NATIONAL RADIO ASTRONOMY OBSERVATORY
Green Bank, West Virginia

Spectral Processor Memo No. 31

Arpil 5, 1988

To: Spectral Processor Group
and Mailing List

From: R. Lacasse

Subj: **Accumulator Memory Controller 1:
Communications with Rack Controller**

A description of the above is attached.

RJL/cjd

Enclosure

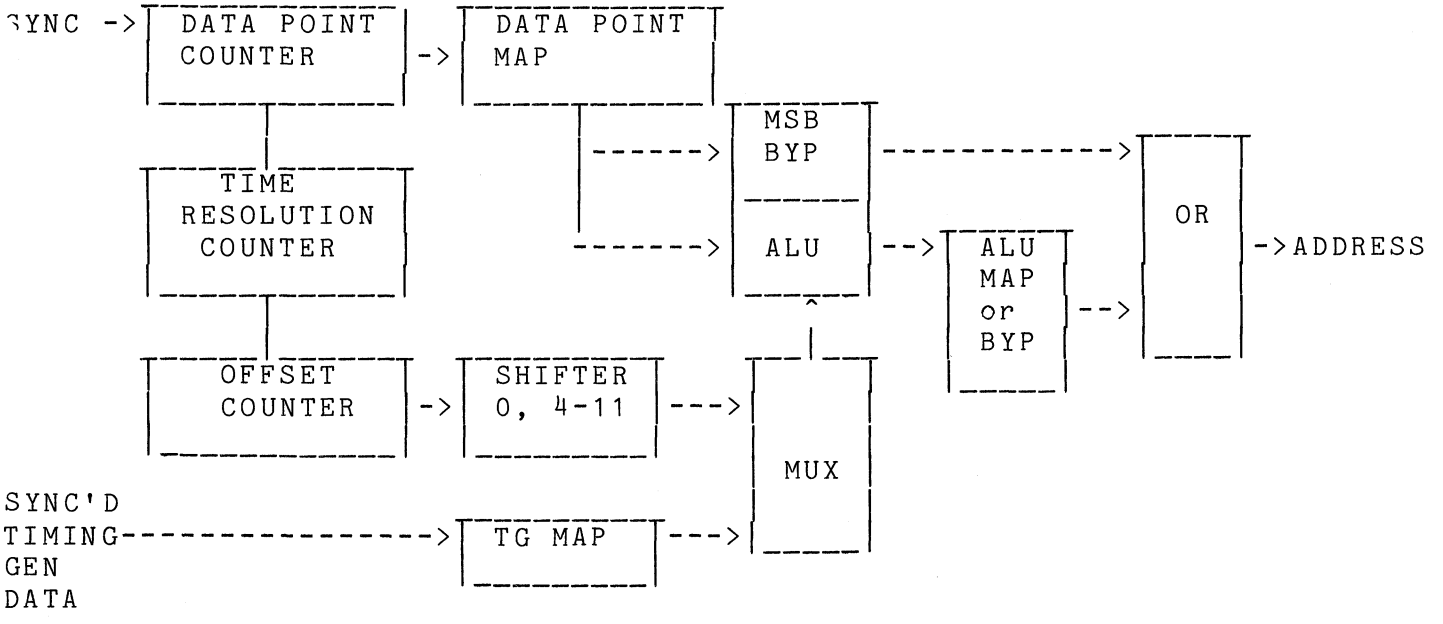
Accumulator Memory Controller 1
Communications with Rack Controller

Introduction:

Accumulator Control Card 1 generates addresses for the accumulator's memories; it also contains the interface to the rack controller, effectively acting as a clearinghouse for rack controller information to and from the entire accumulator. This memo presents a brief functional description of address generator, as a help in understanding the effect of the various control memories and bits. This is followed by a description of what the rack controller can control on the card, including various memories, control pulses, and control bits. Next, a description of the bits in the status word is provided. Finally, the details of how to gain access to the various control pulses, control and status bits, and memories on the card is presented.

Functional Overview:

This card's primary function is to generate addresses for the memory cards. Because of the large number of accumulator modes supported, the addressing logic has to be correspondingly complex. The card generates addresses for two memory cards, termed A and B. For now let us consider addressing just the A memory. A simplified block diagram for the address generator part of the card is shown below:



Operation is briefly as follows. The counters keep up with which data point is being processed. The data point number, from the data point counter, is routed through a map RAM to allow for various modes. The output of the data point map is summed with either a shifted version of the offset counter or with a mapped version of the sync'd timing generator data. The purpose of the TG MAP is to route the desired timing generator signals to the desired ALU bits. The output of the ALU is also mapped to allow for yet another addressing mode (synchronous dedispersion); this

ıap can be bypassed, when not required, to save down-loading time. When ınultiple IF's are being processed, the MSB's of the DATA POINT MAP must be bypassed to offset the addresses of the various channels.

In addition to what is shown above, the card includes a BUFFER COUNTER which is driven by the OFFSET COUNTER. Its purpose is, in some modes, to indicate when to dump data. It has an associated RAM, which allows for varying the time between dumps. For example, we might want to integrate signal and reference for 100,000 FFT cycles, and then have two dump cycles in a row to dump both the signal and reference data. Also, in adition to what is shown above, there is logic to clear all the data rams, and some error sensing circuits.

What needs to be controlled:

This section details the RAMs, control pulses, and control bits which the rack controller must control.

ITEM	DESCRIPTION
A DATA POINT RAM	Maps data point number to an address component for the A memory. Also holds DUMP ME and BLANK ME control bits for each data point. The DUMP ME bit flags which points get dumped during a dump cycle. The BLANK ME bit allows for blanking of individual data points. The "address component" is in bits 0 to 13. The BLANK ME bit is bit 14, and the DUMP ME bit is bit 15. (2K X 16)
B DATA POINT RAM	Ditto for B memory. (2K X 16)
TG MAP	Rearranges the sync'd timing generator bits for input to the ALU's. Bit 0 of the RAM goes to ALU bit 8, bit 1 to bit 9, etc. (2K X 8) Address bits to the RAM are as follows:

BIT NO.	DESCRIPTION
0	CAL
1	SIG/REF
2	t0
3	t1
4	t2
5	t3
6	t4
7	T0
8	T1
9	T2
10	T3

A ALU MAP — Maps A ALU outputs to desired addresses. (16K X 16)

B ALU MAP — Ditto for B ALU. (16K X 16)

MCIMR- — When low, resets the Masscomp interface on Accumulator Mem Control Card 2. Required to initiate communications.

ACCINIT- — When low, initializes the Masscomp interface on Accumulator Mem Control Card 2. Required at the beginning of a scan.

CLRDMEM+ — Rising edge initiates data RAM clear. Must be high for at least two accumulator clocks to be sensed.

MEMOE+ — High output enables control memories.

MBUFOE+ — High, output enables the transceivers between the control RAMS and the local data bus.

A TO B+ — Controls the direction of the control RAM transceivers. High sets the direction towards the control memories.
Note: the above three control bits, MEMOE+, MBUFOE+, and A TO B+, must be used together to assure that the memory outputs and the transceiver outputs going to the memory are not on simultaneously. Initially, all are off. To write, set MBUFOE+ and A TO B+ high. To verify, FIRST change A TO B+ to low, then bring MEMOE+ high.

CLMA- — Controls RAM address source. Must be high for write and verify, and low for data taking.

LASTFRAME+ — Bring high to tell Masscomp that the present data frame is to be the last for this scan. Hold high until after ACCINIT- is brought low.

RCIFN2,RCIFN1, RCIFN0 — Determine the effect of the subsequent write or read to/from counter location as follows:

RCIFN 2 1 0	READ	BOTH	WRITE
0 0 0	NOP		CONTROL REGS.
0 0 1		A DATA POINT MEM	
0 1 0		B DATA POINT MEM	
0 1 1		BUF CTR LO MEM	
1 0 0		BUF CTR HI MEM	
1 0 1		TG MAP	
1 1 0		A ALU MAP	
1 1 1		B ALU MAP	

ACCMODE3, ACCMODE2
ACCMODE1, ACCMODE0 — Control the source of the dump commands for accumulator modes as defined in Memo 22, and repeated for convenience in Appendix C.

SPCONFIG3,
SPCONFIG2,
SPCONFIG1,
SPCONFIG0 — Specifies the Spectral Processor's configuration as it affects RFI flag monitoring. Fourteen allowable configurations are defined in Appendix A.

SKIP MSWS, ERRFN1,
ERRFN0, MCITEST+ — Define the format of data to Masscomp as detailed in Appendix B.

TRP<0:15> — Parallel load bits for the time resolution counter. Enter the two's complement of the desired resolution (e.g. hex FFFF gives time resolution of 1).

OFP<0:15> — Parallel load bits for the offset counter. Format as for TRP.

PDLYP<0:15> — Specifies the delay of the data from the A to D Interface module to the Accumulator inputs. This must be calculated by the Masscomp, based on its knowledge of the Spectral Processor configuration. The number that must be entered is the two's complement of the number of accumulator clocks from the A to D interface to the Accumulator. The first attempt at documenting data for this formula is shown in Appendix D

OFCTRSH2,
OFCTRSH1,
OFCTRCH0 — Specifies the desired number of shifts of the offset counter. The number of shifts equals four plus OFCTRSH<2:0>. Zero shift is obtained with BALUS<1:0>, below.

BALUS1, BALUS0 — Selects B input of both address ALU's as follows:

BALUS1	BALUS0	SELECTED INPUT
0	0	Unshifted offset counter
0	1	Shifted offset counter
1	0	Timing generator map for bits 8 to 15. Shifted offset counter for bits 0 to 7; the intent is to have the offset counter shifted at least 8 positions to give bits 0 to 7 equal to zero.

ADMAPCS+ — When high, selects the A and B ALU maps. When low selects the map bypass mode.

ADMSBSEL<0:2> — A high on any of these bits selects one of the MSB address bits to be bypassed, as follows:

ADMSBSEL<0>	DATA POINT MAP<11>
ADMSBSEL<1>	DATA POINT MAP<12>
ADMSBSEL<2>	DATA POINT MAP<13>

CLR OMEMOVFL- — A control pulse, used to reset the output memory overflow flag.

CLR OMEM SYNC- — A control pulse, used to clear an output memory counter sync error.

CLR DPSYNCERR- — A control pulse, used to clear a data point counter sync error.

CLR CSTARTERR- — A control pulse, used to clear a start synchronization error.

Status Bits:

Seven status bits are available for monitoring from the rack controller. They include:

BOMEMOVFLERR+ — High indicates that the output FIFO has overflowed, i.e., wrapped around on itself.

BOMEMSYNCERR+ — High indicates that the output FIFO read, write, and difference counter are out of sync with one another.

BLASTFRAMEDONE+ — High indicates that the last frame of a scan has been transmitted to the Masscomp.

BGOERROR+ — High indicates that a GO pulse was expected but not received from the Masscomp

DPSYNCERR+ — High indicates that the data point counter and the sync accompanying the data to the accumulator are not synchronized.

CSTARTERR+ — High indicates that the sync pulse, marking the first data word in the scan did not arrive at the expected time.

CLRDONE+ — High indicates that the clear memory request has been completed.

How to Access Bits and Memories:

Because of the limited addressing capability from the rack controller to the modules (4 bits), addressing the 40K locations on the card is a bit lumsy. An attempt was made in the design to make communication to the most important, or most often used locations easiest, at the expense of the less often used ones. The locations of the control bits listed above are detailed below. Then the mechanics of how to write to and in some cases read from those location are discussed.

LOCATION		CONTROL BITS
FUNCTION REG	0	MCIMR-
	1	ACCINIT-
	2	CLRMEM+
	3	MEMOE+
	4	MBUFOE+
	5	A TO B+
	6	CLMA+
	7	LASTFRAME+
	13	RCIFN0
	14	RCIFN1
FUNCTION REG	15	RCIFN2
CONREG0	0	ACCMODE0
CONREG0	1	ACCMODE1
CONREG0	2	ACCMODE2
CONREG0	3	ACCMODE3
CONREG1	0	SPCONFIG0
CONREG1	1	SPCONFIG1
CONREG1	2	SPCONFIG2
CONREG1	3	SPCONFIG3
CONREG1	4	SKIPMSWS
CONREG1	5	ERRFN0
CONREG1	6	ERRFN1
CONREG1	7	MCITEST+
CONREG2<0:15>		TRP<0:15>
CONREG3<0:15>		OFP<0:15>
CONREG4<0:15>		PDLYP<0:15>
CONREG5	0	OFCTRSH0
CONREG5	1	OFCTRSH1
CONREG5	2	OFCTRSH2
CONREG5	3	BALUS0
CONREG5	4	BALUS1
CONREG5	8	ACMAPCS+
CONREG5	9	ADMSBSEL<0>
CONREG5	10	ADMSBSEL<1>
CONREG5	11	ADMSBSEL<2>
PULSE BUF	0	CLR OMEMOVFL-
PULSE BUF	1	CLR OMEMSYNC-
PULSE BUF	2	CLR DPSYNCERR-
PULSE BUF	3	CLR CSTARTERR-
STATUS	0	BOMEMOVFLER+
STATUS	1	BOMEMSYNCERR+
STATUS	2	BLASTFRAMEDONE+
STATUS	3	BGOERROR+
STATUS	4	DPSYNCERR+
STATUS	5	CSTARTERR+
STATUS	6	CLRDONE+

A word on notation:

In the read and writes to control registers discussed below, individual bits are called out in some cases. An "x" in the bit location means. An "n" in the location means user selected value. A "u" in the location means leave this bit unchanged. A "0" or "1" entry of means to set the bit to 0 or 1. For example if we want to reset bits 0 to 7 of CONREG3 and leave bits 8 to 15 alone, we express this as

write uuuu uuuu 0000 0000 to CONREG3.

Also, the high order bits of the address are represented only as "xx", as they differ in racks A and B; for rack A, xx = 07, and for rack B, xx = 27.

-Writing to the function register:
Simple write to address xx0.

-Writing to any CONREG:
Step 1: write to function register to make RCIFN0=RCIFN1=RCIFN2=0.
(write 000x xxxx xxxx xxxx to address xx0)
Step 2: write the CONREG number to the address counter.
(write 0000 0000 0000 0nnn to address xx1)
Step 3: write CONREG data.
(write nnnn nnnn nnnn nnnn to address xx2)

Writing to several successive CONREGs:
Step 1: as above
Step 2: write the lowest CONREG number to the address counter
(write 0000 0000 0000 0nnn to address xx1)
Step 3: write first CONREG data as above.
Step 4: successive writes to xx2 will load successive control registers.

-Writing to control RAMs:
Step 1: Write to function register to indicate desired RAM

RCIFN2	RCIFN1	RCIFN0	ADDRESSED MEMORY
0	0	1	A DATA POINT
0	1	0	B DATA POINT
0	1	1	BUFFER COUNTER LOWER
1	0	0	BUFFER COUNTER UPPER
1	0	1	TG MAP
1	1	0	A ALU MAP
1	1	1	B ALU MAP

(write nnnx xxxx xx10 0xxx to address xx0)
(write nnnx xxxx xx11 0xxx to address xx0)

Step 2: Write starting address to address counter
(write nnnn nnnn nnnn nnnn to address xx1)

Step 3: Write desired data to address xx2. The address counter auto-increments after each write.

-Making control pulses:
Simply write to the PULSE BUF. Bits that are low will cause a pulse.
(write 1111 1111 1111 nnnn to address xx3)

-Reading status bits:
Simply read the status word.
(read address xx8)

-Reading control memories:
Step 1: Specify the desired RAM by writing to the RCIFN bits in the Function Register.
(write nnnx xxxx xx00 0xxx to address xx0)
(write nnnx xxxx xx00 1xxx to address xx0)
where nnn are as specified above in the "write to control memory" section.
Step 2: Write starting address to address counter.
(write nnnn nnnn nnnn nnnn to address xx1)
Step 3: Read the desired data from address xxA. The address counter auto-increments after each write.

APPENDIX A

Spectral Processor Configurations

From the point of view of the RFI blanking logic, fourteen configurations of the Spectral Processor affect the way in which the RFI masks are used. Four factors are involved in determining the fourteen configurations. The first is the number of IF's/number of channels; obviously we don't care about RFI flags 2 to 8 in one channel mode, etc. The second factor is the clock rate of the accumulator; it can either be equal to the FFT clock rate, termed 1X, or twice as fast, termed 2X. The third factor is the Real Correction read rate. The Real Correction module can output data at the same rate as it receives data from the last FFT module, or it can output half the data and hold each output twice as long, to allow the Square and Cross Multiply module to square and cross multiply each point. The last factor is the Square and Cross multiply mode. The Square and Cross Multiply module can be in one of three modes: Square only, Cross only, Square and Cross. From the RFI masking point of view, the first two modes are equivalent and can be termed Square or Cross. The table below summarizes the allowable combinations and assigns a number to each configuration; the assigned number is what is entered for the SPCONFIG bits.

#IF/CH	CLOCK	R.C.	SQ/X MUL	MODE NUMBER
2/256	1X	NOR	SQ OR X	0
2/256	1X	HALF	SQ AND X	1
2/256	2X	NOR	SQ AND X	2
1/1024	1X	NOR	SQ OR X	3
1/1024	2X	NOR	SQ AND X	4
2/512	1X	NOR	SQ OR X	5
2/512	1X	HALF	SQ AND X	6
2/512	2X	NOR	SQ AND X	7
4/256	1X	NOR	SQ OR X	8
4/256	1X	HALF	SQ AND X	9
4/256	2X	NOR	SQ AND X	A
8/128	1X	NOR	SQ OR X	B
8/128	1X	HALF	SQ AND X	C
8/128	2X	NOR	SQ AND X	D

APPENDIX B

Communication Format: Accumulator to Masscomp

This appendix presents the general format of the frames sent to the Masscomp from the Accumulator. Then the details of how certain control bits affect the frame are presented.

General Frame Format:

```
SYNC0+ (= hex DDEE)
SYNC0+
SYNC0+
FRAME COUNT
FIRST CYCLE DATA:
          A UPPER LSW
          A UPPER MSW
          A LOWER LSW
          A LOWER MSW
          B UPPER LSW
          B UPPER MSW
          B LOWER LSW
          B LOWER MSW
          A UPPER ERR
          A LOWER ERR
          B UPPER ERR
          B LOWER ERR
NEXT N CYCLES OF DATA:
          Similar to above
STATUS
          BIT 15: GO ERROR+
          BIT 14: LAST FRAME+
CHECK SUM (entire frame through status, modulo 16)
SYNC0- (= hex 2211)
SYNC0- (= hex 2211)
```

Control Bit Effect on Frame Format:

Three control bits affect the format of the data part of the frame sent to the Masscomp. The first of these is SKIP MSWS. When high only the least significant 16 bits of accumulated data are sent; otherwise, all thirty-two bits are sent with the least significant bits sent first. The other two control bits are ERRFN0 and ERRFN1. Their effect is shown in the table below:

ERRFN1	ERRFN0	DESCRIPTION
0	0	Transmit all error data
0	1	Transmit "A Upper" error word in first cycle only
1	0	Do not transmit any error words
1	1	Transmit "A Upper" error word every M data cycles

A fourth control bit, MCITEST+, affects the contents of the data part of the frame. When low, the bit lets the "normal" data be transmitted, and when high, a test counter is substituted for the data.

Setting the Number of Data Cycles:

The number of data cycles, including the first, is set by programming an 8254 timer chip. The text below gives a cookbook approach to this programming chore. A data sheet is available in the Timing Generator write-up, as well as in Intel and other data books.

Timer 0 in the chip is programmed as a rate generator to set the number of data cycles. This is accomplished as follows, with comments inserted with C-like syntax:

```
/*write counter 0 control word, selecting mode 2 and 2-byte load*/
     write(xxxx xxxx 0011 x100 to address xx7)

/*write counter 0 LSB*/
     write(xxxx xxxx nnnn nnnn to address xx4)

/*write counter 0 MSB*/
     write(xxxx xxxx nnnn nnnn to address xx4)
```

In mode 2 the counter starts counting at the loaded value (nn..nn above) and counts down until it reaches 1. While the count is at 1, the counter output is high, and is used as a flag to signal the last data cycle. Therefore, the value loaded to the counter is simply the number of data cycles desired in the frame, including the first cycle, in hex.

Setting the Error Counter Transmit Rate:

When the control bits ERRFN1 and ERRFN2, described above, are both 1, the "A Upper" error word is transmitted every M data cycles. This mode would be used, for instance, when doing spectroscopy with RFI excision. In this mode, error words for each frequency channel are equal, since RFI excision is done on a transform by transform basis. Therefore we provide the means to send just one error word for each spectrum.

This facility is implemented using part of the same 8254 chip mentioned in the above section, this time using counter 1. Again, the counter is set up as a rate generator, mode 2, as follows:

```
/* write counter 1 control word, selecting mode 2 and 2-byte load*/
     write(xxxx xxxx 0111 x100 to address xx7)

/*write counter 1 LSB*/
     write(xxxx xxxx nnnn nnnn to address xx5)

/*write counter 1 MSB*/
     write(xxxx xxxx nnnn nnnn to address xx5)
```

Writing a value of M will result in the "A Upper" error word being transmitted on data cycles M, 2M, 3M, ... etc. to the end of frame.

APPENDIX C

Accumulator Mode Definition

This appendix reiterates the accumulator modes defined in Spectal Processor Memo 22. In addition, a tenth mode is defined.

MODE	DESCRIPTION
1	Dedispersed frequency average
2	Synchronous spectrum average
3	Synchronous dedispersed average
4	No frequency or time average, no excision
5	Partial frequency and time average
6	Untransformed time samples, no excision
7	Fixed integration-time spectrum average
8	Untransformed but squared and averaged time samples
9	Pulse on/pulse off spectrum averaging
10	Synchronous average to end of memory

APPENDIX D

Module Delay Tabulation

Each module in the data pipeline receives a clock signal from the Timing Generator. Modules up to and including the Real Correction receive what we call the "times one" clock, or C1X. Modules after the Real Correction receive what we call the "times one or two" clock, or C1.2X. This clocking arrangement allows data from the Real Correction module to be both squared and cross multiplied, without sacrificing any data when C1X is less than 20MHz.

To mark the beginning of a scan, the Timing Generator emits two copies of a "SYNC" signal. One copy goes to the A/D Interface. It follows the first data point through this interface on to the Buffer Controller, where it synchronizes a counter which cycles at the FFT cycle rate. In turn this counter emits a pulse, "AD=0", that accompanies the first point of each FFT cycle through the rest of the pipeline. The second copy of the SYNC signal goes to the accumulator, where it triggers a counter preloaded with a value, PDYLP. This counter will then count until it reaches terminal count. If the PDLYP value has been properly calculated and down-loaded, this counter should reach terminal count on the same clock pulse as when the AD=0 pulse arrives at the accumulator. This tells the Accumulator that the first point of the new scan has arrived. To calculate the correct value for PDLYP, one must sum the clock delays of each module in the pipeline.

Many modules can have two or more modes, and their delay can be a function of their mode. Moreover, the delays must be summed in units of C1.2X, since this is the Accumulator's clock. If the rate of C1.2X is double that of C1X, the delay contribution of modules running on C1X must be doubled. The table below lists the delays of each module, in all its modes, at single and, where applicable, at double clock rate.

MODULE	MODE	C1X	C1.2X
A/D Interface	1 Chan	4	
	2 Chan	4	
	4 Chan	5	
	8 Chan	6	
Input Buf Contr	Normal	3	
	Test	8	
Input Buf + Window	2/256 NOR	260	
	2/256 BYP	258	
	Others NOR	516	
	Others BYP	514	
FFT	BYP	2	
	NOR stage n(except last)	8 + (2 exp(n mod 9))	
	LAST STAGE	9	

eal Correction	NOR, 2/256	265	
	NOR, not 2/256	521	
	BYP	2	
SQ/X Mult.	NOR	5	4
	BYP	2	1
Stokes	any	5	5
Faraday	any	8	8
Accumulator input	any	1	1

Example:

Assume the following configuration:
- 2/256 mode
- IF bandwidth = 10MHz
  this implies C1X = 10MHz
  and we need 8 butterfly stages
- C1.2X = 2(C1X) = 20MHz
- Only modules bypassed are FFT stages 8, 9, 10.

Then the following delays are associated with each module:

ADINT:	4
Input Buf Contr:	3
Input Buf + Window:	260
FFT, stage 0:	9
FFT, stage 1:	10
FFT, stage 2:	12
FFT, stage 3:	16
FFT, stage 4:	24
FFT, stage 5:	40
FFT, stage 6:	72
FFT, stage 7:	9
FFT, stage 8:	2
FFT, stage 9:	2
FFT, stage 10	2
Real Correction:	265

Sum of C1X delays: 730 = 1460 C1.2X delays

Square and Cross Mult:	5
Faraday:	5
Stokes:	8
Accumulator	1

Total C1.2X delays: 1479

Then PDLYP equals the two's complement of 1479!

Note: This memo is in file accmemc1.txt.