A DATA-LINK BETWEEN TWO COMPUTERS;
DEVELOPMENT OF SOFTWARE AND EXPERIENCE
OF ITS USE WITH A MISSING-MASS SPECTROMETER EXPERIMENT

T.R. Bell, B.C. Levrat, P.J. Marcer and E.M. Palandri

GENEVA
1967
Copyright CERN, Genève, 1967

Propriété littéraire et scientifique réservée pour tous les pays du monde. Ce document ne peut être reproduit ou traduit en tout ou en partie sans l'autorisation écrite du Directeur général du CERN, titulaire du droit d'auteur. Dans les cas appropriés, et s'il s'agit d'utiliser le document à des fins non commerciales, cette autorisation sera volontiers accordée.

Le CERN ne revendique pas la propriété des inventions brevetables et dessins ou modèles susceptibles de dépôt qui pourraient être décrits dans le présent document ; ceux-ci peuvent être librement utilisés par les instituts de recherche, les industriels et autres intéressés. Cependant, le CERN se réserve le droit de s'opposer à toute revendication qu'un usager pourrait faire de la propriété scientifique ou industrielle de toute invention et tout dessin ou modèle décrits dans le présent document.

Literary and scientific copyrights reserved in all countries of the world. This report, or any part of it, may not be reprinted or translated without written permission of the copyright holder, the Director-General of CERN. However, permission will be freely granted for appropriate non-commercial use.

If any patentable invention or registrable design is described in the report, CERN makes no claim to property rights in it but offers it for the free use of research institutions, manufacturers and others. CERN, however, may oppose any attempt by a user to claim any proprietary or patent rights in such inventions or designs as may be described in the present document.
ABSTRACT

Data from a missing-mass spectrometer experiment, collected and prechecked on an SDS 920, was transmitted via data-link to the CERN central computing facilities where it was processed on-line within the environment of the SIPROS multiprogramming operating system of the CDC 6600. The software developed for the initial production runs is described, together with the experience obtained.
CONTENTS

1. INTRODUCTION .............................................. 1

2. SOFTWARE DEVELOPMENT FOR THE DATA-LINK ................. 2
   2.1 A brief description of the hardware .................. 2
   2.2 Software implementation -- the general picture ....... 2
   2.3 Operating system environment of the data-link program 5
   2.4 The CDC 6600 data-link software packages -- a complete description 7
       2.4.1 The complete package ........................... 7
       2.4.2 Operation of the package ....................... 7
       2.4.3 PP utilization and CM buffering ................. 10
       2.4.4 Calling sequences for data-link CM control routines 11
       2.4.5 DSPOUT (ARRAY, NUM, KS) ...................... 12
   2.5 SDS 920 software -- a complete description .......... 13
       2.5.1 Data-link test program ......................... 13
       2.5.2 Operation ....................................... 13
       2.5.3 Data-link production program ................... 15

3. MISSING-MASS SPECTROMETER EXPERIMENT ..................... 16
   3.1 The CERN missing-mass spectrometer ................... 16
   3.2 Data acquisition and data handling for the experiment 19
   3.3 The processing program ................................ 20

4. CONCLUSIONS AND STATISTICS .................................. 22

APPENDIX ..................................................... 27

REFERENCES ................................................... 31
1. INTRODUCTION

Up to the end of 1964, the experience of a number of groups outside CERN and of two groups within CERN \(^1\) had shown the advantages obtained from the use of a computer on-line to an experiment. A small computer, for example the SDS 920, is capable of buffering, checking, and refining data from experimental equipment; thus through simple instrumental checks it can improve the data acquisition rate and the efficiency of an experiment by preventing the blind accumulation of useless data.

However, apart from guarding against trivial malfunctioning of elements of the system, one has to check the effectiveness of the trigger. It is evident that the advantage of a spark-chamber system over bubble chambers is the possibility of selecting only interesting events; but this selectivity represents a major danger in the sense that it can introduce a bias by rejecting some of the good events, or accepting unwanted selected events. Manual control of the trigger usually means stopping production, and running delay and coincidence curves, for example, then resetting everything to resume data acquisition. This is a necessary procedure and is by no means foolproof. A more effective way is to analyse a fraction of events and to compare a certain number of characteristic histograms with as good statistics as one can have, and look for the shaving of tails, unexpected build-ups, or modification of the general shape.

Moreover, during the course of a run one gets ideas about how to improve the trigger; in which case, as for any change, a quick return of results is a necessity in order to estimate the effects of the change. It is clear that one has to process as many events as possible in order to reach a decision unbiased by statistical fluctuations; and the economy in accelerator time justifies the access to powerful central computing facilities, with high priority and good turn-around. This can be achieved by bicycle on-line or, more flexibly, by a direct data-link; and with the arrival of the CDC 6600 computer at CERN the prospect of a powerful on-line facility for experiments appeared, by connecting this computer via data-links to the experimental areas.

The Interim Report of the Study Group on the On-Line Use of the CDC 6600 Computer \(^2\), published in March 1965, recommended that a data-link and associated support programs should be implemented between the SDS 920 computer, to be situated in the South Hall, and the CDC 6600 computer; and that this link should be constructed, as far as possible, from equipment used previously for the Mercury computer data-links, though with an increase in transmission rates to be better suited to the SDS 920 and CDC 6600 computers.

This report describes the programs used for testing the data-link so constructed, and the application of the data-link to the missing-mass spectrometer experiment. The construction and initial testing of the data-link is described in detail elsewhere \(^3\).
2. SOFTWARE DEVELOPMENT FOR THE DATA-LINK

When construction and hardware testing of the data-link was completed, there remained only the problem of testing the feasibility of using the data-link under operational conditions. At this time the missing-mass spectrometer (MMS) experiment was running, using the SDS 920 computer 'on-line', and producing raw data tapes for analysis on the CDC 3600 computer. This appeared to be an admirable arrangement to commission the data-link, in view of the fact that the experiment was established, the data acquisition and read-out system was well understood, and the programs, both data acquisition and analysis, were operational and fully debugged; and finally that the results obtained could be easily and thoroughly checked by comparison with those results obtained from off-line analysis of the same events from the SDS 920 produced raw data tape.

This final testing of the data-link and its software was carried out in two stages. Initially an SDS 920 program which simulated an on-line run, using pre-recorded data, was run in conjunction with an analysis program in central memory, in order to develop and debug the CDC 6600 software package for running the data-link; finally, the SDS 920 data-link subroutines were incorporated into the data acquisition program, and the complete system used and tested during the MMS accelerator time as an integral part of the experimental data processing (see Diagram I).

2.1 A brief description of the hardware

The data-link consists of a control unit situated near the CDC 6600, an interface at the SDS 920, and transmission equipment between these units. This permits the interchange of status information between the computers, which is used for synchronization, and bi-directional transmission of data. The data path is 25-bits wide (i.e. one SDS 920 word, plus its associated parity bit), and the maximum rate for each line is 125 K-bits per second. The length of this data-link is just over 1 kilometre (see Diagram II).

2.2 Software implementation -- The general picture

This consists of two programs; a CDC 6600 data analysis program and an SDS 920 data acquisition program. The latter is in continuous residence in the 920 throughout the on-line experiment, whilst the former is loaded only when required. Communication between the two programs is via a 6600 central memory (CM), peripheral processor (FP), data-link interface and an associated small computer data-link interface. Once loaded in the CDC 6600, the real-time user program is in continuous occupation of CDC 6600 CM throughout the on-line run. Within the software interface between the two computers the sequence of events is initiated by a request from the small computer for a transmission of data to the 6600. As no external interrupts exist in 6600 hardware, this request is in the form of a status word set within the data-link control by the small on-line computer. The hardware permits the monitoring of this status either directly via the data-link control or via a subchannel of the slow peripheral multiplexer. The former requires the permanent allocation of a FP to the 6600 data-link program, as there exists no method in SIPROS of requesting a FP at regular intervals for monitoring purposes. This has the advantage that, except in unusual circumstances, no data
transfer requests are ignored. The latter hardware connection is an attempt to overcome the inefficiency inherent in the permanent removal of a PP from the system 'pool'. The slow peripheral multiplexor package can monitor data-link status every 10 msec or so, and upon detecting a request for data transmissions it can reactivate the central data-link program via a call to the SIFR0S executive (Executive). Once activated, this program requests the allocation of a PP to carry out the data input. However, the program may have to wait several seconds until a PP becomes available, and hence a data transfer request may be ignored for a sufficient time to cause a subsequent loss of data. Once the allocated PP has detected the request for data input, synchronisation is established between the two computers via a dialogue of status and control signals, and data input to the CDC 6600 takes place in record blocks. A header to each block permits a check of the number of words transferred, and determines whether there is a further block to follow. Errors in the input/output dialogue are reported to CM, and may result in calls to the operator at the teletype for the appropriate intervention.

In case of errors, several attempts at transfer are made, before reporting to the operator at the teletype. Once input to the CDC 6600 is complete, the PP alerts the Executive to restart the new CM program. This program analyses the data, stores the results on a data summary tape, and may, if requested by the program or by the on-line user, transmit data summaries back over the link, or histograms to the CDC 6600 CRT display situated in the experimental area. In practice only the display was used, as there was no space available within the SDS 920 for the CDC 6600 to SDS 920 data transfer interface. Diagram III describes the sequence of events which determined the data flow throughout the system.

2.3 Operating system environment of the data-link program

The software picture is incomplete without a short description of the special features available within the SIFR0S operating system for real-time programs.

Within the operating system, two subsystems situated at either end of available core are assigned to real-time user programs in accordance with a control card.

Each subsystem provides the user program with the following:

i) a fast response from the Executive in answer to input/output requests, and the ability to recall the Executive once data is processed;

ii) the facility to load from the card reader to central memory within minutes, provided no compilation is required and any required equipment is available;

iii) the facility to recover from program or hardware failure (or to hand control to an operator at a teletype): this is implemented via the extensive use of system error procedures and complete input/output dialogue checking for the data-link interface;

iv) the facility to transmit information summaries via the on-line high-speed printers during execution.

However, these facilities are still insufficient to suit either the main requirements of the on-line physicist or the general throughput of the CDC 6600. An operating system with more flexible automatic scheduling and a general roll-in/roll-out capability is desirable. A fuller discussion is given in the conclusions.
<table>
<thead>
<tr>
<th>SEQUENCE OF EVENTS</th>
<th>DATA</th>
</tr>
</thead>
<tbody>
<tr>
<td>PS burst</td>
<td></td>
</tr>
<tr>
<td>Data interrupt from experimental scalers</td>
<td>Raw data transfer to SDS 920</td>
</tr>
<tr>
<td>another event</td>
<td>Raw data</td>
</tr>
<tr>
<td>another event</td>
<td>Raw data</td>
</tr>
<tr>
<td>another event</td>
<td>Raw data</td>
</tr>
<tr>
<td>End of PS burst interrupt or buffer in the 920 full</td>
<td>Checking of raw data</td>
</tr>
<tr>
<td>SDS request data-link transfer</td>
<td>Storage of checked data on magnetic tape on SDS 920</td>
</tr>
<tr>
<td>peripheral processor replies OK</td>
<td>Checked data sent to CDC 6600</td>
</tr>
<tr>
<td>SDS replies waiting to send</td>
<td>Data transferred to CDC 6600 CM</td>
</tr>
<tr>
<td>Parity and length of transfer checked to be OK</td>
<td>CDC 6600 CPU processes data</td>
</tr>
<tr>
<td>Peripheral processor sets control word and alerts Executive</td>
<td>Summary data to tape</td>
</tr>
<tr>
<td>CM program request transfer of summary data to tape</td>
<td></td>
</tr>
<tr>
<td>CM program asks if operator has message (via teletype)</td>
<td></td>
</tr>
<tr>
<td>Message</td>
<td></td>
</tr>
<tr>
<td>Yes or No</td>
<td></td>
</tr>
<tr>
<td>Histogram generated as the result of the message or because of program request</td>
<td>Histogram generated</td>
</tr>
<tr>
<td></td>
<td>Data to PP</td>
</tr>
<tr>
<td></td>
<td>Data to display</td>
</tr>
</tbody>
</table>
2.4 The CDC 6600 data-link software packages -- a complete description

2.4.1 The complete package

The software package for running the CDC 6600 end of the SDS 920 link consists of two parts:

1) A CM subroutine LINKSW which
   a) controls the data link PP program and sets up parameters and requests for it;
   b) interfaces by means of several entry points with the main CM processing program providing data from the link for the main program when requested, outputting over the link when necessary, and outputting to the display storage scope;
   c) provides a means of operator control by communicating with a teletype.

ii) A PP program LI which, when loaded and operating,
   a) continuously monitors the data-link and the SDS 920 to determine whether the SDS 920 wishes to transmit data from the CDC 6600, or is ready to receive data from the CDC 6600;
   b) inputs data from the link whenever the SDS 920 is ready and buffer space is available in the CM of the CDC 6600;
   c) outputs data to the link and SDS 920 whenever the SDS 920 indicates it is ready and the CDC 6600 has data it wishes to send;
   d) outputs blocks of coordinates to the display storage scope on receipt of a request to that effect from LINKSW.

2.4.2 Operation of the package

Initialization

The package operates as a subroutine of the CDC 6600 main processing program, and before it can act it needs to be initialized. This is done by a call from the main program to the entry point LINKST. This call has for parameters the location and length of an area of circular buffer in the CM which will act as the input buffer for the PP. This input buffer is large enough to hold several transfers (or several groups of transfers if one logical CM input consists of more than one transfer). After setting up a parameter list for the PP and receiving a "go" instruction from the operator, LINKSW loads the PP, sets it monitoring the data-link, and returns to the calling program.

Input

As far as input is concerned, the PP then proceeds to work independently of the CM program at least to the extent of inputting all the transfers it can from the link, until there is no more space remaining in the CM circular buffer. Whenever the PP detects that the SDS 920 has a transfer ready for input, it checks first to see if CM buffer space is available, and if it is then it inputs the transfer, writes it to CM,
updates a counter in CM to indicate to the CM program that that buffer section is full of new data, and informs the Executive that new data is available, in case the CM has given up control to the Executive waiting for data to process. If no buffer section is available, the PP ignores the SDS 920 until it finds that the CM program has processed and freed another section. In this way input is buffered at least to the extent of the number of transfers that can fit into the buffer space made available by the main CM processing program.

Whenever the main program wants more data to process, it makes a call to the LINKSW entry point LINKIN. LINKIN first checks the buffer counters to see if any data is currently available. If it is, it decodes the data into a form suitable for the main program, and returns with information as to the location and length of the transfer in the circular buffer. If no data is available, LINKSW first checks to see that the PP has encountered no hardware aborts from the link and the SDS, and if all is well gives back control of the CP to the Executive (via a WAIW macro) until the PP indicates to the Executive that it has filled another buffer section with data or that it has discovered a hardware fault.

When the PP detects a hardware fault it sets a bit inside a CM status word to indicate the nature of the fault, and then drops itself after first informing the Executive to restart the CM program if necessary. The CM program, on detecting the abort, examines the status word to determine its exact reason, prints a message on the teletype informing the operator of the abort and its cause, and reloads the PP to try again for a number of times determined by an adjustable CM parameter. If the PP fails consecutively this number of times, the program gives up and awaits further instructions from the operator via the teletype.

The entire dialogue between CM, PP, and operator takes place within LINKIN, and control is only returned to the main program when a good buffer of information is available in a form ready for the main program to process.

**Releasing a buffer section**

When the main program has finished processing the contents of a buffer section, it must release that buffer section back to LINKSW and the PP by making a call to the entry point LINKRS. This causes the CM buffer pointer to be advanced, thus informing the PP that that buffer section is now free to be filled with more data.

**Output over the data link**

When the main program wishes to output data over the link to the SDS, it makes a call to the entry point LINKOT informing it of the location and size of the output, and the way (if any) in which it wishes the output to be encoded for transfer to the SDS 920. LINKOT then sets up a request to the PP to output the data when it can, together with the location of a status word (provided by the main CM program) in which to indicate when the output has been successfully completed. LINKOT also checks that the PP is loaded and running and has not dropped because of a hardware fault. If there is a fault it analyses it, and then reloads to the PP to proceed with the output.
LINKOT does not, however, wait for the output to be completed but returns to the main program as soon as the request has been set up for a live PP program. The PP accepts the request and then proceeds through its loop, checking if the SDS is ready for input or output. On finding the SDS 920 ready for output it attempts to make the output and, if successful, sets to zero the CN status word with which it has been provided. At all times it gives preference to inputting from the link, and will only output when no inputs are available. If an output abort is detected, the reason is indicated and the PP dropped in the usual way -- to be detected by the CN program on a subsequent call to LINKOT or LINKIN.

Any attempt by the main program to call LINKOT and initiate another output before a previously requested output is complete is ignored, except that LINKOT will check again that the PP is loaded and in a position to try the previous output.

Outputs over the link are thus fully buffered and can be sent off by the main program with the minimum of interference with link inputs.

Outputs to the display storage scope

Outputs from the main program to the storage display are made by calls to the LINKSW entry point DSPOUT. The parameters to the call include the location, size, and form of the block of data to be displayed, together with a status word to indicate if the display was correctly sent or not. CN program control remains in DSPOUT until the display has been successfully sent or definitely aborted. This is because displays are usually sent from the main program by a series of rapid consecutive calls to DSPOUT with small amounts of data in small buffers involved in each call.

DSPOUT sets up a request to the PP program for output to the display -- checking also that the PP is alive -- and then gives control to the Executive until told the display is sent or aborted. The PP accepts this request and attempts to send the block of data involved to the display. If it discovers that the display is involved in one of its long erase or write delay sequences it returns to monitor and input from the link to the extent of its buffer until it discovers that the display is ready to accept its block of data. When the data is sent it informs the Executive to restart DSPOUT which returns to the main program with a status word indicating a successful output.

Display aborts are treated differently from link aborts. If when DSPOUT sets up a display request it finds the PP dropped due to a previous link abort, it stores up this abort status for later processing by LINKIN or LINKOT, and then loads the PP specifically to process the display request. Similarly, display aborts do not cause the PP to drop but are indicated directly to DSPOUT, which processes them while the PP returns immediately to monitor the link for input.

When DSPOUT detects a display abort it prints the reason on the teletype and returns to the main program with an abort status. The main program is thus free to try another display or give up as it thinks fit.

Outputting to the display and in particular display aborts are treated in this way because displays can be regenerated at any time while a lost block of input data can never be retrieved.
2.4.3 PP utilization and CM buffering

When in production the CM program operates with a peripheral processor assigned to it almost permanently. Because the maximum PP utilization is of the order of 20 msec every 2 secs (the pulse time of the CERN Proton Synchrotron) and is normally much less, this is in fact an inefficient way to operate. It was adopted for the reason that the SDS 920 data-link seen from the CDC 6600 is a true real-time device (as distinct from the HPD's and other pseudo real-time devices operating at CERN) in that the CDC 6600 must respond at times completely determined by conditions external to it -- in this case the pulse rate of the PS and the rate at which the experimental apparatus generates data during each pulse. At the time this software was put into production, the only operational way of determining when the data-link desired servicing from the CDC 6600 was to have a PP permanently loaded with a program which spent its time in a loop monitoring the data-link status.

The necessity of having a permanently allocated PP makes possible the system of CM buffering adopted by the program. In effect, once the PP is loaded it operates to a large extent independently of the CM program in that it inputs from the link when such an input is wanted, and writes this input into the CM buffer independent of what stage the CM program has reached -- at least until it has completely filled its CM buffer. It is possible for the CM program to be prevented by various conditions from returning to process the buffer for some time (e.g. while waiting for a PP to write a tape -- or while waiting for completion of the many disk transfers involved in printing a results summary). The program is buffered against loss of data from the link due to conditions of this type to the extent of the number of transfers that can be held in the circular CM buffer it has itself allocated to the PP program.

During the operation of the program, the PP may be dropped and the program made to wait by an instruction from the teletype. This is of use if, due to a temporary PS fault or other cause, the operator learns that there will be no data sent for some time and so can give back the PP to CDC SIPROS for other uses for the time being.

Another way of getting the CDC 6600 to react to the data-link is now available through a link between the data-link terminal and the 6600 multiplexer unit. In this system the current data-link status is made available to the CM program through the multiplexer and the permanent multiplexer PP package in the SIPROS system. Then whenever the CM program wants more data to process it calls the multiplexer package in a way similar to a teletype call, and gives up control of the CP until the data-link indicates through the multiplexer that it has another input ready. When this occurs, the system restarts the CM program which loads its own PP package, performs the input, and then drops the PP when the input is complete.

This results in much more efficient PP usage, although it makes the type of buffering described above impossible, and also makes pointless the use of a multi-transfer CM buffer. The CM program will only interrogate the data-link when it has finished with the current buffer contents and with all the other tasks it wishes to perform, and the situation effectively reverts to one in which the CDC 6600 is not reacting to data input on a real-time basis but is sampling the link for data only when it sees fit.
2.4.4 Calling sequences for data-link CM control routines

The CM control routine for the data-link consists of a subroutine with five
entry points.

LINKST (ARRAY, NSETN, NSET)

This is an initial entry point which must be called at least once to supply
parameters for the data-link FP routine and load the PP.

ARRAY is the array into which the data from the data-link must be stored.

NSETN is the maximum expected size of one record to be input from the data-link
(including the flag words). For fixed-length records this is also the only
size.

NSET is the number of records ARRAY can hold at one time, i.e. the number of
sections into which the input buffer is divided.

NSET must be such that 2 \leq NSET \leq 8

and ARRAY must be at least (NSETN-m) * NSET in length, where m is the number
of flag words.

On return NSET = 1 if start is normal

and NSET = 2 if start is after CDC 6600 has been dead-started.

This routine must be called before any input is requested from the link --
failure to call it will result in program stop.

In addition, it can be called if any of the above parameters need to be changed.

LINKIN (IBEG, LENGTH)

Whenever the calling program wants another record to work on, a call must be
made to LINKIN.

On return from LINKIN

IBEG is FORTRAN index to first word of required record in ARRAY;

LENGTH is the number of CM words in the record.

A return will be made only if a record has been successfully input from the
link. Otherwise, control remains in LINKIN which interacts with an operator via a
teletype.

LINKS

Whenever the calling program has processed a buffer section, a call must be
made to LINKS to release the section to the PP program to be filled with further
information. Failure to call LINKS will result in a cessation of transfers from
the link when the circular buffer has been filled once.

LINKOT (BUFFER, N, M, ESTAT)

A call to LINKOT initiates an attempt to output over the link to the SDS 920.
BUFFER is the start of the CM buffer in which the output is stored.

N is the number of words in the buffer.

M is the number of 6-bit characters (right adjusted) that it is desired to send from each word.

KSTAT is a status word in which the status of the transfer attempt is stored:

KSTAT = 0 indicates transfer successful.
KSTAT = 1 indicates output attempt not yet successful.

NOTES

a) The CM program will encode (pack up) the output into BUFFER before sending it. That is, it will take the M rightmost 6-bit characters in each word in BUFFER, reject all other characters, and store this information in a continuous string of characters starting at the leftmost character of BUFFER. BUFFER is thus destroyed by the call (at least in its original form).

b) If M \geq 10 or M < 0 no encoding is done, and all characters of BUFFER are output.

c) If an output is unsuccessful the program automatically attempts to output after each input (single or multiple record) has been made. Only when the attempt is finally successful is KSTAT put = 0. The routine assumes BUFFER still contains encoded information.

d) Until a previously made call is successful further calls to LINKOT will not result in a new output being started but will only result in another attempt being made to complete the previous output.

2.4.5 DSPOUT (ARRAY, NUM, KS)

Whenever an output to the display is wanted, a call to DSPOUT must be made.

ARRAY is the starting address of a buffer containing the coordinates to be output. Coordinates are assumed to be binary integers \(<1024\) and will be truncated by the PP if they are not. They can be packed in ARRAY in two forms:

i) One coordinate to a CM word in the order Y, X, Y, X, Y, X ...

ii) A pair of coordinates packed into the rightmost two 12-bit bytes of each CM word in the order (Y,X), (Y,X), ...

NUM is the number of points, i.e. the number of pairs of coordinates to be output to the display.

KS is a status word. If on entry

KS = 1 this indicates the display is to be viewed after this call and the coordinates are packed one to a CM word;

KS = 3 display is to be viewed after the call and coordinates are packed two to a CM word;
KS = 0 another call will be made before the display needs to be viewed and coordinates are packed one to a CM word;

KS = 2 another call will be made before viewing and coordinates are packed two to a CM word.

These status allow a display to be computed and built up by a number of successive calls to DSPOUT before the display is to be viewed. It means that a complete display need not be computed into a large buffer all at once -- but the display can be computed and sent serially using a smaller buffer.

On return

KS = 1 means the display has been successfully sent;

KS = 2 means that a hardware fault has prevented the display from being successfully sent. The reason for the fault will appear as a teletype message to the operator.

2.5 SDS 920 software -- a complete description

To test the data-link, and to use it with the on-line program developed for the missing-mass spectrometer experiment, two programs have been used; the first is a complete program, which was used during development and testing of the CDC 6600 programs, and the second is, in effect, a subroutine incorporated in the MMS on-line program.

2.5.1 Data-link test program

This program is designed to facilitate the development of the CDC 6600 data-link package and CM analysis program, and to confirm the feasibility of using the data-link when an experiment is running. To do this it uses a 'raw data tape', a magnetic tape previously written during an experimental run, from which to obtain valid data which it transfers to the CDC 6600 via the data-link. A maximum delay of 2 seconds between reading the data and sending it down the link is imposed -- this time, which is the approximate 7S cycle time for counter experiments, enables the program to determine whether the service of the CDC 6600 is sufficiently fast. The CM analysis program performs a complete analysis of the data, and the results can be compared to results obtained from analysis of the same 'raw data tape' by the MMS production program, run on the CDC 3800.

The program also accepts a transfer of data from the CDC 6600, this is in the form of BCD characters generated by the CM analysis program, which are typed when received.

2.5.2 Operation

The first section of the program carries out initialization - setting flags to the appropriate state, and setting up the entry points to the input/output and interrupt handling routines. The program then enters the control section, where it examines the settings of the console breakpoint switches, and accordingly either waits for operator intervention or proceeds to prepare data for transfer. A record is read from the 'raw data tape', and appropriate control words inserted at the
beginning giving the CDC 6600 data-link package information about the length and format of the data. According to the setting of the console switches, a second record may be read, and the two physical records set up as one logical record.

At this point the 'link operational' status is tested; if the link is non-operational, a message to the operator is typed and the program hangs up continuously testing this status; otherwise, the program sets a status to inform the CDC 6600 that it has a record ready to transfer, and enters a loop of duration 2 seconds. If at the end of this period the transfer to the link has not been made, a 'data lost' counter is incremented, and the program tries again.

The reply from the CDC 6600 to the 'ready to transfer' status is an interrupt to the SDS 920 core location 214 (octal), which contains a branch instruction to the link output routine. The output routine makes the transfer of the first physical record to the link, the SDS 920 remaining 'blocked' for the duration of the transfer. Immediately the transmission is terminated, the program resets the 'ready to transfer' status, unless the multiple transfer mode was requested, and waits 200 μsec in case the CDC 6600 detects a transmission error (incorrect parity bit for one of the words transmitted, or record length not corresponding to the data in the control word). If no error is detected the program either continues by reading another record, or in the multiple transfer mode, waits a further 200 msec for the interrupt requesting the second transfer. If the interrupt does not arrive within this time, an error is indicated and the program returns to the control section.

When the CDC 6600 detects an error in transmission, it sets a status to determine whether the error was caused by incorrect parity or by an incorrect control word, and causes an interrupt to core location 215 (octal) of the SDS 920. The error routine entered thus tries to transfer the data again, until a certain number of consecutive errors have occurred, or until a transfer has been successful. If the transfer is not successful after the specified number of tries, a counter is incremented and the program returns to the control section.

The operation of the test program when it is expecting a transfer of information from the CDC 6600 is little different from that described above; the major difference is that in addition to the status 'SDS ready to transfer', a status 'SDS ready to receive' is set.

The reply to this status, when data from the analysis program is available, is an interrupt to location 213 (octal). The input routine thus entered initiates the transfer of a block of data from the CDC 6600, and resets the 'ready to receive' status. The data received, which for the test is in the form of a message in SDS typewriter code, is typed, allowing a visual check on the arrival and accuracy of the data, plus an implicit internal check on the parity generated for the transmission.

If a parity error is detected, the typewriter is immediately disconnected, a counter is incremented, a status is set to inform the CDC 6600 that there was an error, and, if there have been less than a certain (variable) number of consecutive errors, a status is set to indicate that the SDS is ready for another try. If there have been too many successive errors, the program resets all status and returns to the control section, after incrementing a counter.
2.5.3 Data-link production program

The program developed for use of the data-link by the MMS on-line program consists of a subroutine which handles interrupts relating to the data-link, plus sections of coding which have been inserted 'in line' into the appropriate control sections of the MMS program. This approach was adopted in order that the data-link could be made to conform easily and quickly to the conventions used by the MMS program during its previous 12 months of production.

Operation

Since any I/O device (including the data-link) monopolizes the single data channel of the SDS 920, I/O requests must be treated strictly in sequence and by priority. Hence it was decided that, for convenience, the data recorded on magnetic tape by the MMS program should be dispatched down the link as soon as possible after the tape writing terminated. This has the advantages that, in general, there is sufficient time after writing tape, and before the arrival of the succeeding FS burst (which requires highest priority on the I/O channel) to perform the transmission, and also that the data received by the analysis program in the CDC 6600 is in the same format, whether it comes directly via the data-link or indirectly via a magnetic tape.

At the end of a tape-writing operation, the data-link status is tested; if it is "non-operational", the MMS program continues normally. If the data-link is 'operational', then the link buffer status is examined to see if the previous contents have been transmitted (if not, this is recorded) and the contents of the tape output buffer are transferred en bloc to the link buffer, a status 'SDS ready to transfer' is set, and the program continues with its interrupted task.

The response of the CDC 6600 is an interrupt to location 214 (octal) of the SDS 920; the input routine checks the status of the I/O channel, either terminating any I/O of lower priority than the data-link, or setting a flag to inform any higher priority I/O that there is a data-link transmission pending. When the channel is clear, the current data is transmitted down the link, and the program waits for 250 μsec in case of transmission error. If there is no error, the program restarts any interrupted I/O operation, and continues to compute. In the case of a transmission error, the program attempts to transmit the data again, to a maximum of five attempts, before recording the failure, and considering the data as irrecoverable.

If the data has not been transmitted to the CDC 6600 by the time of arrival of the next FS burst, the 'ready to transfer' status is reset; the arrival of the 'End of Burst' interrupt from the FS causes this status to be set once more.

There was no transmission of data from the CDC 6600 to the SDS 920, only because of a shortage of facilities of the latter machine -- lack of core space, and no peripheral equipment suitable for displaying large amounts of data.
3. **MISSING-MASS SPECTROMETER EXPERIMENT**

The MMS experiment has, since its inception, used a computer on-line to provide some feedback of information to the experiment. Initially, the Ferranti Mercury computer was used, connected by a data-link to the experimental area in the South Hall; this computer was used to perform sample calculations and produce statistics, the raw data being recorded by a magnetic tape unit at the experiment.

For the continuation of the experiment, in autumn 1965 the Mercury computer was replaced by the SDS 920 computer, which performed data acquisition and checking, and also recorded the raw data on tape.

During this time, the experiment as a whole was monitored by the 'bicycle on-line' method, first to the IBM 7090 at CERN, then to the IBM 7040 at EPFL, Lausanne (this was, in fact, 'VW 1200 on-line'), and finally to the CDC 3400 and CDC 3800 at CERN. The direct data-link was incorporated in late autumn 1966, since it is a much more efficient way of monitoring the experiment by virtue of its capability of quick turn-around.

3.1 **The CERN missing-mass spectrometer**

This was put into operation in the South Hall in autumn 1964. It has been proposed, designed, and realized by the Maglò group, with the invaluable help of NP and DD specialists through all the stages of the experiment. A list of the main publications pertaining to this experiment will be found in the Appendix; the references refer the reader to this list.

The principle of the method is the detection of peaks on top of the phase space in the so-called "Jacobian peak" region. Although the method can be applied to a variety of processes, the instrument was used only in the negative pion beam \( \Delta \), in order to investigate systematically the spectrum of non-strange bosons of negative charge \( X^- \) in the reaction:

\[
\pi^- + p \rightarrow p + X^-.
\]  

(1)

Incident momenta between 3.0 and 13.0 GeV/c were used covering a missing-mass band from 0.6 to 2.4 GeV.

The kinematics of the two-body process described in reaction (1) gives a relationship between the proton lab. angle and its momentum for a fixed incident pion momentum and a fixed \( X^- \) mass. The only quantity which must be measured with great accuracy is the angle between the direction of the incident pion and that of the recoil proton.

Precision is achieved by using two sets of five overlapping hodoscope counters on the pion line, and a system of two sonic chambers of two gaps each on the proton line. A third set of hodoscopes is intended to provide additional information on the incident momentum of the pions. (Diagram IV.)

A combined range/time-of-flight method allows the identification of the proton in the trigger. This trigger is improved (reduction of background) if one requires that at least one of the V-counters fires, since \( X^- \) yields at least one charged particle in its decay.
EXPERIMENTAL SET UP
<table>
<thead>
<tr>
<th>Symbol</th>
<th>Physical quantity</th>
<th>Type of digitization</th>
<th>Number of scalers</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\theta$</td>
<td>Angle of incident proton</td>
<td>Coincidence H.E.</td>
<td>2 scalers</td>
<td>More accurate determination of the direction of the incident proton</td>
</tr>
<tr>
<td>$N_\mu$</td>
<td>Number of incident kaons/e-protons</td>
<td>Time-to-amplitude converter + pulse-height analyzer</td>
<td>2 scalers</td>
<td>Cross-section measurement</td>
</tr>
<tr>
<td>$E_\gamma$</td>
<td>Angle of recoil proton</td>
<td>Signals from scintillation counters</td>
<td>16 scalers</td>
<td>Recoil proton identification</td>
</tr>
<tr>
<td>$g_\gamma$</td>
<td>Number of recoil products (rough)</td>
<td>Scintillators</td>
<td>1 pattern unit</td>
<td>Very accurate measurement of the recoil proton angle</td>
</tr>
<tr>
<td>$g_\gamma$, $g_\mu$</td>
<td>Number of decay products (better)</td>
<td>Matrix of scintillators</td>
<td>2 pattern units</td>
<td>Information on the resonance produced</td>
</tr>
<tr>
<td>$G_{\gamma,\mu}$</td>
<td>Direction of the decay products</td>
<td>Magnetic analyzer or high-precision crystals</td>
<td>32 scalers</td>
<td>In the case of single-charged pions, possibility of finding the effective mass</td>
</tr>
<tr>
<td>$\mu$, $\kappa$, $\Delta\lambda$, $\Delta\epsilon$</td>
<td>Identification and parameters</td>
<td>Standard pulser units</td>
<td>Manual switches</td>
<td>Miscellaneous information, muon No. calibration parameters, etc.</td>
</tr>
</tbody>
</table>

| Total X (24-bit word) |
The recoil momentum need not be measured with great accuracy, but owing to the large mass bite of the instrument it was desirable to introduce a system of differential range in order to select, for a given mass, the really flat region which provides the best resolution. This information is redundant with the times-of-flight measured between the T-counter and each of the R1 and R2 counters.

Although this system can work as described and has proved its usefulness, it provides but scant information on the resonance produced, namely proof of its existence, its mass, width, and differential production cross-section. In order to improve this situation, two new elements have been introduced: a vertex matrix of 72 scintillators, and a system of wire chambers to provide information about the multiplicity of charged decay products, and, in the special case of three charged pions and no neutrals, i.e.

\[ X^- \rightarrow \pi^- + \pi^- + \pi^+ \] (2)

a complete determination of directions and momenta of the outgoing particles. The detailed specifications and performance of each part of the system are described elsewhere \(^{7-13}\); their functions and means of digitization are presented in Table 1.

3.2 Data acquisition and data handling for the experiment

During the lifetime of the MMS experiment, both the data acquisition and data processing systems changed and expanded; this description of the data processing programs refers to their status in autumn 1966.

For each event a total of 73 24-bit scalers (or the equivalent) must be recorded. The recovery time of the spark chambers, which is about 10 msec to be quite safe, imposes an upper limit to the number of events recorded of about 15 per burst, though rarely did this figure exceed 10 events.

Given the dead-time of the system (\(\Delta t\)), we can estimate the fraction of events lost (\(x\)) as a function of the number of free triggers (\(N\)) and of the burst length (\(t_{\text{burst}}\)). The number of free triggers depends upon the intensity of the beam, the reaction under investigation, and the acceptance angles; the value is obtained by running without the data acquisition system (i.e. the dead-time is zero).

If \(n\) is the number of events lost, the system will be busy for a total time

\[ t_{\text{busy}} = (N-n)\Delta t, \]

with the relation

\[ x = \frac{n}{N} = \frac{t_{\text{busy}}}{t_{\text{burst}}} \]

that is

\[ x = \frac{N\Delta t}{(t_{\text{burst}} + N\Delta t)} \]
Thus with the imposed dead-time of 10 msec, and a burst duration of about 300 msec, we already lose 50% of the triggers, and get only 15 events/burst from 30 free triggers.

The SDS 920, equipped with a direct memory access feature and connected to the experiment by a CERN-designed interface (Diagram V), was able to read the information in the 73 scalers in less than 1 msec, and was programmed to accept up to 15 events per burst, writing this data onto magnetic tape in constant length blocks with a fixed format. In addition, the program kept a constant watch over the performance and efficiency of the sonic chambers, and the distribution of single counts in each scintillator of the vertex matrix, producing a summary of this information at the request of the physicist.

The off-line program existed in two principal versions, a 'bicycle on-line' version and a data-link version; the former was run on the CDC 3800 computer and the latter on the CDC 6600 computer. All events produced pass through the processing program; a data summary tape is produced from the good events, whilst detailed statistics of the types of failure are compiled and recorded. The data-link version had the added facility that statistics could be printed on demand, and certain histograms could be displayed on the TV set situated in the experimental area.

3.3 The processing program

The program described is the version used on the CDC 3800 computer, and subsequently converted to the CDC 6600. The construction is that of a master program, which does all book-keeping, handles error conditions, produces the data summary tape of good events, and passes control, as required, to the following 'blocks':

**SETUP**
which reads constants describing the geometric layout of the experiment, and the constants needed to interpret the digitized electronics measurements.

**INPUT**
which reads and unpacks the raw data tapes produced by the SDS 920 -- these tapes contain 15 events per record. If a redundancy is encountered, no attempt is made to recover the information in that record. Various statistics are returned to the control program; e.g. the numbers of records read and redundancies, and whether an end-of-file has been encountered.

**SONIC**
is a block of subroutines which checks that the sonic information is within prescribed limits, calculates for each gap X, Y, V, and Δt i.e. the spark coordinates, the velocity of sound in the chamber, and the shock wave effect; these must fall within certain limits in order that the gap as reconstructed be accepted. For an event to be acceptable, all four gaps must give an acceptable reconstruction, and the $X^2$ test for the line fitted must be smaller than a given limit. An attempt is made to 'rescue' the event in the case where only three gaps are 'good' by using combinations of three microphones from the faulty gap. The results from this routine are the three direction cosines of the recoil proton path.
is a subroutine which decodes the information from the incident hodoscopes, calculates the coordinates of the interaction point, and checks that this point is within the target.

which makes corrections to the time-of-flight value for the real path, and performs all the relevant kinematic calculations up to the 'missing mass'.

which decodes the information from the matrix hodoscope, and finds the number of decay products of the resonance.

which calculates the positions of multi-sparks in the wire chambers, which were between the target and the vertex matrix; and finds the directions of the outgoing particles.

When either a change of run-number or an end-of-file on the input tape is detected, there are two subroutines used to produce and print for the terminated run statistics on the performance of the equipment, and as much of the physics results as can be obtained at this stage. For the CDC 6600 version of the processing program, these routines produced the data which was displayed on the TV screen at the experimental area.

4. CONCLUSIONS AND STATISTICS

As already pointed out, the production runs with the MMS experiment were conducted as an initial trial to verify the reliability and to determine the usefulness and the deficiencies of the system. A summary is given of the limitations observed and of some statistics obtained from the first on-line run.

i) The dynamic characteristics of the CBHEN work-load with regard to requirements of turnaround and throughput can only be partially satisfied within the automatic scheduling of the SIFROS multiprogrammed operating system. Some over-all manual scheduling is a necessity. This scheduling required that on-line runs be taken in periods of 3-5 hours once or at most twice per day. Use of the facility for longer periods, or more frequently throughout the day, would only be justified in cases where an experiment had extreme urgency or priority within the experimental program, because of the general effect on turnaround and throughput for the remainder of the users.

ii) During operating periods, a processing program of 25 K was permanently resident in CM although on the average only 3-4% of central processor time was used by the program. The amount of central processor usage was limited by the data-taking rate at the experiment which was only 15 events every 10 seconds or so (in terms of data transferred to the CDC 6600). If full-scale use of the facility is envisaged, a central processor usage more commensurate with CM utilization is desirable. A central processor usage of 3-4% is only justified in a system with roll-in/roll-out capabilities.
iii) A PP was permanently allocated to servicing the data-link. (The multiplexer subchannel monitoring was not used initially.) Shortage of pool PP's is an important factor limiting throughput in SIPROS.

iv) Data transfer requests from the SDS 920 were not always fulfilled. The SDS 920 would only transmit data corresponding to a request, provided that the CDC 6600 responded quickly enough after the request flag had been set, i.e. during the time interval between the end of the transfer of data to SDS 920 tape and the beginning of the next proton synchrotron burst. Although the CDC 6600 program had the highest priority within the multiprogramming system, so that responses from the Executive were sufficiently rapid, holdups occurred if either

a) no pool PP was available to transfer the summary data to magnetic tape (SIPROS FORTRAN I/O is unbuffered);

b) there was a request from the teletype operator for a summary of partial results: in this case transfer of printing to disk caused delays.

In both cases, response to the SDS 920 ready signal was not possible and so data was not transferred.

v) A display of a scatter diagram was sent automatically every 250 good events which occurred every 15 minutes or so. Other displays (three were available) were optional on demand from the operator at the teletype. (Examples of the display types are given in diagrams VI and VII.) Significant statistics were available only after one hour of running. However, as has been pointed out, the data-taking rate was low, and more significant statistical quantities could have been devised and displayed had there been time. Improvements in this area are, however, probably only practical in an operating system with backing stores and overlay facilities which are readily and easily available to the on-line user. A library of parametric programs for various types of displays, which may be selected during the on-line run as necessity dictates, is then feasible.

vi) Loss of statistics through any case of stoppage or unexpected breakdown of the on-line system makes the use of backing storage for predigested data, scatter diagram and histogram accumulations very desirable. Runs can then, if necessary, recommence at the place where stoppage occurred, without loss of continuity. This was not possible in the current implementation. A useful by-product would be the ability of the on-line user to reprocess his predigested data according to a different set of parameters and so gain insight into the best way to adjust and optimize his experimental conditions.

vii) The missing-mass experiment was a well-established one with well-defined data analysis programs. In general, however, experiments using this type of facility could have analysis programs in the final stages of development. Provision should therefore be made for improved diagnostic capabilities and possible recovery in case of program errors, and program correction and resubmission.
POLAROID PICTURES OF TV RECEIVER IN THE SOUTH HALL.

DIAGRAM VI

Plot of the missing mass

DIAGRAM VII

Number of particles in each quadrant of the vertex matrix and in total
Statistics

Some figures accumulated during a typical 90 minute period:

<table>
<thead>
<tr>
<th>Description</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Starting time of run</td>
<td>10.45.10</td>
</tr>
<tr>
<td>Real time of run</td>
<td>5385.83 seconds ~ 90 minutes</td>
</tr>
<tr>
<td>No. of transfers</td>
<td>510</td>
</tr>
<tr>
<td>No. of triggers</td>
<td>7644</td>
</tr>
<tr>
<td>No. of good events</td>
<td>3682</td>
</tr>
<tr>
<td>No. of rescued events</td>
<td>295</td>
</tr>
<tr>
<td>No. of bad events</td>
<td>3667</td>
</tr>
<tr>
<td>Execution time of run</td>
<td>206.73 seconds</td>
</tr>
<tr>
<td>Percentage usage of CP</td>
<td>3.84</td>
</tr>
<tr>
<td>Real time per transfer</td>
<td>10.56 seconds</td>
</tr>
<tr>
<td>Execution time transfer</td>
<td>0.4053 seconds</td>
</tr>
<tr>
<td>Execution time per trigger</td>
<td>0.0270 seconds</td>
</tr>
<tr>
<td>Execution time per good event</td>
<td>0.0561 seconds</td>
</tr>
<tr>
<td>Transfers lost</td>
<td>55</td>
</tr>
<tr>
<td>Summary print-outs</td>
<td>2</td>
</tr>
<tr>
<td>Displays</td>
<td>8</td>
</tr>
</tbody>
</table>

* * *

Experience with the current facility shows its feasibility and a limited usefulness, but it also shows that the requirements of the physicist and the general throughput of the CDC 6600 would be better served by the following features.

i) Data collection on to some form of random access backing store preferably separate from the CDC 6600.

ii) Roll-in of the data analysis program only upon request from the physicist, or when sufficient data has been accumulated on the backing store to give statistically significant feedback to the experimentalist. This would be at most several times per hour in current experiments.

iii) A system incorporating remote diagnostic program modification and updating capabilities on an on-line basis compatible with CDC 6600 efficiency.

These requirements will become available with the implementation of FOCUS'5), a Facility for On-Line Computations and Updating Services, and with the changeover of the operation system of the central computer to CERN Scope and its subsequent improvement program. The FOCUS system will use a CDC 3100 with disk storage as a nucleus, providing facilities for the storage, editing, and transmission between remote users and the central computers of both program and data files. Data collected on data acquisition systems (i.e. small computers) situated in the experimental halls will be transmitted to the CDC 3100 disk under the control of the users teletype console. This console will allow the user to input and edit program files within the FOCUS system, and to transmit jobs to the
central computing system. The addition to the CERN Scope of roll-in/roll-out, further
diagnostic capabilities (TRADE), and retransmission facilities to the CDC 3100 will
contribute to the effectiveness of the FOCUS while maintaining the efficiency of the
central computing installation.

Acknowledgements

We wish to acknowledge all the help received from technicians, engineers, and
computer operators whose valuable contributions made this experience possible, and in
particular to thank F. Marciano who designed the link hardware, D. Brown who built the
display, and the NP Electronics Group who built the SDS 920 data link interface.

We wish to thank the physicists and technicians from the Missing Mass Spectrometer
Group, and especially Dr. B. Maglić, for their comprehension and interest, and Division
Leaders G.R. Macleod and P. Preiswerk for their encouragement.
1. The program is loaded from a deck of binary cards in the usual fashion and is normally controlled from teletype 1 by the computer console.

2. Immediately after the program is loaded in CM the message

   ** PROGRAM WAITING
   ZRSPF!Z

   should appear on the teletype.

3. To set the program going it is only necessary to type GO on this teletype. Messages are put in, in the usual form

   Control	Control
   +A	 MESSAGE	+C

   and no blanks must appear between the start of the message and the first letter.

4. If all goes well the program should proceed after typing GO without any further operator intervention. However, there are a number of error conditions and abnormalities that can arise as follows.

5. Action on dead start

   If it is necessary to dead start the 6600 for any reason during a data-link run, the data-link program should be loaded immediately the CDC 6600 is ready (before the Luciole program) and the message

   GODS

   typed on the teletype as soon as it asks for a response.

6. There are a series of other possible error conditions that can arise. When these occur they give rise to teletype messages of the type.

   ** NO INACTIVE AFTER FUNCTION SENT SW = 4400 0000 0000 0000 0200
There are seventeen of these messages. The program will attempt to recover automatically from these errors. However, if too many of them occur in succession the program will stop and ask for operator action with the message

```
** TOO MANY SUCCESSIVE ERRORS
** PROGRAM WAITING
ZRSVPIZ
```

Typing GO will then restart the program. However, if this situation occurs frequently or consistently a programmer should be called.

7. Particular attention is drawn to the message

```
** LINK NOT OPERATIONAL
```

If this occurs the data-link should be checked (it is situated behind the one-inch tape units at the left of the tape light controller), and if necessary the "operational on" button pressed. This is the leftmost button of the row at the top right of the left-hand rack of electronics.

8. A complete list of messages that can be input by the operator to the program is as follows:

<table>
<thead>
<tr>
<th>Message</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>i) GO</td>
<td>Starts the program into normal operation after any interruption. If typed while the program is running will have no effect.</td>
</tr>
<tr>
<td>ii) GODS</td>
<td>Used to restart the program for the first time when the program has been reloaded after a dead start has interrupted running. It causes the program to read a tape to reset its storage to the conditions before the dead start. If typed at any time other than at the beginning of the program, it has the same effect as a GO instruction.</td>
</tr>
<tr>
<td>iii) WAIT</td>
<td>Causes the program to pause, drop its PP, and wait for further operator instructions from the teletype. Should only be used in normal running in a case of absolute necessity as it can result in loss of data from the link.</td>
</tr>
<tr>
<td>iv) STOP</td>
<td>Used to terminate the program at the end of a run.</td>
</tr>
<tr>
<td>v) DUMP</td>
<td>Causes the PP program and the CM control routines to be dumped on the line printer. If typed while the program is running, the program will continue operation immediately. If typed while the program is waiting, the program will wait again after the dump is complete.</td>
</tr>
</tbody>
</table>
vi) PRINT
Causes the current program print-out to be output on the line printer. Has the same effect on program running as the DUMP instruction.

vii) SUMOUT
Causes the CM production program to calculate and print the results of the run up to the moment SUMOUT is typed. Has the same effect on program running as the DUMP and PRINT instructions.

viii) RESTART
Restarts the CM analysis program from the beginning zeroing all variables and histograms. Should only be used under programmer instructions.

If any other messages except these and those connected with the display (below) is typed on the teletype the message

** MESSAGE NOT UNDERSTOOD

will appear. If the program is running when the message is typed in, it will continue to run. If it is waiting, it will wait again.

.ix) Display data
The program has been written so that part of the current output can be displayed on a cathode ray storage tube. Several operator instructions are connected with this display. They are

a) DISPUP
Initially the program is adjusted so that any attempt to display data will be ignored. This is so that displays cannot be initiated automatically from the program when the CRT is not connected. The command DISPUP typed on the tele-type causes the display section of the program to be activated.

b) DISPDOWN
This disconnects the display part of the program if it has previously been connected by a DISPUP instruction.

c) DISP1
d) DISP2
e) DISP3
There are three separate displays which can be initiated from the teletype. If the display is connected these displays can be activated by typing DISP1, DISP2, DISP3, respectively.

E. Palandri
3.11.1966

* * *
REFERENCES


2) Interim report of the Steady Group on the on-line use of the CDC 6600 computer, CERN-DD/Co/65/3 (1965).


4) D. Ball, J. Garratt, E. McIntosh, P.J. Mercer, M. Palandri, N. Spoonley, P. Warn, The SIFROS operating system (CERN report to be published).


9) F. LeFebvre, B. Levrat, H.R. Blieden, L. Dubal, Maria N. Focacci, D. Freytag, J. Geibel, W. Kienzle, B.C. Maglić, Boson spectrum from 1.1 to 1.6 GeV and properties of $\phi$ meson observed by the missing-mass spectrometer (corrected version), Physics Letters 12, 438 (1965).


15) P.M. Blackall, Focus — a development project for remote on-line access to the CERN central computers, DD/PMB/DISP/38 (1967).