5.5

VAC068A Register Map and Descriptions

Base address for the VAC068A register set is $FFFD 00xx. Register size is up to 16 bits wide and accesses are acknowledged by using DSACK1*. The 16-bit registers are NOT byte accessible. For single-byte registers, the unused bits are read as 1s. Register values are listed in *Table 5-1*.

Table 5-1. Register Values

Local Address	Register List	Size
FFFD 00xx	SLSEL1* Address Mask Register	16 bits
FFFD 01xx	SLSEL1* Base Address Register	16 bits
FFFD 02xx	SLSEL0* Address Mask Register	16 bits
FFFD 03xx	SLSEL0* Base Address Register	16 bits
FFFD 04xx	ICFSEL* Base Address Register	16 bits
FFFD 05xx	DRAM Upper-Limit Mask Register	16 bits
FFFD 06xx	Boundary 2 Address Register	16 bits
FFFD 07xx	Boundary 3 Address Register	16 bits
FFFD 08xx	A24 Base Address Register	13 bits
FFFD 09xx	Region 1 Attribute Register	6 bits
FFFD 0Axx	Region 2 Attribute Register	6 bits
FFFD 0Bxx	Region 3 Attribute Register	6 bits
FFFD 0Cxx	IOSEL4* DSACK Control Register	16 bits
FFFD 0Dxx	IOSEL5* DSACK Control Register	16 bits
FFFD 0Exx	SHRCS* DSACK Control Register	16 bits
FFFD 0Fxx	EPROMCS* DSACK Control Register	16 bits
FFFD 10xx	IOSEL0* DSACK Control Register	16 bits
FFFD 11xx	IOSEL1* DSACK Control Register	16 bits
FFFD 12xx	IOSEL2* DSACK Control Register	16 bits
FFFD 13xx	IOSEL3* DSACK Control Register	16 bits
FFFD 14xx	Decode Control Register	16 bits
FFFD 15xx	Interrupt Status Register	8 bits
FFFD 16xx	Interrupt Control Register	16 bits
FFFD 17xx	Device Location Register	6 bits

Table 5-1. Register Values (continued)

Local Address	Register List	Size
FFFD 18xx	PIO Data Out Register	14 bits
FFFD 19xx	PIO Pin Register	14 bits
FFFD 1Axx	PIO Direction Register	15 bits
FFFD 1Bxx	PIO Function Register	16 bits
FFFD 1Cxx	CPU Clock Divisor Register	8 bits
FFFD 1Dxx	UART Channel A Mode Register	12 bits
FFFD 1Exx	UART Channel A Transmit Data Register	8 bits
FFFD 1Fxx	UART Channel B Mode Register	12 bits
FFFD 20xx	UART Channel A Receiver FIFO	11 bits
FFFD 21xx	UART Channel B Receiver FIFO	11 bits
FFFD 22xx	UART Channel B Transmit Register	8 bits
FFFD 23xx	UART Channel A Interrupt Mask Register	6 bits
FFFD 24xx	UART Channel B Interrupt Mask Register	6 bits
FFFD 25xx	UART Channel A Interrupt Status Register	8 bits
FFFD 26xx	UART Channel B Interrupt Status Register	8 bits
FFFD 27xx	Timer Data Register	16 bits
FFFD 28xx	Timer Control Register	8 bits
FFFD 29xx	VAC068A ID Register	16 bits

The base address location for the VAC068A register set is $FFFD 00xx. All VAC068A registers are cleared during a global reset and remain intact during a soft reset. Only interrupts are masked during a soft reset. Unused or reserved bits may read as a 0 or a 1. The VAC068A ID register remains intact through all resets.

The VAC068A registers are accessed from the local address/data signals and acknowledged as a 16-bit access by DSACK1* assertion. They may only be accessed as a 16-bit word. CACHINH* is asserted during accesses to the VAC068A registers. The VAC068A Identification register must be written after reset to enable VAC068A operation.

SLSEL1* Address Mask Register

Local Address	$FFFD 00xx
Bits 31:16	A set bit in any of the positions enables a comparison of the correspondingly numbered local and VMEbus address bits for the purpose of asserting SLSEL1*.

SLSEL1* Base Address Register

Local Address	$FFFD 01xx
Bits 31:16	The contents of this register are compared under a bitwise mask compare to both the local and VMEbus address bits for the purpose of asserting SLSEL1*.

SLSEL0* Address Mask Register

Local Address	$FFFD 02xx
Bits 31:16	A set bit in any of the positions enables a comparison of the correspondingly numbered local and VMEbus address bits for the purpose of asserting SLSEL0*.

SLSEL0* Base Address Register

Local Address	$FFFD 03xx
Bits 31:16	The contents of this register are compared under a bitwise mask compare to both the local and VMEbus address bits for the purpose of asserting SLSEL0*.

ICFSEL* Base Address Register

Local Address	$FFFD 04xx
Bits 31:24	The upper half of this register is compared to VMEbus address [15:8] for the purpose of asserting ICFSEL*. When a match occurs between register bits [31:24] and the VMEbus address signals A[15:8], ICFSEL* is asserted.
Bits 23:16	The lower half of this register is compared to VMEbus address [15:8] for the purpose of asserting ICFSEL*. When a match occurs between register bits [23:16] and VME address signals A[15:8], ICFSEL* is asserted.

When either bits A[31:24] or A[23:16] match the VMEbus address bits A[15:8], ICFSEL* is asserted. These two different 8-bit compares are used for asserting ICFSEL* to the VIC068A for differentiating between module-based or global functions.

DRAM Upper-Limit Mask Register (Boundary 1)

Local Address	$FFFD 05xx

This register contains an address mask used to specify the upper address limit of the DRAM memory area located in region 0. The local address bits LA[31:16] are logically ANDed with the NOT of their respective bits in this register and, if all AND outputs are Low, DRAMCS* is asserted. A simpler way to view this is if any 1 is present on the address bus where a corresponding 0 exists in the DRAMCS* Upper-Limit Mask register, DRAMCS* is not asserted.

The logic function used for this compare is a masking operation rather than a full magnitude compare. It is advised that only exact binary multiples be specified for the DRAM memory

size (i.e., 1 Mbyte, 2 Mbytes, 4 Mbytes, etc.) to avoid having holes in the local address map. While it is not necessary to fully populate a defined area, DRAMCS* will be asserted if an access is attempted to the area, even if there is no memory present.

This register is cleared up on power-up and on global resets. Following exit from the force EPROM mode, accesses to $0000 00xx will assert DRAMCS* even if no value has been loaded into the mask register.

Boundary 2 Address Register

Local Address $FFFD 06xx

This register contains the lower address limit for region 2 and the upper address limit for region 1. Its contents are compared to local address bits LA[31:16]. If the address is less than the value of this register, and neither DRAM nor A24 space (configured in the A24 Base Address register) access occurs, the access is valid for this region.

Boundary 3 Address Register

Local Address $FFFD 070xx

This register contains the upper address limit for region 2 and the lower limit for region 3. The upper address limit for region 3 is the EPROM address space ($FF00 0000). Its contents are compared to the local address bits LA[31:16]. If the address is less than the value in this register and neither DRAM or A24 space is being accessed, the access is valid for this region. If the address is greater than or equal to the value in this register yet less than EPROM space, a region 3 access occurs.

A24 Base Address Register

Local Address	$FFFD 08xx
Bits 31:25	These are compared to local address bits LA[31:25] for the purpose of overlaying an A24 address space in any one of the three regions described by their respective boundaries. Local access to this address space forces a master access to VMEbus A24 space. The area specified must be above the DRAM upper limit. *Note:* Valid values for bits [31:25] are greater than $02 and less than $FE
Bit 24	This bit is decoded along with bit 20 to determine the A24 data path size for the entire 32-Mbyte range. If bit 24 is cleared, a D16 data path is selected. If bit 24 is set, a D32 data path is selected. This bit is only interpreted if bit 20 is cleared.
Bit 23	When set, this bit selects A24 CACHINH* (cache inhibit). When cleared, no CACHINH* for A24 address space accesses.

A24 Base Address Register

Bit 22	When set, this bit enables bit 21 to determine the data path size of the A16 address space (region 6). When cleared, this bit enables the local address bit LA[16] to decode data path size. If LA[16] is High, a D16 data path is enabled (WORD* asserted). If LA16 is Low, a D32 data path is enabled (WORD* deasserted).
Bit 21	When set, this bit along with bit 22 causes region 6 (VMEbus A16 address space) to have a D32 data path (WORD* deasserted). When cleared, this bit causes region 6 to have a D16 data path (WORD* asserted).
Bit 20	When set, this bit enables the local address bit LA[24] to determine the data path size for A24 master accesses. When LA[24] is High, the data path size is D16. When LA[24] is Low, the data path is D32. When bit 20 is cleared, bit 24 decodes the data path size for the entire A24 address space.
Bit 19	When set, this bit enables CACHINH* on accesses to VIC068A register accesses, VAC068A register accesses, and any of the six IOSEL0-5 local I/O address areas. When cleared, CACHINH* is not asserted on access to these address spaces.

Region 1–3 Attribute Registers

Local Address	$FFFD 09xx Region 1 Attribute register.
Local Address	$FFFD 0Axx Region 2 Attribute register.
Local Address	$FFFD 0Bxx Region 3 Attribute register.
Bit 31	When set, this bit enables WORD* to be asserted upon access.
Bit 30	When set, this bit enables ASIZ1 to be driven Low upon access.
Bit 29	When set, this bit enables ASIZ0 to be driven Low upon access.
Bit 28	When set, this bit enables CACHINH* to be asserted upon access.
Bits 27:26	Follow this table:

Bit 27	*Bit 26*	*Mode*
0	0	Inactive
0	1	Shared resources chip select
1	0	VSB resource chip select
1	1	MWB select (VMEbus request)

DSACKi* Control Registers

Local Address	$FFFD 0Cxx – IOSEL4* DSACKi* Control register (I/O Select Address = $FFF8 0000 to $FFF9 FFFF).
Local Address	$FFFD 0Dxx – IOSEL5* DSACKi* Control register (I/O Select Address = $FFFA 0000 to $FFFB FFFF).
Local Address	$FFFD 0Exx – SHRCS* DSACKi* Control register (I/O Select Address is programmable).

DSACKi* Control Registers (continued)

Local Address	$FFFD 0Fxx – EPROMCS* DSACKi* Control register (I/O Select Address = $FF00 0000 to $FFEF FFFF).
Local Address	$FFFD 10xx – IOSEL0* DSACKi* Control register (I/O Select Address = $FFF0 0000 to $FFF1 FFFF).
Local Address	$FFFD 11xx – IOSEL1* DSACKi* Control register (I/O Select Address = $FFF2 0000 to $FFF3 FFFF).
Local Address	$FFFD 12xx – IOSEL2* DSACKi* Control register (I/O Select Address = $FFF4 0000 to $FFF5 FFFF).
Local Address	$FFFD 13xx – IOSEL3* DSACKi* Control register (I/O Select Address = $FFF6 0000 to $FFF7 FFFF).
Bits 31:29	These bits determine the delay from PAS* assertion to assertion of DSACKi* in CPUCLK cycles per the following table: 000 = 1 cycle 001 = 2 cycles . . 111 = 8 cycles
Bit 28	When set, this bit enables DSACK1*. When cleared, DSACK1* is inactive.
Bit 27	When set, this bit enables DSACK0*. When cleared, DSACK0* is inactive.
Bits 26:24	These bits determine the recovery time for IOSELi* in integer multiples of the CPUCLK as follows: 000 = 1 CPUCLK cycle 001 = 2 CPUCLK cycles . . 111 = 8 CPUCLK cycles The VAC068A recovery time (time between assertions of device select outputs) is controlled by two separate timers; one for even-numbered IOSEL5-0* device select outputs and one for odd-numbered IOSEL5–0* device select outputs. Because of the shared usage of these counters, the user must insure that an IOSEL5–0* access to a device that uses the same counter (odd or even IOSEL5–0* address) is not allowed to start (not issued by the local processor) until the previous access has been deasserted for the required number of CPUCLK cycles. These counters operate only on IOSEL5–0* accesses. It is assumed that accesses to EPROMCS* or SHRCS* do not require a recovery time and should set the value of these bits of their DSACKi* Control register to 000.
Bits 23:22	These bits determine the assertion delay for IORD* from PAS* in 1/2 CPUCLK cycles (i.e., 0.5, 1, 1.5, 2).
Bits 21:20	These bits determine the assertion delay for IOWR* from PAS* in 1/2 CPUCLK cycles (i.e., 0.5, 1, 1.5, 2).

Decode Control Register (continued)

Bit 25	When set, qualify SLSEL0* decode with VMEbus AS*. When cleared, no qualification.
Bit 24	When set, qualify SLSEL1* decode with VMEbus AS*. When cleared, no qualification.
Bit 23	When set, qualify ICFSEL* decode with VMEbus AS*. When cleared, no qualification.
Bit 22	When sets, qualify boundary decodes (except DRAM) with PAS*. When cleared, no qualification.
Bit 21	When set, acknowledge DRAM access as 32-bit port (both DSACK0/1 asserted). When cleared, three-state DSACK0/1*s on DRAMCS*.
Bit 20	When set, redirect SLSEL1* area accesses on local bus to local resource specified in bits 29:28. When cleared, no redirect for SLSEL1*.
Bit 19	When set, redirect SLSEL0* area accesses on local bus to DRAM. When disabled, no redirect.
Bits 18:17	Assertion delay for DSACKi* upon access to DRAM (assertion of DRAMCS*) in CPU clock cycles per the following table: 00 = 0 CPUCLK cycles 01 = 1 CPUCLK cycles 10 = 2 CPUCLK cycles 11 = 3 CPUCLK cycles
Bit 16	When set, FPUCS* is asserted on assertion of PAS*. When cleared, FPUCS* is asserted on CPUCLK.
Bit 31	When set, assert DSACKi* on slave accesses to DRAM during VIC068A local bus cycles (except DRAM Refresh). When cleared, three-state

Interrupt Status Register

Local Address	$FFFD 15xx
Bit 31	When set, this bit indicates that a PIO9 interrupt is pending.
Bit 30	When set, this bit indicates that a PIO8 interrupt is pending.
Bit 29	When set, this bit indicates that a PIO7 interrupt is pending.
Bit 28	When set, this bit indicates that a PIO4 interrupt is pending.
Bit 27	When set, this bit indicates that a mailbox interrupt is pending.
Bit 26	When set, this bit indicates that a timer interrupt is pending.
Bit 25	When set, this bit indicates that a UART A interrupt is pending.
Bit 24	When set, this bit indicates that a UART B interrupt is pending.
Bit 26 (Bit 26 below)	When set, compare VMEbus A[15:8] to SLSEL1* base address register [31:24]. When cleared, no compare. This is used to allocate SLSEL1* in

This register is read only. Bits of this register are cleared by accessing the interrupt control register and clearing the control bits for that particular interrupt.

Interrupt Control Register

Local Address	$FFFD 16xx
Bits 31:30	These bits specify the mapping of PIO9 interrupt to one of the three signals as detailed in the following table.
Bits 29:28	These bits specify the mapping of PIO8 interrupt to one of the three signals as detailed in the following table.
Bits 27:26	These bits specify the mapping of PIO7 interrupt to one of the three signals as detailed in the following table.
Bits 25:24	These bits specify the mapping of PIO4 interrupt to one of the three signals as detailed in the following table.
Bits 23:22	These bits specify the mapping of the mailbox interrupt to one of the three signals as detailed in the following table.
Bits 21:20	These bits specify the mapping of the UART A interrupt to one of the three signals as detailed in the following table.
Bits 19:18	These bits specify the mapping of the UART B interrupt to one of the three signals as detailed in the following table.
Bits 17:16	These bits specify the mapping of the timer interrupt to one of the three signals as detailed in the following table.

Odd bit	*Even bit*	*Function*
0	0	Disabled
0	1	Enable to PIO7
1	0	Enable to PIO10
1	1	Enable to PIO11

Note that each interrupt service routine should clear its interrupt's map bits momentarily in order to clear the interrupt request output. Each interrupt is active Low and edge-triggered except UART A and B, which are event triggered and hold until cleared by clearing the interrupt in the Int Mask register. An interrupt request output is asserted only if a falling edge on an interrupt request input occurs while its map bits are non-zero. PIO9 must be held Low for at least 2.8 ms in order to generate an interrupt request.

Device Location Register

Local Address	$FFFD 17xx
Bit 21	When set, IOSEL5* active on the ID bus.
Bit 20	When set, IOSEL4* active on the ID bus.
Bit 19	When set, IOSEL3* active on the ID bus.
Bit 18	When set, IOSEL2* active on the ID bus.
Bit 17	When set, IOSEL1* active on the ID bus.
Bit 16	When set, IOSEL0* active on the ID bus.

This register specifies mapping of the input/output select device on the ID bus. If any bit is set, it indicates that the corresponding device is located on ID[15:8]. This allows the VAC068A to control the internal Buffer and Latch on the ID bus when these devices are accessed. SWDEN* swaps the data from/to ID[31:24] to ID[15:8] and DDIR controls the data direction.

PIO Data Out Register

Local Address	$FFFD 18xx
Bit 29	PIO13 or LD[29] signal output value.
Bit 28	PIO12 or LD[28] signal output value.
Bit 27	PIO11 or LD[27] signal output value.
Bit 26	PIO10 or LD[26] signal output value.
Bit 25	PIO9 or LD[25] signal output value.
Bit 24	PIO8 or LD[24] signal output value.
Bit 23	PIO7 or LD[23] signal output value.
Bit 22	PIO6 or LD[22] signal output value.
Bit 21	PIO5 or LD[21] signal output value.
Bit 20	PIO4 or LD[20] signal output value.
Bit 19	PIO3 or LD[19] signal output value.
Bit 18	PIO2 or LD[18] signal output value.
Bit 17	PIO1 or LD[17] signal output value.
Bit 16	PIO0 or LD[16] signal output value.

This register is used for writing to the PIO signals [13:0] defined as outputs. PIO[13:0] correspond directly to LD[29:16]. When read, the value in the register is driven onto the local data bus LD[29:16]. When written, the value in the register is driven onto those PIO[13:0] signals defined as outputs in the PIO Function register. To set or clear a single PIO[13:0] bit, the register must be read and a logical AND or OR operation performed on the bit, then the value is written back into the register.

PIO Pin Register

Local Address	$FFFD 19xx
Bits 29:16	Reflect the status of PIO signals [13:0] respectively (i.e., bit 29 = PIO 13, etc.).

This register is read-only and reflects the instantaneous value on those PIO[13:0] signals configured as inputs. Reading this register takes the logic value at the PIO[13:0] signal and drives it onto the local data bus LD[29:16]. Writing to this register causes a DSACK1* assertion and has no effect on the contents of the register.

PIO Direction Register

Local Address $FFFD 1Axx

Bit 30 When set, this bit enables FCIACK* assertion upon access to $FFFF FFxx independent of the function codes. This is useful for interrupt acknowledge emulation for non-68K processors.

Bits 29:16 These bits correspond directly to PIO signals [13:0]. When set, the direction of the PIO signals are output from VAC068A. When cleared, the direction of the PIO signals [13:0] are input to VAC068A. These register bits have no effect if the corresponding PIO Function register bits ($FFFD 1Bxx) are set.

PIO Function Register

Local Address $FFFD 1Bxx

Bit 31 When set, this bit asserts FCIACK* upon access to IOSEL5* address space independent of the function codes. Also, access to IOSEL4* address space asserts FPUCS*. When cleared, access to IOSEL4* and IOSEL5* address space does not affect the FCIACK* and FPUCS* signals.

Bit 30 When set, this bit enables the debounce delay associated with PIO9 (i.e., 26.7-ms debounce circuit delay). See PIO9 Debounce delay description for further details. When cleared, the debounce delay is disabled.

Bits 29:16 These bits select whether the shared function of the PIO pins are enabled. If set, the signal is always an output and operates with the shared function per the following table. If cleared, the signals operate in the PIO mode.

Bit	*General Purpose*	*Shared Function*
29	PIO signal 13	IOSEL2* address range $FFF4 0000 select
28	PIO signal 12	Shared resources chip select output
27	PIO signal 11	Interrupt request pin 11 (output)
26	PIO signal 10	Interrupt request pin 10 (output)
25	PIO signal 9	IOSEL5* address range $FFFA 0000 select
24	PIO signal 8	IOSEL4* address range $FFF8 0000 select
23	PIO signal 7	Interrupt request pin 7 (output)
22	PIO signal 6	IOSEL3* address range $FFF6 0000 select
21	PIO signal 5	I/O write signal
20	PIO signal 4	I/O read signal
19	PIO signal 3	UART B receive data signal
18	PIO signal 2	UART B transmit data signal
17	PIO signal 1	UART A receive data signal
16	PIO signal 0	UART A transmit data signal

Interrupt request functions (bits 27, 26, and 23) are mapped in the Interrupt Control register ($FFFD 16xx).

CPU Clock Divisor Register

Local Address $FFFD 1Cxx

Bits 31:24 These bits set the 16X baud rate clock for use with the VAC068A UART. This register is loaded into an up-counter that continuously counts from the loaded value to $FF and reloads on the next clock.

The table below gives examples of some CPU clock frequencies and the respective divisor to generate a baud rate of 9600.

CPU Clock	*CPU Clock Divisor Register*
16 MHz	$96
16.67 MHz	$93
20 MHz	$7C
25 MHz	$5B
30 MHz	$3B
33 MHz	$27

Note: Baud rate = CPUCLK/(Divisor * 16)

UART Channel A and B Mode Register

Local Address $FFFD 1Dxx Channel A Mode register.

Local Address $FFFD 1Fxx Channel B Mode register.

Bit 31 When set, parity check and generate are enabled. When cleared, parity generate and check are disabled.

Bit 30 When set, even parity check and generate are enabled. When cleared, odd parity check and generate are enabled.

Bit 29 When set, 8 data bits per character are enabled. When cleared, 7 data bits per character.

Bit 28:26 These bits set the baud rate for both the transmitter and receiver. The highest baud rate is derived from the CPU Clock Divisor register. The subsequent baud rates are a division of 2 from the previous baud rate. An example follows:

111 = baud rate of 9600
110 = baud rate of 4800
.
.
000 = baud rate of 75

Bit 25 When set, it allows the character receiver to run. When cleared, the receiver is reset.

Bit 24 When set, it allows the character transmitter to run. When cleared, the transmitter is reset.

Bit 23 When set, it enables the transmitter. When cleared, the transmitter is disabled.

Bit 22 When set, it enables the receiver. When cleared, the receiver is disabled.

UART Channel A and B Mode Register (continued)

Bit 21	When set, a continuous break is sent. When cleared, break is disabled.
Bit 20	When set, this bit enables looping of the transmitter output to the receiver FIFO register. When cleared, looping is disabled.

UART Channel A and B Transmit Data Register

Local Address	$FFFD 1Exx Channel A Transmit Data register.
Local Address	$FFFD 22xx Channel B Transmit Data register.
Bits 31:24	These bits are loaded with data to be transmitted via the TXD* output when configured in the PIO Function register and enabled in the UART Mode register. Enable respective transmitters BEFORE loading these registers.

UART Channel A and B Receiver FIFO Register

Local Address	$FFFD 20xx Channel A Receiver FIFO register.
Local Address	$FFFD 21xx Channel B Receiver FIFO register.
Bit 26	When set, this bit indicates that a break error for this byte was detected; otherwise no break error.
Bit 25	When set, this bit indicates that a frame error for this byte was detected; otherwise no frame error.
Bit 24	When set, this bit indicates that a parity error for this byte was detected; otherwise no parity error.
Bits 23:16	Received characters.

The A and B Receiver FIFO registers are read-only.

UART Channel A and B Interrupt Mask Register

Local Address	$FFFD 23xx Channel A Interrupt Mask register.
Local Address	$FFFD 24xx Channel B Interrupt Mask register.
Bit 31	When set, enable interrupt on single character. When cleared, disable interrupt.
Bit 30	When set, enable interrupt on receiver FIFO full. When cleared, disable interrupt.
Bit 29	When set, enable interrupt on break change. When cleared, disable interrupt.
Bit 28	When set, enable interrupt on overrun, framing, or parity error. When cleared, disable interrupt.
Bit 27	When set, enable interrupt on transmitter ready. When cleared, disable interrupt.

UART Channel A and B Interrupt Mask Register (continued)

Bit 26	When set, enable interrupt on transmitter empty. When cleared, disable interrupt.

The pending interrupt must be disabled in this register, serviced, and then cleared in the Interrupt Control register.

UART Channel A and B Interrupt Status Register

Local Address	$FFFD 25xx Channel A Interrupt Status register.
Local Address	$FFFD 26xx Channel B Interrupt Status register.
Bit 31	When set, this bit indicates that an interrupt has occurred because a character in the receiver is ready to be read.
Bit 30	When set, this bit indicates that an interrupt has occurred because the receiver FIFO is full.
Bit 29	When set, this bit indicates that an interrupt has occurred because a break change was detected.
Bit 28	When set, this bit indicates that an interrupt has occurred because a parity error was detected.
Bit 27	When set, this bit indicates that an interrupt has occurred because a framing error was detected.
Bit 26	When set, this bit indicates that an interrupt has occurred because a overrun error was detected.
Bit 25	When set, this bit indicates that an interrupt has occurred because the transmitter is ready for another character.
Bit 24	When set, this bit indicates that an interrupt has occurred because the transmitter is empty.

This read-only register contains the interrupt status conditions causing the interrupt generated.

Timer Data Register

Local Address	$FFFD 27xx
Bits 31:16	This register contains the data for loading the VAC068A internal watchdog timer. This data is loaded into a 16-bit up-counter when RUN/LOAD is Low as well as under control of the reload circuitry when ONCE/CONTINUOUS is Low. When the contents of this register are read, the value of the timer is driven onto the data bus, not the value loaded into the register. The counter clock input is driven from the carry out of the prescale counter.

Note: Refer to the Timer Control register for more information on RUN/LOAD and ONCE/CONTINUOUS.

Timer Control Register

Local Address	$FFFD 28xx
Bit 31	ONCE/CONTINUOUS: When cleared, the timer counts continuous and interrupt at the end of expiration. If this bit is set, the timer counts once and stops.
Bit 30	RUN/LOAD: When set, the count is enabled and dependent on bit 31 for control of count cycles. When cleared, the counter is disabled.
Bits 29:24	Prescale Load Value: These bits are loaded into the prescale counter. Bits 29 through 24 correspond directly to D5 through D0 respectively. The upper two bits of the prescale output (D6, D7) are tied High and not displayed in the register. The prescale counter carry out clocks the count value loaded into the Timer Data register.
Bits 23:16	Prescaler Value: These bits are read-only. They contain the instantaneous value of the prescale counter. Bits 23 through 16 correspond directly to the prescaler counter output Q7 through Q0 respectively.

VAC068A Identification Register

Local Address	$FFFD 29xx
Bits 31:20	Constant: These bits are predefined and cannot be changed. A read or write to this register does not affect these bits.
Bits 19:16	Revision number: These bits contain the chip revision number. VAC068–F5 – 1AC0 VAC068A – 1AC1

Note: After a global reset and the completion of loading all other registers, this register must be written in order for the VAC068A to enable its decode and compare functions.