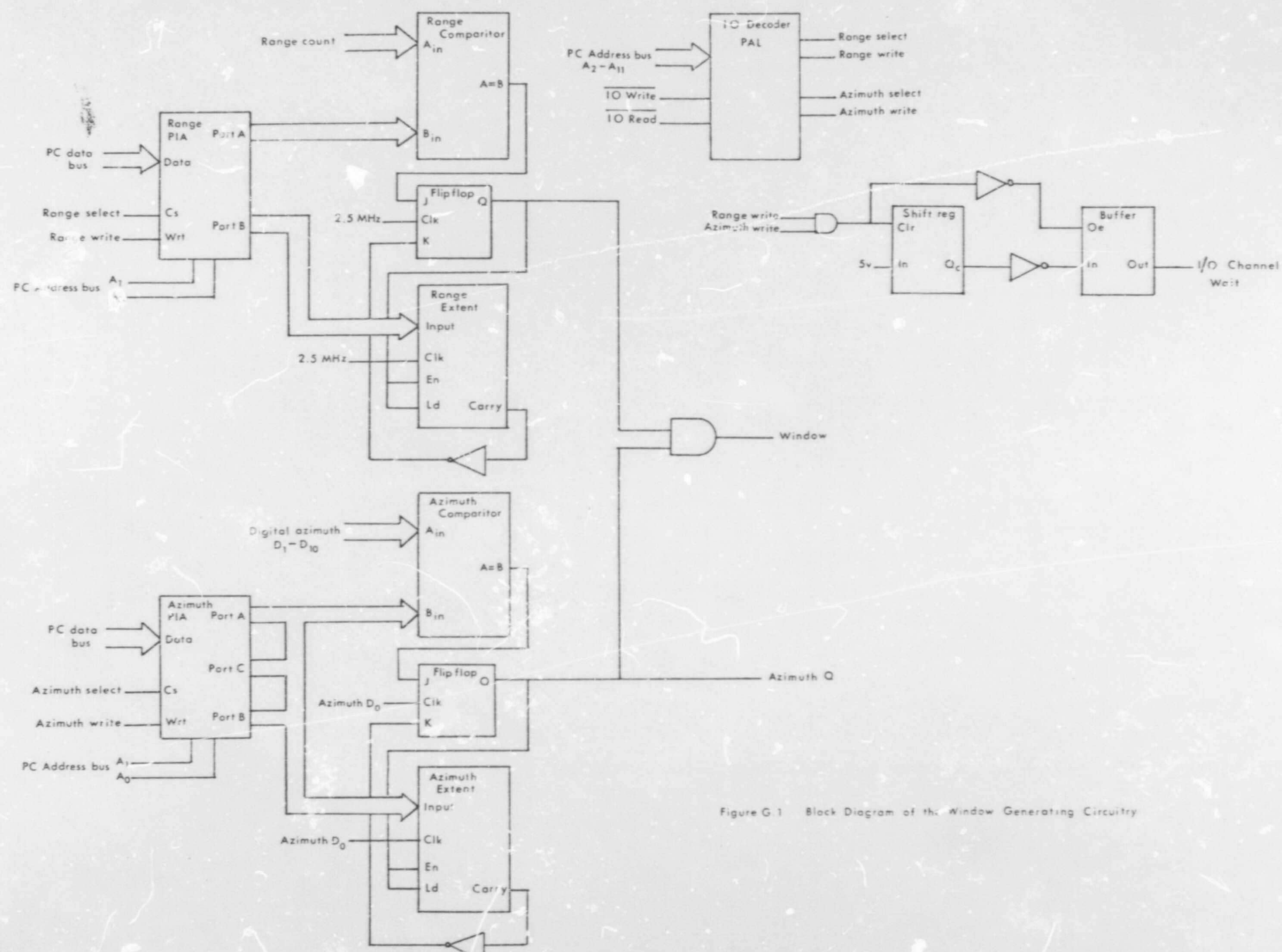


Figure G.1 Block Diagram of the Window Generating Circuitry

Both these transfers are byte-wide transfers. The azimuth information is rather more complicated. Two ten bit words need to be output to the PIA ports. The PIA, however, contains three 8 bit ports. It is thus necessary to output three byte-wide data words and to encode the words in such a way that the correct bits are output at the PIA ports. This is done by writing the lower 8 bits of both the start azimuth and the extent to ports B and A respectively. The two upper bits of the azimuth start are written in bits 4, 5 and the two upper bits of the extent are written in bits 0,1 of the word output by port C.

The two start values are input on the A inputs of the comparators. The B input of the range comparator is derived from the output of the range counter. The B input of the azimuth comparator is derived from the 10 most significant bits of the antenna azimuth position indicator. The two extent values are made avaliable at the inputs to the counters.

As the antenna scans around, the antenna azimuth position is compared to the value contained in the azimuth comparator. When the two values coincide the comparator gives off a pulse. This pulse is detected by the azimuth flip-flop and the Q is set. This loads the azimuth extent counters and enables the count sequence. When the counters reach FFFHex a carry is given which resets the flip-flop Q and disables the count sequence.

A similar operation is carried out for the range comparitor.

The range count is continuously compared to the start range value held in the comparator. When the two values are equal the comparator gives off a pulse. This pulse is detected by the range flip-flop and the Q is set. This loads the range extent counters and enables the count sequence. When the counters reach FFHex a carry is given which resets the flip-flop and disables the count sequence.

The range extent counters are clocked by the 2.5 MHz system clock. The azimuth extent counters need to be clocked by a signal that has a clock rate half that of the antenna position least significant bit. This signal is derived from the 11th most significant bit of the antenna azimuth position indicator.

Because use is made of up-counters for the extent counters the extents output to the counters need to subtracted from the maximum count. The range extent is thus found by subtracting the required extent from FFHex and the azimuth extent is found by subtracting the required azimuth extent from 1FFHex.

The output of the range flip-flop will be a pulse that is high for the time that the range counter lies within the range window. The output of the azimuth flip-flop is a pulse that is high for the period that the antenna lies within the azimuth window. If these two signals are then ANDed the result will be a pulse that is high only while the system lies within the window. This signal is thus used to gate the one-bit video to the memory card.

# APPENDIX H

## THE MEMORY CARD

Figure H.1 details the block diagram of the memory card.

The output of the AND gate from the window generator circuit is ANDed with the one-bit video signal. The output of this AND gate is used to enable a set of counters. These counters are used to count the number of echoes received within the window and to provide an address counter for the memories. Before the commencement of any sweep through the window the counters are loaded with 00Hex.

During the period that the antenna is within the azimuth extent of the window the azimuth flip-flop Q will select the output of the address counters as the address bus for tha memories, it will enable the buffers allowing the range and azimuth values of the echo access to the memory data bus and will disable the memory card from the IBM PC data bus. As each echo is detected the range and azimuth value of that particular echo will be written into a memory location.

When the antenna has scanned over the window, the video is gated off, disabling the counter and the data ready flip-flop is set to indicate to the processor that the data is ready for servicing. When the azimuth Q goes low it also selects the IBM PC address bus for the memories and places the range and azimuth buffers in a tri-state condition.

The memories are situated in an unused area of the IBM PC memory map. All memory transfers from the memory card to the IBM PC are thus implemented as though the memories were part of the IBM

Figure H.1 Block Diagram of the Memory Card

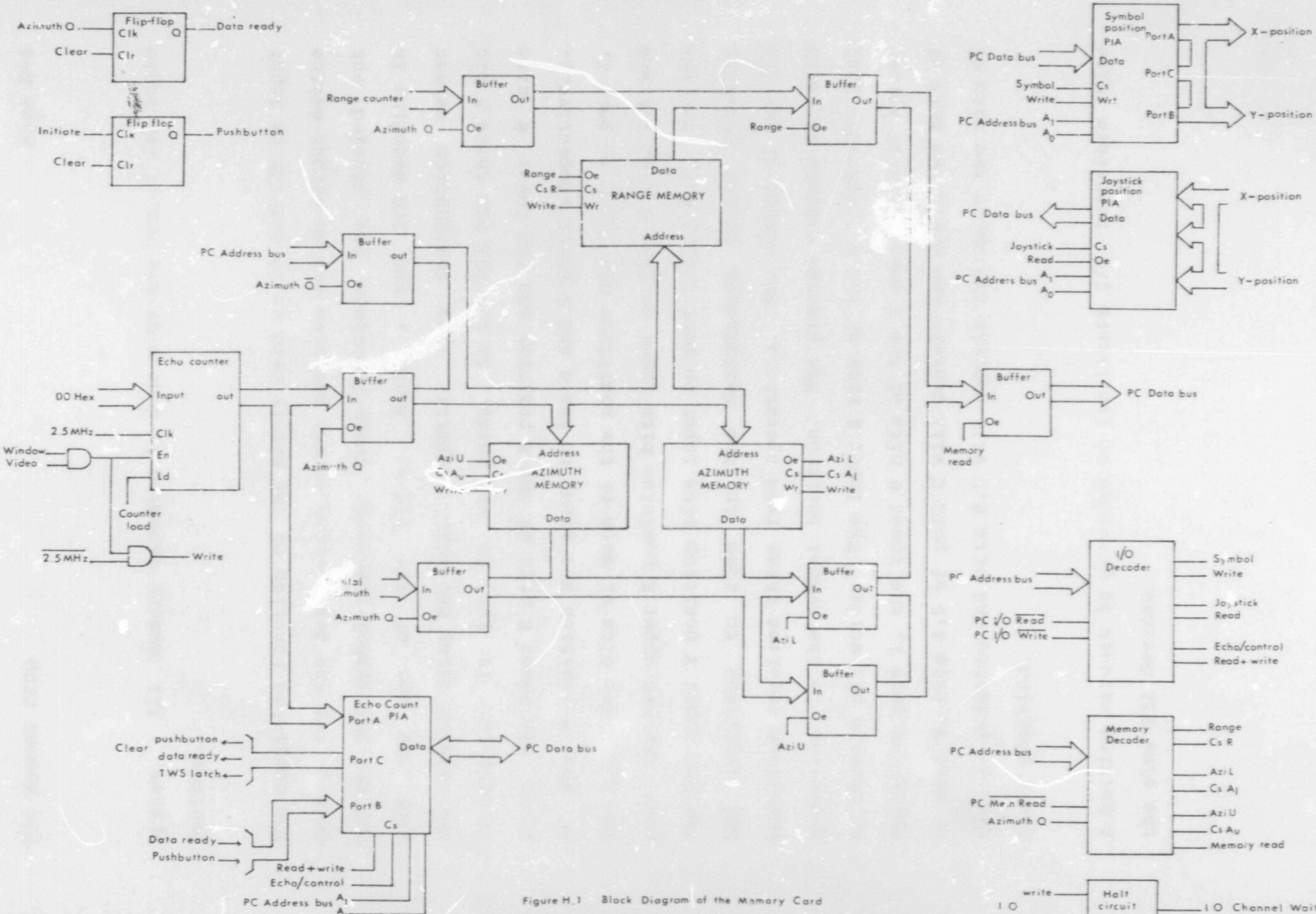


Figure H.1 Block Diagram of the Memory Card system. All memory decoding is done with the use of PAL logic devices.

The remaining function of the memory card is to provide an input device for the designation marker position and an output device for the TWS symbol position. These functions are carried out with the use of two PIA's. Again , the I/O decoding is implemented using PAL logic circuits. The designation marker information is two 10 bit words. As the PIA has three 8 bit ports, the lower 8 bits of the X position and the lower 8 bits of the Y position are input to the A and B ports respectively. The upper two bits of each of the positions are input on port C, with the two upper X position bits input on port C bits 4,5 and the two upper Y position bits input on port C bits 0,1. For the TWS algorithm to read in the designation marker position therefore requires three read operations. The output of the TWS processor is the symbol position. The program outputs the data as three 8 bit words. The lower 8 bits of the X position will appear on port A, the lower 8 bits of the Y position will appear on port B, bits 4,5 of port C will contain the upper two bits of the X position and bits 0,1 will contain the upper two bits of the Y position.

A halting circuit is provided to facilitate I/O transfers with the slow PIA devices.

# APPENDIX I

## SOFTWARE LISTING OF THE TWS PROGRAM

## I.1 THE SETUP PROCEDURE

Figure I.1 details the Setup Procedure.

```
20  ! SETUP PROCEDURE : SET UP THE PIA's
30  !                 : CLEAR ALL FLAGS
40  !                 : SET THE WINDOW SIZE
50      ------------------------------------------
60  !
70 OUT &H303,128
80 OUT &H307,128
90 OUT &H30B,154
100 OUT &H30F,155
110 OUT &H313,128
120 OUT &H310,128
130 OUT &H311,128
140 OUT &H312,48
150 OUT &H30A,1
160 OUT &H30A,14
170 INITIATE=0
180 RIN=900
190 RIN=RIN/60
200 TIN=900
210 !
220 !
```

The out statements configure the relevant PIA's for data input or output, clear all flip-flops and output the static symbol position to the symbol generator circuits. RIN is the required window range extent and this is converted to range counts. TIN is the required azimuth extent in meters. The INITIATE flag is reset.

## I.2 THE MAIN CALLING PROGRAM

Figure I.2 details the Main Calling Routine.

```
230 ! ------------------------------------------------------------------------
240 !
250 ! MAIN CALLING PROGRAM : PUSHBUTTON SET BY OPERATOR
260 !                      : READY      SET BY WINDOW GENERATOR
270 !
280 ! ------------------------------------------------------------------------
290 !
300 PUSHBUTTON=INP(&H309) AND &H2:READY=INP(&H309) AND &H1
310 IF READY=1 AND INITIATE=1 THEN GOSUB 470       ! CALL TRACKING ROUTINE
320 IF PUSHBUTTON=2 AND INITIATE=0 THEN GOSUB 890  ! CALL INITIATE ROUTINE
330 IF PUSHBUTTON=2 AND INITIATE=1 THEN GOSUB 190  ! CALL CANCEL ROUTINE
340 GOTO 300
350 !
```

One may see that this is in fact an endless loop. While the program remains in the loop the PUSHBUTTON and READY flags are continuosly monitored. If the operator depresses the pushbutton the PUSHBUTTON flag is set. If no track has been initiated then the INITIATE flag will be reset and the program will call the Initiate Routine. If however a track was in progress the INITIATE flag will be set and the Cancel Routine will be called. When a track has been initiated the INITIATE flag is set and once the antenna has scanned through the window the READY flag is set and the Tracking Routine will be called.

## I.3 THE TRACKING ROUTINE

Figure I.3 details the Tracking Routine.

The first operations merely ensure that the flags are correctly set and the READY flag is reset. The NUMBER of echoes from the echo counter is then input. If the NUMBER was greater than zero then the COAST counter is reset. The mean azimuth position is

```
360 !
370 ! ------------------------------------------------------------------------
380 !
390 ! TRACKING PROCEDURE : READS IN NUMBER OF ECHOES
400 !                    : CALCULATES THE TARGET POSITION
410 !                    : PERFORMS THE TRACK FILTERING
420 !                    : OUTPUTS THE PREDICTED POSITION TO THE DISPLAY
430 !                    : OUTPUTS THE PREDICTED POSITION TO THE WINDOW
440 !                      GENERATOR
450 ! ------------------------------------------------------------------------
460 !
470 INITIATE=1:READY=0:PUSHBUTTON=0:FOR I=1 TO 100:NEXT I
480 NUMBER=INP(&H308):NUMBER=INP(&H306):IF NUMBER=0 THEN GOTO 610
490 COAST=0:DEF SEG=&H8000
500 AZIMIN=PEEK(&H1000)*4+PEEK(&H1800)/64 AND &H3FF
510 AZIMAX=PEEK(&H1000+(NUMBER-1))*4+PEEK(&H1800+(NUMBER-1))/64 AND &H3FF
520 AZIMEAN=(AZIMIN+AZIMAX)/2
530 RMIN=255:RMAX=0
540 FOR I=0 TO NUMBER-1
550 RVAL=((PEEK(I)*256+PEEK(I+&H800))/128 AND &H1FF)-256
560 IF RVAL<RMIN THEN RMIN=RVAL:IF RVAL>RMAX THEN RMAX=RVAL
570 NEXT I
580 RMEAN=(RMIN+RMAX)/2:TVAL=AZIMEAN*.006136
590 XMEASG=RMEAN*2*SIN(TVAL)+511:YMEASG=RMEAN*2*COS(TVAL)+511
600 GOTO 630
610 IF COAST=5 THEN GOTO 880
620 COAST=COAST+1:XMEASG=XPREDG:YMEASG=YPREDG:N=1
630 ALPHA=(2*N+1)^2/(N*(N+1))-3.7:BETA=ALPHA^2/(2-ALPHA)
640 XERROR=XMEASG-XPREDG:YERROR=YMEASG-YPREDG
650 XSMOOTH=XPREDG+ALPHA*XERROR:YSMOOTH=YPREDG+ALPHA*YERROR
660 XVEL=XVEL+BETA*XERROR/1.5:YVEL=YVEL+BETA*YERROR/1.5
670 XPREDG=XSMOOTH+XVEL*1.5:YPREDG=YSMOOTH+YVEL*1.5
680 XPREDC=XPREDG-511:YPREDC=YPREDG-511
690 RANGE=SQR(XPREDC^2+YPREDC^2)/2:THETA=ATN(XPREDC/YPREDC)*57.2958
700 IF XPREDC>=0 AND YPREDC>=0 THEN GOTO 740
710 IF YPREDC<0 THEN GOTO 730
720 THETA=THETA+360:GOTO 740
730 THETA=THETA+180
740 GOSUB 1230:GOSUB 1530        ! CALL SYMBOL OUTPUT ROUTINE
750                              ! CALL WINDOW OUTPUT ROUTINE
760 OUT &H30A,0:OUT &H30A,14:BEEP:N=N+1:RETURN
770 !
```

calculated using the mean of the initial azimuth position and the final azimuth position. Once this is completed the minimum range and the maximum range are calculated using a sort routine. This then allows one to calculate the mean of the range extremes. The mean azimuth and the mean range position then form the measured target position.

If the NUMBER of echoes was zero then one of two possibilities exist. If the COAST counter indicated that the echo counter had not received echoes for fewer than the last five scans then the COAST counter is incremented by 1 and the target predicted position is used as the present measured position.

The program then enters the filter algorithm. The value of alpha is calculated according to the scan number, N. The value of beta is then calculated. The output of the tracking filters will be the X and Y positions of the target predicted position. These values are then converted to R-Θ coordinates. Because the azimuth value returned by the IBM PC is an angle lying between +90 and -90 degrees the value of theta output to the subroutines must be adjusted to allow full four-quadrant operation.

The predicted range and azimuth position is then output to the symbol output subroutine and the window output subroutine. The echo counter is then reset, the scan number is incremented by 1 and the routine returns to the Main Calling Routine.

If the COAST counter indicated that no target echoes had been received within the last five scans then the track is cancelled by setting the PUSHBUTTON flag and returning to the main calling routine. Upon returning to the main loop the conditions set up would immediately cause the Cancel Routine to be executed.

## I.4 THE INITIATE ROUTINE

Figure I.4 details the Initiate Routine.

```
780 !
790 ! ------------------------------------------------------------------------
800 !
810 ! INITIATE ROUTINE : SETS THE TRACKING FLAG
820 !                  : READS IN THE JOYSTICK POSITION
830 !                  : SETS UP THE ACQUISITION WINDOW
840 !                  : OUTPUTS THE SYMBOL POSITION TO THE DISPLAY
850 !                  : INITIATES THE TRACKING FILE
860 ! ------------------------------------------------------------------------
870 !
880 INITIATE=1:PUSHBUTTON=2:BEEP:RETURN
890 PUSHBUTTON=0:INITIATE=1:CLS:N=1
900 XLOW=INP(&H30C):YLOW=INP(&H30D).COMB=INP(&H30E)
910 XBLOCK=(XLOW AND &HFF)+(COMB AND &H30)*16
920 YBLOCK=(YLOW AND &HFF)+(COMB AND &H3)*256
930 IF XBLOCK>=511 AND YBLOCK>=511 THEN GOTO 990
940 IF XBLOCK>=511 AND YBLOCK<511 THEN GOTO 980
950 IF XBLOCK<511 AND YBLOCK<511 THEN GOTO 970
960 XPOSC=XBLOCK-526:YPOSC=YBLOCK-496:GOTO 1000
970 XPOSC=XBLOCK-526:YPOSC=YBLOCK-526:GOTO 1000
980 XPOSC=XBLOCK-496:YPOSC=YBLOCK-526:GOTO 1000
990 XPOSC=XBLOCK-496:YPOSC=YBLOCK-496
1000 RANGE=SQR(XPOSC^2+YPOSC^2)/2:THETA=ATN(XPOSC/YPOSC)*57.2958
1010 IF XPOSC>=0 AND YPOSC>=0 THEN GOTO 1050
1020 IF YPOSC<0 THEN GOTO 1040
1030 THETA=THETA+360:GOTO 1050
1040 THETA=THETA+180
1050 GOSUB 1230:GOSUB 1530         ! CALL SYMBOL OUTPUT ROUTINE
1060                               ! CALL WINDOW OUTPUT ROUTINE
1070 XPREDG=XPOSC+511:YPREDG=YPOSC+511
1080 XVEL=0:YVEL=0:COAST=0
1090 OUT &H30A.0:OUT &H30A.14:RETURN
1100 !
```

The first operation is to reset the PUSHBUTTON flag , to set the INITIATE flag and to set the scan number to 1.

The symbol position is then input. Because the position data is input as three 8-bit words the position has to be decoded from the input values. Xlow contains the lower 8 bits of the X position and Comb contains the upper 2 bits in bit positions 4 and 5. Ylow contains the lower 8 bits of the Y position and Comb contains the upper 2 bits in bit positions 0 and 1. The center of the block referred to the (512,512) grid is then calculated. This position is arrived at by adding or subtracting offsets depending on which quadrant the marker is found to lie.

Once the marker center has been calculated the coordinates are transformed to R-θ coordinates. Again the value of theta is adjusted to allow full four-quadrant operation.

The marker position is then output to the symbol output routine and also to the window output routine.

The marker position is found relative to the (0,0) grid and this value is used to initiate the filter. The marker position forms the predicted X and Y target position. The filter velocities are initialised with zero. The pushbutton flip-flop is reset and the routine returns to the Main Calling Routine.

## I.5 THE CANCEL ROUTINE

Figure I.5 details the Cancel Routine.

The PUSHBUTTON flag and the INITIATE flag are reset. The symbol static position is output to the symbol generating circuits. The routine then returns to the Main Calling Routine.

```
1110 !
1120 ! ----------------------------------------------------------------
1130 !
1140 ! CANCEL ROUTINE : CLEARS THE TRACKING FLAG
1150 !                : OUTPUTS THE SYMBOL STATIC POSITION
1160 !
1170 ! ----------------------------------------------------------------
1180 !
1190 PUSHBUTTON=0:INITIATE=0
1200 OUT &H310,126:OUT &H311,128:OUT &H312,48
1210 OUT &H30A,7:OUT &H30A,14:OUT &H30A,6:OUT &H30A,14
1220 RETURN
1230 XPOSC=CINT(RANGE*2*SIN(THETA*.017453)):YPOSC=CINT(RANGE*2*COS(THETA*.
))
1240 !
1250 !
```

## I.6 THE SYMBOL OUTPUT SUBROUTINE

Figure I.6 details the Symbol Output Subroutine.

```
1260 ! ----------------------------------------------------------------
1270 !
1280 ! SYMBOL OUTPUT ROUTINE : OUTPUTS THE INITIAL START POINT
1290 !                         OF THE SYMBOL TO THE POSITIONING
1300 !                         CIRCUITRY
1310 ! ----------------------------------------------------------------
1320 !
1330 IF XPOSC>=0 AND YPOSC>=0 THEN GOTO 1390
1340 IF XPOSC>=0 AND YPOSC<0 THEN GOTO 1380
1350 IF XPOSC<0 AND YPOSC<0 THEN GOTO 1370
1360 XBLOCK=XPOSC+526:YBLOCK=YPOSC+496:GOTO 1400
1370 XBLOCK=XPOSC+526:YBLOCK=YPOSC+526:GOTO 1400
1380 XBLOCK=XPOSC+496:YBLOCK=YPOSC+526:GOTO 1400
1390 XBLOCK=XPOSC+496:YBLOCK=YPOSC+496
1400 XLOW=XBLOCK AND &HFF:XUP=(XBLOCK-XLOW)/16
1410 YLOW=YBLOCK AND &HFF:YUP=(YBLOCK-YLOW)/256
1420 COMB=XUP OR YUP
1430 OUT &H310,XLOW:OUT &H311,YLOW:OUT &H312,COMB
1440 OUT &H30A,15:OUT &H30A,14:RETURN
1450 !
1460 !
```

The first operation is the conversion of the input position from R-θ coordinates to X-Y coordinates. The lower left hand position of the symbol is then calculated with reference to the

(0,0) grid. The two 10 bit words then need to be converted into three 8 bit words. The lower 8 bits of the X-position are output as Xlow. The lower 8 bits of the Y-position are output as Ylow. The upper 2 bits of the X-Position are then written to bits 4 and 5 of Comb and the upper 2 bits of the Y-position are written to bits 0 and 1 of Comb. The three 8 bit words are then latched out to the symbol positioning circuits. The routine then returns to the calling routine.

## I.7 THE WINDOW OUTPUT SUBROUTINE

Figure I.7 details the Window Output Subroutine.

```
1470 ! -----------------------------------------------------------------
1480 !
1490 ! WINDOW OUTPUT ROUTINE : OUTPUTS THE WINDOW DIMENSIONS TO
1500 !                         THE  WINDOW GENERATOR
1510 ! -----------------------------------------------------------------
1520 !
1530 DELTAR=&H1FF-CINT(RIN):ROUT=CINT(RANGE-RIN/2)
1540 IF ROUT<84 THEN GOTO 1560
1550 DELTAT=849.89/ROUT:ANGLE=THETA-DELTAT/2:GOTO 1570
1560 DELTAT=10.24:ANGLE=THETA-5.12
1570 IF ANGLE<0 THEN ANGLE=ANGLE+360
1580 DELTAT=&H3FF-CINT(DELTAT*2.845):ANGLE=CINT(ANGLE*2.845)
1590 OUTR=ROUT AND &HFF:OUTDR=DELTAR AND &HFF
1610 OUT &H300,OUTDR:OUT &H301,OUTR
1620 OUTA=ANGLE AND &HFF:OUTDA=DELTAT AND &HFF
1630 AUPPER=(ANGLE-OUTA)/256:DAUPPER=(DELTAT-OUTDA)/16:OUTDAA=AUPPER OR DA
1640 OUT &H304,OUTDA:OUT &H305,OUTA:OUT &H306,OUTDAA
1650 RETURN
```

The first operation is to calculate the range extent, DELTAR. This involves subtracting the required extent in range bins from the value 1FFhex. The initial range value is then calculated as

ROUT. The azimuth extent and initial azimuth value are dependent on the value of ROUT.

If ROUT is greater than 84 (5040 m) then DELTAT is calculated using the trigonometric formula for radius of arc and radius. The initial azimuth value ANGLE is calculated from theta and the azimuth extent.

If ROUT is less than 84 then the azimuth extent DELTAT is fixed at 10 degrees and the initial azimuth value is calculated from theta and the extent.

DELTAT and ANGLE are then converted from angular measure to azimuth counts. The range ROUT and DELTAT are output as is to the window generating circuits. The azimuth information must first be manipulated. The lower 8 bits of DELTAT are output to the window generator as OUTDA and the lower 8 bits of ANGLE are output to the window generator as OUTA. The upper two bits of ANGLE are written to bit positions 4 and 5 of OUTDAA and the upper 2 bits of OUTDA are written to bit positions 0 and 1 of OUTDAA.

The subroutine then returns to the calling routine.