

# **Direct 3-D Measurement of High Energy Elementary Particle Interactions in Nuclear Emulsion Targets**

GERALD P. THOMAS

Department of Physics and Astronomy  
Ball State University  
Muncie, Indiana 47306

## **Introduction**

This work is an outgrowth of work carried out earlier by Thomas (1) which involved a partial automation via micro and macro computers of microscope measurements in nuclear emulsion target pellicles which have been previously exposed to high energy accelerator beams of primary projectile particles at large accelerator installations. The motivation was to reduce the labor in performing measurements as well as provide real-time measurements, calculations, and on-line re-measurement decision making. However, the earlier work suffered from several limitations which have been substantially eliminated here:

1) The Koristka research measurement microscope and computerized automation system was centered upon automation of just one axis; the single optical encoder was attached to just the y-axis. This was sufficient for so-called "multiple scattering" measurements, in which work for the x-y plane (projection plane) is sufficient for comparison with theory. But other types of measurement algorithms such as measurement of primary beam particle deflection angles (described below and due to sub-nuclear interactions) involve the depth (z) dimension as well. Yet, both of these latter algorithms involve only fixed x-axis coordinate separations (fixed cell lengths) between measurement points and so no automation of x-measurements is as greatly desirable. However, the system described herein allows for complete 3-d (x,y,z) measurement flexibility for arbitrary measurement algorithms within the general scope of the emulsion technique.

2) The counting of the square wave pulses outputted by the optical encoders was done by the micromcomputer (KIM-1) itself. This led to only marginally good pulse count totals when the y-axis microfilometer screw and attached scale (1) was rotated too rapidly, due to the finite number of machine level program steps needed to detect, read in, and store each pulse. The digital hardware counter system described below was specially constructed to alleviate this problem.

3) The macrocomputer (DEC-10 university computer) was a necessary part of the automation architecture: Once the KIM microcomputer system counted up the pulses whose number is proportional to the y (crosshair) motion per x (cell) change, and stored this number temporarily, it then had to dump its stored data block of count totals and demographic parameters for the event to a temporary DEC-10 data file which had already been opened up from the TTY over the modem just prior to the dump. A manual make-before-break switch was needed to go to full duplex TTY mode such that calculations, using a DEC-10 program, could be made and sent to the TTY for possible re-measurement prompting. All this was subject to university computer downtimes not to mention the inconvenience of setting up files that needed to be run for each event's measurement. To circumvent this, more RAM was obtained so that measurement and calculational programs in BASIC could be run on the KIM. Also an HDE dual floppy disk system was obtained so as to load programs and store output data for future reference, collations and graphing, and comparison with theory predictions.

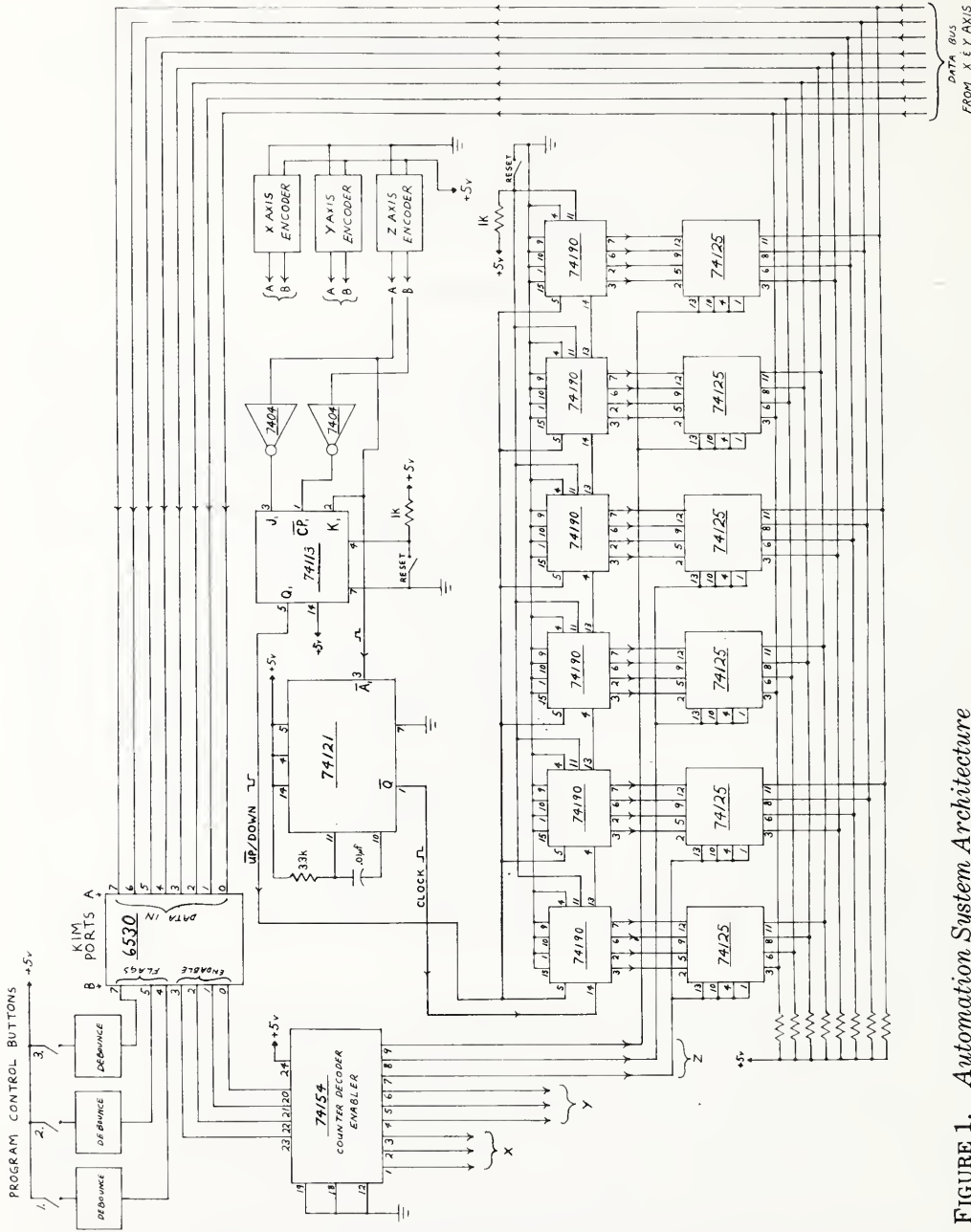


FIGURE 1. Automation System Architecture

Method

A schematic of the 3-d automation hardware architecture used is seen in Figure 1. More on-line hardware is used than cited before by Thomas (1); namely, an external digital counter module as well as two more optical encoders (for the x and z axis, although the angle measurement algorithm described below only reads the y and z axis encoders). Conventionally, the y-axis measurement uses an eyepiece microfilometer scale described in (1) which gives the y displacements that are needed in multiple scattering theory. The change of depth (z), as seen in the eyepiece, is measured by reading the z-collar scale; whereas the cell lengths between x-coordinate positions for various y,z data pairs are given on the x-collar scale or more simply by the audible “clicks” of a notched sleeve for definite lengths

of ex. 50, 100 microns, etc. The three A to D converters, namely the 3 optical encoders, are attached concentrically to the 3 microscope shafts. One whole detailed function is shown for one of the 3 duplicate axis systems, viz., the z-axis. Each encoder will supply a staggered pair of square wave pulses A and B (A leading the B train, if the collar is turned clockwise, looking into the scope, and visa versa) to the angular direction discriminator chips (74113 and 74121).

One function of the 74121 chip component of the direction discriminator is to pass only the first of the 2 pulse trains (A or B) from the encoder to the clock input of the up-down BCD digital counters (six U/D 74190 counter chip exist for each of the 3 axes). The other of the 2 outputs from the discriminator section enables (by using the 74113 chip) either the up or the down function of the counters. The capacitor and resistor connected to the 74121 are tuned to insure a small enough surviving pulse width such that the largest practical revolution speeds of the shaft encoder will not cause any pulse to be missed by the counter. Also, since the rear end of the leading pulse triggers the release of a pulse from  $\bar{Q}$  of the 74121, the requirement that any changes in the up/down status pulse take place before the clock pulse from the 74121 reaches the 74190's is met such that no pulses are lost during changeover. On each axis board there are thus mounted six 74190 counters in series to get the six nibbles (3 bytes) of resolution per axis sought after (up to decimal 999,999) in this application. Each of the 6 decimal nibbles pass to corresponding 74125 tri-state buffers on the same board which are in turn connected by the 8-bit data bus to Port A of the KIM microcomputer. These 6 latched nibbles per axis are gated out to the bus, 2 nibbles(1 byte) at a time (Lo, then Mi, and finally Hi), by means of a series of 3 consecutive enable pulses (06,07, and 08 for the example of the z axis case) sent to the buffers by the MANG machine language subroutine of the calculation program used for the angle algorithm here and written in BASIC called RANG. In particular, for each enable nibble from bits 0-3 of the KIM control of Port B, the 74154 decoder chip produces, for each axis, the correct set of 3 gate pulse enables to get the data byte from each pair of the six 74125 tri-state buffer latches. Both MANG and RANG are stored on the systems disk and the disk data file MUGET on the applications disk as shown in Table 2. Machine level files are designated in the HDE/KIM DOS system by % and programs in BASIC by @.

The machine program (MANG) is shown in detail in Table 2. In the angle measurement algorithm described here, the machine subprogram MANG, written in the 6502 machine language program set (see the left side of Table 2), communicates with the "FOR-NEXT" data INPUT loop of the aforementioned BASIC computation program RANG via a flag system in which one of 3 possible flags are set (using 3 de-bounced button switches next to the KORISTKA microscope, see Figure 1). In particular, one or another of the series of 3 flag buttons next to the scope is depressed at appropriate times during an angle measurement (discussed below) by the measurer in order to "direct", via memory flags, the activities of the RANG program in the "read" (INPUT) stage of that computation program. MANG tests, using AND's, bits 7,5, and 4 of Port B of the KIM for a high (5 volts) for buttons 3 (secondary done), 2 (primary done) and 1 (segment done) respectively, where reference is made to stages of measurement of the lepton angle in lepton induced inelastic, electromagnetic scattering interaction events in nuclear emulsion pellicle targets exposed to a high energy lepton beam. A typical event is seen in Figure 2. Vertical marks on the portion of the figure showing the incident primary lepton track separate x-coordinate "cells", which are typically chosen to be 50 microns in size for beam energies of around 150 GeV (giga electron volts) because the scattering angles are quite small (typically



TABLE 1. *System and Data Disk Files*

SECONDARY STORAGE:

Dual 5" Floppy Disks:

| File    |        | RAM                       |           | File   |        | RAM                  |                        |           |        |    |
|---------|--------|---------------------------|-----------|--------|--------|----------------------|------------------------|-----------|--------|----|
| Number  | Name   | Location                  | Track     | Sector | Number | Name                 | Location               | Track     | Sector |    |
| DIR 1   |        |                           |           |        |        |                      |                        |           |        |    |
|         | ✓DOS   |                           |           |        |        |                      |                        |           |        |    |
| 01      | %FM307 | F300 FFFF                 | 01        | 01     | 02     | %TED                 | E000 EBCI              | 02        | 11     |    |
| 03      | %FMT   | ED00 F0C9                 | 04        | 03     | 04     | %SOR                 | ED00 EE7C              | 04        | 11     |    |
| 05      | %BLD   | ED00 F02B                 | 04        | 14     | 06     | %CPY                 | ED00 EDEA              | 05        | 05     |    |
| 07      | %DEL   | ED00 ED89                 | 05        | 07     | 08     | %DIR                 | ED00 EDC6              | 05        | 09     |    |
| 09      | %*FM   | ED00 EFEC                 | 05        | 11     | 10     | %REA                 | ED00 EDE2              | 06        | 01     |    |
| 11      | %PAK   | ED00 EFD6                 | 06        | 03     | 12     | %LDN                 | ED00 EE48              | 06        | 09     |    |
| 13      | %NAM   | ED00 EE13                 | 06        | 12     | 14     | %BLM                 | ED00 EF07              | 06        | 15     |    |
| 15      | %FRE   | ED00 EE4F                 | 07        | 04     | 16     | %BASIC               | 2000 55FF              | 07        | 07     |    |
| 17      | %BAS   | E000 EBFF                 | 14        | 03     | 18     | <BILIB               | 2000 38FF              | 15        | 11     |    |
| Machine | 19     | %SNP                      | ED00 EE48 | 18     | 13     | 20                   | %VOICE                 | 5E00 5FFF | 18     | 16 |
| Counter | 21     | %MANG                     | 9F00 9FFF | 19     | 04     | 22                   | %BANG                  | 9F00 9FFF | 19     | 06 |
| Data    | 23     | @DANG                     | 5000 6470 | 19     | 08     | 24                   | @CTEST                 | 5000 51CC | 22     | 01 |
| Program | 25     | @DION                     | 5000 5A4B | 22     | 05     | 26                   | @SOUP                  | 0000 0000 | 23     | 10 |
|         | 27     | @RTEST                    | 4FF8 510c | 23     | 13     | 28                   | @MUGET                 | 5000 52A5 | 23     | 16 |
|         | 29     | @RANG                     | 5000 6B80 | 24     | 06     |                      | Data Retrieval Program |           |        |    |
|         |        | Angle Measurement Program |           |        |        | Counter Test Program |                        |           |        |    |
| DIR 2   |        |                           |           |        |        |                      |                        |           |        |    |
| 01      | @MUDAT | 0000 0000                 | 01        | 01     |        |                      |                        |           |        |    |
|         |        | Random Record Data File   |           |        |        |                      |                        |           |        |    |

around 1/2 degrees). Thus, if we have completed one segment (after one “click” of the x-axis collar), and have pushed button 1, then the first BEQ branches to the section of MANG labeled PRIMARY DONE? within which section another BEQ branches to the section labeled SEGMENT, whence the AND #10 avoids LOOP (waiting point for no button pushed), and places, instead, a flag value of 01 to a safe place in RAM (here, \$9FFA) into which the BASIC program can PEEK later on (to see if more segments should be read), whereupon, the 9 bytes of encoder data are “fetched” from the KIM data port (at \$1700) and stored into RAM at \$9FF0-\$9FF8, and then return (RTS) is made to the FOR-NEXT loop in the BASIC program RANG. A close up of RANG’s read in of the 9 bytes of segment data for a given point on the primary beam particle track is seen in Table 3, line 360. RANG goes to machine subroutine MANG, via BASIC’S jump to machine subroutine instruction USR(X), and proceeds to get 9 BCD bytes on each successive iteration (provided the “segment done” flag has been set so as to permit

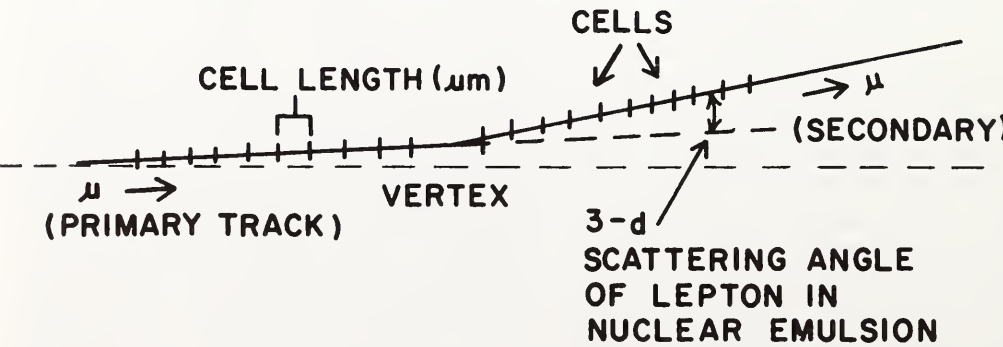


FIGURE 2. *Typical Lepton Track Scattering Topology*

TABLE 2. *Machine Subroutine % MANG*

| INITIALIZE: |                 |                  |                                                               |                                            |  |
|-------------|-----------------|------------------|---------------------------------------------------------------|--------------------------------------------|--|
| 9F00        | D8              | DECOFF           | CLD                                                           | ; Clear Decimal                            |  |
|             | A9 00           | INIT             | LDA #00                                                       | ; Clear Flags:                             |  |
|             | 8D FA 9F        |                  | STA 9FFA                                                      | ; Segment Flag = 0                         |  |
|             | 8D FB 9F        |                  | STA 9FFB                                                      | ; Primary Flag = 0                         |  |
|             | 8D FC 9F        |                  | STA 9FFC                                                      | ; Secondary Flag = 0                       |  |
|             | 8D 01 17        | MASK             | STA 1701                                                      | ; Port A (all bits) = IN                   |  |
|             | A9 0F           |                  | LDY #0F                                                       | ; Port B bits 0-3 = OUT<br>bits 4-7 = IN   |  |
|             | 8D 03 17        |                  | STA 1703                                                      |                                            |  |
|             | EA EA EA EA     | DUMMY            |                                                               |                                            |  |
|             |                 |                  | FLAGS:                                                        | Flags Mask<br>IN OUT                       |  |
| AD 02 17    | SECONDARY DONE? | LDA 1702         | ; Port B 76543210 1702<br>Port A 76543210 1700                |                                            |  |
|             |                 |                  |                                                               | IN<br>DATA                                 |  |
| 29 80       |                 | AND #80          | ; Bit 7 (3rd Flag Button)                                     |                                            |  |
| F0 08       |                 | BEQ Primary Done | ; Go to Primary Done?                                         |                                            |  |
| A0 01       |                 | LDY #01          | ; Set Secondary Flag                                          |                                            |  |
| 8C FC 9F    |                 | STY 9FFC         |                                                               |                                            |  |
| 4C 48 9F    |                 | JMP FETCH        | ; Get last ENCODER data                                       |                                            |  |
| AD 02 17    | PRIMARY DONE?   | LDA 1702         |                                                               |                                            |  |
| 29 20       |                 | AND #20          | ; Bit 5 (2nd flag button)                                     |                                            |  |
| F0 08       |                 | BEQ Segment      |                                                               |                                            |  |
| A0 01       |                 | LDY #01          |                                                               |                                            |  |
| 8C FB 9F    |                 | STY 9FFB         | ; Set Primary Flag                                            |                                            |  |
| 4C 48 9F    |                 | JMP FETCH        | ; Get last ENCODER data                                       |                                            |  |
| 9F36        | AD 02 17        | SEGMENT          | LDA 1702                                                      |                                            |  |
|             | 29 10           |                  | AND #10                                                       | ; Bit 4 (1st flag button)                  |  |
|             | F0 08           |                  | BEQ LOOP                                                      | ; No button yet pushed                     |  |
|             | A0 01           |                  | LDY #01                                                       | ; Set Segment Flag                         |  |
|             | 8C FA 9F        |                  | STY 9FFA                                                      |                                            |  |
|             | 4C 48 9F        |                  | JMP FETCH                                                     |                                            |  |
|             | 4C 18 9F        | LOOP             | JMP SECONDARY<br>DONE?                                        | ; Wait for any flag button<br>to be pushed |  |
| DATA FETCH: |                 |                  |                                                               |                                            |  |
| A2 00       | FETCH           | LDX #00          | ; Mask to fetch first ( $X_{LO}$ ) data<br>byte from counters |                                            |  |
| 8E 02 17    | NEXT            | STX 1702         |                                                               |                                            |  |
| AD 00 17    |                 | LDA 1700         | ; Fetch data byte for track<br>segment                        |                                            |  |
| 9D F0 9F    |                 | STA,X 9FF0       |                                                               |                                            |  |
| E8          |                 | INX              | ; Increment byte fetch mask                                   |                                            |  |
| E0 09       |                 | CPX #09          | ; 9 byte yet?                                                 |                                            |  |
| F0 02       |                 | BEQ RETURN       | ; Done fetching 9 bytes                                       |                                            |  |
| D0 F0       |                 | BNE NEXT         | ; Get next byte                                               |                                            |  |
| 60          | RETURN          | RTS              | ; Return to BASIC FOR-NEXT<br>Data Input Loop                 |                                            |  |

a machine level fetch from the counters) of the FOR-NEXT loop until a PEEK at memory location 40955 (\$9FFB) reveals a "primary done" flag value of 01 to have been set by the machine program (MANG) and detected by the PEEK in statement 400, at which time, transfer out of the FOR-NEXT loop is made and computations on the primary track portion of the event are begun. Thereafter, the same procedure is carried out for the secondary track portion of the event, at statement 510, with the loop escape made in 550 if the PEEK reveals that

TABLE 3. *Print Out of RANG Data Input Sections*


---

```

>LOAD"RANG", 1
>List 340,410
340 INPUT"INPUT CELLP"; CP(K):CELL=CP(K):NN=0
350 PRINT"INPUT YP,ZP PAIRS"
360 FOR I=0 TO 100:X=USR(K):GOSUB 1790
370 Y(I)=LD(3) +LD(4)*100+LD(5)*10000:REM PULSES
380 Z (I) =(LD(6) + LD (7)*100 + LD(8)*10000)*100/UZ:REM MICRONS
390 NN=NN+1
400 IF PEEK (40955)=01 THEN 420
410 NEXT I
>LIST 1790,1870
1790 REM SUBROUTINE DEC TO HEX
1800 PRINT
1810 SPOT=40944
1820 FOR J=0TO8
1830 LH=PEEK(SPOT+J)
1840 LD(J)=INT(LH/16)*10+(LH-INT(LH/16)*16)
1850 PRINT LD(J)
1860 NEXT J
1870 RETURN
>LIST 490,560
490 INPUT"INPUT CELLS";CS(K):CELL=CS(K):NN=0
500 PRINT"INPUT YS,ZS PAIRS"
510 FOR I=0TO100:X=USR(K):GOSUB 1790
520 Y(I)=LD(3)+LD(4)*100+LD(5)*10000:REM PULSES
530 Z(I)=(LD(6)+LD(7)*100+LD(8)*10000)*100/UZ:REM MICRONS
540 NN=NN+1
550 IF PEEK(40956)=01 THEN 570
560 NEXT I

```

---

MANG has placed the "secondary done" value of 01 into memory location 40956 (\$9FFC, see Table 2). The BASIC program RANG does not make use of the "segment done" flag to discontinue and exit the data INPUT loop like the "primary done" or "secondary done" flags do but rather uses it to alert RANG to continue with its INPUT loop iteration, i.e., RANG doesn't proceed to the next "FOR-NEXT" iteration if MANG doesn't first reach its "return from subroutine" (RTS) machine instruction unless the latter's SEGMENT section detects the first button's depression when an AND is made with bit 4 of the control port B of the KIM and a branch to FETCH is thus made in which the up/down digital counters are strobed for the 3 x-axis bytes, the 3 y-axis bytes, and the 3 z-axis bytes and the final return from subroutine (RTS) is thus made. Then the 9 BCD bytes, strobed to KIM port A and put by MANG into RAM at 9FF0-9FF8, are PEEK'd at by RANG's subroutine "DEC to HEX", statement 1830. Unfortunately, RANG's PEEK is programmed by BASIC's interpreter to convert HEX to DECIMAL, but the 9 counter bytes are already in BCD from the counters, so statement 1840 (see Table 3) reverses the PEEK'd values "back", i.e., "back" to BCD, whence the main program, at 370 and 380, assembles together the 3 y-axis bytes (Lo, Mi, and Hi) to a single decimal number N, i.e.,

$$N = (\text{Lo}) \times (0) + (\text{Mi}) \times (100) + (\text{Hi}) \times (10,000) \text{ pulses.}$$

During an actual angle measurement, the BCD up/down counters are reset (see Figure 1) before starting either the primary or secondary for ease of data

printout checking. In any event, at any measurement point during an angle measurement, RANG checks to see if a count downwards from  $y = 0$  pulses has been made and, if so, RANG resets 0 to 1,000,000 pulses so that ex., an incorrect "first difference" in the y-axis coordinates of  $0-999,900 = -999,900$  pulses is replaced by the correct difference of  $999,900-1,000,000 = -100$  pulses. The eyepiece microfilometer cross hair scale, seen in the eyepiece view, visually denotes the y value, in mm, of the movable horizontal member of the eyepiece cross hair (1) as the operator has translated the x-axis screw of the KORSTKA microscope stage over one x-segment (one "cell length" or "click" of the notched x axis sleeve) of the primary or secondary track of the sub-nuclear interaction, adjusting the z-axis control to keep the track portion, being moved along, in focus.

Since one turn of the z-axis microscope collar amounts to 100 microns but one turn of the attached optical encoder amounts to precisely 2000 pulses, then RANG converts (see line 380 of Table 3 for the case of the primary track, and 530 for the case of the secondary track) the z-measurement to

$$z(\text{microns}) = z(\text{pulses}) * 100 \text{ microns/rev/UZ}$$

where  $UZ = 2000$  pulses/rev. The y-axis measurements are based on a 1440 pulse/rev encoder and the x-axis on a 2000 pulse/rev encoder. The x-axis information is not used for the angle measurement algorithm since the x-cell length advance is changed as frequently as twice per event (once for the primary and possibly again for the secondary) and can be easily done manually on the x-axis collar. Its numerical value forms part of the initially typed in demographic data for this particular angle algorithm.

Table 4 shows the Memory Map chosen for the KIM 6502 based system. We see that the 12K HDE microsoft BASIC workspace begins at \$5000, leaving 20K of RAM for the computation program RANG's workspace and its machine

TABLE 4. *KIM System Memory Allotment Map*

| PRIMARY STORAGE:            |                      |                                                            |
|-----------------------------|----------------------|------------------------------------------------------------|
| Memory Board:               | Hex Address Range:   | Content of Function:                                       |
| 8K RAM                      | FFFF<br>E000         | Disk Operating System                                      |
| -----                       |                      |                                                            |
| 16K EPROM                   | C000<br>B000<br>A000 | Disk Boot, Extended Monitor                                |
| -----                       |                      |                                                            |
| 16K RAM                     | 9000 } Machine:      | } Work Space: Machine & Basic                              |
|                             | -----                |                                                            |
|                             | 8000 } Basic:        |                                                            |
|                             | 7000 } Basic:        |                                                            |
|                             | 6000 } Basic:        |                                                            |
| -----                       |                      |                                                            |
| 16K RAM                     | 5000 }               | } Work Space: Machine & Basic                              |
|                             | -----                |                                                            |
|                             | 4000                 |                                                            |
|                             | 3000                 |                                                            |
| 16K RAM                     | 2000                 | Disk BASIC: HDE Microsoft                                  |
|                             | -----                |                                                            |
| 8K RAM & ROM<br>(CPU Board) | 1000<br>0000         | Disk Port<br>CPU Monitor, Stack User RAM,<br>Ports, Timers |



subroutine MANG which was placed in RAM starting at \$9000 with its 9 bytes of data and 3 flag bytes starting at \$9F00. The Disk Operating System by HDE facilitates secondary (disk) storage of each event's principal results along with its deomgraphic data into the random record data file labeled MUDAT (see Table 1). For ease of the event's data retrieval, the record number that is to hold the principal event data is set by RANG to the event number of the scattering event being measured. Thus, data may be retrieved for several previously measured events in a systematic way by some other programs written in BASIC such as are needed, for example, to compare experiment with theory predictions, etc.

A typical session begins by warming up the microscope booting up the KIM system, loading BASIC, running the interactive program RANG up to the first data prompt, centering the eyepiece crosshair, locating and bringing into the center of the eyepiece's field of view the "star" whose beam particle track's scattering angle is to be measured, accessing the event demographics and the topography of the scattering beam track (primary and secondary sections), typing into the TTY the demographics in response to RANG's first prompt, and finally, positioning a point several cells (in x) upstream along the incident primary beam track section from the vertex into the center of the eyepiece field of view by moving the x, and y (eyepiece microfilometer x-axis cross hair) and z controls until the cross hair lateral axis lies coincident with one field of view's worth of track grains (the cross hairs are rotatable about an axis optically concentric to the z-direction such that the lateral, internally movable cross hair can be very well lined up with the general direction of the usually slanted event primary or secondary track). Then, the reset button is pushed followed by a push of the first button ("segment done") whence a series of nine zero bytes is printed by RANG to the TTY for these initial x,y,z coordinates. Then, while the microscope stage's x-coordinate is advanced one x-cell length (usually the 50 micron notch is used at 150 billion electron volt beam energies) closer to the vertex, the z (focus) and y (crosshair) is readjusted until the downstream end of the first cell length (or segment) is reached at which time button 1 ("segment done" button) is pushed whereupon RANG outputs the advance in y and z in terms of raw coordinate data (BCD pulse counts). When the last of the primary's cell lengths is finished (the upstream side of the vertex will now be in view; see left side of Figure 2), button 2 ("primary done") is depressed signaling RANG to make calculations of the primary track angle elements and print them to the TTY. Then RANG has the TTY prompt the measurer with the bell to type in the cell length of the secondary section (scattered section) of the scattering particle track (usually the same value as the primary cell length).

Then the stage is moved to a new initial position just downstream from the vertex and in view of the vertex, whence the reset button is pushed and then the first ("segment done") button is pushed getting a set of nine zero bytes on the TTY. The cross hair is set along the secondary at several positions downstream (each separated by the chosen x-cell length) and pushes of the first button ("segment done") are made as with the primary. When the last position is reached, the third button ("secondary done") is depressed whence RANG now calculates and prints the secondary's angle elements and then goes on to find the net 3-d scattering "space angle" (as well as the next x-y projection angle and the net r-z "dip angle" where r is the projected horizontal displacement of a segment). Then RANG has the TTY print a prompt asking if a total print out of the detailed measurement results is desired. At this point, the measurer may feel that he has not exercised quite enough care in measuring the coordinates or a quick perusal of the print out might tell him this and so when he is asked next if he



wishes to retain this measurement, he will type no in response, whence RANG's running average of angle results will not include that particular measurement. Then, when he is asked next if he wishes to re-measure, he might do so if he wishes to increase his precision statistics on that event further, or he may not do so. When he has done his last re-measurement, RANG computes the grand average of that event's principle angle results and demographic data and stores them to the record in the random storage data disk (in the applications drive) file MUDAT (see TABLE 1) that has the same number as the event itself. At this point, one can close the session (RANG closes out the data file MUDAT) or begin a new event.

### Results and Conclusions

The system works quite well. Before, it was not time feasible to make several re-measurements (whether to correct marginal measuring techniques or to increase precision statistics) on a given event, because hundreds of such events might need to be measured before comparison with theory predictions can be made meaningful. But now, the measurement time has been reduced from upwards to an hour (for just one measurement) down to 15 minutes (including several re-measurements if wished). Further, the automation in data taking has greatly enhanced the reduction of error in contrast to before when there were reading errors (parallax, etc.), copying errors, typing errors, etc., not to mention the time required for all these latter tasks.

It is hoped that future use of this system for other types of measurement algorithms in the context of the nuclear emulsion technique of elementary particle research, whether using y and/or z and/or x dimension optical encoders, will only require a modification of the calculational source program (like RANG here), not the hardware.

### Literature Cited

1. THOMAS, GERALD P. AND COGGINS MARK. 1980. Microcomputer Assisted Multiple Coulomb Scattering Measurements in Nuclear Emulsion Pellicles. *Proceedings of the Indiana Academy of Science*. 89:375-81.