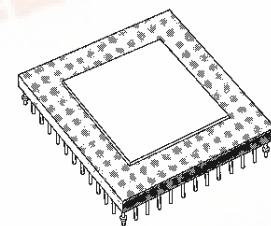


**IMSA110**

## IMAGE AND SIGNAL PROCESSING SUB-SYSTEM

- 1-D/2-D SOFTWARE CONFIGURABLE CONVOLVER/FILTER
- ON-CHIP PROGRAMMABLE LINE DELAYS (0 — 1120 STAGES)
- 8-BIT DATA AND 8.5-BIT COEFFICIENT SLICE
- 21 MULTIPLY-AND-ACCUMULATE STAGES
- 1-D (21) OR 2-D (3 x 7) CONVOLUTION WINDOW
- ON-CHIP POST PROCESSOR FOR DATA TRANSFORMATION
- FULLY CASCADABLE IN WINDOW SIZE AND ACCURACY
- 20 MHZ DATA THROUGHPUT (420 MOPS)
- SIGNED/UNSIGNED DATA AND COEFFICIENTS
- MICROPROCESSOR INTERFACE
- HIGH SPEED CMOS IMPLEMENTATION
- TTL COMPATIBLE
- SINGLE +5V  $\pm$  10% SUPPLY
- POWER DISSIPATION < 2.0 WATTS
- 100 PIN CERAMIC PGA



**PGA100**  
(Ceramic Grid Array Package)

### APPLICATIONS

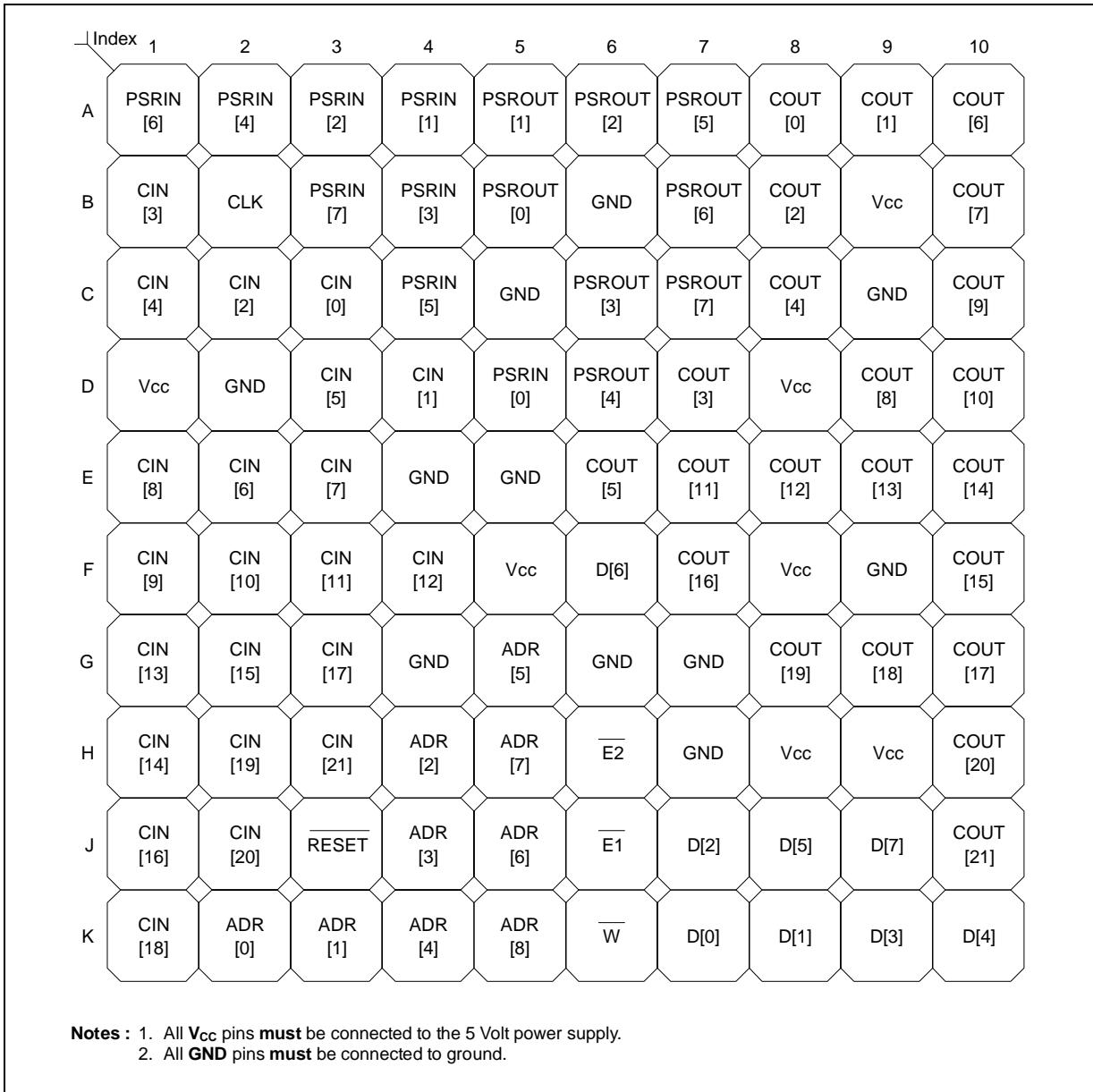
- 1-D and 2-D digital convolution and correlation
- Real time image processing and enhancement
- Edge and feature detection
- Data transformation and histogram equalisation
- Computer vision and robotics
- Template matching
- Pulse compression
- 1-D or 2-D interpolation

### ORDERING INFORMATION

Part Number	Package	Clock Speed	Military/commercial
IMSA110-G20S	PGA100	20MHz	commercial

A11001.TBL

## PIN CONNECTIONS



A110-01.EPS

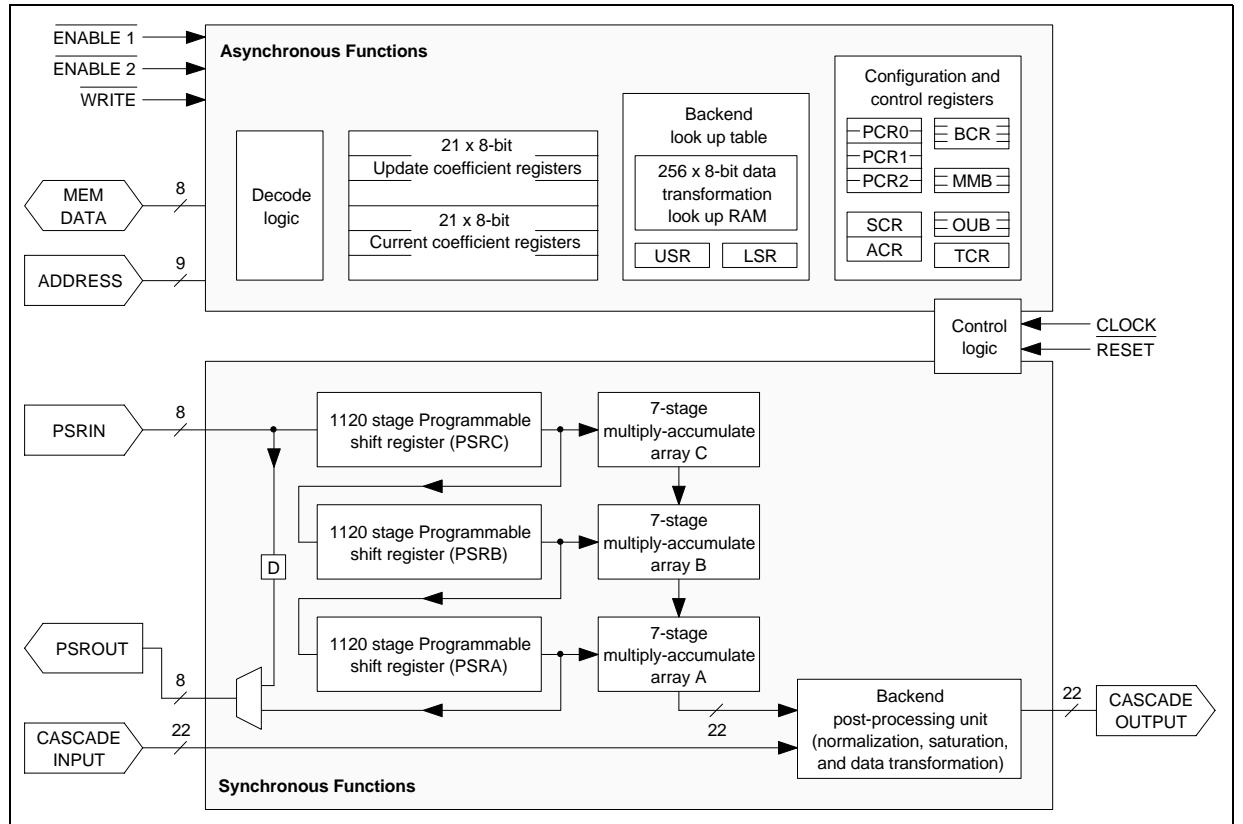
## 1. INTRODUCTION

The IMSA110 is a single-chip reconfigurable and cascadable subsystem suitable for many high speed image and signal processing applications. Apart from its powerful multiply-accumulate capability (420 MOPs), the strength of the IMSA110 lies in its extensive programmable support for data conditioning and transformation.

## 2. DESCRIPTION

The IMSA110 consists of a configurable array of multiply-accumulators, three programmable length 1120 stage shift registers, a versatile post-processing unit and a microprocessor interface for configuration and control purposes. The comprehensive on-chip facilities make a single device capable of dealing with many image processing operations.

Figure 1 : IMSA110 Users Model



The IMSA110 has five interfaces through which data can be transferred, Figure 1. The microprocessor interface allows access to the coefficient registers, the configuration and status registers, and the data transformation tables. The remaining four interfaces allow high speed data input and output to the IMSA110 and the cascading of several devices. A typical IMSA110 system is shown in Figure 3. If  $N$  devices are used in the cascade, they can be configured, entirely under software control, as a  $21N$  stage 1-D transversal filter or as a  $7X$  by  $3Y$  2-D window, where  $X$  and  $Y$  are any integers satisfying  $N \leq XY$ . For example 4 cascaded devices can be software configured as: an 84-stage 1-D filter, a 7 by 12 2-D window, a 28 by 3 2-D window, or a 14 by 6 2-D window.

The final output of the chip is 22 bits wide in two's complement format.

Figure 2 shows the distribution of the delays inside the part.

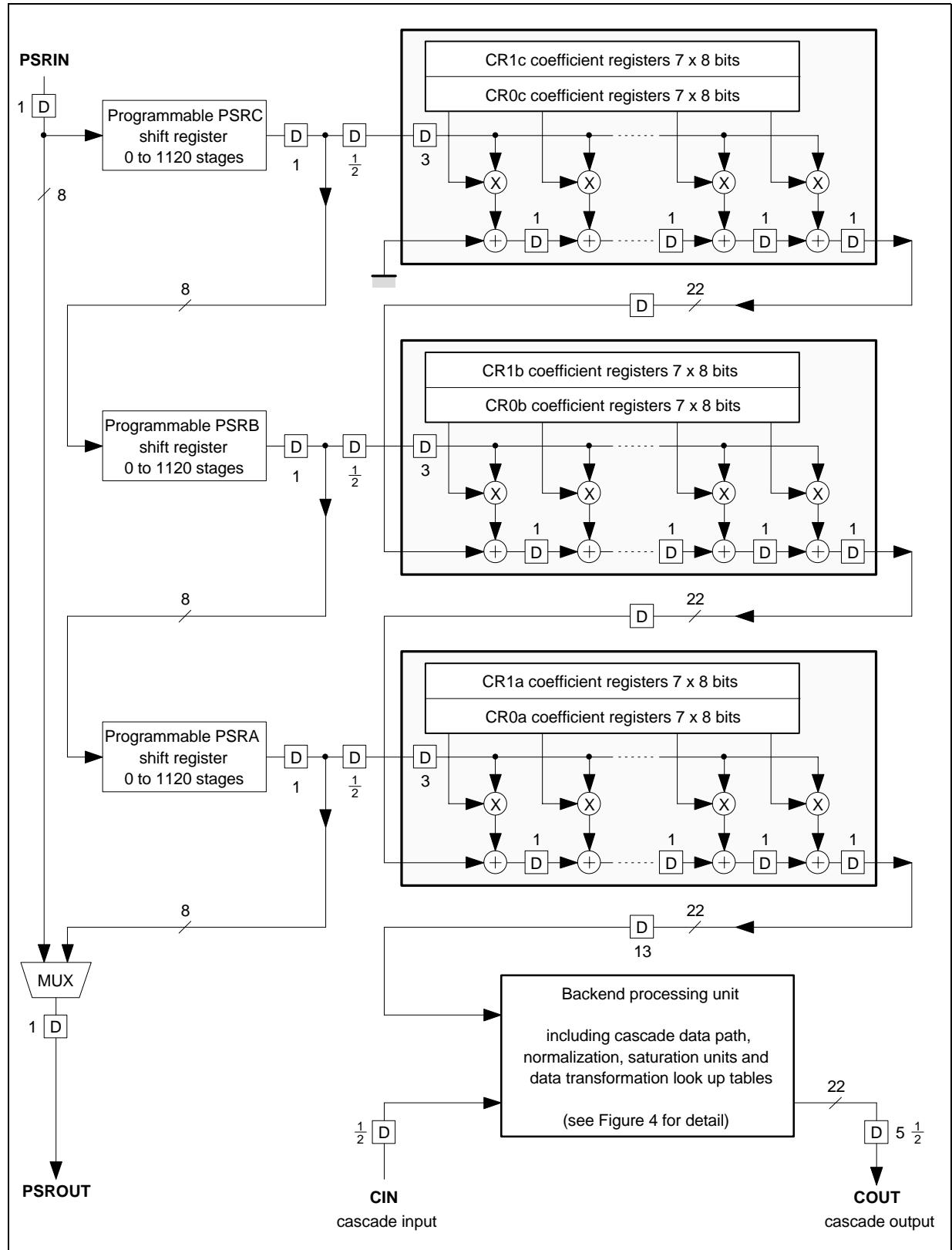
The latency between PSRin and COUT is dependent upon the length of PSRc. For example, with PSRc set to 0, and all coefficients set to zero except CR0c[6] (so the data passes through all MAC stages), the COUT bus will correspond to the PSRin bus delayed by 47 clock cycles.

The latency between PSRin and PSROUT is 5 cycles PLUS the lengths of PSRc, PSRb and PSRa. If the shift registers are bypassed by setting SCR[1] to 1 then PSROUT will be PSRin delayed by 2 clock cycles.

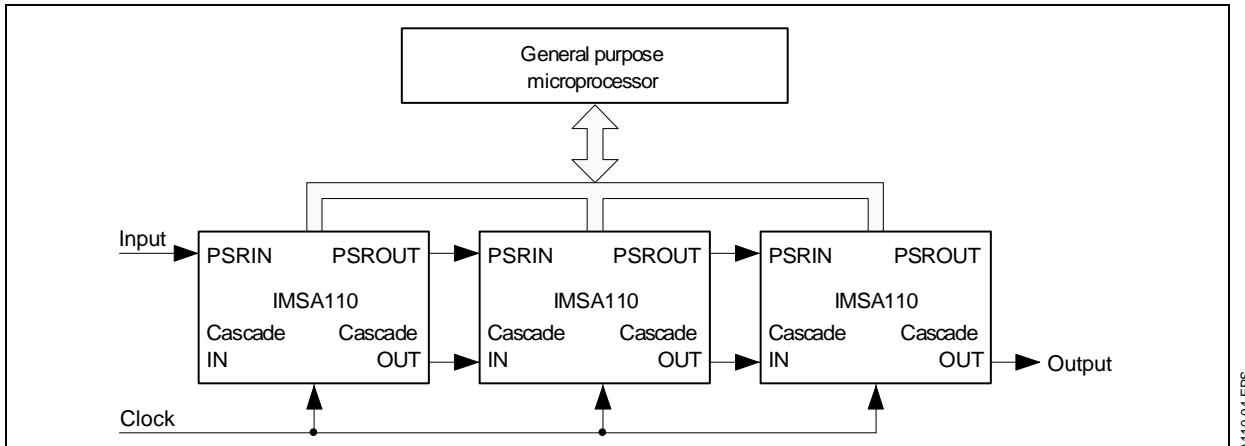
The Latency between the cascade input (CIN) and cascade output (COUT) is 6 cycles. This is shown lumped at the cascade input and cascade output pads in Figure 2. Figure 4 gives details of the data pipelining through the backend datapath.

## IMSA110

Figure 2 : Synchronous Functions of the IMSA110



A110-03.EPS

**Figure 3 : A Typical IMSA110 Based System**

### 3. PROGRAMMABLE SHIFT REGISTERS

The three shift registers are 8 bits wide and are each programmable from 0 up to 1120 clock cycles in length. The lengths are programmed into control registers via the microprocessor interface.

Data is clocked into the device via the PSRin bus (Programmable Shift Register in) at a maximum rate of 20MHz. On-chip, the input data is then fed through a pipeline of the three shift registers. The output of the first shift register passes to the first 7-stage mac array and also to the input of the second shift register. Having passed through all three shift registers the data is output on the PSRout bus and can be used for cascading. Alternatively, as shown in Figure 2 the shift registers can be bypassed and the input data transferred to the PSRout bus after two delay stages. This mode can be controlled via the on-chip control registers and significantly simplifies software configuration of a cascade arrangement.

### 4. MAC ARRAY

As shown in Figure 2, the processing core of the device consists of a configurable array of multiply-accumulators (macs). The mac array consists of three 7-stage transversal filters which can be configured either as a 21-stage linear pipeline or as a  $3 \times 7$  two-dimensional window. The input data is 8 bits wide and is fed to the mac array via three programmable shift registers.

The output of each shift register is supplied as input to one of the three 7-stage transversal filters. For each of the three transversal filters the associated input data is fed simultaneously to all 7 mac stages. At each stage the input sample is multiplied by a coefficient stored in memory, and added to the output of the previous stage delayed by one clock cycle. The output of each 7-stage mac is fed, via a delay stage, to the first stage in the next transversal

filter.

The coefficient word width in the mac array is 8 bits wide. Two banks of coefficients are provided. At any instant one set of coefficients is in use within the mac array. The set in use is defined by the state of the 'Current Bank' bit, ACR[0]. The other set can be altered via the microprocessor interface. Once a new set of coefficients has been loaded, the activities of the two coefficient banks can be interchanged without interrupting the flow of data. Alternatively, by setting the 'continuous bank swap' bit SCR[0], the two coefficient banks are swapped automatically after each data input. In this case the 'Current Bank' bit only determines which bank is used first. Both data input and coefficients can be programmed independently to support two's complement or positive unsigned formats allowing multiple devices to be used as a 'slice' in higher accuracy systems.

Within the mac array no truncation or rounding is performed on the partial products. The mac array output is fed to the backend post-processing unit which is responsible for data transformation / normalisation and cascading function.

### 5. BACKEND POST-PROCESSOR — hardware description

The Backend Post-Processor consists of four major blocks : The input block (shifter, cascade adder and rectifier unit), a statistics monitor, the data conditioning unit which itself consists of the data transformation unit and the data normaliser, and the output block (output adder and multiplexers).

A detailed diagram of the Backend Post-Processor is given in Figure 4.

All operations performed in the backend are on two's complement signed numbers unless otherwise stated.

### 5.1 Shifter, Cascade Adder and Rectifier

Data from the mac array enters the datapath via a programmable shifter. The shifter is capable of arithmetic right shifts (divides) of up to 8 bits with rounding, and left shifts of up to 8 bits. The size of this shift is controlled by the status bits BCR0[5-1]. The output of the shifter passes into the cascade adder where it is added, along with any rounding generated by the shifter, to either the cascade input bus (BCR0[0] = 0), or a zero value (BCR[0] = 1).

If the result of this 22-bit signed addition is greater than  $2^{21} - 1$ , ( $2097151_{10}$ ) then the adder will generate a positive overflow. Likewise, if it is less than  $-2^{21}$ , ( $-2097152_{10}$ ) a negative overflow will be generated. In other words, a positive overflow is generated if the result of adding two positive numbers (both MSBs = 0) is negative (resulting MSB = 1). Conversely, a negative overflow is generated if the result of adding two negative numbers (both MSBs = 1) is positive (MSB = 0). Adding two numbers of different signs cannot cause the adder to overflow.

The output of the cascade adder can optionally be full-wave or half wave rectified under the control of BCR0[7,6]. The output of the rectifier passes onto the X bus. Overflows on the X bus are signalled to both the statistics monitor and the data conditioner.

### 5.2 Statistics Monitor

The statistics monitor allows the user to set up watch dogs on the dynamics of the data on the X bus. It cannot affect the data on the X bus. The statistics gathered provide information on the system behaviour which can be used to ensure correct data scaling and normalisation. The information is also useful in the control of the overall system's analogue frontend.

#### Hardware/Functions

The statistics monitor consists of a 24 bit Min/Max register (MMR), a 24 bit Min/Max Buffer (MMB), a 22 bit Over/UnderShoot Counter (OUC), a 22 bit Over/UnderShoot Buffer (OUB) and a 22 bit twos complement comparator.

It can perform one of four functions :

- **MAX REGISTER** : Capture the maximum value of data and store it in the MMR.
- **MIN REGISTER** : Capture the minimum value of data and store it in the MMR.

- **OVERTSHOOT COUNTER** : Increment the OUC each time the data value exceeds the preset value in the MMR.

- **UNDERSHOOT COUNTER** : Increment the OUC each time the data value is less than the preset value in the MMR.

The mode of operation is determined by the Max/Min switch BCR1[0], and the Static Threshold switch BCR1[1].

#### Operation

Each sample on the X bus is compared against the threshold stored in the MMR.

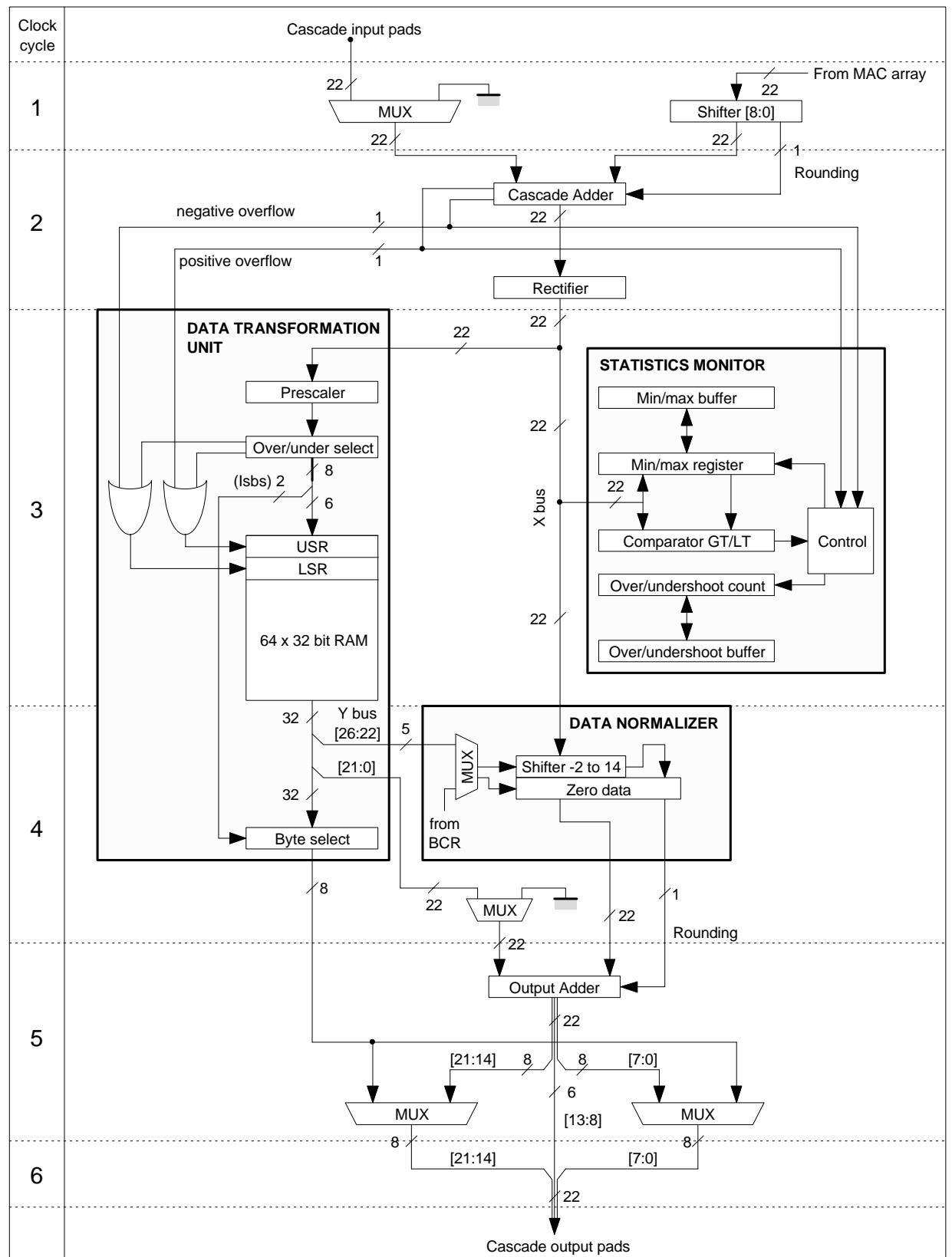
If the unit is configured as an **overshoot counter** and the data on the X bus exceeds the threshold in the MMR, then the counter (OUC) is incremented. If the data is less than or equal to the threshold, then no action will occur. The OUC is unsigned and will not wrap around. Thus it behaves as a saturating counter with a maximum value of  $2^{22} - 1$ , ( $3FFFF_{16}$ ,  $4194303_{10}$ ). If there is a positive overflow on the X bus, then the counter will increment since the correct X bus value must exceed the threshold. Similarly a negative overflow on the X bus will not increment the counter since the correct X bus value cannot exceed the preset threshold.

If the unit is configured as an **undershoot counter** then the counter will be incremented whenever the sample is less than the preset threshold. In this case a negative overflow will cause the counter to increment.

If the unit is configured as a **max register** and the X bus exceeds the current threshold in the MMR, then the value on the Xbus is loaded into the MMR and becomes the new threshold and the counter is incremented. If the threshold is not exceeded then no action occurs. Thus the value in the MMR is the maximum value that has appeared on the X bus, and the value in the OUC has been incremented by the number of times that the threshold has been updated.

If the unit is configured as a **min register** then the threshold is updated and the counter incremented whenever the X bus is less than the current threshold.

When operating as a min/max register, overflows on the X bus can never cause the threshold to be updated as this would load an erroneous value into the MMR.

**Figure 4 : Detailed Block Diagram of the Backend Post-processing Unit**

A110-05.EPS

### Overflows

Bit 22 of the MMR records the history of positive overflows on the X bus. Similarly bit 23 records the history of negative overflows. These bits in the MMR are set to zero by writing to the MMR copy location and are active independently of whether the **Static Threshold** bit is set. When the MMR is read, then bits 22 and 23 are interpreted as follows:

bit 23	bit 22	condition
0	0	No overflow has occurred
0	1	One or more positive overflows have occurred
1	0	One or more negative overflows have occurred
1	1	Both positive and negative overflows have occurred

Detailed block diagram of the Backend Post-processing Unit

### Access to registers

The MMR and OUC are accessed, through the memory interface, only via their associated buffers (MMB and OUB respectively) and are not accessible directly. In order to load the MMR with a value, the host must first write the value to the MMB and then transfer the data from the MMB to the MMR by performing a WRITE to the **copy MMR** location, 0B4<sub>16</sub>. To read the MMR the host must first perform a READ cycle from location 0B4<sub>16</sub> (which transfers the contents of the MMR into the MMB) and then read the MMB. The OUB is accessed in the same way except that the dummy writes and reads are done to and from location 0BC<sub>16</sub>.

Copies from MMR to MMB and OUC to OUB (reads) can be performed at any time giving a snapshot of the contents of the MMR and OUC respectively. Copies from MMB to MMR and OUB to OUC (writes) can also be performed at any time allowing the threshold and counter to be updated dynamically.

### 5.3 Data transformation unit

The data transformation unit consists of a prescalar, an under/over select detector, a look up table and a byte selector. It can be used in isolation to perform arbitrary data mappings, or in conjunction with the data normaliser to implement sophisticated dynamic range compression functions.

#### Prescalar

This allows an 8-bit field anywhere within the 22-bit X bus to be selected as the address to the LUT. This is performed by right shifting the X bus so that the required 8 bits are at the least significant end. The

amount of right shift is programmed in BCR2[4-0] and can have a value from 0 to 16.

#### Over/under select detector

With PosLUTAddr (SCR[6]) set to zero, this unit monitors whether the amount of right shift performed by the prescalar is sufficient to include all significant bits in, and maintain the sign of, the selected 8 bit field (i.e. an over or under select is generated if the most significant bit of the selected 8 bit field differs from any subsequent bit right up to and including the most significant bit of the right shifted X bus). This will be an **overselect** if the X bus is positive (Bit 21 = 0), and an **underselect** if the X bus is negative (Bit 21 = 1). In other words the LUT address is always deemed to be signed with an address range of -128 to 127.

If however the control bit PosLUTAddr (SCR[6]) is set to one, the unit monitors whether the amount of right shift performed by the prescalar is sufficient to include all significant bits in the selected 8 bit field AND that all unselected bits are zero (i.e. an over or under select is generated if the first selected bit (bit 9) is not zero OR differs from any subsequent bit right up to and including the most significant bit of the right shifted X bus). This will be an **overselect** if the Xbus is positive and an **underselect** WHENEVER the Xbus is negative. Thus, in this mode, the address range of the LUT is 0 to 255.

Prescalar under/over selects and X bus positive/negative overflows are passed to the LUT along with the selected 8 bit address field.

#### Look up table (LUT) and byte select

The LUT consists of 64 words, 32 bits wide plus two special 32 bit locations called the **upper** and **lower saturation** registers (USR and LSR respectively). Thus the LUT is actually 66 words by 32 bits. The 32 bit output of the LUT is called the Y bus.

The most significant 6 bits of the 8 bit address field are used to address one of 64 words in the LUT. The least significant pair of bits in the 8 bit field are used to control a byte select on the output. Thus in addition to operating as a 64+2 word look up table of 32 bit words, it can be used as an 8 bit, 256+2 byte LUT providing 8bit — 8bit transformations.

Positive overflows on the X bus, and over selects in the prescalar cause the LUT to access the USR overriding the address given by the prescalar. Likewise negative overflows and under selects cause the LUT to access the LSR. Any sort of overflow on the X bus or prescalar will cause the byte select control to be overridden and the **most significant byte** (byte 3) of the appropriate Saturation Register will appear on the byte wide output of the data transformation unit.

If there are simultaneous overflows on the X bus and in the prescalar then the overflow from the X bus takes priority.

The USR and LSR can thus be used to model the saturating behaviour of analogue circuits instead of the usual ‘wrap around’ encountered in digital systems. Alternatively the USR and LSR could signal error conditions within the backend directly on the output pins via one of the output multiplexers.

The LUT is loaded via the memory interface. The addressing for the LUT corresponds to the 8 bit field, assuming that the byte selector is being used. In order to access the look up table, USR and LSR from the microprocessor interface, the **LUT Access** control bit ACR[1] must be set to zero. This will force the Y bus to zero and the normaliser to be controlled by BCR3[7-3] regardless of the setting of the dynamic normalisation bit, BCR3[2]. The LUT, USR and LSR can then be loaded with any arbitrary value via the microprocessor interface. Setting the LUT access control bit to one will then allow the LUT to be used in the data transformation unit.

#### 5.4 Data normaliser

This unit consists of a shifter capable of right shifts of up to 14 bits and left shifts up to 2 bits, followed by a *zero data unit* and an adder. The shifter is controllable from one of two 5 bit sources : control bits BCR3[7-3] or bits 26 to 22 of the Y bus. The control bit **Enable Dynamic Normalisation** (BCR3[2]) determines which source is in control of the normaliser. If this bit is set to zero the normaliser is controlled by BCR3[7-3]. The five bit field is a two's complement number between 14 and -2. This indicates the amount of right shift (negative meaning left shift). Any value outside this range causes the output of the shifter to be forced to zero. The output of the shifter, with any rounding generated by the shifter, goes into the output adder.

#### 5.5 Output adder

This is a 22 bit adder with one of its inputs coming from the data normaliser. The other input is either bits 21 to 0 of the Y bus from the data transformation unit, or set to zero under the control of BCR3[1]. Note that any overflow occurring due to left shifting in the normaliser or the subsequent addition in the output adder is not detected by the IMSA110.

#### 5.6 Output multiplexers

These two multiplexers allow the currently selected byte from the LUT to be optionally selected to drive either the most significant byte and/or the least significant byte of the Cascade Output pins. This is

controlled by the state of BCR2[5] and BCR2[6]. Enabling either of these multiplexers overrides the state of the Cascade Output pins only on the relevant 8 pins. The remaining pins will continue to represent the output of the output adder.

### 6. BACKEND POST-PROCESSOR — Modes of Operation

The backend post-processing unit is capable of performing many functions including data scaling, transformation, dynamic range compression and histogram equalisation.

#### 6.1 Default mode (after Reset)

At power up or after reset the state of the backend post-processor is such that data from the MAC array and the cascade input are added and pass straight through the datapath unaffected.

The default mode for the statistics monitor is **min register** although the values in the OUB, OUC, MMR and MMB will be undefined. Likewise the contents of the LUT, USR and LSR will be undefined, the **LUT Access** control bit will be zero forcing the Y bus to zero and allowing the microprocessor interface to access the LUT, USR and LSR.

Note that the cascade output pins and the PSR output pins are tristated.

#### 6.2 Cascade adder / MAC data scalar

These units allow the cascading of IMS A110s where the output of the MAC array may be scaled before it is added to the cascade input data. The shifter can also be used for combining devices to obtain extended precision in input data, coefficient word length or both.

The ability to zero the cascade input provides a simple means of controlling the number of ‘active’ devices cascaded as well as a means of debugging large systems.

#### 6.3 Rectification

Rectification, the removal of negative results, is needed in several image processing functions.

For example, edge detection using a Sobel operator usually requires full wave rectification due to the different signs obtained at differing edge transitions. Edge detection using a Laplacian operator produces a change of sign at an edge. In this case, removing negative numbers using half wave rectification can produce better results as full wave rectification can lead to some blurring of the edge transition.

#### 6.4 Static scaling

This can be performed using one of two units: the MAC array output shifter (as above), and the data normaliser. In the second case the data undergoes a simple scaling operation (with rounding) within the normaliser. The normaliser can be used to scale (multiply) the data by the factors 0, 1/16384, 1/8192, 1/4096 ..., 1/2, 1, 2, 4. By controlling the normaliser from the control bits BCR3[7-3], this provides a means for simple scaling of the data before it is output. Setting BCR3[1] and BCR2[6,7] to zero ensures that the data transformation unit takes no part in the operation and the output of the normaliser is passed unchanged to the output pins.

#### 6.5 Dynamic scaling

In this mode the scaling is controlled by the data itself. i.e. the scalar is controlled from the LUT (Ybus bits 26-22) by setting BCR3[2] to one, the Ybus input to the output adder being set to zero either by setting BCR3[1] to zero or programming the LUT accordingly. This mode can provide a discontinuous non-linear transformation.

#### 6.6 Simple transformation

This mode allows the user to apply arbitrary transformations to the data before it is output. Here the LUT is treated as 256 by 8, addressed as either -128 to 127 if PosLUTAddr is set to zero or 0 to 255 if PosLUTAddr is set to one. The 8 bit field selected by the LUT prescalar is used to address a byte in the LUT which is passed directly to the output pins via one of the output multiplexers. Ybus control of the data normaliser is disabled, BCR3[7-3] are set out of range so as to zero the normaliser output and the Ybus input to the output adder is set to zero by BCR3[1]. One (or both) of the output multiplexers are enabled and so the addressed byte from the LUT passes straight to the cascade output pads. Only the most significant byte of the USR and LSR are applicable in this mode as overflows override the byte select control and force it to select the most

significant byte.

#### 6.7 Dynamic normalisation

In this mode the normaliser and transformation units in the output conditioner are used together to perform sophisticated non-linear dynamic range compression and transformations. As in the simple transformation case the prescalar selects an 8 bit field anywhere within the X bus. The most significant 6 bits, and overflows, are fed as an address to the LUT. In this case the look up table is treated as 64+2 by 32. Bits 26 to 22 of the Y bus are used to control the normaliser block so that the input to the normaliser is dynamically scaled. The output of the normaliser is then added in the output adder to the least significant 22 bits of the Y bus (Note that only 28 bits of the 32 bit Y bus are actually used).

Thus the data is scaled, rounded, and then an offset is added to the scaled result. Each operation can be viewed as

$$\text{output} = \text{input} \times \text{scale} + \text{offset}$$

Where scale and offset are both programmable functions of input. One way to view this operation is to consider that the original data range is divided into 64 equal sized levels and in each level a different scale and offset is applied. The scale and offset stored in the USR and LSR would be chosen to give the desired behaviour under overflow conditions.

Note that in the case of cascade adder overflows, the data on the X bus is invalid, so the scale here would usually be set out of range so as to zero the normaliser output. The offsets in the USR and LSR would then provide the cascade output directly.

Note also that if the 5 bit scale field in the LUT is programmed so that the normaliser always zeros the data, then the output will correspond to the 22 bit offset field in the LUT. This can be viewed as a coarse transformation with wide dynamic range which is useful for applications such as image contour emphasis and equalisation.

**Figure 5 : Bit Format of Data Stored in LUT, USR and LSR**

31	24	23	16	15	8	7	0
	BYTE 3		BYTE 2		BYTE 1		BYTE 0

scale if  
BCR3 [2] = 1      offset if BCR3 [0] = 1

## 7. GLOSSARY

This section defines the meaning of terms used elsewhere in this data sheet.

### Arithmetic Shift

For a right shift, the most significant bit is always copied into the most significant end of the result. For example shifting right by 2:

$$\begin{array}{lll} 01000101 & \rightarrow & 00010001 \\ 11000101 & \rightarrow & 11110001 \end{array}$$

For a left shift, the least significant bit will become zero.

Note that left shifting can cause overflows and these are not detected in the MAC output scalar or the data normaliser.

### Rounding

All rounding done within the IMS A110 is equivalent to truncating after adding 1/2 LSB. (Rounding is always applied in the positive direction). For example for 8 bit twos complement numbers undergoing a two bit right shift:

$$\begin{array}{ll} 00000011 \rightarrow 00000000 + 1 = 00000001 & \text{(rounded up)} \\ 00000010 \rightarrow 00000000 + 1 = 00000001 & \text{(rounded up)} \\ 11111110 \rightarrow 11111111 + 1 = 00000000 & \text{(rounded up)} \\ 00000001 \rightarrow 00000000 & \text{(no rounding)} \\ 11111101 \rightarrow 11111111 & \text{(no rounding)} \end{array}$$

Left shifts do not generate rounding.

### Transversal Filter

A transversal filter is a calculation consisting of the sum of products of successive points of input data. For input data  $x_i, x_{i+1}, \dots$ , and a set of coefficients,  $c_6, c_5, \dots$ , the result,  $Y$  is:

$$Y = \sum_{i=0}^6 c_i \times x_{6-i}$$

### Two's Complement

Two's complement numbers allow both positive and negative numbers. For example in 8 bit numbers the most positive number is 127, the most negative is -128:

two's complement	decimal
10000000	-128
10000001	-127
11111111	-1
00000000	0
00000001	1
01111111	127

### Rectification

Rectification is a method of removing negative numbers. There are two methods: Full wave and Half wave. In either case all positive numbers and zero are unaffected. In Full wave rectification, any negative numbers are negated (i.e. multiplied by 1) so that they become positive. In Half wave rectification, all negative numbers are replaced by zero.

### Dynamic Range Compression

When Dynamic is used in this context, it is to indicate a change of behaviour for each data point. For example, a dynamic shift is one where the size of the shift may change on each successive clock cycle. Dynamic range compression is range compression making use of an offset and shift, which can change depending on each data point. This allows the essential non-linear transformations required in image processing to be implemented on the IMS A110.

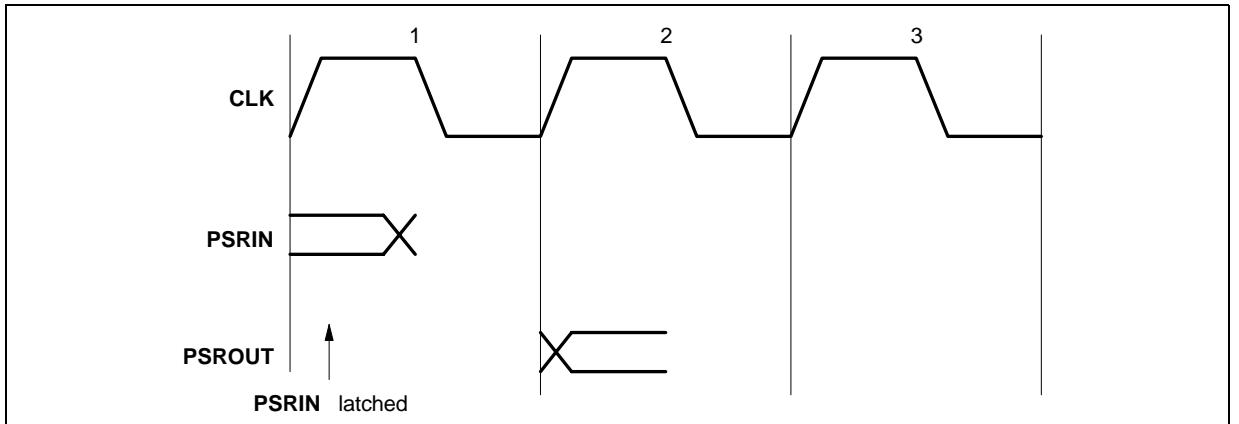
### Bit Fields

Bits, words and addresses in this data sheet are little-endian; The lowest order byte of a multiple byte word is referred to as byte 0, and is addressed in the same way. Similarly, the least significant bit of any bit field is that with the lowest bit number. For example, 'bits 26-22' refers to a 5 bit field where bit 22 is treated as the least significant, and bit 26 as the most significant.

### Latency

Within the IMSA110 the latency is the number of clock cycles from an input to its corresponding output. For instance, with the programmable shift registers bypassed by setting SCR[1] to 1, the latency from PSRin to PSRout will be 2 as shown in Figure 6.

Figure 6



A110-07.EPS

## PIN DESIGNATIONS

### System services

Pin	In/out	Function
<b>V<sub>cc</sub>, GND</b>		Power supply and return
<b>CLK</b>	in	Input clock
<b>RESET</b>	in	System reset

### Synchronous input/output

Pin	In/out	Function
<b>PSRin[7-0]</b>	in	Programmable shift register input
<b>PSRout[7-0]</b>	out	Programmable shift register output
<b>Cin[21-0]</b>	in	Cascade input port
<b>Cout[21-0]</b>	out	Cascade output port

### Asynchronous input/output

Pin	In/out	Function
<b>E1, E2</b>	in	Memory interface enable signals
<b>W</b>	in	Memory interface write enable
<b>ADR[8-0]</b>	in	Memory interface address bus
<b>D[7-0]</b>	in/out	Memory interface data bus

Note : Signal names are shown with an overbar if they are active low, otherwise they are active high.

### 8.1 System services

System services include all the necessary logic to start up and maintain the IMS A110.

#### Power

Power is supplied to the device via the **V<sub>cc</sub>** and **GND** pins. Several of each are provided to mini-

mise inductance within the package. All supply pins must be connected. The supply must be decoupled close to the chip by at least one 100nF low inductance (e.g. ceramic) capacitor between **V<sub>cc</sub>** and **GND**.

#### CLK

The clock signal **CLK** controls the timing of input and the output on the four dedicated interfaces, and controls the progress of data through the shift registers, multiply-accumulate array and post-processing unit. The A110 is fully static so the clock can be slowed down or stopped in either state without corrupting data.

#### RESET

If this pin is taken low for at least 2 clock cycles, the control logic within the IMS A110 will be reset and all of the control and configuration registers will be initialised to their default values. All other register, memory locations, datapath registers and shift registers will not be reset by this signal.

A reset is initiated automatically when power is first applied to the device. This reset will be completed once four cycles of **CLK** have occurred after **VCC** is valid.

### 8.2 Synchronous services

#### PSRin[7-0]

This 8-bit wide bus supplies input data to the device. The input data enters the first of the three shift registers in the chain. The timing of this input is controlled by the **CLK** signal. The data on the **PSRin** port is sampled on the rising edge of the clock. In a cascade arrangement, this bus will be connected to the **PSRout** port of the previous device. In such an arrangement the **PSRin** port on the first device will be the input to the overall cascaded system.

**PSRout[7-0]**

This bus outputs the data from the last programmable shift register in the chain. The data on this bus is synchronously clocked by the rising edge of **CLK**. In a cascade arrangement this port will be connected to the **PSRin** port of the next device. At power up, or after a reset, the **PSRout** pins are tristated. They are enabled by SCR[5].

**Cin[21-0]**

The Cascade Input port allows IMSA110s to be cascaded. It also can be used for combining an external signal (e.g. a reference image or an offset) with the processed result. In a cascade arrangement, this bus will be connected to the Cascade Output of the previous device. The data on the **Cin** bus is sampled on the rising edge of **CLK**.

**Cout[21-0]**

This bus outputs the processed result from the IMSA110 and can also be used for cascading. The 22-bit result is synchronously clocked by the rising edge of **CLK**. In a typical cascaded system this bus will be connected to the Cascade Input port of the next device. On the last device in the cascade, this bus will be the output of the overall system. At power up, or after a reset, the **Cout** pins are tristated. They are enabled by SCR[4].

**8.3 Asynchronous input/output****E1, E2**

If both of these signals are low, then the microprocessor interface is enabled. The operation of these enable signals is very similar to those found on static RAMs. When either of these signals are high

the Write Enable and the address inputs are ignored and the microprocessor interface Data signals are high impedance. When both Enable signals are low a read or write access is made to registers or the RAMs within the IMSA110. Access to the microprocessor interface can occur asynchronously to the synchronous pins (**PSRin**, **PSRout**, **Cin**, **Cout**) of the device.

**W**

Write Enable indicates whether the access to the IMS A110 memory interface is to be a read or a write. If **W** is low a write access is indicated.

**ADR[8-0]**

The nine bit binary value applied to the address inputs of the IMSA110 indicates which register or RAM location within the device is to be accessed.

**D[7-0]**

During a write to the microprocessor interface an 8-bit word is applied to the Data pins which is written to the appropriate location. During a read cycle the contents of the location accessed are placed on the Data pins. When either of the Enables are high the Data pins are high impedance.

**9 REGISTER DESCRIPTION****Memory map**

Within the IMSA110 addresses are fully decoded. Reading from locations not defined in the memory map will produce zero data. Data written to such locations is ignored. This allows the part to be fully programmed using a ROM with an address incrementer. In this case, for future compatibility, zero should be written to all undefined locations.

Register	Address decimal	Address hex	Function
<b>CR0a</b>	0—6	000—006	Coefficient Registers Bank 0a
<b>CR0b</b>	16—22	010—016	Coefficient Registers Bank 0b
<b>CR0c</b>	32—38	020—026	Coefficient Registers Bank 0c
<b>CR1a</b>	64—70	040—046	Coefficient Registers Bank 1a
<b>CR1b</b>	80—86	050—056	Coefficient Registers Bank 1b
<b>CR1c</b>	96—102	060—066	Coefficient Registers Bank 1c
<b>PCRA</b>	128—129	080—081	PSRA Control Register
<b>PCRB</b>	130—131	082—083	PSRB Control Register
<b>PCRC</b>	132—133	084—085	PSRC Control Register
<b>SCR</b>	144	090	Static Control Register
<b>ACR</b>	146	092	Active Control Register
<b>BCR</b>	160—163	0A0—0A3	Backend Configuration Register
<b>MMB</b>	176—178	0B0—0B2	Maximum/Minimum Buffer
<b>CMM</b>	180	0B4	Copy MMR
<b>OUB</b>	184—186	0B8—0BA	Overshoot/Undershoot Buffer
<b>COU</b>	188	0BC	Copy OUC
<b>TCR</b>	208	0D0	Test Control Register
<b>USR</b>	248—251	0F8—0FB	Upper Saturation Register
<b>LSR</b>	252—255	0FC—0FF	Lower Saturation Register
<b>LUT</b>	256—511	100—1FF	Look up Table

## 9.2 Registers

### CR0a      Coefficient registers bank 0a

These seven 8-bit locations contain coefficients which can be used by the third, of the three, 7-stage mac arrays. CR0a(0) (address #000) corresponds to the coefficient register of this mac array nearest to its output. Similarly CR0a(6) (address #006) corresponds to the coefficient register of this mac nearest to its input. These Coefficient registers can be written to provided that the other register bank is in use. Whether the coefficient written is signed or unsigned is determined by the 'Unsigned Coefficient' bit SCR[3]. Once a value is written to a coefficient register, its value can be read back from

an internal duplicate register. These registers will be used by the mac array, when ACR[0], 'Current Bank' is set to zero. Writing to these Coefficient Registers while in use will result in an undefined operation of the mac array.

### CR0b      Coefficient registers bank 0b

These seven 8-bit locations contain coefficients which can be used by the second, of the three, 7-stage mac arrays in the chain. CR0b(0) (address #010) corresponds to the coefficient register of this mac array nearest to its output. Similarly CR0b(6) (address #016) corresponds to the coefficient register of this mac nearest to its input. Their behaviour is otherwise identical to CR0a.

Figure 7 : IMSA110 Memory Map

Address (Hex)	Name	bit										
		7	6	5	4	3	2	1	0			
1FF	<b>LUT</b>	Look Up Table										
100	<b>LSR</b>	Lower Saturation Register										
0FC-0FF	<b>USR</b>	Upper Saturation Register										
0F8-0FB	<b>TCR</b>											
0D0	<b>COU</b>	Copy Over/UnderShoot Buffer										
0BC	<b>OUN</b>	Over/UnderShoot Buffer										
0B8-0BA	<b>CMM</b>	Copy Min/Max Buffer										
0B4	<b>MMB</b>	Min/Max Buffer										
0B0-0B2	<b>BCR3</b>	Normaliser Control					Dynamic normali- sation	LUT to output adder	0			
0A3	<b>BCR2</b>	0	LS output byte	MS output byte	Look Up Prescaler							
0A2	<b>BCR1</b>	0	0	0	0	0	0	Static threshold	Greater Than			
0A0	<b>BCR0</b>	Full Wave	Half Wave	MAC Output Scaler					Zero Cascade			
092	<b>ACR</b>	0	0	0	0	0	0	Backend LUT Access	Current Bank			
090	<b>SCR</b>	0	PosLUT Addr	PSR Out Enable	Cascade Enable	Unsigned Coef Load	Unsigned Data	Bypass PSRs	Cont Swap			
085	<b>PCRC</b>	0	0	0	0	0	Shift Length (Upper Bits)					
084	<b>PCRC</b>	Shift Length (Lower Bits)										
083	<b>PCRB</b>	0	0	0	0	0	Shift Length (Upper Bits)					
082	<b>PCRB</b>	Shift Length (Lower Bits)										
081	<b>PCRA</b>	0	0	0	0	0	Shift Length (Upper Bits)					
080	<b>PCRA</b>	Shift Length (Lower Bits)										
066	<b>CR1c</b>	Bank 1 Coefficient Register										
060	<b>CR1b</b>	Bank 1 Coefficient Register										
056	<b>CR1a</b>	Bank 1 Coefficient Register										
050	<b>CR0c</b>	Bank 0 Coefficient Register										
046	<b>CR0b</b>	Bank 0 Coefficient Register										
040	<b>CR0a</b>	Bank 0 Coefficient Register										
026												
020												
016												
010												
006												
000												

**CR0c      Coefficient registers bank 0c**

These seven 8-bit locations contain coefficients which can be used by the first, of the three, 7-stage mac arrays in the chain. CR0c(0) (address #020) corresponds to the coefficient register of this mac array nearest to its output. Similarly CR0c(6) (address #026) corresponds to the coefficient register of this mac nearest to its input. Their behaviour is otherwise identical to CR0a.

**CR1a      Coefficient registers bank 1a**

These seven 8-bit locations contain coefficients which can be used by the third, of the three, 7-stage mac arrays in the chain. CR1a(0) (address #040) corresponds to the coefficient register of this mac array nearest to its output. Similarly CR1a(6) (address #046) corresponds to the coefficient register of this mac nearest to its input. These registers will be used provided that ACR[0], 'Current Bank' is set to one, or continuous bank swap mode is in operation (SCR[0] set to one).

**CR1b      Coefficient registers bank 1b**

These seven 8-bit locations contain coefficients which can be used by the second, of the three, 7-stage mac arrays in the chain. CR1b(0) (address #050) corresponds to the coefficient register of this mac array nearest to its output. Similarly CR1b(6) (address #056) corresponds to the coefficient register of this mac nearest to its input. Their behaviour is otherwise identical to CR1a.

**CR1c      Coefficient registers bank 1c**

These seven 8-bit locations contain coefficients which can be used by the second, of the three, 7-stage mac arrays in the chain. CR1c(0) (address #060) corresponds to the coefficient register of this mac array nearest to its output. Similarly CR1c(6) (address #066) corresponds to the coefficient register of this mac nearest to its input. Their behaviour is otherwise identical to CR1a.

**PCRA      PSRA Control register**

This is a 16-bit register, with least significant byte at location #080, and is used to set up the length of the last shift register in the chain. Programmed lengths outside the range 0 to 1120 will cause undefined behaviour of the shift register.

**PCRB      PSRB Control register**

This is a 16-bit register, with least significant byte at location #082, and is used to set up the length of the second shift register in the chain. Programmed lengths outside the range 0 to 1120 will cause undefined behaviour of the shift register.

**PCRC      PSRC Control register**

This is a 16-bit register, with least significant byte at location #084, and is used to set up the length of the first shift register in the chain. Programmed lengths outside the range 0 to 1120 will cause undefined behaviour of the shift register.

**SCR      Static control register**

The Static Control Register contains the control bits which set up parts of the IMS A110 which are likely to not need reconfiguration during processing. The contents of this register are not affected by the IMS A110 and can be read at any time. Modifying the Static Control register during processing will result in undefined behaviour. Normal operation will start to occur between 0 and 3 clock cycles after the completion of the write cycle.

**ACR      Active control register**

The Active Control Register contains status and control bits which are likely to be accessed during normal operation of the IMS A110.

**BCR      Backend configuration register**

The Backend Configuration Registers consist of four byte-wide registers BCR0, BCR1, BCR2, and BCR3 which are located at addresses #0A0, #0A1, #0A2, and #0A3 respectively. These four registers are used to control the backend post-processing unit. None of the control bits in these registers can be modified by the IMS A110. Modification of the values in these registers during processing may result in undefined behaviour. Normal operation will start to occur between 0 and 3 clock cycles after the completion of the write cycle.

**MMB      Maximum/minimum buffer**

These three locations hold a 24-bit wide word, with the least significant byte at the lowest address, and act as a buffer between the MMR and the micro-processor interface. All the transactions between the MMR and the host processor must take place through this register. When the MMR is not in use, the value of this buffer is undefined.

**CMM      Copy MMR**

This location is used to enable the data transfer between the MMB and MMR. A write to this location causes the contents of MMB to be copied into the MMR and bits 23 and 22 of the MMR (the cascade adder overflow flags) to be set to zero. A read from this location causes the reverse, i.e the contents of the MMR are copied into the MMB. The value written to this location is ignored, the value read back is undefined.

## OUN Overshoot/undershoot buffer

These three memory locations hold a 22-bit word, with the least significant byte at the lowest address, and act as a buffer between the OUC and the microprocessor interface. All the transactions between the OUC and the host processor must take place through this register. When the OUC is not in use, the value of this buffer is undefined.

COU Copy OUC

This location in the memory is used to enable the data transfer between the OUB and OUC. A write to this location causes the contents of OUB to be copied into the OUC. A read from this location causes the reverse, i.e the contents of the OUC are copied into the OUB. The value written to this location is ignored, the value read back will be undefined.

## TCR Test control register

This register is used for testing, and should be loaded with zero for normal operation.

## USR Upper saturation register

This is a 32-bit value with the least significant byte at the lowest address. Its contents are used to replace the LUT output if positive overflow(s) occur in the look up prescaler and / or in the cascade adder. Accesses from the microprocessor interface can only be made while ACR[1] is set to zero.

## LSR Lower saturation register

This is a 32-bit value with the least significant byte at the lowest address. Its contents are used to replace the LUT output if negative overflow(s) occur in the look up prescaler and / or in the cascade adder. Accesses from the microprocessor interface can only be made while ACR[1] is set to zero.

## LUT Look up table

These locations are for the 256-byte look up table which is used for data mapping and transformation operations. From the microprocessor interface,

these locations are addressed in the same way as that seen by the 8-bit output of look up prescaler. When used in 32 bit mode, the locations are treated in the same way as other 32 registers: Word 0 has its most significant byte at #103, its least significant byte at #100, Word 12 has its most significant byte at #133, its least significant byte at #130. Accesses from the microprocessor interface can only be made while ACR[1] is set to zero.

## 10. REGISTERS — BIT ALLOCATION

This section describes the register details bit by bit. Each section commences with the name of the register with the bit number(s) followed by the default value, in the general format:

Name REGISTER [MSB—LSB] Default : MSB...LSB

The least significant bit of a register is bit 0.

\* in the tables indicates the default state of the register bit(s).

## 10.1 PSR control registers (PCR)

PSRA control PCRA[10-0] Default: 0...0

These eleven least significant bits of the PCRA are used to specify the length of the last Programmable Shift Register (PSRA). The length of the shift register will be numerically equal to the binary value loaded in these bits. The value loaded in must be in the range of 0 to 1120 decimal. If a value outside this range is written to these bits the behaviour of the shift register will be undefined. After updating this register, the behaviour of the delay is undefined for 22 clock cycles. Hence changing the length from 1000 to 1001 delays, will result in correct output only after 1023 cycles. This will also have to propagate through the backend before the cascade output values will be correct.

Reserved PCRA[15-11] Default: 00000

These 5 most significant bits of the PCRA are reserved. The user should write zero to these locations to maintain compatibility with future products. The value read from these locations will be zero.

## PSRB control      PCR[10-0]      Default: 0...0

These eleven least significant bits of the PCR[10-0] are used to specify the length of the second Programmable Shift Register (PSRB). The length of the shift register will be numerically equal to the binary value loaded in these bits. The value loaded in must be in the range of 0 to 1120 decimal. If a value outside this range is written to these bits the behaviour of the shift register will be undefined. After updating this register will also have to propagate through PSRA and the backend before the cascade output values will be correct

## Reserved      PCR[15-11]      Default: 00000

These 5 most significant bits of the PCR[15-11] are reserved. The user should write zero to these locations to maintain compatibility with future products. The value read from these locations will be zero.

## PSRC control      PCRC[10-0]      Default: 0...0

These eleven least significant bits of the PCRC[10-0] are used to specify the length of the first Programmable Shift Register (PSRC). The length of the shift register will be numerically equal to the binary value loaded in these bits. The value loaded in must be in the range of 0 to 1120 decimal. If a value outside this range is written to these bits the behaviour of the shift register will be undefined. After updating this register will also have to propagate through PSRB, PSRA and the backend before the cascade output values will be correct

## Reserved      PCRC[15-11]      Default: 00000

These 5 most significant bits of the PCRC[15-11] are reserved. The user should write zero to these locations to maintain compatibility with future products. The value read from these locations will be zero.

## 10.2 Static control register (SCR)

### Reserved      SCR[7]      Default: 0

This location is reserved. The user should write zero to this location to maintain compatibility with future products. The value read from this location will be zero.

### Positive Look up table address      SCR[6]      Default: 0

This bit affects the way in which the over/under select detector checks the LUT address. It determines whether the address range of the LUT is signed (-128 to 127) or positive (0 to 255). A one at this location indicates a positive LUT address.

### PSR out Enable      SCR[5]      Default: 0

A zero at this location will force the PSR Output pins into the tristate mode.

### Cascade Enable      SCR[4]      Default: 0

A zero at this location will force the Cascade Output pins into the tristate mode.

### Unsigned coefficient load      SCR[3]      Default: 0

If this bit is set to one, the format of subsequently loaded coefficients become unsigned, with coefficient value assuming a range between 0 and 255 decimal. An 8-bit coefficient with all its bits set to one will represent +255 decimal. When this bit is zero the format of subsequently loaded coefficients will be twos complement and the corresponding numerical value will have a range between -128 and +127. By changing this bit whilst coefficients are being loaded, coefficients between -128 and +255 can be used. The unsigned format on all coefficients is suitable when IMS A110s are combined to obtain wider coefficients for extended precision.

SCR[3]	Coefficient type
0	Signed coefficients *
1	Unsigned coefficients

### Unsigned data      SCR[2]      Default: 0

If this bit is set to one, the IMS A110 input data format will become unsigned, with input data value assuming a range between 0 and 255 decimal. An 8-bit value with all its bits set to one will represent +255 decimal. When this bit is zero the input data format will be twos complement and the corresponding numerical value will have a range between -128 and +127. Unlike SCR[3], this bit cannot be used to dynamically alter the data format. The unsigned format is suitable when IMS A110s are combined to obtain wider input data for extended precision.

SCR[2]	Data type
0	Signed data *
1	Unsigned data

### Bypass shift registers      SCR[1]      Default: 0

This bit is used to program the path between the PSRin and PSRout ports. A zero at this location will cause the output from the last programmable shift register to be sent to PSRout port. Writing a one to this bit will cause the three programmable shift registers to be bypassed, and the data entering the port PSRin to be fed directly, via a delay of 2 clock cycles, to the port PSRout. This bit allows full programmability of a cascade arrangement so that the same hardware can be operated in a variety of ways.

**Continous bank swap SCR[0] Default: 0**

The continuous bank Swap bit selects whether the two banks of coefficient registers are used alternately after each data input or if this is controlled solely by the state of the 'Current Bank' bit in the Active Control Register ACR[0].

SCR[0]	Swap mode
0	Swap on asserting ACR[0] *
1	Swap after end of each input cycle

**10.3 Active control register (ACR)**

**Reserved ACR[7-2] Default: 00000**

These 6 most significant bits of the ACR are reserved. The user should write zero to these locations to maintain compatibility with future products. The value read from these locations will be zero.

**Enable look up table ACR[1] Default: 0**

Writing a zero into this control bit allows the memory interface to access the Look up table; the output to the data transformation unit will be zero. The normaliser will be controlled by BCR3[7-3], regardless of the state of BCR3[2]. Writing a one to ACR[1] allows the IMS A110 to use the Look up Table. After changing this bit, 2 clock cycles must occur before the Look up Table can be accessed.

ACR[1]	LUT mode
0	Memory interface access *
1	Data transformation unit

**Current bank ACR[0] Default: 0**

When the 'Continuous Bank Swap' bit is set to zero, writing a zero into this control bit instructs the IMS A110 to use the set of coefficient registers at addresses 0 to #X26. Setting a one to this bit instructs the IMS A110 to use the set of coefficient registers at addresses #40 to #X66. If the 'Continuous Bank Swap' bit is set to one, then this bit only indicates the bank selected for the first cycle of the continuous swap mode. Writing to this bit whilst in continuous bank swap mode (SCR[0]=1) will result in undefined behaviour of the mac array.

ACR[0]	Coefficient bank
0	Use coefficient registers at 0 to #X26 *
1	Use coefficient registers at #40 to #X66

**10.4 Backend control register 0 (BCR0)**

**Enable full-wave BCR0[7] Default: 0 rectification**

If this bit is set the output of the cascade adder is

full-wave rectified (absolute value operation) before it is fed to the remainder of the backend. This bit will override the function of the BCR0[6].

**Enable half-wave BCR0[6] Default: 0 rectification**

Writing a one in this bit will cause the negative values from the cascade adder to be replaced with zero. Note that writing a one into BCR0[7] will override the function of this control bit.

BCD0[7-6]	Rectifier mode
0 0	Straight through *
0 1	Half wave rectification
1 0	Full wave rectification
1 1	Full wave rectification

**Mac array output BCR0[5-1] Default: 00000 scaler**

The contents of these five bits control the amount of right or left shift applied to the data at the output of the mac array. This field is interpreted as a two's complement number. A positive number represents a right shift (divide). Any shift in the range -8 (11000) to +8 (01000) is legal. Values outside this range will result in undefined behaviour of the mac output scaler.

**Zero cascade input BCR0[0] Default: 0**

This bit controls the Cascade Input Multiplexer. Writing a one to this bit will cause a zero, instead of the cascade input data, to be fed to the cascade adder.

BCR[0]	Cascade input mode
0	Cascade data *
1	Zero

**10.5 Backend control register 1 (BCR1)**

**Reserved BCR1[7-2] Default: 00000**

These locations are reserved. The user should write zero to these locations to maintain compatibility with future products. The values read from these locations will be zero.

**Static threshold BCR1[1] Default: 0**

If this bit is set to one, the signals from the comparator will be used to increment the Over / Under- shoot Counter only. If this bit is zero, the signals from the comparator will be used to latch the output of the Cascade Adder into the Maximum / Minimum Register (MMR), and to increment the counter. In this case the counter will have been incremented by the number of times that the threshold has been updated.

**Enable greater than      BCR1[0]      Default: 0**

This control bit determines whether the comparator in the statistics monitor behaves as a 'greater than', or as a 'less than' comparator. The signal from this comparator is used to drive the Over / Undershoot Counter and the Max / Min Register. A one at this location selects 'greater than'.

BCR1[1-0]	Statistics monitor mode
0 0	Min. register *
0 1	Max. register
1 0	Undershoot counter
1 1	Overshoot counter

**10.6 Backend control register 2 (BCR2)****Reserved      BCR2[7]      Default: 0**

This location is reserved. The user should write zero to this location to maintain compatibility with future products. The value read from this location will be zero.

**Pass LUT data to      BCR2[6]      Default: 0  
least significant  
output**

This bit controls the output multiplexer. If this bit is set to one, the selected byte from the LUT is output on the least significant byte (bits 7 to 0) of the Cascade Output pins.

**Pass LUT data to most      BCR2[5]      Default: 0  
significant output**

This bit controls the output multiplexer. If this bit is set to one, the selected byte from the LUT is output on the most significant byte (bits 21 to 14) of the Cascade Output pins.

**Look up prescaler      BCR2[4-0]      Default: 00000**

The contents of these five bits control the amount

of (arithmetic) right shift applied to the data, by the Look up Prescaler. Writing a numerical value between 0 and 16 (binary 10000) into these bits, will cause the data to be right-shifted by a corresponding number of places. For example, if the bit pattern 00101 is written to these five bit positions, a right shift of 5 places will occur. Writing any value outside the range (0 to 16) will result in undefined behaviour of the look up Prescaler.

**10.7 Backend control register 3 (BCR3)****Normalizer control      BCR3[7-3]      Default: 00000**

These five bits control the number of places, that the normaliser shifts the data to the right or to the left. This field is interpreted as a twos complement number. A positive number is taken to be a right shift. Any shift in the range -2 (11110) to +14 (01110) is legal. Any other value will cause the number zero to be output from the normaliser.

**Enable dynamic      BCR3[2]      Default: 0  
normalization**

If this bit is set to one, the normaliser will be controlled by bits 26 to 22 from the output of the look up table, instead of BCR3[7-3].

**Feed LUT data to      BCR3[1]      Default: 0  
output adder**

One of the inputs of the Output Adder can be either supplied by the Look up Table or forced to zero. Setting this control bit to zero selects zero. Setting this control bit to one selects bits 21 to 0 of the Look up Table.

**Reserved      BCR3[0]      Default: 0**

This location is reserved. The user should write zero to this location to maintain compatibility with future products. The value read from this location will be zero.

## 11. ELECTRICAL SPECIFICATION

### 11.1 DC electrical characteristics

#### ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Min.	Typ.	Max.	Units	Notes (1,2)
$V_{CC}$	DC supply voltage	0		7.0	V	3
$V_I, V_O$	Voltage on any other pin	-1.0		$V_{CC}+0.5$	V	3
$T_A$	Temperature under bias	-40		85	°C	
$T_{STG}$	Storage temperature	-65		150	°C	
PDmax	Power dissipation			2.0	W	

Notes : 1. All voltages are with respect to **GND**.

2. This is a stress rating only and functional operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not implied. Stresses greater than those listed may cause permanent damage to the device. Exposure to absolute maximum rating conditions for extended periods may affect reliability.
3. This device contains circuitry to protect the inputs against damage caused by high static voltages or electrical fields. However, it is advised that normal precautions be taken to avoid application of any voltage higher than the absolute maximum rated voltages to this high impedance circuit. Unused inputs should be tied to an appropriate logic level such as  $V_{CC}$  or **GND**.

#### DC OPERATING CONDITIONS

Symbol	Parameter	Min.	Typ.	Max.	Units	Notes (1)
$V_{CC}$	Supply Voltage	4.5	5.0	5.5	V	
$V_{IH}$	Input Logic '1' Voltage <b>CLK</b> Input Logic '1' Voltage other pins	4.0 2.0		$V_{CC}+0.5$ $V_{CC}+0.5$	V V	2 2
$V_{IL}$	Input Logic '0' Voltage <b>CLK</b> Input Logic '0' Voltage other pins	-0.5 -0.5		0.5 0.8	V V	2 2
$T_A$	Ambient Operating Temperature	0		70	°C	3

Notes : 1. All voltages are with respect to **GND**.

2. Input signal transients, up to 10ns wide, are permitted in the voltage ranges (**GND** - 0.5V) to (**GND** - 1.0V) and  $V_{CC}$  + 0.5V to  $V_{CC}$  + 1.0V.
3. 400 linear ft/min transverse air flow.

#### DC CHARACTERISTICS

Symbol	Parameter	Min.	Typ.	Max.	Units	Notes (1,2)
$V_{OH}$	Output Logic '1' Voltage	2.4		$V_{CC}$	V	4
$V_{OL}$	Output Logic '0' Voltage	0		0.4	V	5
$I_{IN}$	Input leakage current(any input current)			$\pm 10$	$\mu A$	3
$I_{OZ}$	Off state output leakage current			$\pm 10$	$\mu A$	3
$I_{DD}$	Average power supply current			350	mA	

Notes : 1. All voltages are with respect to **GND**.

2. Parameters measured over full voltage and temperature operating range.
3.  $V_{CC} = V_{CC}(\text{max})$ ,  $\text{GND} \leq V_{IN} \leq V_{CC}$
4.  $I_{OZ} \leq -4.4 \text{ mA}$
5.  $I_{OZ} \leq 4.4 \text{ mA}$

#### CAPACITANCE

Pin	Min.	Typ.	Max.	Units	Notes
<b>CLK</b>		12		pF	1,2
All other pins		5		pF	1,2

Notes : 1. This parameter is supplied for engineering guidance and is not guaranteed.

2.  $TA = 25^\circ\text{C}$ ,  $F = 1 \text{ MHz}$ .

## IMSA110

### 11.2 Thermal Characteristics

#### PIN GRID ARRAY THERMAL CHARACTERISTICS

Symbol	Parameter	Min	Nom	Max	Units	Notes
$\theta_{JA}$	Junction to ambient thermal resistance			35	°C/W	1,2

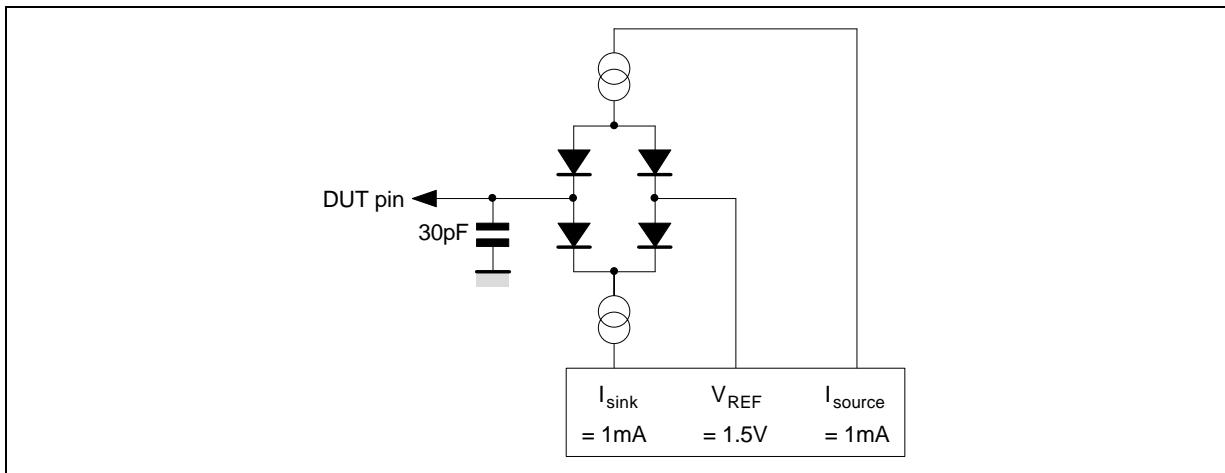
Notes : 1. Measured at 400 linear ft/min transverse air flow.  
 2. This parameter is sampled and not 100% tested.

### 11.3 AC timing characteristics

#### AC test conditions

OUTPUT LOADS (except output turn-off tests) : 30pF for all outputs.

Figure 8 : Output Load (output turn-off tests)



#### TIMING REFERENCE LEVELS

Pin	Reference levels	Notes
INPUTS	0.8V, 2.0V	1
CLK	0.5V, 4.0V	
OUTPUTS	0.4V, 2.4V	2,3
OUTPUTS	$\pm 100\text{mV}$ change from previous steady output voltage	4

Notes : 1. Except CLK.  
 2. Output continuously driven.  
 3. Timings are tested using  $V_{OL}=0.8\text{V}$  and with a suitable allowance for the time taken for the output to fall from 0.8V to 0.4V.  
 4. Output turn-off tests.

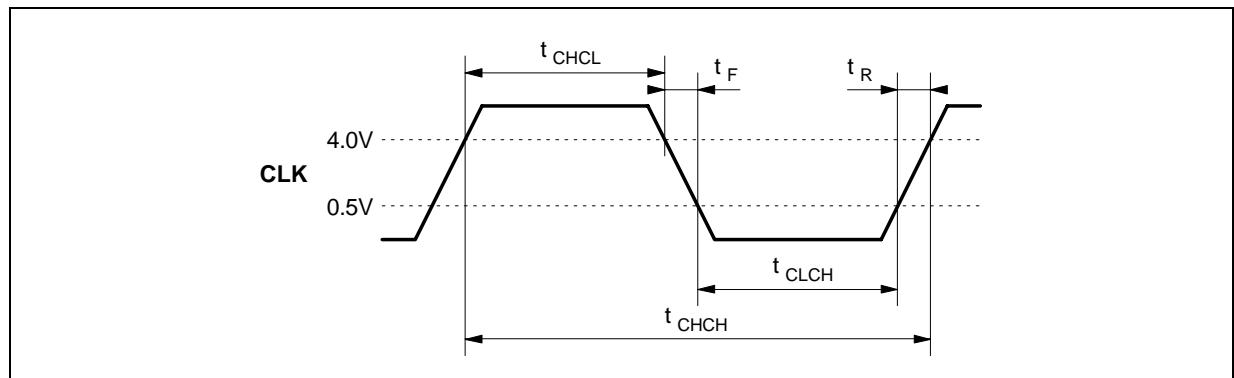
### 11.4 Timing diagrams

#### CLOCK REQUIREMENTS

Symbol	Parameter	Min	Typ.	Max	Units	Notes
$t_{CHCL}$	Clock Pulse High Width	20			ns	2
$t_{CLCH}$	Clock Pulse Low Width	20			ns	2
$t_{CHCH}$	Clock Period	50			ns	2
$t_R$	Clock rise time	0		50	ns	1
$t_F$	Clock fall time	0		50	ns	1

Notes : 1. Clock input transitions should be monotonic between the input thresholds of 0.5 V and 4.0 V.  
 2. For Rev.A parts  $t_{CHCL}$ ,  $t_{CLCH}$  and  $t_{CHCH}$  have **maximum** values of 50 000ns, 50 000ns and 100 000ns respectively.  
 (A minimum clock frequency of 10kHz.)

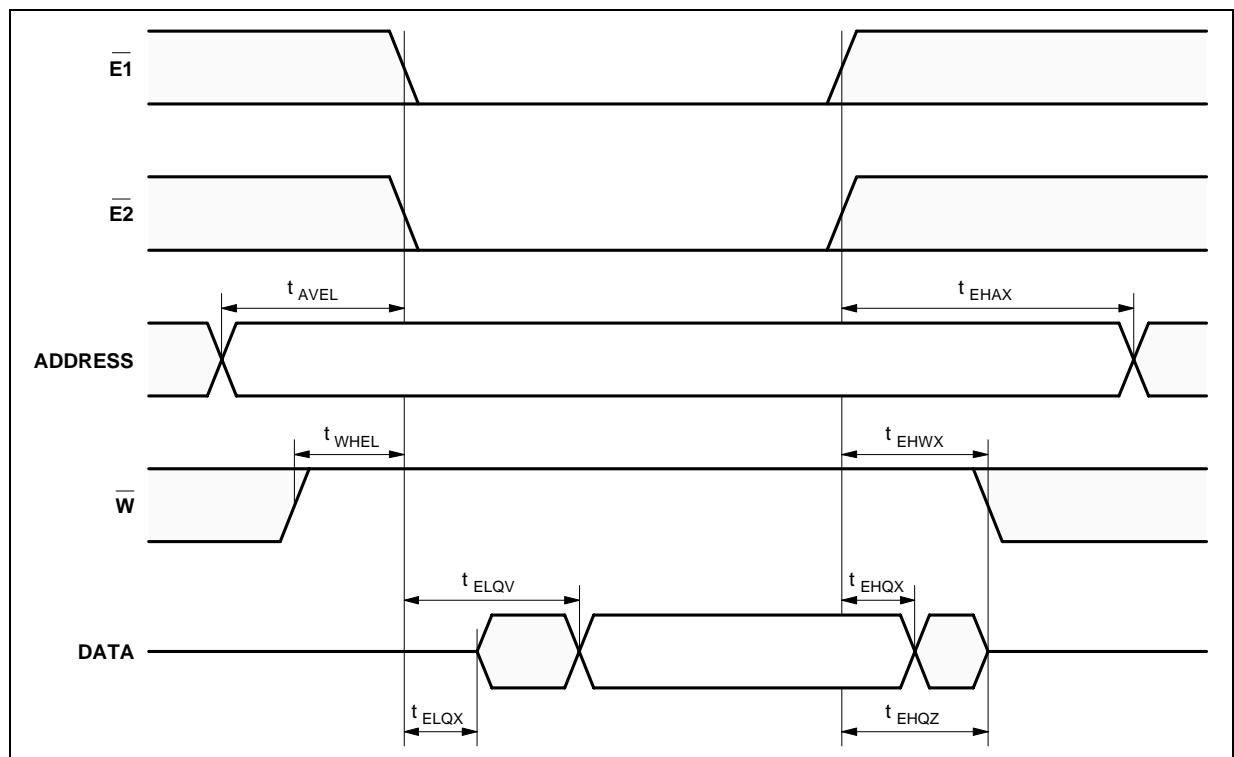
Figure 9



MICROPROCESSOR INTERFACE READ CYCLE

Symbol	Parameter	Min	Max	Units	Notes
$t_{AVEL}$	Address setup	0		ns	
$t_{EHAX}$	Address hold	0		ns	
$t_{WHEL}$	Read Command Setup	0		ns	
$t_{EHWX}$	Read Command Hold	0		ns	
$t_{ELQX}$	Output turn-on	0		ns	
$t_{ELQV}$	Read data access		100	ns	
$t_{EHQX}$	Read data hold	0		ns	
$t_{EHQZ}$	Output turn off		25	ns	

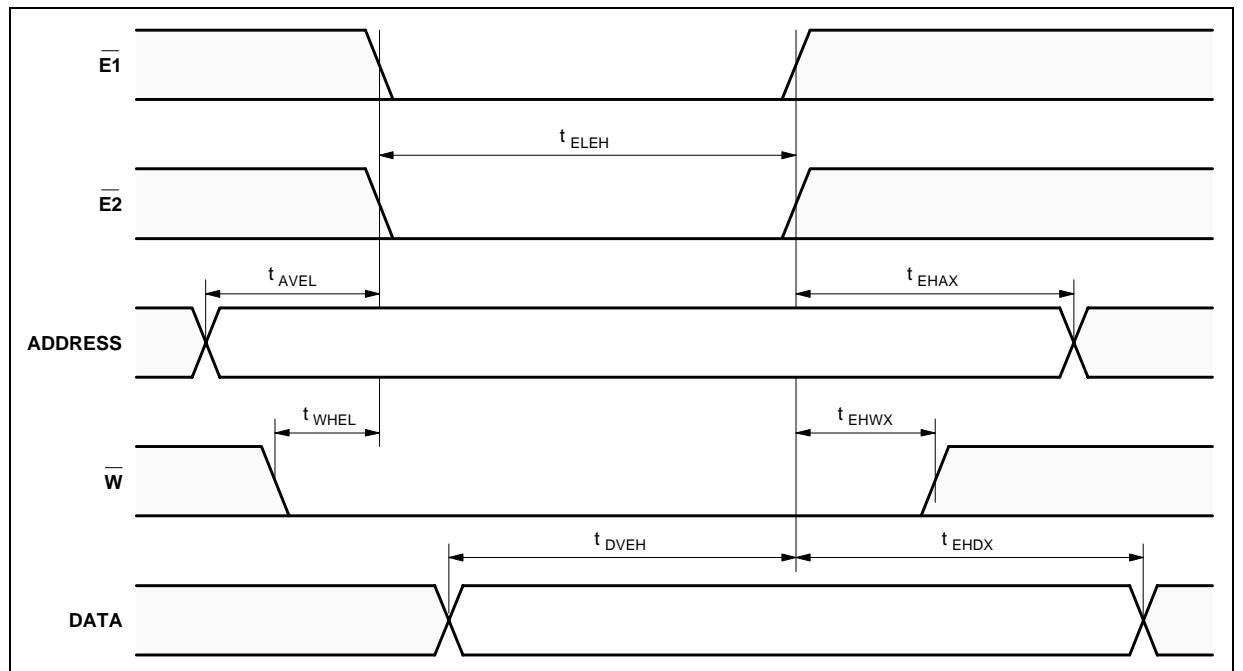
Figure 10



## MICROPROCESSOR INTERFACE WRITE CYCLE

Symbol	Parameter	Min	Max	Units	Notes
$t_{ELEH}$	Enable Width Low	100		ns	
$t_{AVEL}$	Address setup	0		ns	
$t_{EHAX}$	Address hold	0		ns	
$t_{WLEL}$	Write Command Setup	0		ns	
$t_{EHWX}$	Write Command Hold	0		ns	
$t_{DVEH}$	Write data Set up	50		ns	
$t_{EHDX}$	Write data hold	0		ns	

Figure 11

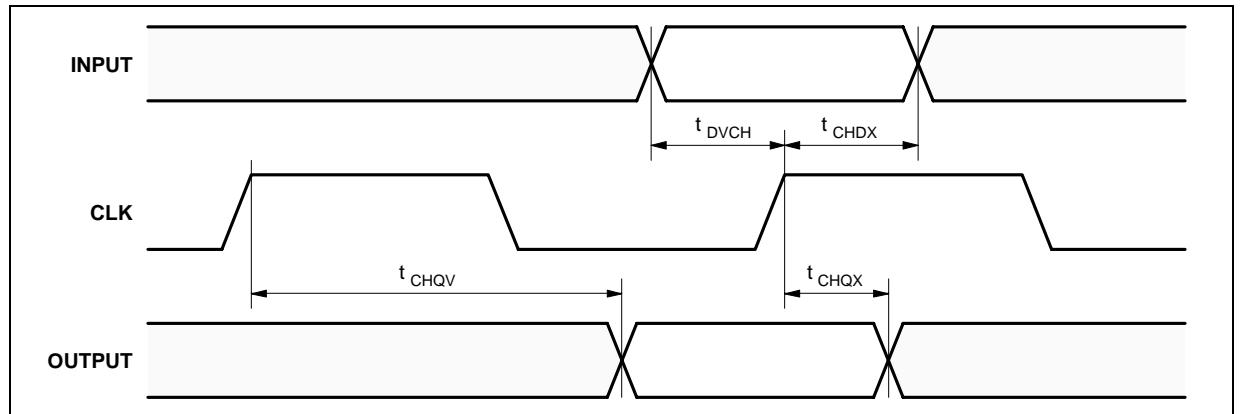


A110-11.EPS

## SYNCHRONOUS INPUT AND OUTPUT

Symbol	Parameter	Min	Max	Units	Notes
$t_{CHQV}$	CLK high to Output Valid		40	ns	
$t_{CHQX}$	Output hold time after CLK	2		ns	
$t_{DVCH}$	Input setup time to CLK high	8		ns	
$t_{CHDX}$	Input hold time to CLK high	0		ns	

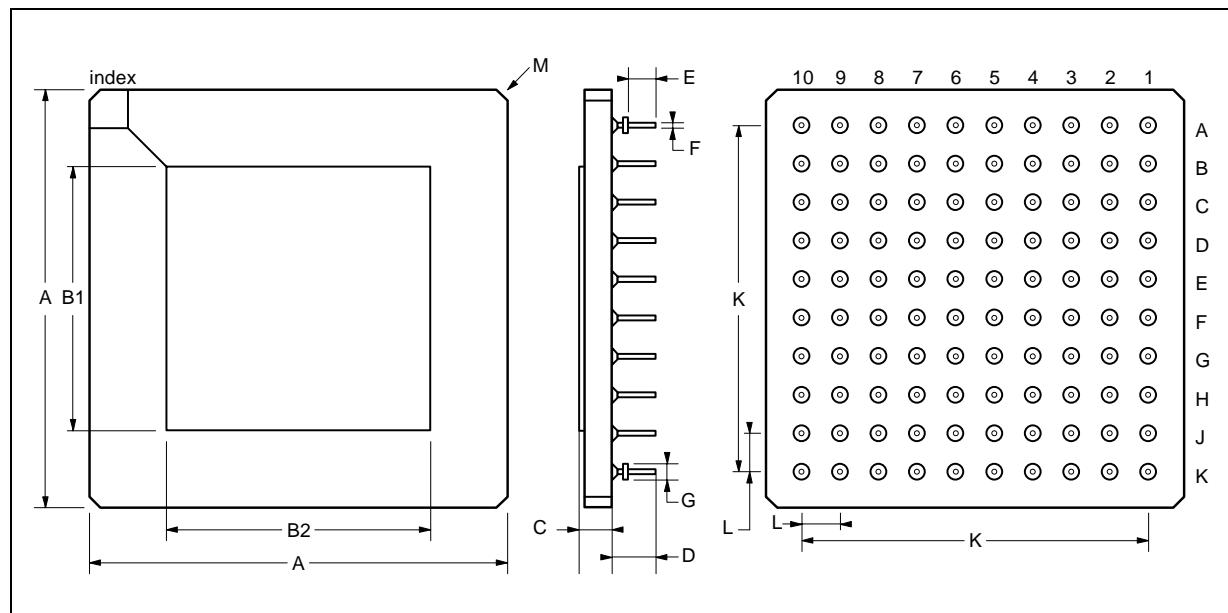
Figure 12



A110-12.EPS

## IMSA110

### PACKAGE MECHANICAL DATA 100 PINS - GRID ARRAY PACKAGE



PMFGA100.EPS

DIM	Millimetres		Inches		Notes
	Nom	Tol	Nom	Tol	
A	26.924	$\pm 0.254$	1.060	$\pm 0.010$	
B1	17.019	$\pm 0.127$	0.670	$\pm 0.005$	
B2	18.796	$\pm 0.127$	0.740	$\pm 0.005$	
C	2.456	$\pm 0.278$	0.097	$\pm 0.011$	
D	4.572	$\pm 0.127$	0.180	$\pm 0.005$	
E	3.302	$\pm 0.127$	0.130	$\pm 0.005$	
F	0.457	$\pm 0.051$	0.018	$\pm 0.002$	Pin diameter
G	1.143	$\pm 0.127$	0.045	$\pm 0.005$	Flange diameter
K	22.860	$\pm 0.127$	0.900	$\pm 0.005$	
L	2.540	$\pm 0.127$	0.100	$\pm 0.005$	
M	0.508		0.020		Chamfer

PGA100.TBL

Information furnished is believed to be accurate and reliable. However, SGS-THOMSON Microelectronics assumes no responsibility for the consequences of use of such information nor for any infringement of patents or other rights of third parties which may result from its use. No licence is granted by implication or otherwise under any patent or patent rights of SGS-THOMSON Microelectronics. Specifications mentioned in this publication are subject to change without notice. This publication supersedes and replaces all information previously supplied. SGS-THOMSON Microelectronics products are not authorized for use as critical components in life support devices or systems without express written approval of SGS-THOMSON Microelectronics.

© 1994 SGS-THOMSON Microelectronics - All Rights Reserved

Purchase of I<sup>2</sup>C Components of SGS-THOMSON Microelectronics, conveys a license under the Philips I<sup>2</sup>C Patent. Rights to use these components in a I<sup>2</sup>C system, is granted provided that the system conforms to the I<sup>2</sup>C Standard Specifications as defined by Philips.

SGS-THOMSON Microelectronics GROUP OF COMPANIES

Australia - Brazil - China - France - Germany - Hong Kong - Italy - Japan - Korea - Malaysia - Malta - Morocco  
The Netherlands - Singapore - Spain - Sweden - Switzerland - Taiwan - Thailand - United Kingdom - U.S.A.