1.0) Introduction

1.1 General Description
This specification gives the requirements for the 4096-lag correlator ASIC to be used in the ALMA correlator. This chip runs on a 125 MHz clock. The chip operating voltage will be 3.3 VDC or less (the final chip supply requirement will be determined later). There could be a lower core voltage inside the chip. The chip package will be a surface mount, industry standard package.

Each individual lag of the correlator chip consists of a 2-bit, 4-level, times 2-bit, 4-level, multiplier whose output is integrated in a 25-bit accumulator. Each accumulator has secondary storage for the 16 most significant bits. All 25-bits of the accumulator can be asynchronously cleared before an integration. A single synchronous Blanking signal input to the chip can be used to stop the accumulation in all correlator circuits.

The ALMA correlator chip has many modes of operation. A serial program word on the chip provides the configuration information for the chip. The serial program word register on the chip must be loaded by a microprocessor external to the chip before operation can begin. The program word register has a secondary storage register to prevent the chip from seeing the serial program word shift-through.

1.2 Doubling the Size from a 4096 Lag to a 8192 Lag Correlator Chip
A larger version of the chip is also being considered. The primary difference is that each 64 lag sub-block, would be replaced with a 128 lag sub-block. Almost all other aspects of the chip will remain the same. This would approximately double the number of gates in the chip.

Due to cooling considerations, it is desirable to maintain the maximum chip power at 2 watts, despite the doubling of the number of gates. Thus, for the 8192 Lag Chip, 0.18 micron technology would probably be required.

1.3 Chip Quantities
64 chips/correlator card * 4 cards/plane * 32 planes * 4 Baseband pairs = 32,768 chips.
Thus the ALMA Correlator total requirement is 39,322 chips, plus 20% spares, equals 39,322 chips.
1.4 Work Breakdown
NRAO provides this detailed chip specification. Innotech will do the detailed design. NRAO will provide a close overview of the process, to assure the desired results. Innotech will do the layout, and arrange fabrication. NRAO will evaluate the prototypes.

2.0) Block diagram
Figures 1 and 2 give a block diagram of the ALMA correlator chip. This chip is a 4-by-4 matrix of correlator blocks where each block consists of 256 multiplier/accumulator/secondary storage circuits (referred to as lags). Each of the correlator blocks can be configured as a single 256-lag correlator, as two 128-lag correlators, or as four 64-lag correlators. Each 256-lag block is thus comprised of four 64-lag sub-blocks. There are a total of 64 of the 64-lag sub-blocks on the chip.

Figure 2 shows the 4-by-4 array of correlators being driven by samples from 4 antennas on each axis of the matrix (shown in the Figure as antennas Y, Y+1, Y+2, and Y+3 on the Vertical axis of the matrix, and antennas X, X+1, X+2, and X+3 on the Horizontal axis). Each antenna supplies 2-bit samples from each of two digitizers (M0 and M1 as seen in Figures 1 and 2).

The correlator chip thus requires eight, 2-bit driving signals (for a total of sixteen bits) for the Vertical axis of the internal 4-by-4 matrix, and eight, 2-bit driving signals on the Horizontal axis.

The correlator chip has two, separate sixteen bit busses entering the bottom of the chip, called the Main Bus and the Aux Bus (See Figure 1).

Figure 1 also shows eight, two bit, bi-directional drives on the horizontal axis.

The Horizontal axis to the internal matrix, as seen in Figure 1, can be driven from one of the following four sources:

- Drive from the Main Bus input pins at the bottom of Figure 1.
- Drive from the Aux Bus inputs pins at the bottom of Figure 1.
- Drive from the Left I/O pins used as chip inputs, on the left side of Figure 1.
- Drive from the Right I/O pins used as chip inputs, on the right side of Figure 1.

The input busses are selected with tri-state drivers. The drivers are controlled by the CENTERBUS[1:0] bits of the program word (See Figure 3). When the Horizontal axis of the correlator matrix is driven from either the Main Bus or the Aux Bus, these signals can also drive both the right and left I/O pins out of the chip using the tri-state drivers seen in Figure 1. These modes are further explained later in this document.

Readout of the 16-bit accumulator secondary storage registers is over a 16-bit tri-state bus. Any of the 64, 64-lag sub-blocks may be selected for results readout with the application of a 6-bit code plus two permissive, output enable signals X OE and Y OE. The 64-lag results shift out in lag order, lag zero first, lag one next, lag two next, etc. Any number from 1 to all 64 lags may be read from a sub-block at a time. The exact number of results read out is determined by the number of transitions of the READCLK input. See the explanation of Figure 7, in section 2.5.
**2.1) Data input-output**

Figure 3 shows data (samples) into and out of the correlator chip. The chip is to be connected in a matrix configuration on the correlator card in the ALMA application. The chip supports the Horizontal and Vertical matrix structure of the card.

Figure 3 is described further in the Card Level discussion, in Section 5.

**2.2) The Basic Correlator**

Figure 6 shows one of the 4096 (64x64) correlator lag circuits on the correlator chip. A correlator lag includes a 2-bit, 4-level, by 2-bit, 4-level, multiplier, using the Biased Multiplication Table shown. The multiplier output is summed in a 25-bit accumulator, at the 125 MHz chip clock rate (that is, one multiplication and summation into the accumulator occurs on each positive clock transition). The 25-bit accumulator consists of a 5-bit synchronous stage, a 4-bit ripple-through pre-scalar, and a 16-bit ripple through counter.

The 5 bit synchronous stage consists of the Multiplier, 5 Bit Adder, and 5 Bit Accumulator. Since the maximum output from the Multiplier is 9, that makes the maximum toggle rate of the MSB of the 5 bit stage to be approximately every 4\(^\text{th}\) clock cycle. Shown below the 5 bits are incrementing by 9.

```
00000, 01001, 10010, 11011, 00100, 01101, 10110, 11111, 01000
```

The MSB transitioning from 1 to 0 provides the toggle edge for the subsequent ripple counter. Each T flip-flop asynchronously toggles on the 1 to 0 transition of the previous stage. Since there is an interval between the blanking, and the dump to storage signal, the ripples will have time to settle (See the timing diagram on Figure 6).

The Accumulator can function as a 25 bit or 21 Bit Accumulator, as determined by the FULLACC line at each block. FULLACC is logically defined on Figure 4, since it effects the whole 256 lag Block. The FULLACC line is a function of only the RxFULLACC control bit in non-diagonal blocks on the chip. In RxFULLACC, x equals 0, 1, 2, or 3 to indicate the row number. Figure 2 illustrates the Row definition. FULLACC is a function of a combination of RxFULLACC control bit, and the DIAGONALyAUTO control bit, in the chip diagonal blocks (y equals block 0,5,10, or 15). Note in the present ALMA Correlator Board plan, RxFULLACC will always equal 1.

The 16BITCNTR program word bit is provided to allow for faster functional test. Otherwise it would take too large a number of test vectors to increment the 16 bit counter through all states.

16BITCNTR=0 bypasses eight bits, to allow rapid sequencing of the upper eight bits. The lower eight bits are frozen, to limit power dissipation (See Figure 6). Without limiting the power, this would provide a mode of operation with higher power dissipation than any normally occurring mode. This might provide a destructive worst case.

The DIAGONALyAUTO program word bit indicates the 256 lag block “y” in the chip is on the diagonal of the matrix of four cards, and in Auto Correlation Mode. The y corresponds to 256 lag block 0, 5, 10, or 15. Auto Correlation mode implies 21 bit accumulation, and 1 ms. dump times. Since only the diagonal blocks are of interest in Auto Correlation mode, more rapid data retrieval is possible. The integration time for the diagonal blocks could be 1 ms, while the other blocks could be 16 ms. In Auto Correlation mode, only the self products are of interest. Note if we are in Cross Correlation mode, the self products would have the 16 ms dump time, just like the cross products.
If FULLACC=0, the 4-bit pre-scalar is bypassed and the accumulator acts as a 21-bit accumulator. In this mode, the pre-scalar is held static to reduce the power consumption of the chip. If FULLACC=1, the pre-scalar is used in a normal way and the correlator has the full 25-bit accumulator for integration.

Figure 6 shows the 16-bit secondary storage register for the 16 most significant bits of the accumulator. The contents of the 16 bit secondary storage registers are shifted out of the chip on a 16 bit bus. During read-out, the 16-bit secondary storage registers shift broadside off the chip. All 64 lags from a lag sub-block connect as a 16-bit wide, 64-bit deep, broadside shift register.

In typical operation, the BLANKING signal to the chip will be high (inhibiting correlation/integration) for approximately 256 clock cycles (2.048 microseconds) every 1.00 milliseconds (See Figure 6 and Figure 8). This is accomplished via a gated clock (See Figure 5). Alternatively, there could be clock enables on the affected flip-flops. Occasional Blanking cycles will be marked, via an externally generated DUMP ENABLE signal. DUMP ENABLE is an input pin to the chip (Logic shown on Figure 4). Two signals, the DUMP TO STORAGE signal to the secondary storage register, and the ACCUMULATOR RESET, will be generated from the rising edge of the marked Blanking signal (See Figure 8 and Sections 2.4 and 2.6.3). These signals implement dumping to secondary storage of all 16 upper accumulator bits.

2.3) 64-Lag Correlator Sub-Block

Figure 5 shows how 64 of the basic correlator circuits are connected together to build a 64-lag correlator sub-block. All 64, 2-bit P (prompt) inputs to the 64 correlators are connected in parallel while the 64 2-bit D (delayed) inputs are connected to the correlators from a 64 stage (1 bit or 2 bit) lag generator. The 2 bit output of the sub-block lag generator connects to an adjacent 64-lag correlator sub-block input multiplexer (See Figure 2).

The lag generating shift register has two modes of operation (See Figure 5). In the normal mode, there is one stage of shift register between each correlator of the 64-lag correlator sub-block (for each bit of the 2-bit D signals). When the input signal is twice Nyquist sampled, the lag generator supplies two stages of shift register per signal bit between each correlator. The R0OVERSAMP through R3OVERSAMP bits from the serial program register determines the one-or-two bit operation of lag generators for one row of correlators (four bits for the four rows of the chip). For power conservation, the alternate flip-flop should be static when not in use. This could be accomplished in the flip-flop multiplexer component.

There is a digital delay line at the input end of the Delayed (D-DATA) Input. The delay line can be set for 1, 2, or 3 clock cycle(s) delay, as programmed by the M0 and M1 bits. The table at the top of Figure 5, reproduced here, defines these bits.

<table>
<thead>
<tr>
<th>LEAD-BLK</th>
<th>CONCAT</th>
<th>OVERSAMP</th>
<th>M0</th>
<th>M1</th>
<th># DELAYS</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>X</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Pipeline register only.</td>
</tr>
<tr>
<td>X</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>CONCAT, Non-Oversampled</td>
</tr>
<tr>
<td>X</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0 *</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>CONCAT, Oversampled</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0 *</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>LEAD, Non-Oversampled</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>LEAD, Oversampled</td>
</tr>
</tbody>
</table>
* Note that an M0 M1 of 0 1 could be used instead of the shown 1 0. This would still give the required 2 delays.

The LEAD-BLK signals adds an extra one (or two bits delay if Oversampled), if the 256 lag block is a Lead. This is discussed further in the next section. A pipeline register is required, since the parallel, PROMPT line has one. This makes for a 2 or 3 bit delay for a LEAD section.

The purpose of the LEAD-BLK signal is as follows. For a cross product of two antennas, the leads and lags are concatenated, and an FFT performed. Without the LEAD-BLK signal present, the leads and lags share a common signal. The common signal is the product of the two antennas with no delays. Removing the common signal leaves an odd number of components, which are difficult to do an FFT on. The extra delay inserted by LEAD-BLK solves the problem.

The CONCAT signal indicates the D-DATA INPUT is wrapped around from a previous 64 lag block. An extra 1 bit (or 2 bit if Oversampled) delay is required. There is no pipeline register, since the prompt data would be in parallel with that from the previous, 64 lag block. If CONCAT is 1, LEAD-BLK is don’t care (X), since the LEAD bits are only inserted at the beginning of the Delayed chain, not in the middle.

If LEAD-BLK=0 and CONCAT=0, then only the pipeline register is required. OVERSAMP is don’t care, since the pipeline register does not count as a delay element.

The BLANKING is not applied to the D-DATA chain of shift registers. This allows new data to shift into the registers for the next 1 ms. period. The BLANKING pulse width correlates with how many registers require data shifted into them. For example the width would be at least 512 clocks for four 64 lag blocks, in series, in oversample mode. The BLANKING pulse width can be greater than required. That would entail the unnecessary loss of a small fraction of the data.

**2.4) 256-Lag Correlator Block**

Figure 4 shows how four 64-lag sub-blocks interconnect to make a 256-lag correlator block. Each 256-lag correlator block is driven by four 2-bit digitizer signals. The exact mode of operation is controlled by 8 multiplexer stages that are programmed by 17 bits from the program word register. Each row of 256-lag correlator blocks in the 4-by-4 chip matrix (See Figure 2) receives the same 16 program word bits, with the exception that each 5-1 MUX receives an independent bit. Figure 4 shows the detailed implementation of the 5-1 MUX. Figure 2 shows the multiplexers for the whole chip. A total of 4*(16 + 4) = 80 program word bits are used to control all input multiplexers on the chip.

The X-M0 and X-M1 inputs connect to every one of the input multiplexers, allowing either X input to drive either the prompt or delayed input of any 64 lag correlator block. The Y-M0 and Y-M1 inputs do not connect to every one of the input multiplexers, due to the need to connect the delayed path from sub-block 0 to sub-block 1, etc. Thus there is some restriction on which Y inputs can connect to the delayed inputs of the 64 lag correlator blocks. Within these limitations, the source of inputs to the prompt and delayed inputs of the 64 lag correlator block, could be swapped for test purposes.

The LEAD-BLK signal into the 64 lag blocks provides an extra stage of delay, for the LEAD portion of the array. The equations for all four, 64 lag blocks share the element LEAD-BLK. For each 256 lag block, LEAD-BLK is determined as shown in the tables of Figure 4. The choice of the four tables is determined by the program word bits LEAD-CTRL[1:0].
The CONCAT signal into each 64 lag block specifies if there is a wrap around input from a previous stage. For 64-LAG CORRELATOR 0, CONCAT = WRAP-BLK0 indicating the delay data comes from the previous 256 lag block (See Figure 2). For 64-LAG CORRELATORS 1, 2, and 3, CONCAT indicates the mux setting for the delay data coming from the previous 64-LAG CORRELATOR (See Figure 4). CONCAT is used in the 64 LAG CORRELATOR to adjust the delay, as previously explained.

In Figure 8, the signals SEQ DUMP TO STORAGE and SEQ ACCUMULATOR RESET are generated. These provide for transferring the accumulator contents to secondary storage, then resetting the accumulator. In Figure 4, these signals are used in each 256 lag block.

If FULLACC = 0, the 21 bit accumulator is used. This forces a DUMP TO STORAGE and ACCUMULATOR RESET every blanking cycle, since the accumulator would overflow otherwise. The multiplication table is biased, since they are all positive numbers (See Figure 6). Thus even totally uncorrelated inputs could cause an overflow, if the integration period is too long.

If FULLACC = 1, the 25 bit accumulator is used. Then the DUMP ENABLE signal controls for which Blanking cycles, the DUMP TO STORAGE and ACCUMULATOR RESET are asserted.

A 16-bit wide, 4-to-1 output multiplexer is used to allow selection of any of the four 64-lag correlator sub-blocks for read-out. The two bits to control these multiplexers come from chip input pins.

2.5) Results Read-Out

Figure 7 shows the final 16-bit wide, 16-to-1 multiplexer stages required for result read-out. Sixteen bit broadside outputs, from each of the sixteen 256 lag correlator blocks, come together in these multiplexers. One 16-bit result stream may be selected for shift out. The shift-out control logic block is in Figure 7. The Control Logic decodes the x- and y-axis select bits, 6-bit correlator select lines (four block select bits, and two correlator sub-block select bits), and the READCLK line.

The X Axis and Y Axis input pins output enable a single correlator chip onto a correlator card sixteen bit, results output bus. Multiple correlator chips can have their output busses tri-stated together.

The four, correlator block select input pins select one of the sixteen baseline correlator blocks for readout. A baseline correlator block is a 256 lag block, obtained when an antenna x is multiplied by an antenna y.

The two bit, sub-block input pins select one of the four 64 lag sub-blocks, in a 256 lag block.

The input pin READCLK is used for shifting out the data. READCLKOUT provides a version of READCLK for subsequent chips on the card. The rising edge produces the next result. Selecting a 64 lag sub-block, and output enabling the chip, produces lag 0 from the selected block at the chip output pins. SEL[0..5] selects which of the 64 lag sub-blocks to route to the 16 bit output bus. SEL[0..5] also determines which of the 64 Results Shift Out Clocks will be activated. This serially shifts out the 64 lags in the block, over the 16 bit parallel bus.

The maximum data rate allowed for READCLK is 125 MHz. This means the data output must be able to be captured in the next chip, despite going through the several output multiplexers. It would be best if there were no flip/flop stages between the secondary storage registers and the chip output.
pins. Otherwise, we will have to issue one or more clocks before the first result is output from the chip. The pipeline registers would be acceptable if needed, but it would be better without them.

The four diagonal blocks (0,5,10, and 15) have their Result Shift Out Clocks generated as shown in the logic of Figure 7. The DIAGONALyAUTO program word bit (where y = 0,5,10, or 15), tells if the diagonal Block is in a subarray doing Auto correlation mode(not Cross Correlation mode).

Readout of diagonal blocks will be gated with the AUTO input signal. AUTO = 1 indicates Auto Correlation Mode. AUTO = 0 indicates Cross Correlation mode. If the diagonal block has its DIAGONALyAUTO bit set, it will only respond to readout if the AUTO signal is set. If the diagonal block is in Cross Correlation Mode, it will only respond to readout if the AUTO signal is 0, indicating Cross Correlation Mode.

The Long Term Accumulator (LTA) readout circuitry reads the Self product blocks in both Self and Cross mode, as specified by the AUTO input pin. The DIAGONALyAUTO bit determines which readout is effective, with the other readout ignored. This arrangement allows the LTA to blindly read all the diagonal blocks in Time Slot 0 (AUTO=1) and Time Slot 1 (AUTO=0).

2.6) Control Logic

Figure 8 shows control logic required for the correlator chip.

2.6.1) Program Word

A serial in program register can be programmed from an external source to set the mode of operation for the chip. The interface to this program register is via the PGM DATA and PGM CLK input pins. Bit 0 is shifted in first. Data clocks on the clock rising edge. The serial clock is buffered and output from the PGM CLK OUT pin. Data from the last stage of the shift register is output from the PGM DATA OUT pin, so correlator chips on the correlator card may be connected in series. Care must be taken in the chip design, so the setup and hold times into subsequent chips are satisfied.

The program register has a secondary register so all program bits to the chip can be made to change at once. The secondary storage register is loaded via the PGM STB input pin. The elements in this register can be a latch. This strobe input is buffered and drives the PGM STB OUT output pin.

See 08070100n001.xls Sheet 2 for a detailed listing of the program word bit assignments.

2.6.2) Clock Logic

The 125 MHz clock enters the chip on the C125 input pin. A buffered version of C125 is output on the C125OUT output pin, for use on the correlator card.

2.6.3) Storing the Results for Readout

The BLANKING and the DUMP ENABLE signals into the chip control the correlator accumulator secondary storage registers. A logic high on BLANKING will synchronously stop all correlators on the chip. BLANKING is implemented via gating the clock (See Figure 5). Alternately, BLANKING can be implement by clock enables in the LAG accumulator, D flip-flops (See Figure 6).
Dump of the correlator accumulator results, into the correlator secondary storage, will occur after
the assertion of BLANKING, if DUMP ENABLE is high. SEQ DUMP TO STORAGE (See Figure
12 for timing) and SEQ ACCUMULATOR RESET are implemented via delays in the Control
Logic of Figure 8. These signals are gated on Figure 4, with FULLACC and DUMP ENABLE.
ACCUMULATOR RESET can be seen on Figure 6, to reset the 25 bit accumulator.

3.0) Input-Output Pins
See 08070100n001.xls Sheet 1 for a detailed listing of the input-output pins.

4.0) Serial Program Word
See 08070100n001.xls Sheet 2 for a detailed listing of the program word elements.

5.0) Card Level Considerations

5.1) Reason for Card Considerations
Understanding how the chip is used in the Correlator Card will allow the chip design to take into
account card level effects. It could also allow the recognition of errors, or improvements in the
specification of the chip.

5.2) Signal Distribution to Set of Four Correlator Cards
See Figure 9. Each Correlator Card services the correlation of 32 by 32 antennas. For a plane of
64 antennas, four cards are needed as shown.

The dashed diagonal line on Figure 9, represents the 64 self products, or autocorrelations. The Self
Cards, Cards 0 and 3, are along the diagonal. Card 0 multiplies antennas 0-31 times antennas 0-31.
Card 3 multiplies antennas 32-63 times 32-63. Only chips along the diagonal of the Self Cards,
will possibly have the DIAGONALyAUTO program word bit asserted, if the y block is in AUTO
mode.

Cards 1 and 2 are called Cross Cards. Card 1 multiplies 32-63 times 0-31, for the LEADS. Card 2
multiplies 0-31 times 32-63 for the LAGS. The Self Cards contain LAGS above the diagonal and
LEADS below the diagonal.

The upper right of Figure 9 shows the signal distribution to the cards. Note the Cross Cards (Card
1 and Card 2) have twice as many input signals as the Self Cards. The Cross Cards and Self Cards
will be physically identical and interchangeable. Card functionality will be determined by its
position in the rack. The Self Cards only have signals coming in on the Main Bus, with the Aux
Bus unused. The Cross cards use both busses. Different chip modes are necessary to distribute the
signals from the Main Bus in the Self Cards, and to use the Aux Bus in the Cross Cards.

5.3) Chip Modes
Figure 3 portrays the chip data bus structure. Each bus represents sixteen identical data paths.
Sixteen is obtained by multiplying four antennas, times two digitizers, times two bits. The chip
program word bits determine the flow of data between the six busses. Figure 3 identifies the six,
sixteen bit busses as DBL[15:0], DBR[15:0], DTL[15:0], DTR[15:0], DL[15:0], and DR[15:0].
These data paths connect into (or out of) the correlator chip, and into the horizontal and vertical axes of the internal array of correlators. All sixteen of the data paths are programmed identically in all cases.

Horizontal data flow is controlled by the LTOR (Left to Right) bit of the serial program word. LTOR controls the right output bus via OE-RO. LTOR is output via the LTOR-OUT pin. This is input into the next chip to the right via the LTOR-IN pin. LTOR-IN controls the left output bus via OE-LO. This arrangement prevents bus contention, regardless of the programming of the LTOR bits in the chips.

The CENTER BUS in the chip can be driven from one of four sources. In the Mode Logic, the two bit CENTERBUS code from the program word is decoded to specify which of the four sources drives the CENTER BUS. This arrangement precludes bus contention. The CENTERBUS decoder outputs should be break before make to prevent a momentary bus contention on transitions. That is there will be a period when all four outputs are zero, whenever the inputs change states.

The remaining Program Word bit, which determines the chip mode, is AUXENBL. This enables data passage out the top of the Aux Bus. Zeroing this data when not used saves power. On the Self Card, AUXENBL would always be zero, since the Aux Bus is not used.

Thus the input signal LTOR-IN, and the program word bits LTOR, CENTERBUS[1:0], and AUXENBL determine the data flow in the chip. The state of these bits for useful modes is tabulated in the lower left of Figure 3.

Above the table are data flow diagrams for a partial, horizontal row of chips in the Self Card and Cross Card.

5.4) **Self Card Description**

5.4.1) Data Distribution

See Figure 10. The signals from 32 antennas enter the card. Sets of four antennas enter into each of eight correlator chips. The signals are passed chip to chip vertically up the card. Along the diagonal, the signals spread out horizontally as well. To the left of the diagonal, chips pass data horizontally to the left. To the right of the diagonal, chips pass data horizontally to the right.

The Vertical axis signals into the internal correlator matrix must be applied through adjustable delay lines (See Figure 1). A delay of from zero to 31 bits is available to compensate for the position of the correlator chip in the correlator card chip matrix. Delay is set by DLY[4:0] from the Program Word. The current card design would use delays of 0,2,4,6,8,10,12, and 14 bits. The higher delay values would allow for a card containing 256 Correlator Chips.

5.4.2) Self Card Chip Modes

The five different chip modes required are shown on the right side of Figure 10. The chip modes are also shown on the left in Figure 3. They are labeled T0 through T7. There are more possible modes, but they are not useful.

Program word bit LTOR determines the horizontal data flow. LTOR leaves the chip as LTOR-OUT, to become LTOR-IN of the next chip to the right. This arrangement guarantees there will not be bus contention between chip outputs. Tying chip pin LTOR-IN high on the far left turns off the
left output bus, since there are no more chips to drive. This saves power. Setting program word bit LTOR = 0 on the far right chip, similarly turns off the output to the right.

Since the inputs may be floating, the chip should provide internal pull-ups.

CENTERBUS (a two bit field from the program word) selects the tri-state drivers to drive the Center Bus (See Figure 3). Also this determines the Horizontal input into the correlator array in the chip.

### 5.5) Cross Card Description

#### 5.5.1) Data Distribution

See Figure 11. One set of 32 antennas enters the card into the chip Main Busses. Each data bus shown contains the signals from four antennas. The signals are passed chip to chip vertically up the card. These signals provide the drive to the Vertical (Y) axis of the internal correlator matrix.

A second set of 32 antennas enters the chip Aux Busses. These pins were unused on the Self Card. The signals propagate vertically using the Aux Bus, until they reach the diagonal. The signals are not processed in chips that they travel vertically through. Upon reaching the diagonal, the signals spread out horizontally. To the left of the diagonal, chips pass data horizontally to the left. To the right of the diagonal, chips pass data horizontally to the right. These signals provide the Horizontal (X) axis drive to the internal correlator matrix.

The Vertical (Y) axis signals into the actual correlator matrix must be applied through adjustable delay lines. A delay of from zero to 31 bits is provided to compensate for the position of the correlator chip in the correlator card chip matrix.

#### 5.5.2) Cross Card Chip Modes

The five different chip modes required are shown on the right side of Figure 11. The chip modes are also shown on the left in Figure 3.

As in the Self Card Modes, LTOR determines the horizontal data flow, and CENTERBUS determines the drive source for the tri-state drivers to drive the Center Bus (See Figure 3). The difference from the Self Card modes is that now the Aux Bus is used for the Horizontal input. Also, Program Word bit AUXENBL is set to propagate the data upward. After the diagonal, AUXENBL is not set, to save power. Note that even though the Aux Bus data is not needed past the diagonal, it will still be wired on the card, for the sake of versatility. The chip should have internal pull-up resistors, since the Aux Bus card inputs will not be driven on the Self Cards.

#### 5.6) Test Modes

The left most chips will output their left bus to test points. Grounding a test point that has the LTOR-IN held high will enable the outputs.

The right most chips will output their right bus to test points. Setting LTOR =1 in these chips program word will enable the outputs.

The top chips will have their Main Bus and Aux Bus outputs to test points. The Main Bus outputs will be enabled all the time. The Aux Bus outputs can be enabled by setting AUXENBL in the
Program Word. Note that AUXENBL would have to be set for the chips between the diagonal and the top to give full data propagation.