) we get

$$S = \left\lfloor \frac{T}{1.278 + .209 \left\lceil \frac{20R+6}{256} \right\rceil + .0105(20R+6)} \right\rfloor *(20R+6) \qquad (5)$$

Obtaining the value of R which maximizes S is not possible by direct method [Calculus] due to the presence of $\lceil\ \rceil$ and $\lfloor\ \rfloor$. For that reason a computer program was used to determine the optimum value of R. This value is 365 which gives a file size of 7306 bytes.

## 4. SYSTEM FEATURES

### 4.1 System Management Functions

The aim of system management is to ensure error-free handling of data and to recover from some exceptional cases. The system management programs handle:

(i) Tape identification and listing: this is used to identify a tape as a scratch tape, nonMMW tape, MMW program tape or MMW data tape. It is also possible to list the contents of any tape. Listing could be done in full or partially.

(ii) Conversion table maintenance and reporting.

(iii) Recovering a corrupted MMW data tape: three reasons can cause a MMW data tape to be corrupted.

(a) Immature end of the data collection run which could be due to power failure.

(b) Tape parity error.

(c) Tape physical tearing.

An error trapping is enabled when perofrming an I/O operation with the tape. This will transfer control to an error handling recovery routine. The recovery procedure itself is dependent on the cause of error. If powerfailure occurs; a memory dump is done for the whole readings in the computer memory and an end of file mark is written on the same tape. However, for physical tearing of tape a new tape is needed and a copying procedure starts from the old tape to the new one.

## 4.2 On-line Real-time Tracing

Sometimes, it is useful to trace the values of the chanels being scanned in real-time i.e. during the scanning procedure. The reason why we are interested in getting on-line real-time tracing of the values being read is that it could be useful to detect any fault in the links being scanned which cannot be detected by the system. It is also useful during rain events to check that the rain gauge is functioning.

To enable this facility the user sets a software switch which is initially unset. To stop tracing the user unsets this switch again. Note that when tracing is enabled the computer will print the channel number, the read value, the current date and time. The thermal printer must be ON to get this printout. Note that the size of printout is large since the scanning rate is in milliseconds; it means that several lines will be printed per second. However, since the printing speed is relatively slow the actual number of printed lines in a particular time span will be less.

## 4.3 Forced End of Run

Sometimes, it happens that a measuring run ought to be aborted for various reasons. A facility is provided which ensures orderly end of run before the actual time being specified in the run parameters as the end time. This is accomplished by setting another software switch which is initially unset.

The following actions will be done when detected by the system:

The data acquisition activity will be stopped.

The current file in memory will be ended and copied to the MMW data tape.

The reel header will be updated and the summary log will be printed.

Control is returned to main menu again after insertion of the MMW program tape in the drive.

This facility proved to be very useful during the course of long data acquisition runs.

### 4.4 Forced End of Tape

This is similar to the previous one except that the measuring run will be continued after the insertion of a new scratch tape in the tape drive. This is particularly useful when there is significant activities on the radio links and it is expected that the current reel will not last long before being full. In such cases it may be better to make a forced end of tape by setting a software switch to simulate an end of tape. Note however, that the original run will resume after putting a new scratch tape.

### 4.5 Back-up Utility

Since tapes are not very reliable media for storing valuable data, backup operations seem to be a must. In this utility program, the input data tapes are copies to new scratch tapes, for backup purposes. Two files can be backed up: MMWRAN and MMWFAD which contain the abstracted data.

### 4.6 Unified End of Run for Programs

All the programs in the system display the same message to return to the main menu again. "Put the program tape". The program will wait indefinitely until the tape is inserted and the continue key is pressed.

### 4.7 Data Abstraction

**Motivation:**

Here all the data obtained so far are scanned and according to pre-determined thresholds, the relevant data are kept on two new files. This procedure is to be continued as new data is collected.

Two new types of files are created and updated. These are the MMWRAN and the MMWFAD files. Both of them are formatted according to the MMW standards. In the first file, relevant radio links data during rain are recorded while in the second one, the data during non-rain periods are stored. Large variations in radio signals are identified by a difference of more than 1 dB. Any rain measurements under 1 mm/hr are disregarded.

**File organization:**

Two new files are created: the MMWRAN and MMWFAD. They are structured according to the MMW standards. Since both of them contain the same type of data they have similar structure.

The MMWRAN contains data during rain while the MMWFAD contains data when there is no rain. Since these files will be updated each time new data tapes are created; there should be a means to optimize their processing.

Storing data on cassettes or magnetic tapes necessitates sequential processing. Updating files stored on these storage devices is always done by copying. But we have one cassette tape drive only. This was really a challenge. However, an intelligent, yet simple solu tion was done. This is to mark several files on each reel to be used for MMWRAN or MMWFAD files. Marking files means writing EOF marks on the data cartridge. In the same time we keep track of the last written file and the exact portion which of where data is already exist. If a reel is full a special flag is set in the header file. This method of organization has the following advantages:

i. It is possible to update the file using one drive only. This is done by retrieving the last partially filled file in memory, updating it and re-writing it back on the same cartridge. This is possible and reliable since all files have exactly the same size.

ii. Accessing of the last file is fast because in the header file there are two pointers; one refers to the last file n say, which can be accessed by skipping n-1 EOF marks. The other pointer is used to refer to the last part written within this file.

iii. No "read-test EOF" loop procedure is necessary to append data to either file.

iv. A full reel is detected by a special flag in the header file.

v. Formatting of new reels required for the MMWRAN or the MMWFAD files is done automatically during normal operation.

The only disadvantage of this approach is that the elaborating "housekeeping" procedure necessary to create these files which complicates programs design and testing.

The organization of file MMWRAN is as follows:

i. On track 0; a header file is created automatically which contains:

- The file name = MMWRAN
- Full reel indicator:

  1 = full reel
  0 = partially full

- Reel sequence number
- Last file written
- Last row number within this file which contains data.

ii. On track 0; 20 files used to hold data are formatted. Each file can hold up to 365 data lines; the same as that of the original data file.

iii. On track 1; more 20 files are formatted similar to those on track 0.

The file MMWFAD has similar structure as that of MMWRAN except the file name which is MMWFAD, of course.

The data files on both MMWRAN and MMWFAD are 7400 bytes each. Total of 40 files could be stored on each reel. Each file consists of 365 different events with 18 bytes for each event. Each entry is as follows:

2 bytes for channel number stored in integer format
6 bytes for reading date in YYMMDD format
6 bytes for reading time in HHMMSS format
4 bytes for the physical value stored in short real format.

The physical value is in dB for radio links channels and in mm/hour for rain channel. The definition of channels is maintained in the channel information table as reported before.

**Memory organization:**

Since the available memory is limited and the data abstraction subsystem requires the handling of four different types of files: MMW data files, MMWRAN, MMWFAD and Conversion Tables; a careful memory organization and management is a must.

First of all, the creation and updating of MMWRAN is done separa tely from that of MMWFAD. So in a single run we are dealing with three files only: MMWRAN or MMWFAD, the Conversion Table file and the original MMW data file.

The relevant conversion tables are read from conversion tables file and stored in a fixed part of the computer memory. This is done because we need these tables frequently. Every radio link data read, is converted immediately to a dB reading. Another fixed part of the computer memory is needed for the program itself. The remaining part is divided between two buffers; one for the input file, the MMW data file and the other for the output file; the MMWRAN or MMWFAD file. Two pointers are used to keep track of the input and output buffers. Since only relevant data are kept on output file we expect that the input buffer will be exhausted much earlier than the output one. Free space of input buffer could thus be utilized.

**Data abstraction algorithm**

Two programs are developed for the data abstraction subsystem, one for the creation and updating of the MMWRAN file and the other for the MMWFAD file. The general logic of first program is as follows:

1. Initialize pointers, buffers and other working areas and unset rain switch.

2. Read the relevant conversion tables and store them in the computer memory.

3. Ask for the output tape.

4. If the output is a new reel; then format it according to its file organization and goto (6).

5. If the output is an old one; check its label and retrieve the pointers of file and row.

6. Ask for the Input tape.

7. Check whether this tape contains rain data or not. This is done by checking the run parameters file.

8. If no rain data; print a message and go to (6) or stop if no more tapes are to be processed.

9. Read a record at end goto (19).

10. If the record is a rain channel and reading > 1 goto (11) else goto (9).

11. Set rain switch "Start of rain event".

12. Write output record.

13. Read a record at end goto (19).

14. If the record is a rain channel and reading < 1 goto (18).

15. If the record is a rain channel and reading > 1 goto (12).

16. If the record is a radio link channel and readings difference is significant (> 1 dB) goto (12).

17. If the record is a radio link and readings difference is not significant goto (13).

18. Unset rain swtich "End of rain event" and goto (9).

19. Close output tape and stop.

The logic of the other program is similar to the previous one. The only difference is that we are interested in the data when there is no rain.

**4.8 Reporting**

Different reports can be produced from the system. They could be grouped into two major groups:

(i) Data retrieval; and
(ii) Statistical analysis.

In the data retrieval reports the measured physical quantities are reported against date, time and channel number. Interpolation is used when mapping from sensed values to actual physical ones. It is possible to have selective reports based on particular channels and/or particular ranges of data. Different statistical analysis can be done on selective basis, details of system reporting and analysis capabili ties can be found elsewhere [5].

## 5. CONCLUSION

The paper has presented algorithms of a computer system for real-time data acquisition. Among other features, the system can scan up to 40 channels with different scan modes and speeds. Only significant information are stored on tapes to optimize tape utilization and scan rate. The system is provided with conversion tables to map sensed DC voltage values into the actual data domain. The system is capable of detecting channel failures and/or errors. A warning message is issued and automatic restoration is established when the faulty conditions are ended on-line real-time tracing is provided. The computer system is suitable for implementation on the simplest micros and has been working satis factorily on a field study run by the authors in Riyadh city, Saudi Arabia. The study is intended for identifying the effects of meteoro logical conditions on the propagation of millimetric radio waves used for communications.

## ACKNOWLEDGEMENT

This research is sponsored by grant # AR-5-29 from King Abdulaziz City for Science and Technology and grant # 5/403 from the Research Center at the College of Engineering, King Saud University.

## REFERENCES

1. Millimetric Wave Propagation and Communication - A field study in Riyadh," First Progress Report SANCST project AR-5-29, May 1983.

2. Sharaf Eldin, A., Alhaider, M.A. and Ali, A.A., "A Micro-computer- based Real-time Data System for Radio Wave Propagation Studies," Journal of Microcomputer Application, Academic Press, 1986.

3. Sharaf Eldin, A., "A Micro-computer-based Real-time Data Acquisition System for Solar Energy Studies," The 17th Annual Simulation and Modeling Conference, Pittsburgh, USA, April 1986.

4. HP9825 Operating and Programming Reference, HP, May 1980.
5. Millimatric Wave Propagation and Communication, "A Field Study in Riyadh", Progress Report # 5, KACST Project AR-5-29, May 1986.

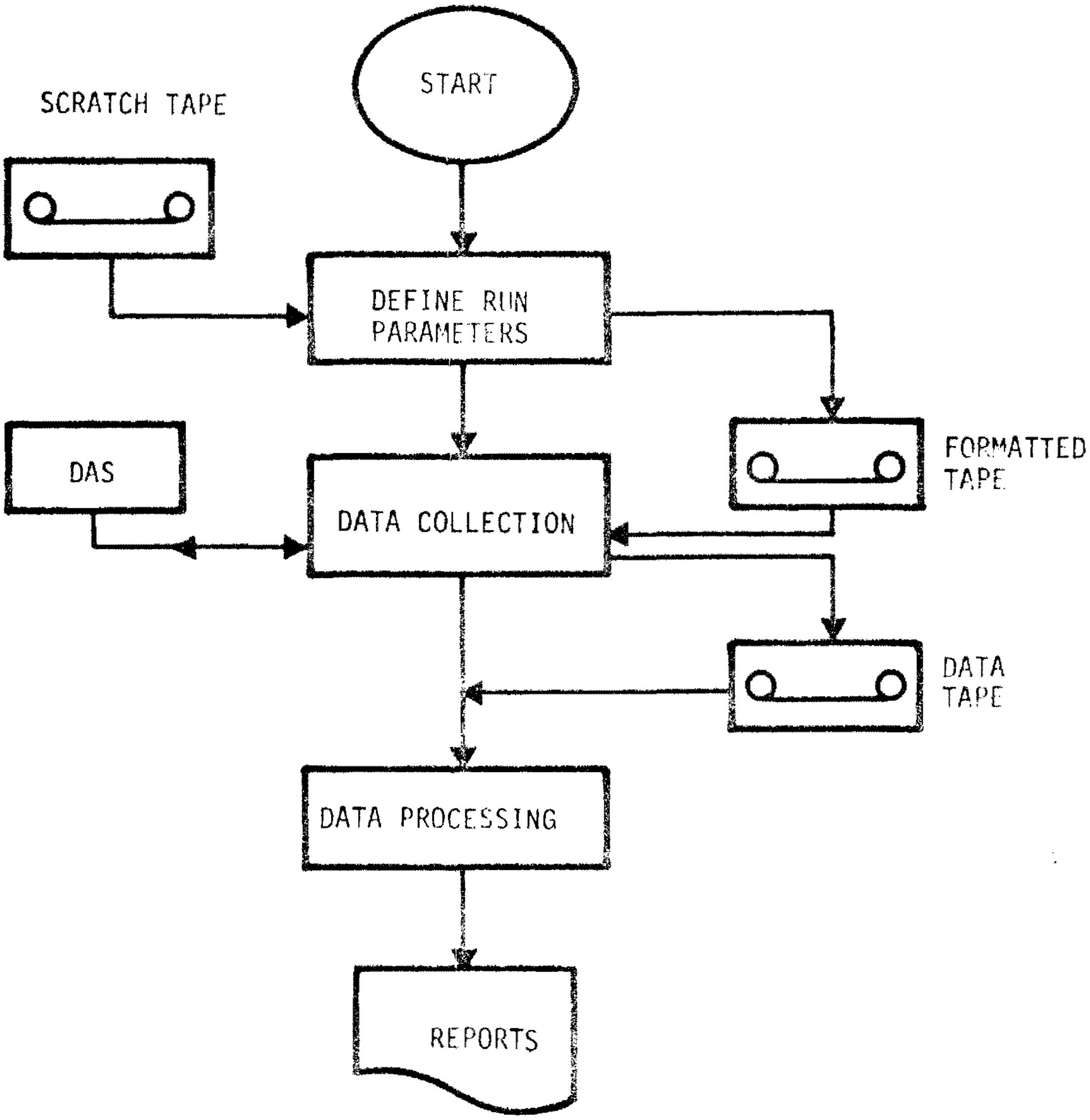


FIGURE 1 : GENERAL SYSTEM BLOCK DIAGRAM

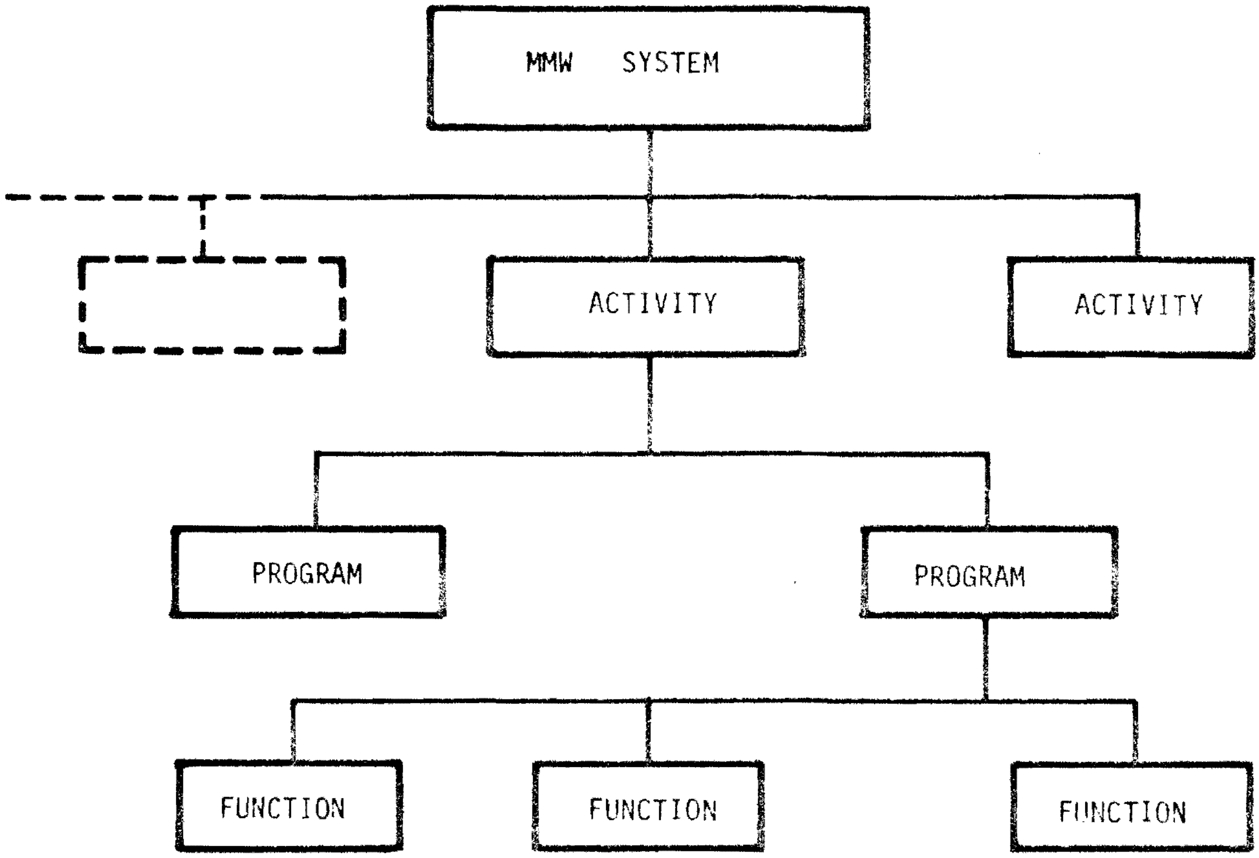


FIGURE 2 : MMW SYSTEM HIERARCHY