digital EQUIPMENT CORPORATION
MAYNARD·MASSACHUSETTS

PEPR And OTHER SYSTEMS

A. H. Rosenfeld

University of California
Berkeley, California
and

H. D. Taft
Yale University
New Haven, Connecticut

MODULES · COMPUTERS · SYSTEMS

PePR

RAPID DATA PROCESSING

E. C. Fowler

October 22, 1962

Original

A number of reports concerning the development of the Flying Spot Digitizer (FSD or Hough-Powell Device), the Scanning Measuring Projector (SMP), and the Precision Encoding Pattern Recognition (PEPR) device have been given previously. In particular, the people who are leading these developments gave progress reports at the recent CERN Conference on instrumentation. These same people reported again at a Rapid Data Processing Conference held at the Brookhaven Laboratory on September 20 and 21, 1962.

All of us who hope to use some of these devices or exact copies of them can see clearly that we shall owe a great debt to the people who are developing them, and we are grateful to them for devoting so much of their time to updating our knowledge of these developments.

Several of us thought that it would be worthwhile to produce a written record of this informal meeting which would contain the most recent developments which have not been described in previous reports. Enclosed you will find the results of our effort to achieve this.

12

PEPR and Other Systems

A. H. Rosenfeld

University of California
Berkeley, California

and

H. D. Taft

Yale University
New Haven, Connecticut

This report represents an attempt to present a logical sequence of programs leading from bubble chamber film through the complete kinematical analysis of events. Since a number of systems already exist to handle the latter part of this problem, the emphasis here is placed upon the automatic scanning and measuring problem as well as upon the preparation of the data for the existing systems. In particular, Part I of this report is a detailed proposal for programming the Precision Encoder and Pattern Recognition device (PEPR) at present under development at MIT.(1) Part II contains a review of the leading automatic scanning and measuring systems as well as a proposal for unifying the output of these systems and processing this output. It is interesting to us that it now appears to be possible to outline a complete program which encompasses automatic scanning, measuring, and pattern recognition by relatively straight forward methods without recourse to any very sophisticated ideas or any theorems about the "discipline" of pattern recognition.

PART I - A Program of PEPR

Overall Program:

The proposed preliminary program for PEPR consists of three principal phases. Phase 1 consists of two orthogonal area scans designed to establish one bank of data (less than 20 words) in core for each track segment. This scan will be designed to have a high efficiency for detecting track segments and may even detect more track segments than are present in order to make sure that none are missed. This situation is remedied in Phase 2 where each track segment is followed its full length. The end points of each segment are thus determined and at each end a rotary scan is made and the angles of linked segments are recorded. Redundant segments may now be eliminated and clusters of track segments may be formed by matching end points and angles. Phase 3 then consists of the precision encoding. At the present time, a possible procedure would be to obtain and record roughly 10 precision points on each track segment involved in a cluster. Not all of this data would necessarily be used later, but the "scanning event type" programming could then be deferred to a more generally available computer such as a 7090. These three phases would then be repeated for each of three views. Before writing an output tape a number of consistency checks could be made to insure that the same topological pattern had been found in all views and that all vertices lay inside the visible volume of the chamber. At this point the output data tape for this frame can be written while the film is being advanced. In a one view PEPR,the consistency checks must also be deferred until all three views are present simultaneously in the 7090.

Definitions (illustrated in Figure 2)

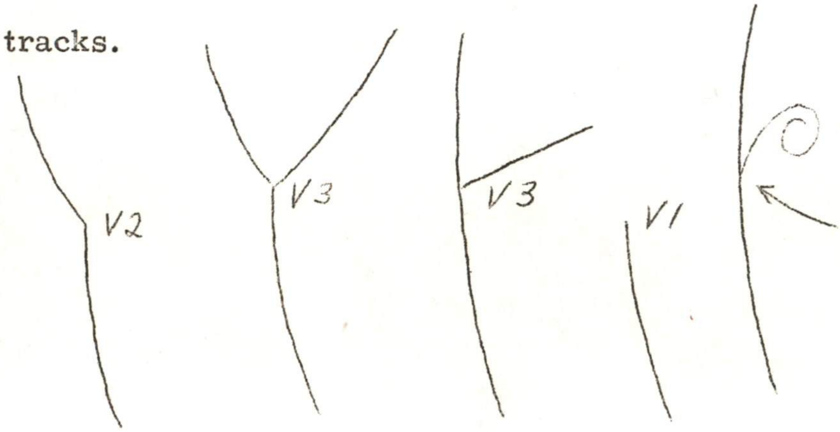
Track element - A length of track less than 2 mm long on the film and straight enough to be recognized by the PR scan as a single correlated set of "hits" at 1^o angular intervals.

Track Segment - A curved length of track which has no kink detectable by PEPR.

Track (i. e., projected track) - The result of following a track segment in both directions until it meets a "vertex".

Vertex - A track ending or the intersection of two or more track segments or a kink of $> 1^{\circ}$, provided all these are in a fiducial area. See Figure 1.

Cluster (i. e., a cluster of tracks made by charged particles) - An assembly of linked tracks.



Not a vertex if δ ray rejected as a track.

Figure 1. Vertices

Fiducial Area - For each view a different fiducial area will be defined. Area scanning covers this area. Any track which is track followed outside this area will be ~~chopped off and~~ defined as ending where it crosses this boundary.

The two orthogonal scans are given separate suggestive names: the scan which sweeps ~~in x~~ across the beam and detects beam tracks and tracks tending in the beam direction will be called the "active" scan because it finds most of the data. The orthogonal scan is called the "inactive" scan.

PEPR CONTROLLER INPUT REGISTERS
PR MODE

Figure 3

<table>
<tr><th>Bit
Word</th><th>0</th><th>1</th><th>2</th><th>3</th><th>4</th><th>5</th><th>6</th><th>7</th><th>8</th><th>9</th><th>10</th><th>11</th><th>12</th><th>13</th><th>14</th><th>15</th><th>16</th><th>17</th></tr>
<tr><td>A</td><td colspan="12">X COORDINATE
(STATIC)</td><td colspan="2">VIEW</td><td>Mode
1</td><td colspan="3"></td></tr>
<tr><td>B</td><td colspan="12">Y COORDINATE
(STATIC)</td><td colspan="3">X OR Y BITE</td><td colspan="3"></td></tr>
<tr><td>C</td><td colspan="7">UNBLANK
(STATIC)</td><td colspan="7">BLANK
(STATIC)</td><td colspan="4"></td></tr>
<tr><td>D</td><td colspan="8">INITIAL ANGLE
(DYNAMIC)</td><td colspan="8">FINAL ANGLE
(STATIC)</td><td colspan="2"></td></tr>
<tr><td>E</td><td colspan="18"></td></tr>
<tr><td>F</td><td colspan="14"></td><td colspan="4">READOUT
COMMUTATOR</td></tr>
</table>

PEPR CONTROLLER INPUT REGISTERS

PE MODE

FIGURE 4

Bit / Word	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
A	X COORDINATE (DYNAMIC, NOT PLANTED)												VIEW		Mode	PE Data 0 for 1 for	Tracks Rasters	
B	Y COORDINATE (DYNAMIC, NOT PLANTED)												X or Y BITE					
C	DISABLE PHOTOMULTIPLIER (STATIC)														Init. Raster Counters XR 0 for clear 1 for set	YR	PE Sweep Direction 0:X 1:Y	0:up 1:down
D	INITIAL ANGLE (STATIC)																	
E	COUNT-OVER ADDRESS AND/OR ENABLE PHOTOMULTIPLIER												Count-Over Rate 0 for 1 for	Counting Last rate & servo			Count-over Direction 0:AX 1:BY	0:up 1:down
F															READOUT COMMUTATOR			

All these definitions are now summarized in Fig. 2. Consider the right-hand track of the V. This whole track is actually track-followed twice, once by starting downbeam from the segment found in the active scan, once by tracking up from that in the inactive scan. Later these two tracks are found to be identical.

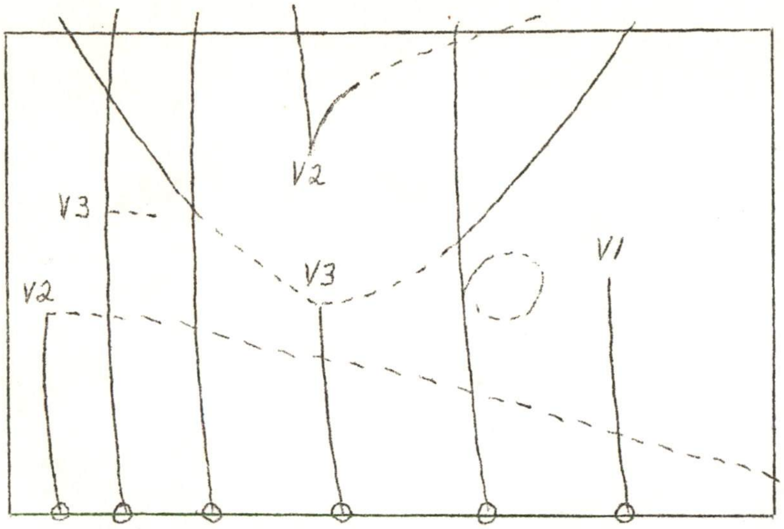


Figure 2.

————————	found in active scan
-----------	" " inactive scan
•	Vertex
o	Beam track entry
vN	Vertex of N tracks

PEPR Commands and Registers

The circuitry and overall lay-out of PEPR have been described previously.(1) We present here details of the flow of information from the PDP-1 computer(2) to the PEPR Controller which are necessary to program for this system.

Six input registers labelled A, B, C, D, E, and F exist in the Controller and have the format shown in Figs. 3 and 4. The function of each of these registers is described on the figures. These registers must be loaded from the I$\overline{\text{O}}$ Register of the PDP-1 using the following set of commands. Note that the IOT instructions are all of the 72 rather than 73 class. This is because no return pulses are sent from PEPR to signify completion of its task; therefore, the computer must not stop and wait.

Load A register IOT (72)NN12

NN=

Segment	Bits	NN (octal)
1	16, 17	01
2	14, 15	02
3	12, 13	04
4	0 thru 11	10

Load B register IOT (72)NN13

NN=

Segment	Bits	NN (octal)
1	15 thru 17	01
2	12 thru 14	04
3	0 thru 11	10

Load C register IOT (72)NN14

NN=

Segment	Bits	NN (octal)
1	16, 17	01
2	14, 15	02
3	0 thru 13	10

Load D register IOT(72)NN15

NN=

Segment	Bits	NN (octal)
1	0 thru 15	10

Load E register IOT(72)NN16

NN=

Segment	Bits	NN (octal)
1	16, 17	01
2	12 thru 15	04
3	0 thru 11	10

Load F register IOT(72)NN17

NN=

Segment	Bits	NN(octal)
1	14 thru 17	01

Within the IOT set for any given register, any combination of segments can be loaded by a single instruction by simply adding the NN bits for the segments involved. For example, segments, 1, 2, and 4 of the A register can be loaded simultaneously by IOT 1312. A given IOT instruction, however, can operate only on a single register.

The F register is used exlusively to control readout from the Controller; the high-order bits are not used and bits 14 → 17 control the readout commutator which selects the contents of 1 of 16 different registers for transmission back to the PDP-1. As things stand at the moment, only ~~14~~ 15 positions of the readout commutator are employed and the words will appear in approximately the following order:

Readout commutator position

0: PR flag~~s~~ word

1: PE flag~~s~~ word

2: D_{0-7} (angle for PR mode)

3: Readout buffer No. 1
4: Readout buffer No. 2
5: Readout buffer No. 3
6: Readout buffer No. 4
7: Readout buffer No. 5
8: Readout buffer No. 6

The cordinates of up to six tracks in one scan (i. e. one PE sweep in the PE mode; or one high frequency sweep at one particular angle of line-segment orientation in the PR mode).

10: A_{0-11}
11: B_{0-11}
12: X Raster Counter, bits 0 - 11
13: Y Raster Counter, bits 0 - 11
14: Servo Counter, bits 0 - 11

In PE mode Only

The PR flag word consists of 18 bits of which the first 4 have the following significance:

1. Angle Count Stopped (i. e., either 2 or 4 happened)
2. PR Data
3. Overflowed Buffers
4. Angle = Final Angle

The PE flag word consists of 18 bits of which the first 8 have the following significance:

1. E Identity (Open gate in PE mode)
2. C Identity (Close gate in PE mode)
3. PE track Data
4. Overflowed Buffers
5. Count Over Counter Overflow
6. Servo Counter Overflow

7. Raster Error

8. PE Raster Data

The Readout Buffers, 1 through 6, in the Controller have the following format:

For future 40 Mc Interp.

$$\overbrace{0\ 1\ 2\ 3\ 4\ \underbrace{5\ 6\ 7\ 8\ 9\ 10\ 11}_{\text{Interpolation Bits (PR)}}}^{\text{Course Bits (PE)}}\ \ \overbrace{12\ 13\ 14\ 15}^{\text{Interp. Bits (PE)}}\ \ 16\ \ 17$$

1 indicates last piece of data (PR mode only)

To read information from the Controller into the PDP-1 IŌ register the Readout Commutator (F Register bits 14 ⟶17) must be set for the desired output buffer and the command IŌT (72) 4013 must be given. The command IŌT (72) 2013 is also provided to advance the Readout Commutator by one. These commands are given this structure so that the readout command will simultaneouly advance the Readout Commutator using the command "or" feature of the PDP-1.(2) A further command IŌT(72) 4112 is provided to start the CRT PR sweep. Six additional commands IŌT(72) 4212 thru 4712 have also been provided but are as yet unassigned.

Input register segments in Figs. 3 and 4 are labelled "Not Planted," "Static" or "Dynamic" to indicate whether or not the information must be entered from the PDP-1 and whether or not this information is changed internally by the Controller. In particular ~~for instance~~, the initial angle (D bits 0 →7) is actually used in the PR mode as a counter by the controller and the PR scan is stopped when a coincidence is achieved between D bits 0 → 7 and D bits 8 → 15 (the final angle). At any point during the PR scan D bits 0 → 7 represent the current angle of scan. The Controller is so wired that the angle counter will be advanced by 1° at the end of the scan on which data is found unless the Readout Buffer overflow has been set. Similar remarks apply to the other registers which are "dynamic."

An important feature of the PDP-1 computer is the "sequence break" system.(2) Essentially, at a given signal the status of the machine is stored and a transfer is made to a predetermined location in memory. The interrupted computation can then be resumed at any later time determined by the part of the program responding to the sequence break signal. Sequence breaks will be initiated in PEPR by the following conditions:

1) Data present in PR mode
2) Angle equals final angle
3) Data present in PE mode
4) E Identity (PE mode) (Open gate)
4) C Identity (PE mode) (Close gate

Each of these sequence breaks will operate through a separate channel and may be handled separately by the program according to the requirements of each situation.

Phase 1. The Area Scan

a. Scan Procedure

We define a coordinate system as shown in Fig. 5. The fiducial region of the chamber is divided into square cells 2 mm on a side (all dimensions are on the film). Typical large chamber pictures would then have 30 cells across the beam and 60 cells along the beam.

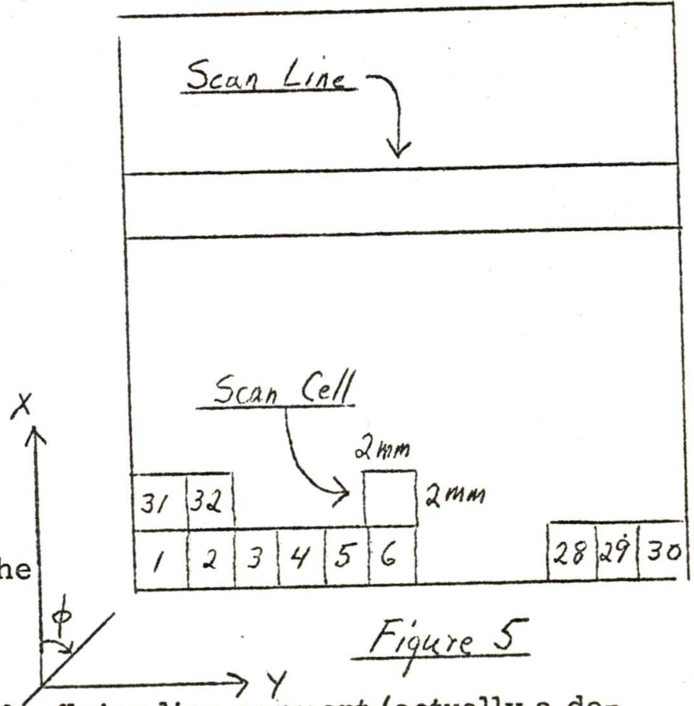


Figure 5

An important feature of PEPR is its use of a flying line segment (actually a defocussed spot) for scanning in the PR mode. This line can be rotated in 1° angular intervals and will describe the pattern illustrated in Fig. 6. The X or Y bite can be adjusted from 2 mm to 0 and the lengthening of the line to achieve this constant bite is automatically done by PEPR. Using this patter, the cells will be scanned one at a time starting at X = Y = 0 and proceeding first across the film increasing Y by 2 mm for each cell and increasing X by 2 mm for each scan line. It is then a property of the PEPR scan pattern shown in Fig. 6 that if a track element has a direction between -45° and +45° it will be recognized as a track element once and only once on a scan line. It is, therefore, possible, using the position and slope of the element, to predict in which cell (or in which of two cells at worst) the element will be found in the next scan line. When a new track element is recognized, a track bank is set up for it in core and a prediction is made. In scanning any cell (other than cells in the first scan line) all predictions are checked against the new track elements obtained and one of three alternatives is applied:

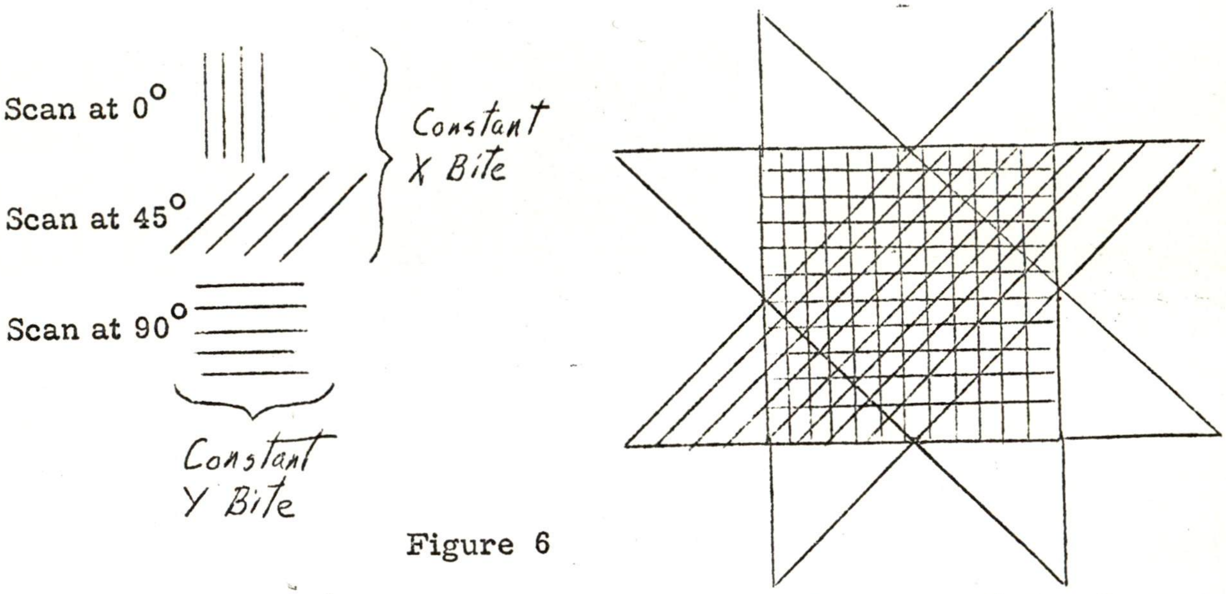


Figure 6

a) An element is found which was not predicted. In this case, a new track

Typical Track Bank

Track Bank No.
X (beginning)
Y (beginning)
ϕ (beginning)
X (current)
Y (current)
ϕ (current)
Predictions
Predictions
Curvature Information

Figure 7

bank is set up and a new prediction is made.

b) An element was predicted, but is not found. In this case, the track bank is "closed" and is not altered until the next track following phase. To handle ambiguous predictions one can refrain from closing a track bank until both predictions are proven false (that is, until the track is found to be absent from both cells in which is was predicted).

c) An element was predicted and is found. In this case, the track bank is updated (new X, Y, and ϕ values are entered) and a new prediction is made. To increase the accuracy of the prediction, it may be desirable to store a running correction factor to be applied to the slope. This correction factor is related to the curvature in an obvious way and may also be used to eliminate delta rays on the basis of curvature versus range. Such delta ray banks will be kept open, as long as the slope remains between -45° and $+45^{\circ}$, in order to absorb as many elements as possible. At the end of phase 1, however, all such banks may be erased from memory in order to speed up the next phase (track following).

The scan pattern described above, namely covering the entire area, but only one half the angular range, is called the "active" scan defined above. This scan is then to be followed by an orthogonal "inactive" scan in which the roles of the X axis and Y axis are interchanged. Breaking the scan up into these two parts serves two main purposes. Firstly, the prediction logic described above is made considerably simpler since predictions need only be made on one end of a track and only into the following scan line. Secondly, the timing problems involved in processing the data during the scan are considerably simplified since predictions do not have to be made immediately but only by the end of a period of time corresponding to one scan line.

Also, if pile-up occurs during the active scan it is possible to occupy the CRT with the inactive scan while the data from the active scan is processed and thus to avoid stopping the scan for any longer than necessary. The additional time introduced by breaking the PEPR scan into two parts increases the scanning time by only about 0.5%, i.e., the time necessary to reposition the spot to the center of 1800 cells at 10 μsec per position.

As the flying line segment sweeps at a fixed angle across the pattern shown in Fig. 6, an interpolation counter in the PEPR Controller counts from 0 to 100 in 10 μsec. The contents of this interpolation counter are dumped into the Readout Buffers whenever a track element is encountered. The least count in the PR mode is therefore 20 microns or about 1 bubble diameter. It is estimated that the long term jitter (i.e. minutes) in the overall position of the pattern will be no more than 60 to 80 microns and the short term jitter (i.e. milliseconds) is expected to be much less. As described by B. Wadsworth,(3) a given track element may be detected on a scan at any angle within perhaps ±3° of the true direction of the element. Thus one such element may give rise to up to six "hits" (one on each of six successive scans). This means that the program must decode this information and correlate all the "hits" from any one element as described below.

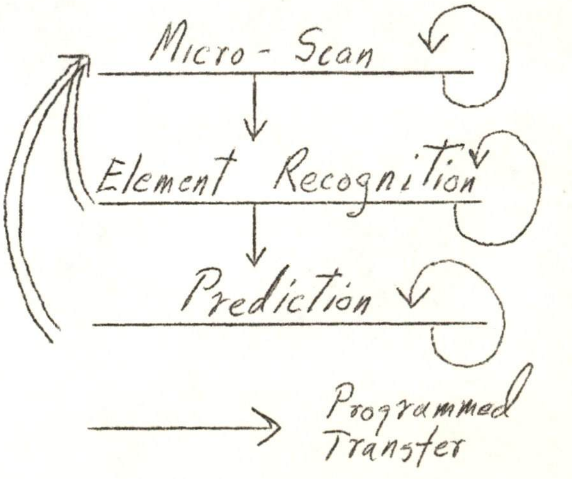


Figure 8

b. Scan Program Logic

The scanning programming takes place on three levels shown in Fig. 8. We shall discuss these in the order of their priority.

1. Micro-scan

Let us assume that the program is processing data from previous scans and a "hit" occurs, i.e., at some angle from one to six pulses (from one to six adjacent tracks at the same angle) are received by the Controller, the appropriate interpolation bits are stored in the Readout Buffers and at the end of the current sweep the scan is automatically stopped. At this point a sequence break(2) will occur which transfers control to the Micro-scan program. It is the function of this subroutine to store the data from the Controller as fast as possible and to restart the PR scan. Since PEPR will in general give from 3 to 5 "hits" on any one track at adjacent angles and since each angle scan takes only 10μsec, Micro-scan will wait 3 machine cycles after restarting the scan before transferring control to the Element Recognition Level It is estimated that the entire delay in the PR sweep to remove and store data will be less than 100 μsec per hit or 500 μsec per track element. This means that if there is on the average one track per scanning cell, the total scanning time will be ~~14%~~ 25% longer than the time necessary for the CRT sweep due to the data handling requirements. It is interesting that the speed of the PDP-1 computer is approximately matched to the CRT cweep speed in this device. A possible flow for Micro-scan is shown in Fig. 9

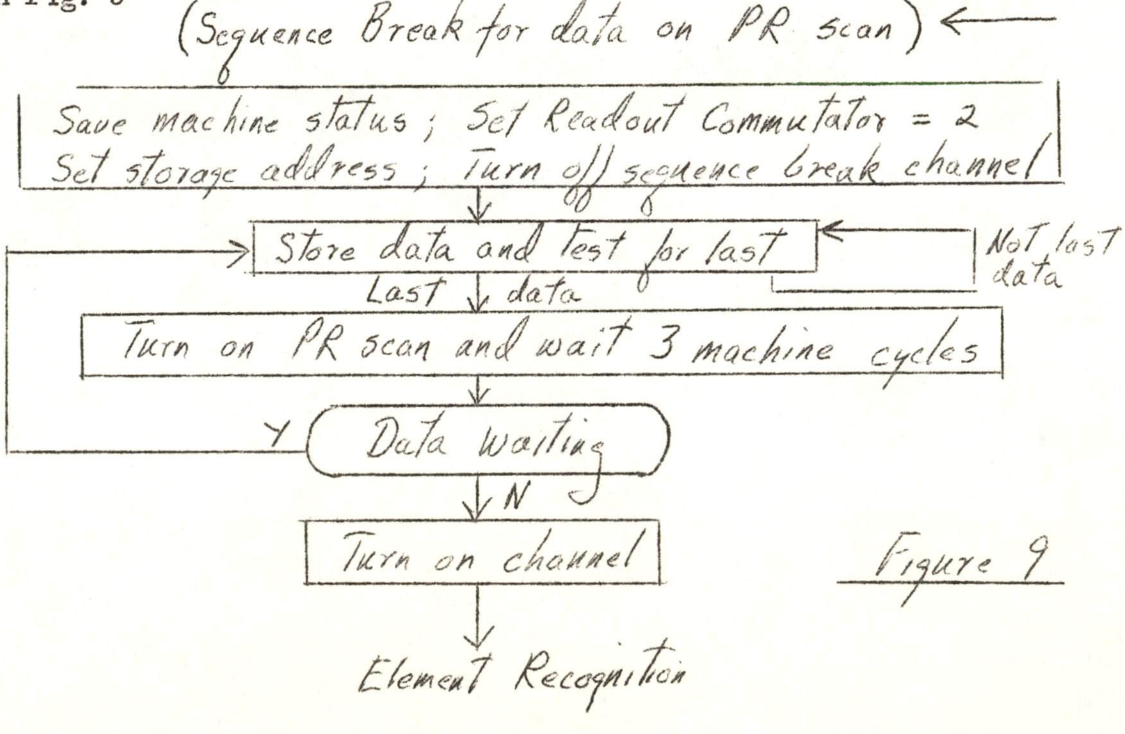


Figure 9

2. Element Recognition

The multiple (2 to 4) "hits" which a single track will produce in the PR scan combined with the possibility of "jitter" of up to three track widths will produce ambiguities in certain cases which will complecate the recognition of track elements. Groups of hits which are not separated by a blank scan will be stored in up to 90 banks (B1 ⟶ B90) of six words each. The number of each bank will represent the angle at which the hit accurred. A typical pattern of data produced by three nearly parallel tracks is shown in Fig. 10.

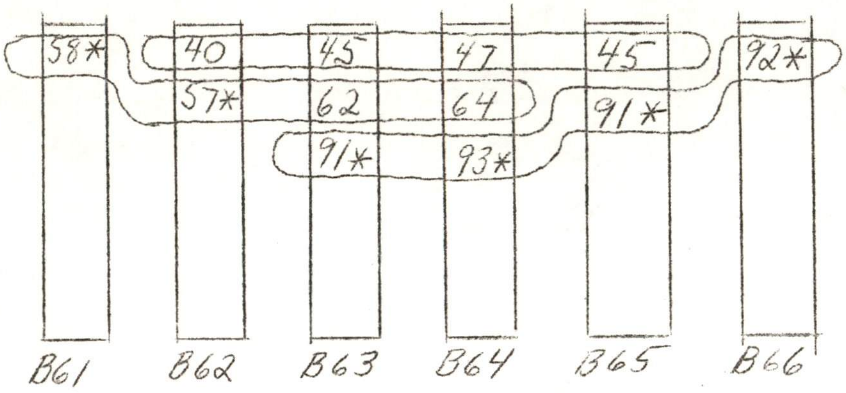


Figure 10

where the numbers are PR interpolation counts and * indicates the last piece of data (the last bit of the appropriate readout buffer = 1). The numbers which are to be associated are encircled. A possible flow is shown in Fig. 11. It is estimated that this program could handle a single track (with 4 "hits" on the average) in less than 200 machine cycles. Since the 90° scan requires 180 machine cycles this is roughly matched to 1 track per 2 mm square cell. A picture containing 20 to 30 beam tracks could thus be handled without interrupting the scan. It is true that the spacing between tracks will be more accurate than the absolute location due to the "jitter" and this information is not used in this method. A more sophisticated program utilizing this fact may be added later if it is found to be necessary.

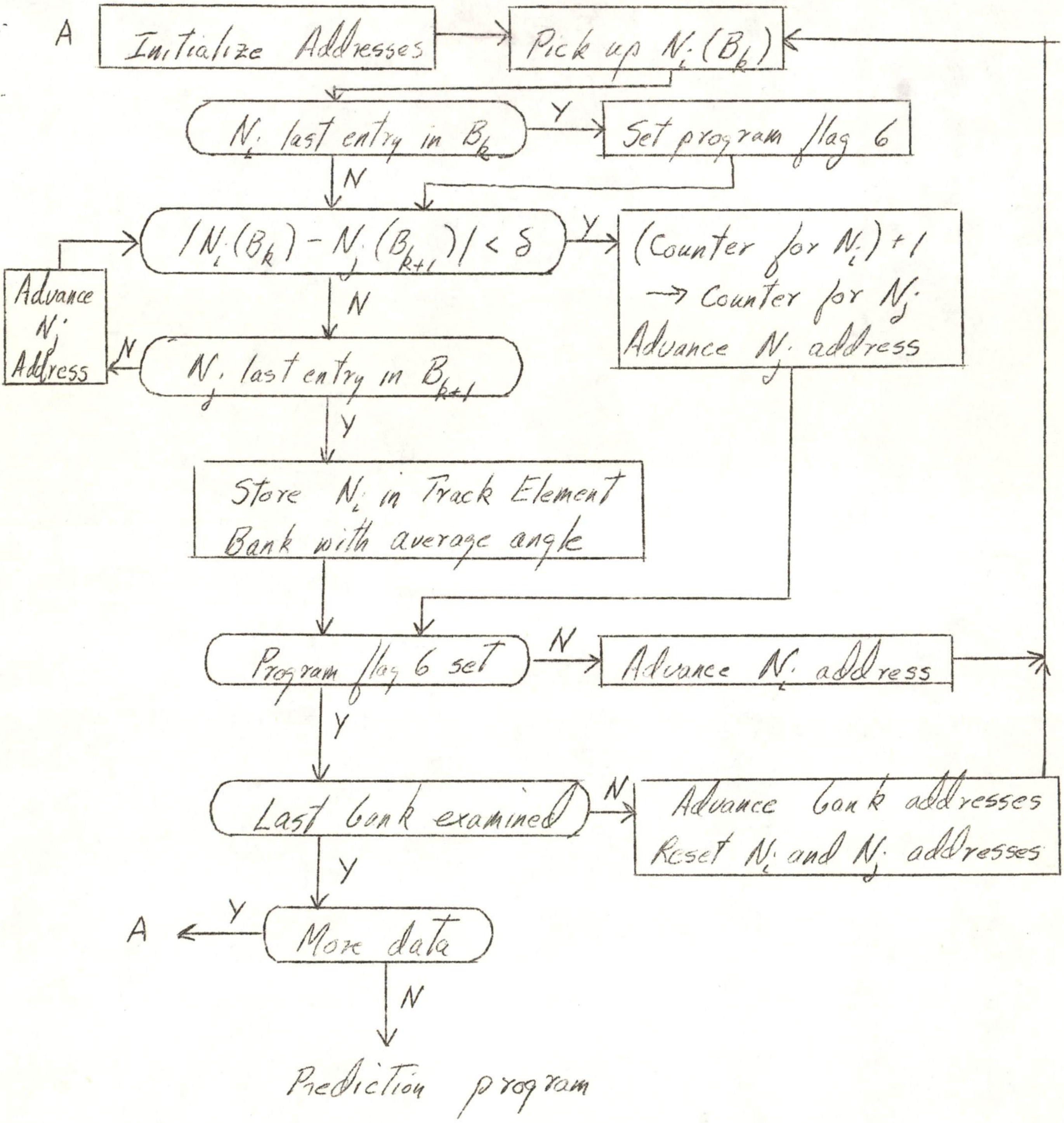


Figure 11

3. Prediction

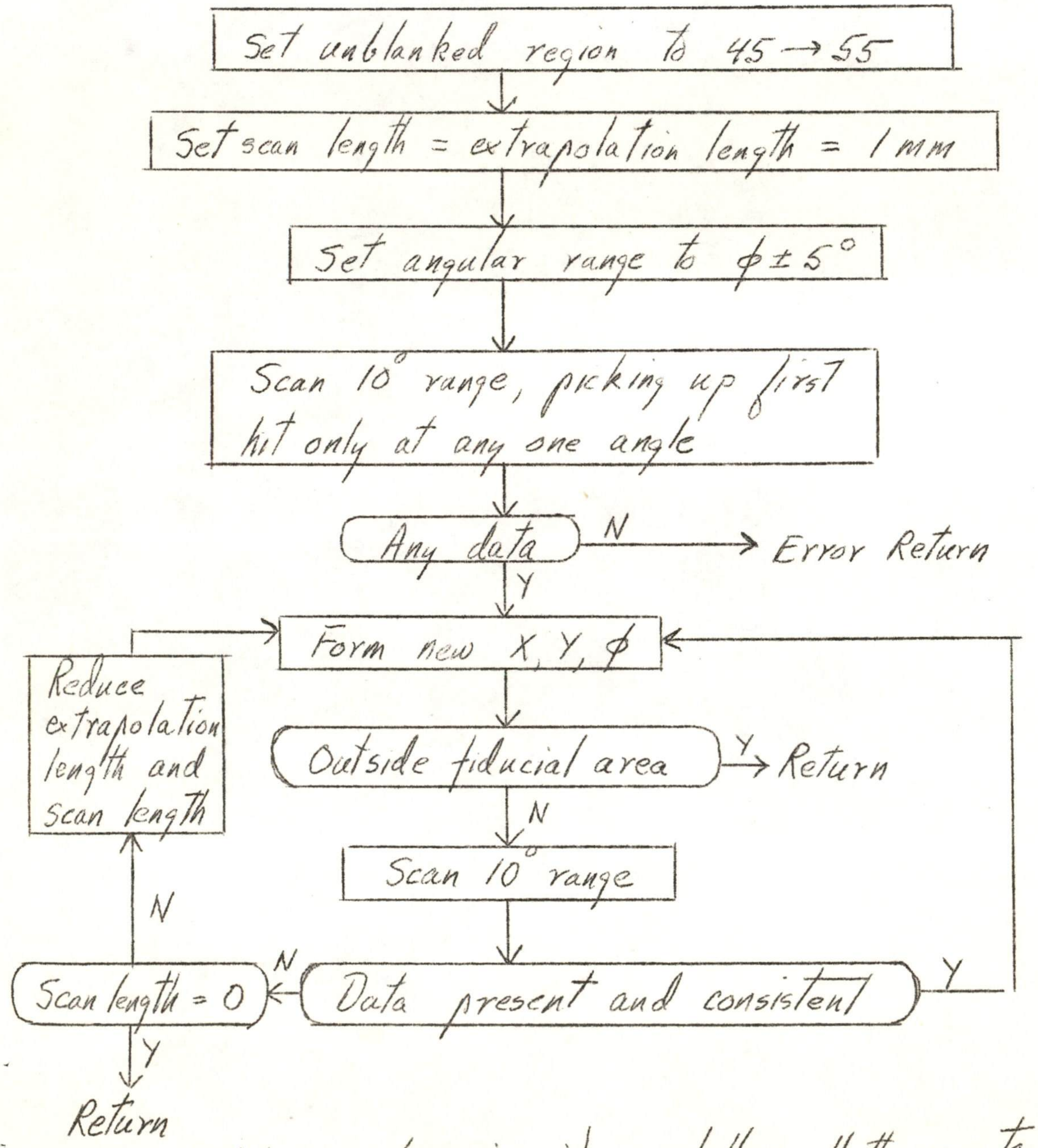
This level of the area scan program is probably the simplest to program. Track elements with their average angles and cell numbers will be picked up in order from the track element bank. For each track element a search is made ~~of~~ for a suitable ~~the~~ track bank~~s~~ and either a track bank is updated and a new prediction made or a new track bank is opened. At the end of a scan line, track banks still containing predictions in the completed line are closed. Predictions can be made with a table look-up procedure relating interpolation count with angle so that the timing on this level is estimated to be not critical.

An overall area scan control routine will also be required to adjust the cell position and to test for the end of a scan line and the end of scan. This control routine will be operated by a special sequence break which will be enabled when the scan angle equals the final angle.

Phase 2 Track Following and Filtering

a. Subroutine TSC

The function of the basic track following subroutine is to take input data such as X, Y, and ϕ for a given track segment and to follow this segment to either a vertex or to the edge of the fiducial region.* A possible flow for this subroutine is shown in Fig. 12



*Later matching may be easier if we follow all the way to the end.

Figure 12

Even this rather rough method of assimilating the data will require approximately 300 μ sec per mm including scanning compared with the actual scanning time of 100 μsec. However, one can still follow 50 track segments each 120 mm long in 1.8 sec so that this problem should not seriously limit the performance of PEPR.

b. Filtering Logic

Each track segment is thus followed its entire length and both end points are located with a precision of a few bubble diameters. On each end of each segment except where segments leave the fiducial area, a rotary scan will be made and a list of angles at which elements are found will be associated with each end. The track banks will then be searched and duplicate track segments will be discarded. After this, end points will be matched and whenever end points on two tracks agree the angles will be matched to establish linkages. Thus the angles found at the end of a given segment will be replaced by the numbers of the segment whose initial angles and end points match. By this means, a set of linked segmentsis produced for which a "scanning event type" may easily be written. It is interesting to note that this technique will also detect 2 prong vertices in which one of the prongs continues in the direction of the beam track with an undetectable change in direction. The other track will then be detected as a segment with two linkages (180° apart) at one end which have not been matched. These two elements may then be track followed and the end points and angles may be matched with an existing "straight through" track in order to establish the linkage.

Phase 3 Precision Encoding

a. Programming for Precision Encoding

The basic program for the PE mode can be devided into two parts. The first part is concerned with locking the spot onto the desired raster and the second with moving the spot, which is servoed onto one edge of this raster, across the film and locating the precision coordinates of a track by counting raster pulses as well as by examining an interpolation counter which is restarted at each raster. Details of the design are discussed in the original PEPR proposal.(1) The corner of the picture at which the count-over is to begin as well as the initial direction of motion are specified by entering bits 16 → 17 in the Controller E register. The initial rate and initial count-over address are entered in E bits 12 → 15 and E bits 0 → 11 respectively. As the count-over progresses, the program must reduce the rate and advance the address until the final address is reached and the servo is turned on. The purpose of this rather elaborate procedure is to minimize the time necessary to perform the count-over while avoiding the danger of overshoot when the servo is turned on. The detailed timing of this program must be determined by trial and error. In addition, in order to make sure that no rasters are missed during this count due to dust spots, etc., the count-over will begin using a line segment whose length is controlled by B bits 12 → 14 and this line will be reduced by the program to a spot as the final count-over address is approached.

The second part of the PE program will simply set the gates (E bit 0 → 11 and C bits 0 → 13) and control the processing of the data from the PE scan. This control will be simplified by the sequence breaks which have been provided and seems to present very little difficulty. As soon as the gate is closed the program will set up and initiate the next PE scan.

b. PE Control and Timing

At the present time emphasis is to be placed on a "one-view" PEPR system in which the scanning event type logic would be handled subsequent to measurement by a 709 or 7090 computer. Under these circumstances a simple procedure for

PEPR would be to measure approximately 10 "average" points on every track involved in a cluster. Tracks not so involved will presumably be of no interest in later analysis. Depending on the average bubble density in the chamber such ~~a measurement~~ an "average" point will require perhaps 6 PE scans per point in order to be certain that the points measured are really part of a track and not background. Since the estimated time for one PE scan is 3 milli-seconds, the total time for 10 average points on each of 10 tracks would be 1.8 seconds. This procedure seems to be satisfactory in relationship to the rest of the program and it has a further interesting feature. The tape produced by PEPR operated in this fashion may be used as a " scanning tape" described by H. White(4) and in Part II of this report.

Ionization Measurements

In most experiments handled by this system it will be very important to obtain accurate ionization measurements on all tracks of interest. PEPR may be programmed to shrink the flying line segment to a spot in the PR mode and thus to obtain ionization measurements extremely rapidly. Let us assume that the bubble density on the film is 10 per mm. In ~~.02~~ ~ 0.2 sec one can make 1000 independent sweeps across a track (separated by 60 μ say in order to avoid overlap due to jitter) with a 25 μ spot. The data thus received (bubble or no bubble) should yield sufficiently precise information for most purposes in a time short compared with the total processing time for PEPR.

Conclusion

The total area scanning time to cover a frame 60 mm by 120 mm including the data processing is estimated to be approximately 4 sec, the total track following time should be no more than 2 sec, and the measuring time approximately 2 sec.

It therefore seems safe to assume that PEPR can process a triad in less than 30 sec and should therefore be capable of handling 10^6 pictures per year. These estimates seem to be fairly conservative at the present time and the principal uncertainties involved in the PEPR system would seem to be connected with the hardware. Questions such as what angular resolution and what accuracy will really be obtainable in both the PR and PE modes should be settled only after a period of months.

Part II

A Data Analysis System using PEPR (or PEPR Simulation with FSD)

A. PEPR and PEPR-Simulator

We have talked with George Rabinowitz at BNL about the Pasta-Marr-Rabinowitz program which can sort Hough-Powell-FSD data into line-segments at roughly the FSD digitizing rate. Let us call this program PMR.

Once we became familiar with both PMR and the PEPR program described above, we realized that they are logically already very similar and can produce identical output, very much like the abstraction tape proposed by Howard White.

We thus propose a system in which PMR can be treated as a pseudo-PEPR. This system can probably start operating using FSD and PMR before the PEPR hardware is ready.

As described in II.G the PMR mode will be more expensive in computer time than the PEPR mode, but it has the advantage that different labs can switch to PEPR if, and when, PEPR - availability and individual economics permits. Moreover, if the CDC-6600 or some other new computer should ever turn out to be >5 times as cheap (per machine cycle) as the 7094 it might well throw the economic balance back in favor of PMR.

For Berkeley, where it is likely that a CDC-3600 may be acquired in mid 1963, it should be borne in mind that all the programs described in this section as written for an

IBM 7094, could instead be run on a CDC 3600.

We also discussed the first version of PEPR, with only enough lenses (4) to scan a single view at a time. Since FSD also deals with only one view, this makes PMR and PEPR even more similar. We realize that eventuallyone may want to program PEPR to work its way out of difficulties by consulting all three views with "random access", but this requires seven off-axis lenses and six months delay in hardware and considerable additional programming. We prefer, therefore, to get the one-view program system working and to gain experience before going to three views.

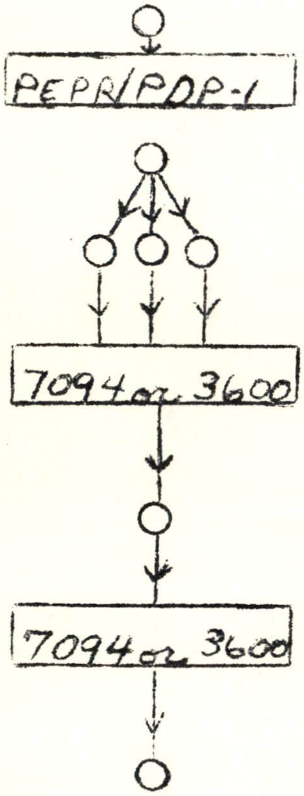
We will be most satisfied if the programming described in both parts of this note will at first handle about 90% of the events of an easy experiment (~ 10 tracks/frame), with the indigestible residue measured on a Franckenstein or SMP.

B. SYSTEM

In this section we discuss the system assuming that the input comes from PEPR. In II-C we will take up the PMR program and its use as a PEPR-simulator.

The proposed overall flow is shown in Fig. M.

Fig. M. Flow of Data



Film, one roll only (one view at a time).

PEPR program, ~ 5 hour/600 triads

Single-view data tape ≈150 feet, Low Density

Single-view data tapes { 3 for BNL 3-camera system
1 for Berkeley 1-camera system

Triad Program - matching of most vertices consistency checks. ~10 min./600 triad

Main Data Tape (Primary tape library, high density 150 ft/600 triad)

Scanning Program, controlled by Scanning Event Type Control Program, ~1 min/600 triad

Standard Input to Package, Fog, YAP, Grind, etc.

On Fig. M we have indicated times to process 600 triads. We chose 600 as a unit because

a. Berkeley 72" film comes in 850 ft. rolls of 600 triads.

b. BNL 80" " " " (we think) 400 ft. rolls of 600 views.

PDP-1 can write only low density magnetic tape as illustrated in Fig. M and discussed in Part I. PEPR will go as far as it can go dealing with a single view, and write out a Single-View Data Tape while it advances film. (PEPR output from Berkeley 3-view cameras will have all the views of a frame close together on the tape. For other cameras PEPR should have its output tapes changed when films of different views are loaded.)

In the next few paragraphs we make some definitions and use them to explain certain lists for our proposed Data-Tape, which we introduce in Table III.

For Vertices we shall use the notation VN, where N is the "prong number" (where we count the incoming as well as the outgoing prongs). Thus

a V1 is any track beginning or ending inside the fiducial area; e.g. stopping track, track hitting top glass, charge exchange, etc.

a V2 could be a decay in flight, or a lambda decay.

a V3 could be an elastic scatter; etc.

With our linkage information from the rotary scan we can group the tracks into clusters of linked vertices.

Let us call b1, b2, etc. the clusters orginating with beam tracks and call other clusters c1, c2, etc. On cluster b1, call the vertices b1a, b1b, etc.

To illustrate this classification, Fig. 2 is represented now as Fig. Z with through tracks dropped. For this example, our proposed <u>vertex</u> list for the data tape would be

Number of beam clusters nb=4
b1a (V2, x, y)
b2a (V3, x, y), b2b (V1, x, y)
b3a (V3, x, y), b3b (V2, x, y)
b4a (V1, x, y)
Number of non beam clusters nc=2
c1a (V1, x, y)
c2a (V2, x, y)

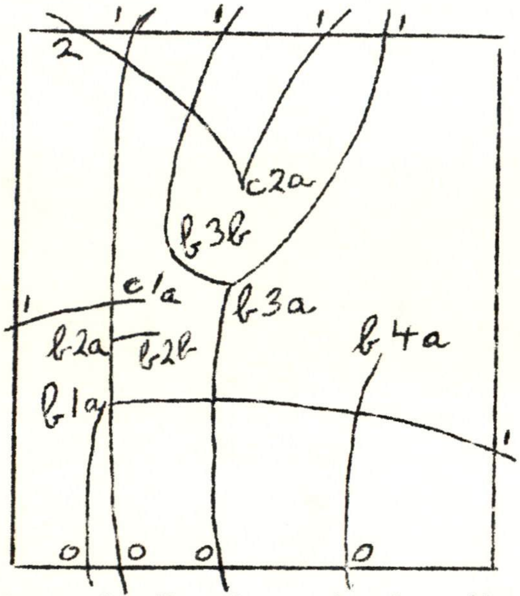


Fig. Z Tracks entering the beam "window" are numbered 0
" crossing the fiducial boundary are numbered 1, 2, ...

A "kinkless vertex" is one in which the directions of two of the tracks, as viewed from the vertex, are opposite. In many cases they may be caused by the chance coincidence of a through-track and the end of another track starting or stopping within the fiducial volume. In some cases they are real vertices (low-momentum-transfer collisions, K^- capture at rest, etc.). Data from two views must be compared to tell whether it is a real vertex or an accident. Hence, kinkless vertices should have an extra identifying mark to help the Triad Program.

Track <u>names</u> are also illustrated in Fig. Z. We propose a track list in which for each end of the track we write the coordinates and the angle; these data to be followed with the length L of the track and an estimate of the ionization I, thus:

O-bla (x, y, θ at O; x, y, θ at bla; L, I).

bla-1 (x, y, θ at bla, x, y, θ at 1, L, I).

We are aware that the coordinates of a vertex now appear on the tape both in the vertex list and again $\geq$ 1 time in the track list, but the redundancy appears convenient for the later programs.

TABLE III. CONTENTS OF DATA TAPES

I. SINGLE-VIEW FORMAT	Approx. No. of 36-bit words.
A. Fixed Data: Chamber, Expt. No., Roll, Frame, View	10
B. Fiducials, to full Precision Encoding Accuracy (17 bits)	3
C. Single View PR (Pattern Recognition) data (12 bit coordinates).	
1. Number of "through tracks", i.e. tracks which enter and leave the fiducial boundary and have no ordinary or even kinkless vertices.	

Approx. No. of 36-bit words.

2. Vertex list, e.g.

   b1a (V2, x, y)

   b2a (V3, x, y), b2b (V1, x, y)

   c2a (Va, x, y)

   One 36-bit word/vertex x about 5 vertices — 5

3. Track list

   0-b1a (x, y, d at 0, x, y, d at b1a, L, I)

   c2a -2

   Three 36-bit words/track x about 10 tracks — 30

4. List of Rejects and Doubts — 3

D. Single-View PE Track Data

   (10 avg.-pts./track)

   20 17-bit wd/track x 10 track — 100

Total = 150 wd at 200 bit/in.=5 in./view — ~150

II. TRIAD FORMAT

   A. Same fixed Data — 10

   B. Vertex Summary

      Highest Prong Number (N) present

      Number of V1, V2, VN

   C. Track summary (3-D reconstruction, first order optics)

		Approx. No. of 36-bit words.
	For each track give vector momentum and whether it leaves chamber of ends inside. If it ends, did it stop or charge - exchange, or can't we tell?	20
D.	Triad Rejects and Doubts	~3
E.	Possible origins for each V2 (see text)	10
F.	Items B, C, and D or View I, same as before	140
G.	Items B, C, and D of Views II and III same, except Vertex and track names rearranged to agree with View I	140
	Total = 500 wds, this time 540 bits/inch which is 90 wd/inch: Still 5"/triad	~450

Note that the information on this Tape is similar to the abstraction tape proposed by Howard White*, except that we feel that the P.E. information should be attached; i.e., unlike White we do not favor going back for a second pass to precision-digitize either for PEPR or for PMR. Particularly, for PMR the fine-digitizing has been done, so why go back and do it later? Apart from this disagreement we too extol the virtues of an abstraction tape.*

C. "Triad" 709 Program

This is a program which will read Single-View Data tapes and write a Triad Data Tape which will then be easy to scan many times for many different experiments. With all three

*White informs us that he has also come to favor including P.E. data.

views in core, it is possible to match most tracks without recourse to more than 1st order spatial reconstruction. (A few tracks of similar curvature, direction, and length, may have to await final reconstruction).

Probably it is economic to improve first-order optical reconstruction so that V1's can be classified as true V1's inside the chamber, or as "L" tracks (leaving the top or bottom of the chamber). "Zero prong" tracks 0-L or 1-L should be dropped since they are just "through" tracks. We want to keep all true V1 tracks, even if they are not beam tracks.

List H of Table III (Possible origins for each V2), can be obtained without precise stereo reconstruction (actually in a single view) by making a construction (which we learned from Nick Samios and is sketched in Fig. N). The construction is to extend both tracks of a V2 to meet at x. To a good approximation the line x-V2 is the line of flight of any particle which could have decayed into V2. For each V2 we would list any other vertices which in all three views fall within the sector of uncertainty shown shaded in Fig. N.

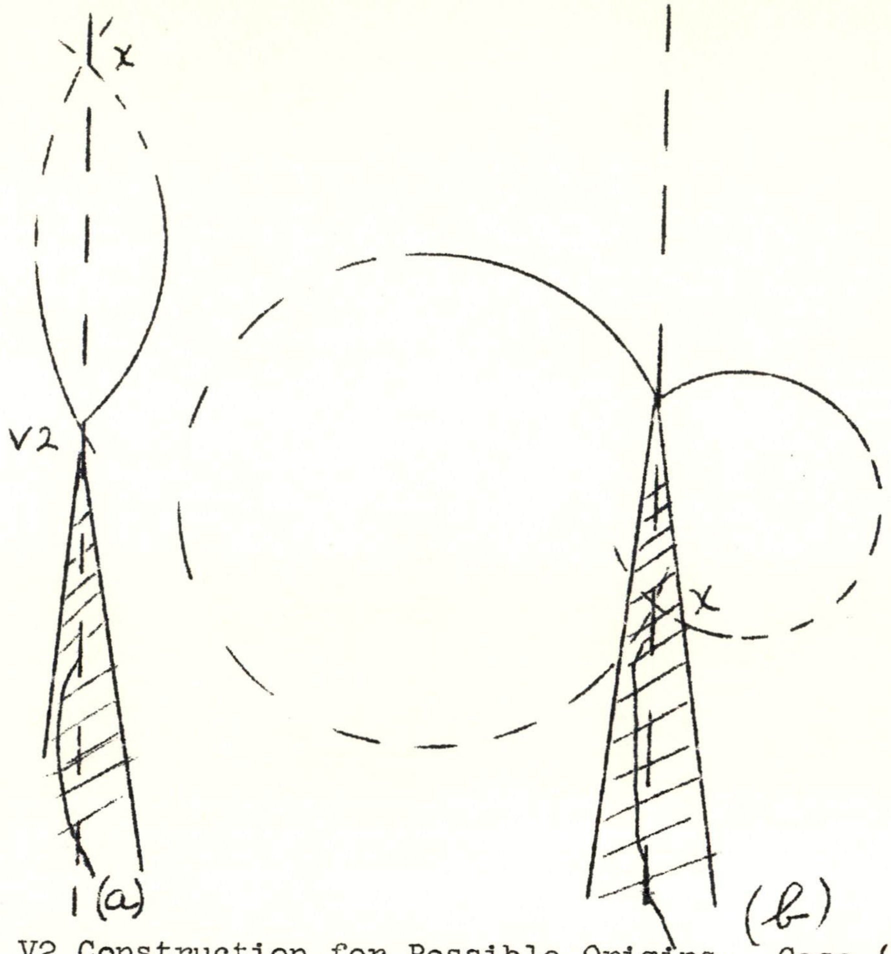


Fig. N. V2 Construction for Possible Origins. Case (a) x falls beyond vertex; Case (b), x falls in front of vertex.

D. "Scan" 709 Program

This program will be organized like kinematics programs with processing routines and Scanning-Event-Type Control Programs.

Using the Vertex and Track lists on the Primary Data Tape written by the Triad Program, the Event Type looks for the various topologies a given reaction can have. Scan-Event-Types will have instructions similar to those now given to a human scanner.

Thus when SCAN finds a 0 prong (V_1) and a V (V_2), which points at an origin, it will classify this event as event type 30 (for the Alvarez group), just as a human now does.

SCAN can write out tapes which can be read by any of the current systems: Package, Fog, etc.

TRIAD and SCAN could be put into core together and write the permanent Data Tape for the library and the selected SCAN output simultaneously.

II-E PMR Program (Pseudo-PEPR)

Talks on the Pasta - Marr - Rabinowitz work have just been presented at Munich, and will soon be available, but we summarize the program briefly.

PMR sorts the FSD digitizings directly out of the input buffer. FSD supplies these data at an average rate of about 10,000 digitizings per sec., and PMR barely keeps up with this 10Kc average rate on an IBM 7090. Actually the program is now written to read the data not from an on-line FSD but from high-density tape, which also reads at 10Kc, (i.e. 10,000 36-bit words/sec). This makes it easy to use and debug the early versions of the program.

The program has two major ideas:

1. The perception that one can sort at 10Kc.
2. The definition of track banks and predictions just as we have described in Part I (we copied the idea from them).

For the first 50 scan lines it proceeds like PEPR'S Area-Search Scan except that it has no option except to cover about ±45°.

Every 50 scan lines all track banks are tested to see if they have received enough hits to be kept on the "active" list. Inactive banks are summarized and cleared out. Active banks are further tested to see if they are beam tracks.

Beam tracks are moved to special B-Banks. Since B-tracks represent a large portion of all digitizations, PMR have developed programming tricks to test all incoming data first very quickly against an ingenious B-track "Map." If the coordinate falls on this map it is efficiently stored away.

Active non-beam ("crossing") track banks are left where they were. We shall refer to these banks as C-Banks.

After 16 hits in <u>any</u> bank, PMR summarizes them as an "average-point" and saves the result. Note that they correspond to our similar averages of P.E. data, although we suggested averaging only four points. Note further that 16 hits corresponds to 1 to 2 mm of track length, so their basic element is similar to that we have been discussing for PEPR. By the time PMR has finished the active scan it supplied track segments, each with a beginning and end point and about L/2 average points, when L is track length on film, in mm.

The above is what PMR does already. They use only a few thousand words of real FAP coding, although core is 2/3 full of banks, input buffer, diagnostic output routines, etc.

What steps remain before they can write the single-view Data Tape of Section II-B?

Let us assume they have just finished the main scan on View I. They must still

1. Move stage to orthogonal scan position,
2. Meanwhile, they may want to do an equivalent to PEPR's track following: i.e. fit the average points to a smooth curve and look for kinks.
3. Do the orthogonal scan and sorting.

3a. Join all the overlapping track segments found in the main and orthogonal scans.

4. Find and list vertices. Note that there are vertices with kinks, and those without. A kinkless vertex is sketched in Fig. P.

Fig. P. Kinkless vertex from high energy peripheral collision.

It seems fastest to compare the position of every segment-ending with all other segment-endings to see if they form a kink-type vertex. After a kink-vertex list is established each kink-vertex (i.e. all V1 Vn in turn must be checked against an approximate curve representing each track segment to see if it falls near that segment. If it does, check in more detail to see if it is a kinkless vertex. We suggest that kinkless V3s be put into some <u>tentative</u> classification until views II and III are available, since most of them will be accidental coincidences and not real V3s. (Note that this Kink-finding is one task done much more easily by PEPR, with its rotary scan.)

5. Eliminate (but tally) all "through" tracks. (Both beam (B) and crossing (C)) By "through" tracks we mean tracks that enter the fiducial area from the outside, pass all the way through without any tentative kinkless vertices, and leave the fiducial area.

After these steps we should have everything ready to write the single-view data tape so that at this point PEPR an PMR have merged.

We should point out that experience should show that it is necessary to make an orthogonal scan only in two of the three views, namely views II and III according to the Berkeley numbering convention, which is sketched in Fig. Q. However, to be conservative, in the time estimates we shall assume that all three views are orthogonal-scanned.

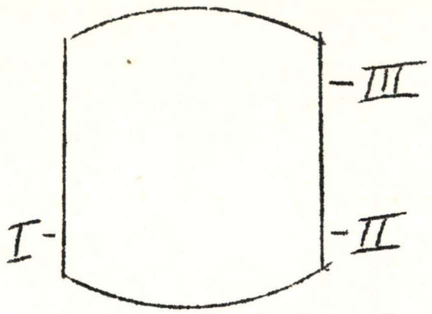


Fig. Q. View Numbers, Berkeley Convention

Actually by the following scheme one might be able some of the time to eliminate the full orthogonal scan even on View III: Measure first View I, main scan only. Then View II, both scans. Then view III, main scan. Then compute the View III vertex list. Then have the computer read the vertex lists made by View II. If it has already found from the View III main scan only all the right vertices, it needs the orthogonal scan on View III only for P.E. data on a few tracks; the computer can decide what parts, if any, of the orthogonal scan it needs.

II.F: <u>MORE DETAILED ESTIMATE OF RATES</u>

I. PEPR

A. Pattern Recognition Mode.

<u>P.R. Mode Area Scan (4 sec, arrived at as follows)</u>: One sweep at fixed θ, 10 μsec. So one cell (2mm x 2mm on film) equals 90° x 10 μsec = .9 msec for "active" scan, plus of course another 0.9 msec for "inactive" scan. Delays while executing microscan coding all ~20 percent, so 2.2 msec per cell for total area scan.

How many cells/view? Shutt 80-inch film: (about 120 mm x 60 mm) equals 60 x 30 cell, equals 2000 cells. Alvarez 72-inch chamber film ($\approx$110 x 32 mm) = 1000 cells.

We'll assume 2000 cells in this section, then area scan of view ~4 sec.

P.R. Mode Track Follow (2.4 Sec):

Assume 1 mm x-bite (y-bite for transverse tracks). Assume length of track to be followed equivalent to 30 beam tracks, but most of them followed at least twice, call it 50 x 120 mm equals 6000 mm equals 6000 cell of 1 mm bite. But each cell needs only about 10 sweeps (i.e. 10 values of $_\theta$), so 100 μsec. But each cell needs also about 200 μ-sec of PDP-1 time, so the total is 400 μsec. 6000 x 0.4 msec equals 2.4 sec.

P.R. Mode Rotary Scan (about .1 sec):

Assume 50 vertices, so 50 rotary scan of full 180°, so 50 x 2 msec or 0.1 sec.

P.R. Mode Ionization Measurement on Non-Through Tracks (1 sec).

Shrink line to spot, make 500 sweeps/track, each with a different x, y. Maybe 20 PDP-1 cycle to set up a sweep, so .2 msec/sweep, so 0.1 sec/track. 10 tracks take 1 sec.

Total P.R. Mode ~6 sec/view

B. P.E. Mode (3 sec).

Basic timing per "probe" (i.e. to count "across" and then "down") = 3 msec. If track is half gaps, 6 msec/hit. For each track we want 10 average points of 4 hits each equals 40 hits equals 1/4 sec. So for 10 tracks we need 3 sec.

Total P.E. Mode ~ 3 sec/view.

C. Film advance and fiducial measurement (4 sec).

D. Totals. I view ~ 13 sec, 1 triad 40 sec for 80-inch chamber (maybe 30 sec 72-inch chamber).

Conclusions:	72-inch chamber	80-inch chamber
Triad/min	2	3/2
Triad/hr	120	90

II. Computer-times.

A. PDP-1 time for making vertex lists, track lists, etc., and writing tape. We have not yet thought about this question, but we guess a few sec/view is safe. We have this much time during precision encoding and film advance so we have added nothing to time estimates. Tape writing goes on in <<1 second, so can be done during film advance.

B. 7094 Time for Triad and Scan. (Both PMR and PEPR) There is considerable doubt in our minds about the Triad program, since we have little experience in this sort of matching of vertices and tracks. But we guess that Triad will run a few seconds/triad on a 7094. Scan should run not much slower than tape reading speed, which is tens of events/sec.

C. FSD-PMR on a 7094:

We stated earlier that PMR now can sort data from the input buffer at about 10 Kc on a 7090. We assume that the 7094 runs 3/2 faster, and guess that PMR, after it is hampered with a few more chores, might run at 12 Kc on a 7094.

Howard White has given us the following rates for FSD:

It is set to digitize a view of either the 72-inch or 80-inch chambers in about 4 sec; yielding about 40,000 digitization on a typical main scan, and perhaps 15,000 on the orthogonal scan(at least for Berkeley film where the bubble density is high). Thus he hopes it will take about 3 sec of computer time for PMR to handle the main scan, and close to one sec of the orthogonal scan; or about 4 sec. for both scans of one view. We add another second to handle the input of data and the making of vertex lists, and multiply by three to get time per triad, arriving at 15 sec. per triad. Note that this is computer time, not real time, since a single FSD will actually have to work for 30 to 40 sec to digitize a triad.

We will summarize the FSD times by pointing out that in real time FSD-PMR is about the same speed as PEPR, but that a 7094 is fully occupied only about 50% of the time; i.e. it will have time to do Fog-Cloudy-Fair and whatever else can be time-shared.

D. FSD with External Roads

This question does not really enter into this note; but in the next section we compare several current systems with PEPR and with FSD-PMR, so we state here Howard White's estimates.

Time to digitize the main scan is still 4 sec., and film advance is still 4 sec., but only part of the view has to be orthogonal-scanned, and that not all the time, allow 2 sec for orthogonal scan and 4 sec for frame advance, then probably a view can be handled in 10 sec., or a triad in 30 sec; again about the speed of PEPR. But with external roads, the 7094 is needed only a few sec. per view, let's say 25% of the time.

II-G. COSTS OF OPERATION.

I. Pless gave us the following rough costs for PEPR:

Basic PDP-1	$120,000
PDP-1 Accessories	100,000
Controller	~110,000*
Lenses	5,000
PEPR Oscilloscope	5,000
Yokes, etc.	5,000
Film Transport	20,000
Total System	~365,000

*One can only get an exact price by consulting DEC about a specific controller.

However, since more engineering might have to be done, we shall be conservative and round up to $400,000.*

How many triads per year shall we discuss? We have seen in Section F that both PEPR and FSD/PMR can process about two triad/min., for the 72-inch chamber. Thus, in one year, four shifts (i.e. all the time) both systems can process about 10^6 triads. This seems a convenient unit which we shall use for the rest of this section.

---

*We are aware that when physicists estimate that some apparatus should cost ~$365,00 it usually ends up costing $1 million. In the case of PEPR the situation is different in two respects:

1. The PDP-1 and the Controller would both be purchased from Digital Equipment Corp. The PDP-1 at $220,000 is commercial. The controller, at ~$110,000 might grow in price if we ask for new features, but it is not likely to grow very much. The remaining equipment estimate amounts only to $35,000, and would have to grow in cost by a factor 7 before it doubled the overall estimate of ~$365,000.
2. In any case we do not propose that PEPR be developed anywhere except MIT, or that other laboratories should purchase one until there is confidence that it works as expected for an overall cost of not much more than $400,000.

PEPR and PMR can process about 2 triads/minute for the 72-inch chamber. Thus in one year 4 shifts (1 shift-year equals 120,000 minutes) both systems can process about 10^6 triads. So our discussion in this section is all in terms of a million triads per year. Note that 10^6 triads scanned and measured implies $\sim 10^6$ events (perhaps 500,000) going into present computational systems (PACKAGE-through-SUMMEX, FOG through FAIR, etc.). 10^6 events/year needs a bare minimum of another shift on a 7094. (Present Alvarez group experience (or lack of it!) indicates we'll actually use much more, perhaps 3 shifts).

In Table IV we want to compare several current systems, operational or proposed.

We use the following time estimates. If the film is scanned by people:

To scan a frame, whether or not it has an event: 1 min.

If the frame has an event, and is to be measured by:

...Franckenstein; to record the data for the master list, 1 min. more

...FSD or SMP, to record the data and digitize, 2 min. more

We assume that all the projectors mentioned above are used three shifts.

To Measure an event with

Franckenstein: 10 min. more.

FSD (after external roads have been made): 1/2 min, but 7094 needed only 1/8 min.

SMP: no more time.

To Scan and Measure a 72" triad assumed on the average to have one event

PEPR: 1/2 min. on PDP-1.

FSD/PMR: 1/2 min. on 7094, but PMR needs only 1/4 min. of processing time, so 1/4 minute free for kinematics. Note that this estimate of 1/2 minute is an optimistic rounding of the estimate of Sect. F where we said 30 to 40 sec., however it is easier to construct and remember Table IV if all three measuring times (FSD with and without external roads, and PEPR) are the same value of 1/2 min.

We then calculate Table IV assuming that all systems scan 10^6 triads/year and measure 10^6 events/year. This does not imply that each frame yields exactly one event. Table IV ignores the fact that all systems have rejects which need remeasurements; it also ignores the need for maintenance personnel for all systems.

TABLE IV Times required to scan 10^6 frames/year and measure 10^6 events/year.

System	Scanning		Measuring		Shifts on a Computer	
	Salaries	Projectors	Salaries	Projectors	Digitize through writing single-view data tapes	Computation on 7094 (shifts)
F'stein	8	3	80	30	0	1-3
FSD-External Roads	24	8	4[b]	1	1/4x4, on 7094	1-3
FSD/PMR	0	0	4[b]	1	1/2x4, on 7094	1-3
SMP	24	8	0	0	4, on PDP-1[a]	1-3
PEPR	0	0	4[b]	1	4, on PDP-1	1-3

a) One or perhaps two PDP-1 could handle 8 SMP up to analysis programs. Alternatively 8 SMP could be run on a time-sharing basis by a large computer.

b) Operators, 1 each for 4 shifts.

To compare costs on various systems we assume that it costs $8,000/year to employ a scanner

so the 24 girls to run SMP's (or FSD with roads) cost~$200,000/year. Hence it seems that PEPR, at $400,000 once, is cheaper than FSD with roads and even cheaper than SMP. (After all 8 SMP's at $40,000 each involve some capital cost too!) And as we pointed out earlier, this economy is not **the** only virtue of the PEPR or PMR system; equally attractive to physicists is its ability to produce tapes which can be rescanned at perhaps 20 events/sec. to solve new problems as they arise.

Although PEPR looks cheaper than SMP and can produce a useful Data tape, we do not suggest that PEPR should ever entirely replace SMP's or Franckenstein's. This is all the more true, since the PDP-1 which runs PEPR can also handle SMP's or Franckenstein's. Consider for example, a laboratory with one PEPR and three SMP's (or Franckenstein's) on-line to the same PDP-1. PEPR could operate during the two loneliest shifts, scanning and measuring about half a million events/year. Let us suppose that in its youth the PEPR program has doubts about 10 % of the triads it scans, so asks for human inspection of perhaps 50,000 triads/year (perhaps 10,000 of these will actually have to be measured). This can easily be done during the two day and evening shifts, with the SMP's (or Franckenstein's) free most of this time for other interesting physics, for training graduate students, etc. Remember of course that some time on a larger computer must still be available for Kinematics. One could either run lines from PDP-1 to a 7090 and interrupt when an event is ready to be fitted, or one could carry over tapes several times/day.

The comparison of PEPR and FSD/PMR is more complicated. Even if we believe the time estimates given in section F, it is not clear how much to count the cost of a 7094. We shall give two extremes. The lower estimate is to use the Berkeley rental cost, taking advantage of IBM's educational contribution, and not counting overhead. This rental is about $45,000/month, i.e. a little over half a million dollars/year. The higher estimate is to take Brookhaven's rental, and add overhead, in which case we guess $1.5 million/year. Then FSD/PMR needs for half a 7094 and this amounts either to $250,000 or $750,000/year. Even at $250,000/year it seems that a $400,000 one-time investment in PEPR is wiser.

What happens with the arrival of a CDC 3600? If one does not reprogram to take advantage of its special features it will run about twice as fast as a 7094. With reprogramming (but that is expensive), it might run four times as fast. Let us compromise and assume that it runs three times as fast. A 3600 with educational discount costs about $2.2 million, plus perhaps 0.1 million/year for service. If we depreciate it over 4 years, we find a cost of about $3/4 million/year, i.e. 3/2 the lower estimate of rental on the 7094. Instead of 2 shifts on a 7094 let us assume PMR would take only 2/3 shifts (i.e. 3-1/3 shifts would be available to other programs, on a time-sharing basis). This 2/3 shift would cost $125,000/year, as compared with $250,000 to 750,000 for half time on a 7094. Now it is no longer clear that PEPR is cheaper for a laboratory that is large enough to afford a 3600 and keep it busy 4 shifts.

In conclusion we feel that PEPR is not likely for some time to turn out to be more expensive than FSD/PMR, and promises to be flexible and particularly useful for smaller laboratories. We have proposed a program system which can use either PEPR or FSD/PMR input so we can gain more experience without having to make binding decisions. The ultimate choice must, of course, be determined by which system works better.

H. Acknowledgements and Remarks:

We wish to repeat that many of the ideas in this report have come, with little modification, from many people who are working on the Hough-Powell device and on PEPR.

We think it is worth pointing out how much each of these groups has stimulated the other (of course the Hough-Powell effort is older and has contributed more). Thus the precision-encoding scheme for PEPR is taken over from FSD. Conversely we suspect that Pasta, Marr, and Rabinowitz were influenced by the area-search and track-follow procedures of PEPR. Certainly we have been greatly influenced by the present Marr-Rabinowitz approach, and have taken the orthogonal scan idea over directly from FSD (there it was a necessary inconvenience, but it turns out to be a convenience for us).

References

1. Proposal to U.S. Atomic Energy Commission for the Development of a Precision Encoder and Pattern Recognition Device (PEPR) in the Massachusetts Institute of Technology Laboratory for Nuclear Science, February 10, 1962.
2. Programmed Data Processor - 1 Manual, Digital Equipment Corporation, Maynard, Massachusetts.
3. Line-Segment Detector for Line Patterns, B.F. Wadsworth, Technical Report No. 75 (NYO-2667) Massachusetts Institute of Technology Laboratory for Nuclear Science.
4. A Proposal for an Automatic Scanning System (unpublished), H.S. White, University of California Radiation Laboratory Physics Note, Memo No. FSD-11, May 8, 1962, and Proceedings of International Conference on Instrumentation for High-Energy Physics, CERN, 1962.
5. A Method for Faster Analysis of Bubble Chamber Photographs, P.V.C. Hough and B.N. Powell, CERN, Geneva, Switzerland.
6. SMP-1 Scanning and Measuring Projector (unpublished), L.W. Alvarez, P.O. Davey, R.I. Hulsizer, J.N. Snyder, A.J. Schwemin and R. Zane, University of California Lawrence Radiation Laboratory, UCRL-10109, April 10, 1962.