Welcome to E-XFL.COM

Understanding Embedded - DSP (Digital Signal Processors)

Embedded - DSP (Digital Signal Processors) are specialized microprocessors designed to perform complex mathematical computations on digital signals in real-time. Unlike general-purpose processors, DSPs are optimized for high-speed numeric processing tasks, making them ideal for applications that require efficient and precise manipulation of digital data. These processors are fundamental in converting and processing signals in various forms, including audio, video, and communication signals, ensuring that data is accurately interpreted and utilized in embedded systems.

Applications of Embedded - DSP (Digital Signal Processors)

Details	
Product Status	Active
Type	Fixed Point
Interface	Serial Port
Clock Rate	26MHz
Non-Volatile Memory	ROM (6kB)
On-Chip RAM	8kB
Voltage - I/O	5.00V
Voltage - Core	5.00V
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	144-LQFP
Supplier Device Package	144-LQFP (20x20)
Purchase URL	https://www.e-xfl.com/product-detail/analog-devices/admc401bstz

ANALOG-TO-DIGITAL CONVERTER

(V_{DD} = AV_{DD} = 5 V ± 5%, GND = AGND = 0 V, T_{AMB} = –40°C to +85°C, CLKIN = 13 MHz, VIN0 to VIN7 = 4.0 V p-p, V_{REF} = 2.0 V, unless otherwise noted)

Parameter		Test Conditions	Min	Typ	Max	Unit
AC SPECIFICATIONS						
SNR	Signal to Noise Ratio	f_{IN} = 1.0 kHz	68	70		dB
SNRD	Signal to Noise and Distortion	f_{IN} = 1.0 kHz	66	69		dB
THD	Total Harmonic Distortion	f_{IN} = 1.0 kHz		–76	–70	dB
CTLK	Channel-Channel Crosstalk	f_{IN} = 1.0 kHz		–89	–72	dB
CMRR	Common-Mode Rejection Ratio			–90	–72	dB
PSRR	Power Supply Rejection Ratio			0.025	0.1	% FSR
ACCURACY						
INL	Integral Nonlinearity			±0.6	±1.5	LSB
DNL	Differential Nonlinearity			±0.5	±1.0	LSB
	No Missing Codes			12		Bits Guaranteed
	Zero Error			0.1	0.25	% FSR
	Gain Error[1]			0.4	1.0	% FSR
TEMPERATURE DRIFT						
	Zero Error			0.025		% FSR
	Gain Error[1]			0.025		% FSR
INPUT VOLTAGE						
V_{IN}	Voltage Span				4.0	V p-p
C_{IN}	Input Capacitance[2]			10		pF
CONVERSION TIME						
t_{CONV}	Total Conversion Time	All 8 Channels			1.88	µs

NOTES
[1]Excludes Internal Voltage Reference Error.
[2]Analog Input Pins VIN0 to VIN7.

Typical values are neither tested nor guaranteed.
Specifications subject to change without notice.

VOLTAGE REFERENCE

(V_{DD} = AV_{DD} = 5 V ± 5%, GND = AGND = 0 V, T_{AMB} = –40°C to +85°C, CLKIN = 13 MHz, VIN0 to VIN7 = 4.0 V p-p, V_{REF} = 2.0 V, unless otherwise noted)

Parameter		Test Conditions	Min	Typ	Max	Unit
V_{REF}	Output Voltage Reference	SENSE = REFCOM	1.96	2.0	2.04	V
	Output Voltage Tolerance[1]	SENSE = REFCOM		6		mV
	Output Current			1.0		mA
	Load Regulation	1.0 mA Load Current		0.3	1.5	mV
	Power Supply Rejection Ratio			0.1	1.5	mV
	Reference Input Resistance			8		kΩ

NOTES
[1]Relative tolerance due to temperature change, T_{MIN} to T_{MAX}.

Specifications subject to change without notice.

POWER-ON RESET

(GND = AGND = 0 V, T_{AMB} = –40°C to +85°C, CLKIN = 13 MHz, unless otherwise noted)

Parameter		Test Conditions	Min	Typ	Max	Unit
V_{RST}	Reset Threshold Voltage		3.25		4.0	V
V_{HYST}	Hysteresis Voltage			75		mV

Specifications subject to change without notice.

ABSOLUTE MAXIMUM RATINGS*

Supply Voltage . . . . . –0.3 V to +7 V
Input Voltage . . . . . –0.3 V to V_{DD} + 0.3 V
Output Voltage Swing . . . . . –0.3 V to V_{DD} + 0.3 V
Operating Temperature Range (Ambient) . . . . –40°C to +85°C
Storage Temperature Range . . . . . –65°C to +150°C
Lead Temperature (5 sec) . . . . . +280°C

*Stresses above those listed under absolute maximum ratings may cause permanent damage to the device. These are stresses only; functional operation of the device at these or any other conditions above those indicated in the operational section of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

ORDERING GUIDE

Model	Temperature Range	Instruction Rate	Package Description	Package Option
ADMC401BST	–40°C to +85°C	26 MHz	144-Lead Plastic Thin Quad Flatpack (LQFP)	ST-144
ADMC401-ADVEVALKIT			Development Tool Kit	
ADMC401-PB			Evaluation/Processor Board	

CAUTION

ESD (electrostatic discharge) sensitive device. Electrostatic charges as high as 4000 V readily accumulate on the human body and test equipment and can discharge without detection. Although the ADMC401 features proprietary ESD protection circuitry, permanent damage may occur on devices subjected to high-energy electrostatic discharges. Therefore, proper ESD precautions are recommended to avoid performance degradation or loss of functionality.

Timing Parameters

GENERAL NOTES

Use the exact timing information given. Do not attempt to derive parameters from the addition or subtraction of others. While addition or subtraction would yield meaningful results for an individual device, the values given in this data sheet reflect statistical variations and worst cases. Consequently, you cannot meaningfully add up parameters to derive longer times.

TIMING NOTES

Switching characteristics specify how the processor changes its signals. You have no control over this timing; it is dependent on the internal design. Timing requirements apply to signals that are controlled outside the processor, such as the data input for a read operation.

Timing requirements guarantee that the processor operates correctly with another device. Switching characteristics tell you what the device will do under a given circumstance. Also, use the switching characteristics to ensure any timing requirement of a device connected to the processor (such as memory) is satisfied.

MEMORY REQUIREMENTS

This chart links common memory device specification names and ADMC401 timing parameters for your convenience.

Parameter Name	Function	Common Memory Device Specification Name
t_{ASW}	A0–A13, $\overline{DMS}$, $\overline{PMS}$ Setup before $\overline{WR}$ Low	Address Setup to Write Start
t_{AW}	A0–A13, $\overline{DMS}$, $\overline{PMS}$ before $\overline{WR}$ Deasserted	Address Setup to Write End
t_{WRA}	A0–A13, $\overline{DMS}$, $\overline{PMS}$ Hold after $\overline{WR}$ Deasserted	Address Hold Time
t_{DW}	Data Setup before $\overline{WR}$ High	Data Setup Time
t_{DH}	Data Hold after $\overline{WR}$ High	Data Hold Time
t_{RDD}	$\overline{RD}$ Low to Data Valid	$\overline{OE}$ to Data Valid
t_{AA}	A0–A13, $\overline{DMS}$, $\overline{PMS}$, $\overline{BMS}$ to Data Valid	Address Access Time

Parameter		Min	Max	Unit
Memory Read				
Timing Requirements:				
t_{RDD}	$\overline{RD}$ Low to Data Valid		$0.5t_{CK} - 11 + w$	ns
t_{AA}	A0–A13, $\overline{PMS}$, $\overline{DMS}$, $\overline{BMS}$ to Data Valid		$0.75t_{CK} - 12 + w$	ns
t_{RDH}	Data Hold from $\overline{RD}$ High	0		ns
Switching Characteristics:				
t_{RP}	$\overline{RD}$ Pulsewidth	$0.5t_{CK} - 5 + w$		ns
t_{CRD}	CLKOUT High to $\overline{RD}$ Low	$0.25t_{CK} - 5$	$0.25t_{CK} + 7$	ns
t_{ASR}	A0–A13, $\overline{PMS}$, $\overline{DMS}$, $\overline{BMS}$ Setup before $\overline{RD}$ Low	$0.25t_{CK} - 6$		ns
t_{RDA}	A0–A13, $\overline{PMS}$, $\overline{DMS}$, $\overline{BMS}$ Hold after $\overline{RD}$ Deasserted	$0.25t_{CK} - 3$		ns
t_{RWR}	$\overline{RD}$ High to $\overline{RD}$ or $\overline{WR}$ Low	$0.5t_{CK} - 5$		ns

w = wait states × t_{CK}.

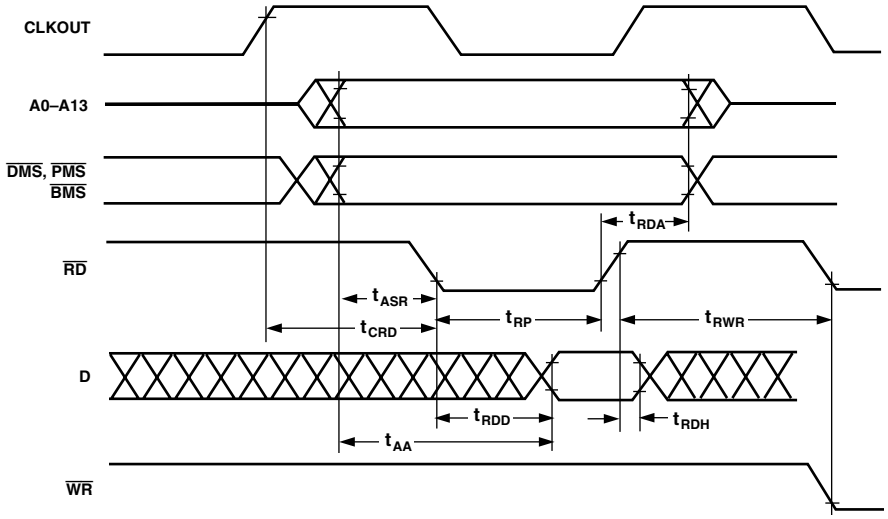


Figure 4. Memory Read

Parameter		Min	Max	Unit
Serial Ports				
Timing Requirements:				
t_{SCK}	SCLK Period	50		ns
t_{SCS}	DR/TFS/RFS Setup before SCLK Low	5		ns
t_{SCH}	DR/TFS/RFS Hold after SCLK Low	10		ns
t_{SCP}	$SCLK_{IN}$ Width	20		ns
Switching Characteristics:				
t_{CC}	CLKOUT High to $SCLK_{OUT}$	$0.25t_{CK}$	$0.25t_{CK} + 15$	ns
t_{SCDE}	SCLK High to DT Enable	0		ns
t_{SCDV}	SCLK High to DT Valid		20	ns
t_{RH}	TFS/RFS_{OUT} Hold after SCLK High	0		ns
t_{RD}	TFS/RFS_{OUT} Delay from SCLK High		20	ns
t_{SCDH}	DT Hold after SCLK High	0		ns
t_{TDE}	TFS(Alt) to DT Enable	0		ns
t_{TDV}	TFS(Alt) to DT Valid		20	ns
t_{SCDD}	SCLK High to DT Disable		20	ns
t_{RDV}	RFS (Multichannel, Frame Delay Zero) to DT Valid		20	ns

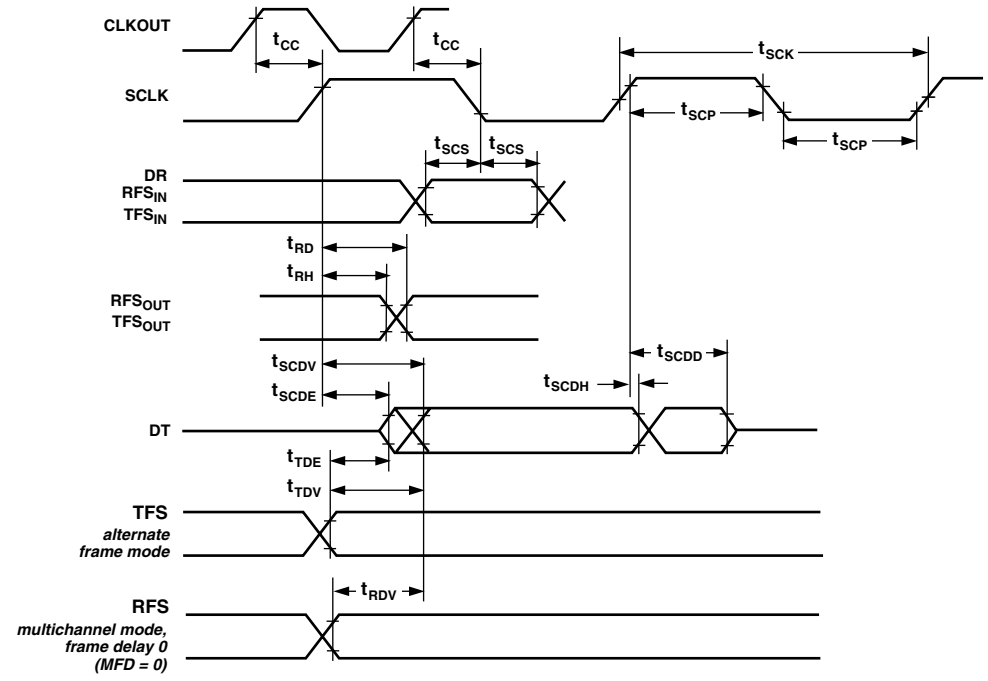
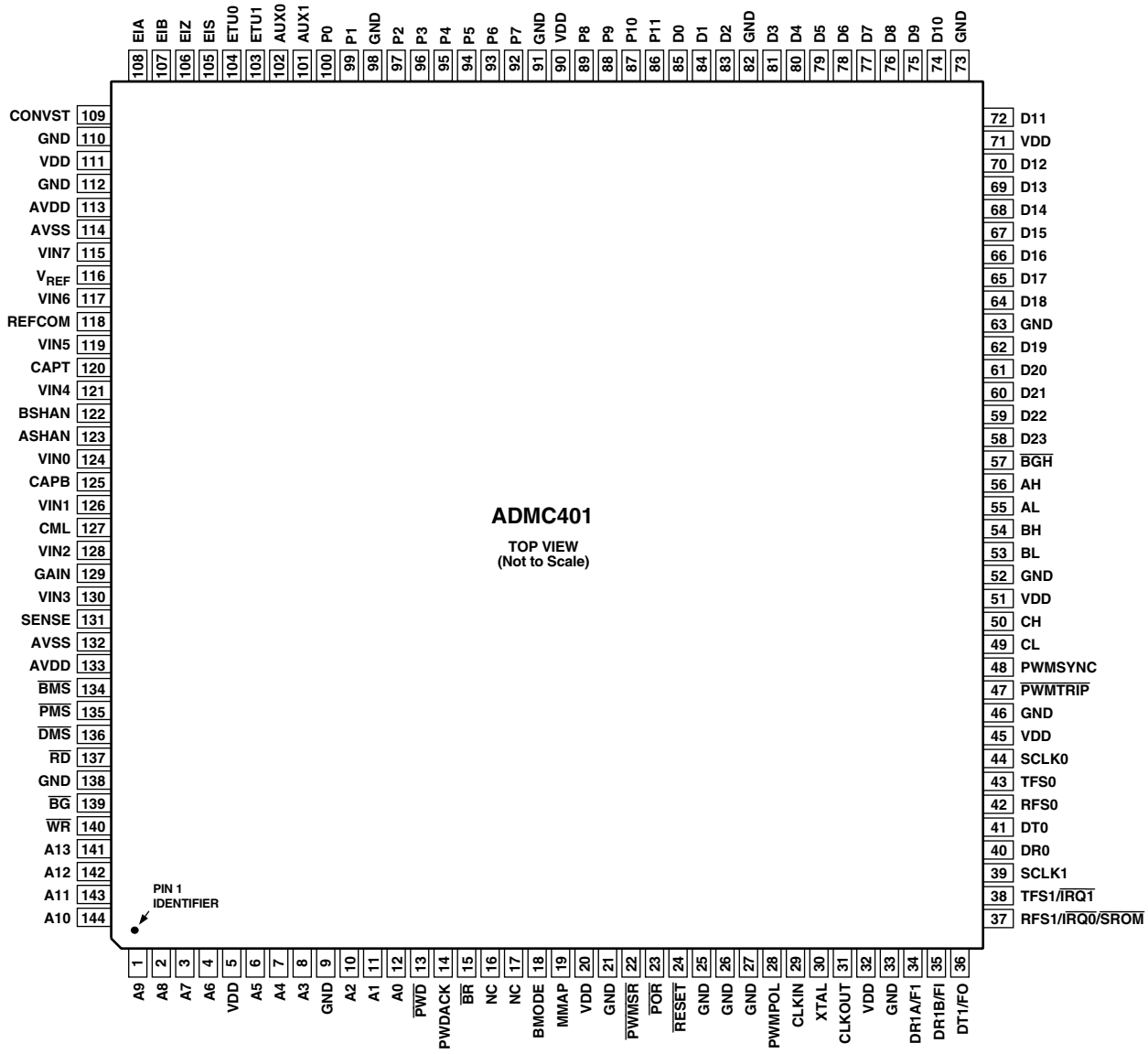


Figure 6. Serial Ports

PIN CONFIGURATION



NC = NO CONNECT

(Continued from Page 1)

- **Programmable Digital I/O (PIO) Port**
  - **12-Pin Configurable Digital I/O Port**
  - **Flexible Interrupt Generation**
  - **Four Dedicated PIO Interrupt Vectors**
  - **Each I/O Line Configurable as PWM Shutdown**
- **Two 8-Bit Auxiliary PWM Outputs**
  - **Programmable Switching Frequency**
  - **Independent or Offset Modes**
- **Two-Channel Event Timer (Capture) Unit**
  - **Configurable Event Definition**
  - **Single-Shot or Free-Running Modes**
- **Peripheral Interrupt Controller**
  - **Manages Peripheral Interrupts**
- **16-Bit Watchdog Timer**
- **Internal Power-On Reset System**
- **Programmable 16-Bit Interval Timer with Prescaler**
- **Two Double Buffered Synchronous Serial Ports**
  - **Boot Load Protocols via SPORT1:**
    - **Synchronous E^{2}PROM/SROM Booting**
    - **UART Boot Loader with Autobaud**
    - **Synchronous Master or Slave Boot Loader**
  - **Debugger Interface via SPORT1:**
    - **UART Interface with Autobaud**
    - **Synchronous Master or Slave Interface**
- **Full Debugger for Program Development**
- **Industrial Temperature Range –40°C to +85°C**
- **Operating Voltage 5.0 V ± 5%**
- **Package: 144-Lead LQFP**

GENERAL DESCRIPTION

The ADMC401 is a single-chip DSP-based controller, suitable for high performance control of ac induction motors (ACIM), permanent magnet synchronous motors (PMSM), brushless dc motors (BDCM) and switched reluctance (SR) motors in industrial applications. The ADMC401 integrates a 26 MIPS, fixed-point DSP core with a complete set of motor control peripherals that permits fast motor control in a highly integrated environment.

The DSP core of the ADMC401 is the ADSP-2171 which is completely code compatible with the ADSP-21xx DSP family (as well as other members of the integrated motor controllers of the ADMC3xx family) and combines three computational units, data address generators and a program sequencer. The computational units comprise an ALU, a multiplier/accumulator (MAC) and a barrel shifter. The DSP core also adds instructions for bit manipulation, squaring (x^2), biased rounding and global interrupt masking. In addition, two flexible double-buffered, bidirectional synchronous serial ports are included in the ADMC401.

The ADMC401 provides 2K × 24-bit internal program memory RAM, 2K × 24-bit internal program memory ROM and 1K × 16-bit internal data memory RAM. The program and data memory RAM can be boot loaded through the serial port from either a serial E^{2}PROM, through a UART connection (either from external host microprocessor or from the Motion Control Debugger) or via a synchronous serial interface from a host microprocessor. Alternatively, the internal program and data memory RAM may be booted from an external device across the address and data buses. The program memory ROM includes a monitor that adds software debugging features through the serial port.

Additionally, the ADMC401 device adds significant external memory and peripheral expansion capabilities by making available the full address and data bus of the DSP core. This feature permits expansion of both external program and data memory and means that the DSP core can address up to 14K × 24 bits of external program memory and up to 13K × 16 bits of external data memory.

The ADMC401 contains a number of special purpose, motor control peripherals. The first is a high performance, 8-channel, 12-bit ADC system with dual channel simultaneous sampling ability across 4 pair of inputs. An internal precision voltage reference is also available as part of the ADC system. In addition, a three-phase, 16-bit, center-based PWM generation unit can be used to produce high-accuracy PWM signals with minimal processor overhead. The ADMC401 also contains a flexible incremental encoder interface unit for position sensor feedback; two adjustable-frequency auxiliary PWM outputs, 12 lines of digital I/O; a 2-channel event capture system; a 16-bit watchdog timer; two 16-bit interval timers (one of which can be linked to the encoder interface unit) and an interrupt controller that manages all peripheral interrupts. Finally, the ADMC401 contains an integrated power-on-reset (POR) circuit that can be used to generate the required reset signal for the device on power-on.

PIN FUNCTION DESCRIPTION

The ADMC401 is available in an 144-lead TQFP package. Table I contains the pin descriptions.

Table I. Pin List

Pin Group Name	# of Pins	Input/ Output	Function
A13–A0	14	O	Address Lines
D23–D0	24	I/O	Data Lines
$\overline{PMS}$, $\overline{DMS}$, $\overline{BMS}$	3	O	External Memory Select Lines
$\overline{RD}$, $\overline{WR}$	2	O	External Memory Read/Write Enable
MMAP	1	I	Memory Map Select
$\overline{POR}$	1	O	Internal Power On Reset Output
$\overline{RESET}$	1	I	Processor Reset Input
CLKOUT	1	O	Processor Clock Output
CLKIN, XTAL	2	I, O	External Clock or Quartz Crystal Input
$\overline{BR}$	1	I	Bus Request
$\overline{BG}$, $\overline{BGH}$	2	O	Bus Grant and Bus Hang Control
BMODE	1	I	Boot Mode Select
$\overline{PWD}$, PWDACK	2	I, O	Power-Down and Power-Down Acknowledge
SPORT0	5	I/O	Serial Port 0 Pins (TFS0, RFS0, DT0, DR0, SCLK0)
SPORT1	6	I/O	Serial Port 1 (TFS1/$\overline{IRQ1}$, RFS1/ $\overline{IRQ0}$/$\overline{SROM}$, DT1/FO, DR1A/FI, DR1B/FI, SCLK1)
VIN0–VIN7	8	I	Analog Inputs
ASHAN, BSHAN	2	I	Inverting Inputs to Sample and Hold Amplifiers
GAIN	1	I	Analog Input for Gain Calibration
V_{REF}	1	I/O	Reference Voltage Input/Output
REFCOM	1	GND	Reference Common
CML	1	O	Common-Mode Level (Midsupply)
CAPT, CAPB	2	O	Noise Reduction Pins
SENSE	1	I	Voltage Reference Select
CONVST	1	I	External Convert Start
AH-CL	6	O	PWM Outputs
$\overline{PWMTRIP}$	1	I	PWM Shutdown Signal
PWMPOL	1	I	PWM Polarity Control
PWMSYNC	1	O	PWM Synchronization Output
$\overline{PWMSR}$	1	I	PWM Switched Reluctance Mode Control
PIO0–PIO11	12	I/O	Digital I/O Port
ETU0, ETU1	2	I	Event Timer Inputs
AUX0–AUX1	2	O	Auxiliary PWM Outputs
EIA, EIB, EIZ, EIS	4	I	Encoder Interface Inputs and External Registration Inputs
NC	2		No Connect
AVDD	2	SUP	Analog Power Supply
AVSS	2	GND	Analog Ground
VDD	8	SUP	Digital Power Supply
GND	16	GND	Digital Ground

INTERRUPT OVERVIEW

The ADMC401 can respond to different interrupt sources, some of which are internal DSP core interrupts and others from the motor control peripherals. The DSP core interrupts include a:

- Power up (or $\overline{RESET}$) interrupt.
- A peripheral (or $\overline{IRQ2}$) interrupt.
- A SPORT0 receive and a SPORT0 transmit interrupt.
- A SPORT1 receive (or $\overline{IRQ0}$) and a SPORT1 transmit (or $\overline{IRQ1}$) interrupt.
- Two software interrupts.
- An interval timer timeout interrupt.
- A power-down interrupt.

In addition, the motor control peripherals add other interrupts that include:

- A PWMSYNC interrupt.
- An ADC end of conversion interrupt.
- An encoder loop timer timeout interrupt.
- Five peripheral input/output (PIO) interrupts.
- An event timer interrupt.
- An encoder count error interrupt.
- A PWM trip interrupt.

The interrupts are internally prioritized and individually maskable except for the nonmaskable power-down interrupt.

Memory Map

The ADMC401 has two distinct memory types; program memory and data memory (in addition to external boot memory). In general, program memory contains user code and coefficients, while the data memory is used to store variables and data during program execution. Both program memory RAM and ROM is provided internally on the ADMC401. The program memory map of the ADMC401 can be altered depending on the state of the MMAP and BMODE pins. The various program memory maps are illustrated in Figure 11 for the permissible settings of MMAP and BMODE. The state of these pins also impact the way in which the internal memory of the ADMC401 is booted, as described later.

There is 2K of internal ROM on the ADMC401. Setting the ROMENABLE bit on the Data Memory Wait State Control Register (at address DM (0x3FFE)) enables the ROM. When the ROMENABLE bit is set to 1, addressing program memory in the ROM range will access the on-chip ROM. When ROMENABLE is set to zero, addressing program memory in this range will access external program memory. The ROMENABLE bit is initialized to zero after reset unless MMAP and BMODE = 1.

When MMAP = BMODE = 0, the ADMC401 provides 2K × 24 bits of internal program memory RAM starting at address 0x0000 that is booted from a byte-wide interface on the address and data buses. Following boot loading, program execution starts at address 0x0000. In this mode, the remainder of the program memory space, a 12K × 24-bit block starting at address 0x1000, is assigned to external memory.

When MMAP = BMODE = 1, the program memory map is identical to the previous case, but ROMENABLE defaults to 1 at reset, and execution starts from the internal program memory ROM located at address 0x0800. This permits the internal (and external if desired) memory to be boot loaded across the various serial interfaces on SPORT1.

The SENSE pin controls whether the A/D system operates with an internal or an external reference. For operation with the internal reference, the SENSE pin should be tied to the REFCOM pin. In this mode, the internally derived 2 V voltage reference appears at the V_{REF} pin. To operate with an external voltage reference, the SENSE pin should be tied to the AV_{DD} pin and the external voltage reference may be applied at the V_{REF} pin.

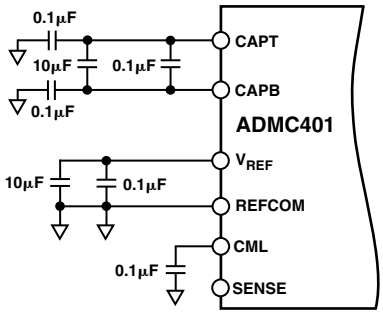


Figure 20. Recommended Capacitor Decoupling Networks for the ADMC401

OPTIMIZING ADC PERFORMANCE

The optimum noise and dc linearity performance is achieved with the largest input signal voltage span (i.e., 4 V input span) and with matching impedance in series with each of the analog inputs (VIN0 to VIN7, ASHAN and BSHAN). Additionally, the operational amplifier must exhibit source impedance that is both low and resistive, up to and beyond the sampling frequency. When a capacitive load is switched onto the output of the operational amplifier, the output will momentarily drop, due to its effective output impedance. As the output recovers, ringing may occur. To remedy this situation, a series resistor can be inserted between the op amp output and the ADC input (R_S as shown in Figure 18). Recommended configurations include using the OP27 amplifiers with an R_S of 20 Ω. Alternative recommended op amps are the AD8051 and AD8054.

Figure 18 shows ASHAN driven by the internally generated reference voltage at V_{REF}. When driving ASHAN with an internally generated V_{REF}, better performance will result if the driving impedance of ASHAN matches the driving impedance of the other analog inputs. This can be implemented with the addition of a second amplifier to Figure 18, between V_{REF} and ASHAN, to match the amplifier on VIN0.

For noise sensitive applications, it may also be beneficial to add some shunt capacitance between the inputs (VIN0 and ASHAN of Figure 18) and analog ground. Since this additional capacitance combines with the equivalent input capacitance of the analog inputs, a lower series resistance may be possible. The input RC combination also provides some antialiasing filtering on the analog inputs. To optimize performance when noise is the primary consideration, increase the shunt capacitance as much as the transient response of the input signal will allow. Increasing the capacitance too much may adversely affect the op amp's settling time, frequency response and distortion performance.

ADC REGISTERS

The configuration and structure of the ADC registers is described at the end of this data sheet.

THE PWM CONTROLLER

OVERVIEW

The PWM generator block of the ADMC401 is a flexible, programmable, three-phase PWM waveform generator that can be programmed to generate the required switching patterns to drive a three-phase voltage source inverter for ac induction (ACIM) or permanent magnet synchronous (PMSM) motor control. In addition, the PWM block contains special functions that considerably simplify the generation of the required PWM switching patterns for control of the electronically commutated motor (ECM) or brushless dc motor (BDCM). A special mode for switched reluctance motors (SRM) exists as well, enabled by a dedicated pin.

The PWM generator produces three pairs of PWM signals on the six PWM output pins (AH, AL, BH, BL, CH and CL). The six PWM output signals consist of three high side drive signals (AH, BH and CH) and three low side drive signals (AL, BL and CL). The polarity of the generated PWM signals may be programmed by the PWMPOL pin, so that either active HI or active LO PWM patterns can be produced by the ADMC401. The switching frequency, dead time and minimum pulsewidths of the generated PWM patterns are programmable using respectively, the PWMTM, PWMDT and PWMPD registers. In addition, three duty-cycle control registers (PWMCHA, PWMCHB and PWMCHC) directly control the duty cycles of the three pairs of PWM signals.

Each of the six PWM output signals can be enabled or disabled by separate output enable bits of the PWMSEG register. In addition, three control bits of the PWMSEG register permit crossover of the two signals of a PWM pair for easy control of ECM or BDCM. In crossover mode, the PWM signal destined for the high side switch is diverted to the complementary low-side output and the signal destined for the low side switch is diverted to the corresponding high side output signal. In addition to ease of use of the PWM controller for ECM or BDCM, this crossover mode can also be used to transition the PWM signals into the overmodulation range with relative ease.

In many applications, there is a need to provide an isolation barrier in the gate-drive circuits that turn on the power devices of the inverter. In general, there are two common isolation techniques, optical isolation using opto-isolators and transformer isolation using pulse transformers. The PWM controller of the ADMC401 permits mixing of the output PWM signals with a high-frequency chopping signal to permit easy interface to such pulse transformers. The features of this gate-drive chopping mode can be controlled by the PWMGATE register. There is an 8-bit value within the PWMGATE register that directly controls the chopping frequency. In addition, high frequency chopping can be independently enabled for the high side and the low side outputs using separate control bits in the PWMGATE register. Also, all PWM outputs have sufficient sink and source capability to directly drive most opto-isolators.

The PWM generator is capable of operating in two distinct modes, single update mode or double update mode. In single update mode the duty cycle values are programmable only once per PWM period, so that the resultant PWM patterns are symmetrical about the midpoint of the PWM period. In the double update mode, a second updating of the PWM registers is implemented at the midpoint of the PWM period. In this mode, it is possible to produce asymmetrical PWM patterns that produce

lower harmonic distortion in three-phase PWM inverters. This technique also permits closed loop controllers to change the average voltage applied to the machine windings at a faster rate and so permits faster closed loop bandwidths to be achieved. The operating mode of the PWM block (single or double update mode) is selected by a control bit in MODECTRL register.

The PWM generator of the ADMC401 also provides an output pulse on the PWMSYNC pin, which is synchronized to the PWM switching frequency. In single update mode a PWMSYNC pulse is produced at the start of each PWM period. In double update mode, an additional PWMSYNC pulse is produced at the midpoint of each PWM period. The width of the PWMSYNC pulse is programmable through the PWMSYNCWT register.

The PWM signals produced by the ADMC401 can be shut off in a number of different ways. First, there is a dedicated asynchronous PWM shutdown pin, $\overline{\text{PWMTRIP}}$, that, when brought LO, instantaneously places all six PWM outputs in the OFF state (as determined by the state of the PWMPOL pin). In addition, each of the PIO lines of the ADMC401 (PIO0 to PIO11) can be configured to act as an additional PWM shutdown. By setting the appropriate bit in the PIOPWM register, the corresponding PIO line acts as an asynchronous PWM shutdown source in a manner identical to the $\overline{\text{PWMTRIP}}$ pin. These two hardware shutdown mechanisms are asynchronous so that the associated PWM disable circuitry does not go through any clocked logic, thereby ensuring correct PWM shutdown even in the event of a loss of the DSP clock. In addition to the hardware shutdown features, the PWM system may be shut down in software by writing to the PWMSWT register.

Status information about the PWM system of the ADMC401 is available to the user in the SYSSTAT register. In particular, the state of the $\overline{\text{PWMTRIP}}$ and PWMPOL pins is available, as well as status bits that indicates whether operation is in the first half or the second half of the PWM period.

A functional block diagram of the PWM controller is shown in Figure 21. The generation of the six output PWM signals on pins AH to CL is controlled by four important blocks:

- The Three-Phase PWM Timing Unit, which is the core of the PWM controller, generates three pairs of complemented and dead time adjusted center based PWM signals.
- The Output Control Unit allows the redirection of the outputs of the Three-Phase Timing Unit for each channel to either the high side or the low side output. In addition, the Output Control Unit allows individual enabling/disabling of each of the six PWM output signals.
- The Gate Drive Unit provides the correct polarity output PWM signals based on the state of the PWMPOL pin. The Gate Drive Unit also permits the generation of the high-frequency chopping frequency and its subsequent mixing with the PWM signals.
- The PWM Shutdown Controller takes care of the various PWM shutdown modes (via the $\overline{\text{PWMTRIP}}$ pin, the PIO lines or the PWMSWT register) and generates the correct $\overline{\text{RESET}}$ signal for the Timing Unit.

The PWM controller is driven by a clock at the same frequency as the DSP instruction rate, t_{CK} and is capable of generating two interrupts to the DSP core. One interrupt is generated on the occurrence of a rising edge of the PWMSYNC pulse and the other is generated on the occurrence of any PWM shutdown action.

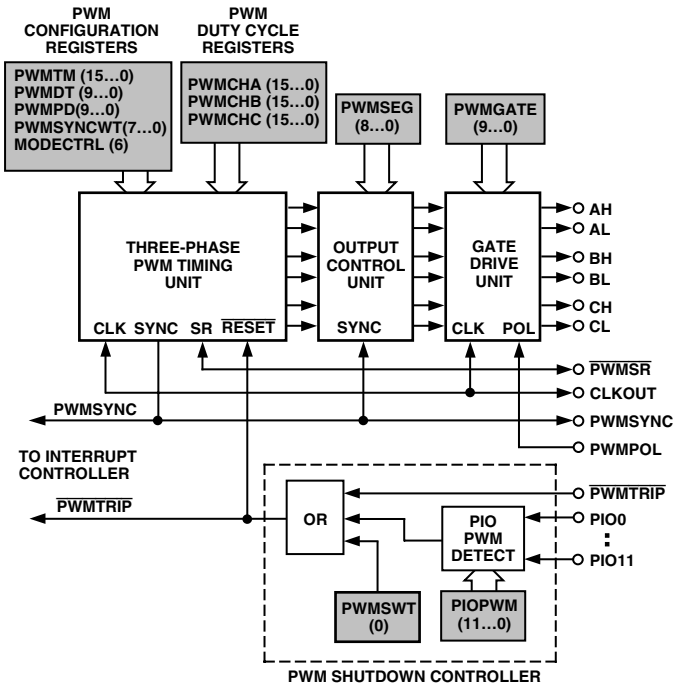


Figure 21. Overview of the ADMC401 PWM Controller

THREE-PHASE TIMING UNIT

The 16-bit three-phase timing unit is the core of the PWM controller and produces three pairs of pulsewidth modulated signals with high resolution and minimal processor overhead. The outputs of this timing unit are active LO such that a low level is interpreted as a command to turn ON the associated power device. There are four main configuration registers (PWMTM, PWMDT, PWMPD and PWMSYNCWT) that determine the fundamental characteristics of the PWM outputs. In addition, the operating mode of the PWM (single or double update mode) is selected by Bit 6 of the MODECTRL register. These registers, in conjunction with the three 16-bit duty cycle registers (PWMCHA, PWMCHB and PWMCHC), control the output of the three-phase timing unit.

PWM Switching Frequency, PWMTM Register

The PWM switching frequency is controlled by the 16-bit PWM period register, PWMTM. The fundamental timing unit of the PWM controller is t_{CK} (DSP instruction rate). Therefore, for a 26 MHz CLKOUT, the fundamental time increment is 38.5 ns. The value written to the PWMTM register is effectively the number of t_{CK} clock increments in half a PWM period. The required PWMTM value as a function of the desired PWM switching frequency (f_{PWM}) is given by:

$$PWMTM = \frac{f_{CLKOUT}}{2 \times f_{PWM}} = \frac{f_{CLKIN}}{f_{PWM}}$$

Therefore, the PWM switching period, T_s, can be written as:

$$T_S = 2 \times PWMTM \times t_{CK}$$

For example, for a 26 MHz CLKOUT and a desired PWM switching frequency of 10 kHz (T_S = 100 μs), the correct value to load into the PWMTM register is:

$$PWMTM = \frac{26 \times 10^6}{2 \times 10 \times 10^3} = 1300$$

The largest value that can be written to the 16-bit PWMTM register is 0xFFFF = 65,535, which corresponds to a minimum PWM switching frequency of:

$$f_{PWM,\,MIN} = \frac{26 \times 10^6}{2 \times 65{,}535} = 198\ Hz$$

PWM Switching Dead Time, PWMDT Register

The second important parameter that must be set up in the initial configuration of the PWM block is the switching dead time. This is a short delay time introduced between turning off one PWM signal (say AH) and turning on the complementary signal, AL. This short time delay is introduced to permit the power switch being turned off (AH in this case) to completely recover its blocking capability before the complementary switch is turned on. This time delay prevents a potentially destructive short-circuit condition from developing across the dc link capacitor of a typical voltage source inverter.

The dead time is controlled by the 10-bit PWMDT register. There is one dead time register that controls the dead time inserted into the three pairs of PWM output signals. The dead time, T_D, is related to the value in the PWMDT register by:

$$T_D = PWMDT \times 2 \times t_{CK}$$

Therefore, for a 26 MHz CLKOUT, a *PWMDT* value of 0x00A (= 10) introduces a 770 ns delay between the turn-off on any PWM signal (say AH) and the turn-on of its complementary signal (AL). The amount of the dead time can therefore be programmed in increments of $2t_{CK}$ (or 77 ns for a 26 MHz CLKOUT). The PWMDT register is a 10-bit register so that its maximum value is 0x3FF (= 1023) corresponding to a maximum programmed dead time of:

$$T_{D,\,MAX} = 1023 \times 2 \times t_{CK} = 1023 \times 2 \times 38.5 \times 10^{-9} = 78.8\ \mu s$$

for a CLKOUT rate of 26 MHz. Obviously, the dead time can be programmed to be zero by writing 0 to the PWMDT register.

PWM Operating Mode, MODECTRL and SYSSTAT Registers

The PWM controller of the ADMC401 can operate in two distinct modes; single update mode and double update mode. The operating mode of the PWM controller is determined by the state of Bit 6 of the MODECTRL register. If this bit is cleared, the PWM operates in the single update mode. Setting Bit 6 places the PWM in the double update mode. By default, following reset, Bit 6 of the MODECTRL register is cleared so that the default operating mode is in single update mode.

In single update mode, a single PWMSYNC pulse is produced in each PWM period. The rising edge of this signal marks the start of a new PWM cycle and is used to latch new values from the PWM configuration registers (PWMTM, PWMDT, PWMPD and PWMSYNCWT) and the PWM duty cycle registers (PWMCHA, PWMCHB and PWMCHC) into the three-phase timing unit. In addition, the PWMSEG register is also latched into the output control unit on the rising edge of the PWMSYNC pulse. In effect, this means that the characteristics and resultant duty cycles of the PWM signals can be updated only once per PWM period at the start of each cycle. The result is that PWM patterns that are symmetrical about the midpoint of the switching period are produced.

In double update mode, an additional PWMSYNC pulse is produced at the midpoint of each PWM period. The rising edge of this new PWMSYNC pulse is again used to latch new values of the PWM configuration registers, duty cycle registers and the PWMSEG register. As a result it is possible to alter the characteristics (switching frequency, dead time, minimum pulsewidth and PWMSYNC pulsewidth) as well as the output duty cycles at the midpoint of each PWM cycle. Consequently, it is possible with double update mode to produce PWM switching patterns that are not symmetrical about the midpoint of the period (asymmetrical PWM patterns).

In the double update mode, it may be necessary to know whether operation at any point in time is in either the first half or the second half of the PWM cycle. This information is provided by Bit 3 of the SYSSTAT register, which is cleared during operation in the first half of each PWM period (between the rising edge of the original PWMSYNC pulse and the rising edge of the new PWMSYNC pulse introduced in double update mode). Bit 3 of the SYSSTAT register is set during operation in the second half of each PWM period. This status bit allows the user to make a determination of the particular half-cycle during implementation of the PWMSYNC interrupt service routine, if required.

The advantage of the double update mode is that lower harmonic voltages can be produced by the PWM process and faster control bandwidths are possible. However, for a given PWM switching frequency, the PWMSYNC pulses occur at twice the rate in the double update mode. Since new duty cycle values must be computed in each PWMSYNC interrupt service routine, there is a larger computational burden on the DSP in the double update mode. Alternatively, the same PWM update rate may be maintained at half the switching frequency to give lower switching losses.

Width of the PWMSYNC Pulse, PWMSYNCWT Register

The PWM controller of the ADMC401 produces an output PWM synchronization pulse at a rate equal to the PWM switching frequency in single update mode and at twice the PWM frequency in the double update mode. This pulse is available for external use at the PWMSYNC pin. The width of this PWMSYNC pulse is programmable by the 8-bit read/write PWMSYNCWT register. The width of the PWMSYNC pulse, $T_{PWMSYNC}$, is given by:

$$T_{PWMSYNC} = t_{CK} \times \left(PWMSYNCWT + 1\right)$$

so that the width of the pulse is programmable from t_{CK} to 256 × t_{CK} (corresponding to 38.5 ns to 9.85 μs for a CLKOUT rate of 26 MHz). Following a reset, the PWMSYNCWT register contains 0x27 (= 39) so that the default PWMSYNC width is 1.54 μs, again for a 26 MHz CLKOUT.

PWM Duty Cycles, PWMCHA, PWMCHB, PWMCHC Registers

The duty cycles of the six PWM output signals on Pins AH to CL are controlled by the three 16-bit read/write duty cycle registers, PWMCHA, PWMCHB and PWMCHC. The integer value in the register PWMCHA controls the duty cycle of the signals on AH and AL, in PWMCHB controls the duty cycle of the signals on BH and BL and in PWMCHC controls the duty

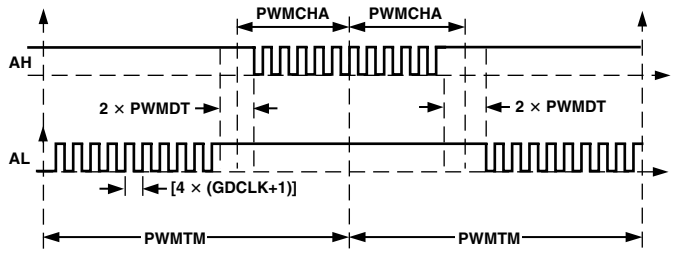


Figure 26. Typical active LO PWM signals with high frequency gate chopping enabled on both high side and low side switches.

PWM Polarity Control, PWMPOL Pin

The polarity of the PWM signals produced at the output pins AH to CL may be selected in hardware by the PWMPOL pin. Connecting the PWMPOL pin to DGND selects active LO PWM outputs, such that a LO level is interpreted as a command to turn on the associated power device. Conversely, connecting the PWMPOL pin to V_{DD} selects active HI PWM and the associated power devices are turned ON by a HI level at the PWM outputs. There is an internal pull-up on the PWMPOL pin, so that if this pin becomes disconnected (or is not connected), active HI PWM will be produced. The level on the PWMPOL pin may be read from Bit 2 of the SYSSTAT register, where a zero indicates a measured LO level at the PWMPOL pin.

SWITCHED RELUCTANCE MODE

The PWM block of the ADMC401 contains a switched reluctance (SR) mode that is controlled by the $\overline{\text{PWMSR}}$ pin. The switched reluctance mode is enabled by connecting the $\overline{\text{PWMSR}}$ pin to DGND. In this SR mode, the low side PWM signals from the three-phase timing unit assume permanently ON states, independent of the value written to the duty-cycle registers. The duty cycles of the high side PWM signals from the timing unit are still determined by the three duty cycle registers. Using the crossover feature of the output control unit, it is possible to divert the permanently ON PWM signals to either the high-side or low-side outputs. This mode is necessary because in the typical power converter configuration for switched or variable reluctance motors, the motor winding is connected between the two power switches of a given inverter leg. Therefore, in order to build up current in the motor winding, it is necessary to turn on both switches at the same time. Typical active LO PWM signals during operation in SR mode are shown in Figure 27 for operation in double update mode. It is clear that the three low-side signals (AL, BL and CL) are permanently ON and the three high side signals are modulated in the usual manner so that the corresponding high side power switches are switched between the ON and OFF states. The SR mode can *only* be enabled by connecting the $\overline{\text{PWMSR}}$ pin to GND. There is no software means by which this mode can be enabled. There is an internal pull-up resistor on the $\overline{\text{PWMSR}}$ pin so that if this pin is left unconnected or becomes disconnected the SR mode is disabled. Of course, the SR mode is disabled when the $\overline{\text{PWMSR}}$ pin is tied to V_{DD}.

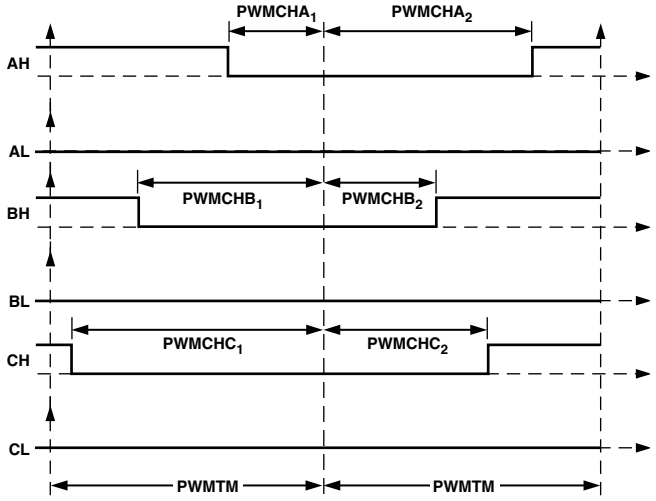


Figure 27. Active LO PWM signals in SR Mode (PWMPOL = $\overline{\text{PWMSR}}$ = DGND) for ADMC401 in double update mode.

PWM SHUTDOWN

In the event of external fault conditions, it is essential that the PWM system be instantaneously shutdown in a safe fashion. A low level on the $\overline{\text{PWMTRIP}}$ pin provides an instantaneous, asynchronous (independent of the DSP clock) shutdown of the PWM controller. All six PWM outputs are placed in the OFF state (as defined by the PWMPOL pin). *Note, however, when the $\overline{\text{PWMSR}}$ pin is in the SR mode, the three low side PWM signals from the three-phase timing unit will remain in the ON state.* In addition, the PWMSYNC pulse is disabled and the associated interrupt is stopped. The $\overline{\text{PWMTRIP}}$ pin has an internal pull-down resistor so that if the pin becomes unconnected the PWM will be disabled. The state of the $\overline{\text{PWMTRIP}}$ pin can be read from Bit 0 of the SYSSTAT register.

The 12 PIO lines of the ADMC401 can also be configured to operate as PWM shutdown pins using the PIOPWM register. The 12-bit PIOPWM has a control bit for each PIO line (Bit 0 controls PIO0, etc.). Setting the control bit enables the corresponding PIO line as a PWM shutdown pin. A falling edge on the PIO line will then generate an instantaneous, asynchronous shutdown of the PWM system, in a manner identical to the $\overline{\text{PWMTRIP}}$ pin. Also like $\overline{\text{PWMTRIP}}$, all of the PIO lines have internal pull-down resistors, so that if a PIO pin becomes unconnected and is configured as a PWM shutdown pin, the PWM will be disabled. Following a reset, all PIO lines are configured as inputs, have pull-downs and are programmed as PWM shut down pins (PIOPWM = 0x0FFF) so that the PWM is shutdown. Correct operation of the PWM is not possible without first correctly configuring the PIO system.

In addition, it is possible to initiate a PWM shutdown in software by writing to the 1-bit PWMSWT register. The act of writing to this register generates a PWM shutdown command in a manner identical to the $\overline{\text{PWMTRIP}}$ or PIO pins. A hardware trip has no effect on the PWMSWT register. It does not matter which value is written to the PWMSWT register. However, following a PWM shutdown, it is possible to read the PWMSWT register to determine if the shutdown was generated by hardware or software. If the PWM shutdown was caused by the PWMSWT register, a 1 will be read back from the PWMSWT register. Reading the PWMSWT register automatically clears its contents.

asynchronous timing of encoder and DSP-reading events. As a result, more accurate computations of the position and velocity of the motor shaft may be performed.

The EET consists of a 16-bit encoder event timer, an encoder pulse decimator and a clock divider. The EET clock frequency is selected by the 16-bit read/write EETDIV clock divide register, whose value divides the CLKOUT frequency. The contents of the encoder event timer are incremented on each rising edge of the divided clock signal. An EETDIV value of zero gives the maximum divide value of 0x10000 (= 65,536), so that the clock frequency to the encoder event timer is at its minimum possible value.

The quadrature signal from the encoder interface unit is decimated at a rate determined by the 8-bit read/write EETN register. For example, writing a value of two to EETN, produces a pulse decimator output train at half the quadrature signal frequency, as shown in Figure 31. The rising edge of this decimated signal is termed a velocity event. Therefore, for an EETN value of two, a velocity event occurs every two encoder edges, or on each edge of one of the encoder signals. An EETN value of 0 gives an effective pulse decimation value of 256.

On the occurrence of a velocity event, the contents of the encoder event timer are stored in an intermediate Interval Time Register. Under normal operation, this register stores the elapsed time between successive velocity events. After the timer value has been latched at the velocity event, the contents of the encoder event timer are reset to one.

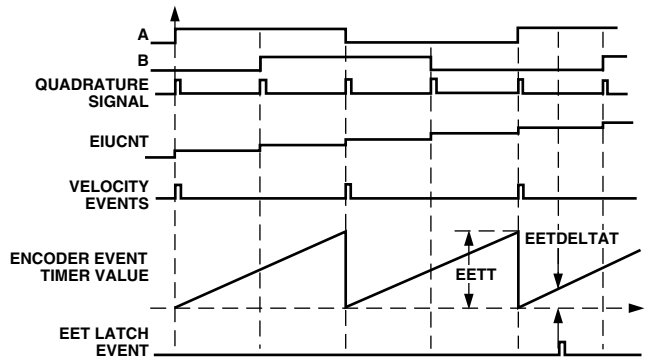


Figure 31. Operation of Encoder Interface Unit and EET of ADMC401 in the Forward Direction with EETN = 2

Latching Data from the EET

When using the data from the Encoder Event Timer, it is important to latch a triplet set of data at the same instant in time. The three pieces of data are the contents of the encoder quadrature up/down counter, the stored value in the Interval Time Register (giving the precise measured time between the last two velocity events) and the present value of the encoder event timer (giving an indication of how much time has passed since the last velocity event).

The data from the EET can be latched on the occurrence of two different events. The particular event is selected by Bit 4 (EETLATCH) of the EIUCTRL register. Setting this EETLATCH bit causes the data to be latched on the timeout of the encoder loop timer (EIUTIMER). At that time, the contents of the encoder quadrature counter (EIUCNT) are latched to a 16-bit register EETCNT. In addition, the contents of the intermediate Interval Time register are latched to the EETT register and the contents of the encoder event timer are latched to the EETDELTAT register. The three registers, EETCNT, EETT and EETDELTAT, then contain the desired triplet of position/speed data required for the control algorithm. In addition, if the timeout of the EIUTIMER is used to generate an EIU loop timer interrupt, the required data is automatically latched and waiting for execution of the interrupt service routine (which may be some time after the timeout instant if there are multiple interrupts in the system). By latching the EIUCNT register to the EETCNT, the user does not have to worry about changes in the EIUCNT register (due to additional encoder edges) prior to servicing of the EIU loop timer interrupt.

The other EET latch event is defined by clearing the EETLATCH bit of the EIUCTRL register. In this mode, whenever, the EIUCNT register is read by the DSP, the current value of the intermediate Interval Time register is latched to the EETT register and the contents of the encoder event timer are latched to the EETDELTAT register. The three registers, EIUCNT, EETT and EETDELTAT now contain the desired triplet of position/speed data required for the control algorithm. Note the difference from before, in that the encoder count value is now available in the EIUCNT register.

It is important to realize that the EETT, and EETDELTAT registers are only updated by either the timeout of the EIUTIMER register (if EETLATCH bit is set) or the act of reading the EIUCNT register (if the EETLATCH bit is cleared). Therefore, if the EETLATCH bit is set, the act of reading the EIUCNT register will not update the EETT and EETDELTAT registers. Following reset, Bit 4 of the EIUCTRL is cleared.

EET Status Register

There is a 1-bit EETSTAT register that indicates whether or not an overflow of the EET has occurred. If the time between successive velocity events is sufficiently long, it is possible that the encoder event timer will overflow. When this condition is detected, Bit 0 of the EETSTAT register is set and the EETT register is fixed at 0xFFFF. Reading the EETSTAT register clears the overflow bit and permits the EETT register to be updated at the next velocity event.

If an encoder direction reversal is detected by the EIU, the encoder event timer is set to 1 and the EETT register is set to its maximum 0xFFFF value. Subsequent velocity events will cause the EETT register to be updated with the correct value. If a value of 0xFFFF is read from the EETT register, Bit 0 of the EETSTAT register can be read to determine whether an overflow or direction reversal condition exists.

On reset the EETN, EETDIV, EETDELTAT and EETT registers are all cleared to zero. Whenever either the EETN or EETDIV registers are written to, the encoder event timer is reset to zero and the EETT register is set to zero.

edge on any of them will instantaneously shut down the PWM. However, based on the particular PIO interrupt that is flagged, the user can easily determine the source of the shutdown. This permits the action of the interrupt service routines following a PWM shutdown to be tailored to the particular fault that occurred.

On reset, all PIO lines are configured as PWM shutdown sources. Because all PIO lines are also configured as inputs and have internal pull-down resistors, any unconnected PIO lines will cause a PWM shutdown. Therefore, prior to using the PWM system of the ADMC401, it is imperative that the PIO stage be correctly configured for the particular application.

PIO REGISTERS

The configuration of all registers associated with the PIO system of the ADMC401 are shown at the end of the data sheet. Each of the registers has a bit directly associated with one of the PIO lines. For example, Bit 0 of all registers affects only the PIO0 line of the ADMC401.

EVENT TIMERS

OVERVIEW

The ADMC401 contains a dual channel event timer (capture) unit (ETU) that may be used to accurately measure the elapsed time between defined instants on a particular channel. The ETU has two dedicated input pins, ETU0 and ETU1. The ETU system contains a set of 16-bit data registers that are used to store the value of the dedicated ETU timer on the occurrence of defined events on the input pins. A configuration register is used to define the nature of the events on each of the input pins. In addition, a control register is used to initiate event capture on the inputs. A status register may be read to determine the state of the two capture channels. A dedicated ETU interrupt may be generated upon completion of a capture sequence on either the ETU0 or ETU1 channels. An event may be defined as either a rising or falling edge on the associated ETU0 and ETU1 input pins. Therefore, the ETU system can be used to compute the frequency, period, duty cycle or on-time of signals applied at the inputs. A block diagram of the ETU system of the ADMC401 is shown in Figure 32.

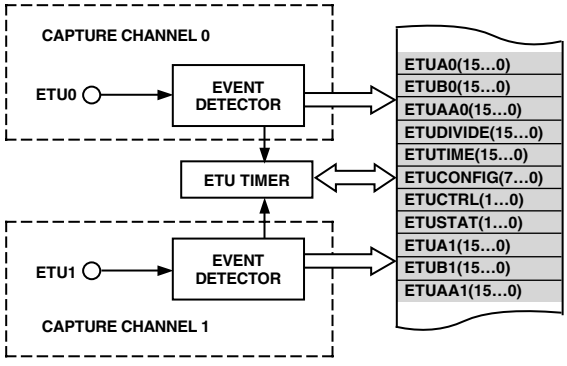


Figure 32. Functional Block Diagram of Event Timer Unit of ADMC401

ETU EVENT DEFINITION

The ETU system of the ADMC401 contains a dedicated 16-bit timer whose clock frequency may be programmed using the ETUDIVIDE register. This register divides the CLKOUT frequency to provide the clock signal for the ETU timer.

The clock frequency of the ETU timer may be expressed as f_{CLKOUT}/ETUDIVIDE and is common to both channels. At any time, the contents of the ETU timer may be read in the 16-bit read only ETUTIME register.

Two events are used to trigger the ETU, termed Event A and Event B. By setting the appropriate bits of the ETUCONFIG register, it is possible to define both events A and B as either rising or falling edges on the appropriate pin. For example, setting Bit 0 of the ETUCONFIG register defines Event A of the ETU0 channel as a rising edge on the ETU0 pin. Similarly, setting Bit 4 of the ETUCONFIG register defines Event A of the ETU1 channel as a rising edge on the ETU1 pin. Event A defines the start of the event capture sequence. Associated with each ETU channel are three data registers, ETUA0, ETUB0 and ETUAA0 for ETU Channel 0 and ETUA1, ETUB1 and ETUAA1 for ETU Channel 1. These data registers store the ETU timer value on the occurrence of the first A event, the first B event and the second A event, respectively. For example, for ETU Channel 0, ETUA0 stores the timer value on the first occurrence of Event A on the ETU0 pin, ETUB0 stores the timer value on the first occurrence of Event B on the ETU0 pin and ETUAA0 store the timer value on the second occurrence of Event A on the ETU0 pin. Registers ETUA1, ETUB1 and ETUAA1 perform the same function for events on ETU Channel 1.

ETU INTERRUPT GENERATION

The completion of the event capture sequence can be defined as either the occurrence of Event B or the second occurrence of Event A by setting the appropriate bits of the ETUCONFIG register. At the end of the capture sequence, the ETU generates an interrupt. For example, if Bit 2 of the ETUCONFIG register is set, ETU Channel 0 will generate an ETU interrupt on the occurrence of Event B on the ETU0 pin. On the other hand, if Bit 6 of the ETUCONFIG register is cleared, ETU Channel 1 will generate an ETU interrupt on the occurrence of the second Event A on the ETU1 pin. Both ETU channels generate the same interrupt to the DSP when capture is complete. If both ETU channels are used simultaneously, the ETUSTAT register can be polled to determine the status of both channels and determine which caused the interrupt. If capture on ETU Channel 0 is complete, Bit 0 of the ETUSTAT register is set. Similarly, if event capture on ETU Channel 1 is complete, Bit 1 of the ETUSTAT register is set. Reading the ETUSTAT register automatically clears all bits of the register.

ETU OPERATING MODES

The ETU channels of the ADMC401 can operate in two distinct modes; single shot and free-running. The particular mode may be selected for ETU Channel 0 by programming Bit 3 of the ETUCONFIG register and for ETU Channel 1 by programming Bit 7 of the ETUCONFIG register. Setting these bits puts the respective ETU channel in free-running mode while clearing the bits enables the single-shot mode. In single-shot mode, upon completion of the capture sequence and consequent generation of the interrupt, further event capture is disabled until the interrupt has been serviced and the appropriate bit of the ETUCTRL register has been set. Setting Bit 0 of the ETUCTRL register restarts the capture for ETU Channel 0, while Bit 1 restarts capture for Channel 1. In the free-running mode, the bits of the ETUCTRL register remain set and the ETU channel continues to capture following the generation of the interrupt.

Interrupt Configuration

The IFC and ICNTL registers of the DSP core control and configure the interrupt controller of the DSP core. The IFC register is a 16-bit register that may be used to force and/or clear any of the eight DSP interrupts. Bits 0 to 7 of the IFC register may be used to clear the DSP interrupts while Bits 8 to 15 can be used to force a corresponding interrupt. Writing to Bits 11 and 12 in IFC is the only way to create the two software interrupts.

The ICNTL register is used to configure the sensitivity (edge or level) of the $\overline{IRQ0}$, $\overline{IRQ1}$ and $\overline{IRQ2}$ interrupts and to enable/disable interrupt nesting. Setting Bit 0 of ICNTL configures the $\overline{IRQ0}$ as edge sensitive while clearing the bit configures it for level sensitive. Bit 1 is used to configure the $\overline{IRQ1}$ interrupt and Bit 2 is used to configure the $\overline{IRQ2}$ interrupt. It is recommended that the $\overline{IRQ2}$ interrupt be always configured for level sensitive as this ensures that no peripheral interrupts are lost. Setting Bit 4 of the ICNTL register enables interrupt nesting. The configuration of both IFC and ICNTL registers is shown at the end of the data sheet.

Interrupt Operation

Following a reset (with ROMENABLE = 1), the ROM code monitor of the ADMC401 copies a default interrupt vector table into program memory RAM from address 0x0000 to 0x005F. Since each interrupt source has a dedicated four word space in this vector table, it is possible to code short interrupt service routines (ISR) in place. Alternatively, it may be required to insert a JUMP instruction to the appropriate start address of the interrupt service routine if more memory is required for the ISR.

On the occurrence of an interrupt, the program sequencer ensures that there is no latency (beyond synchronization delay) when processing unmasked interrupts. In the case of the timer, SPORT0, SPORT1 and software interrupts, the interrupt controller automatically jumps to the appropriate location in the interrupt vector table. At this point, a JUMP instruction to the appropriate ISR is required.

In the event of a motor control peripheral interrupt, the operation is slightly different. For any of the eleven peripheral interrupts, the interrupt controller automatically jumps to location 0x0004 in the interrupt vector table. In addition, the required vector address (between 0x0030 and 0x0058) associated with the particular interrupt source is placed in the PICVECTOR register of the PIC block. Code loaded at location 0x0004 by the monitor on reset subsequently performs a JUMP from location 0x0004 to the address specified in the PICVECTOR register. This operation with the PICVECTOR register results in a slightly longer latency associated with processing any of the peripheral interrupts, as compared with the latency of the internal DSP core interrupts.

The code located at location 0x0004 by the monitor on reset is as follows:

```
0x0004: DM (I4_SAVE) = I4;
        I4 = DM (PICVECTOR);
        JUMP (I4);
```

The default code for each of the motor control peripherals is:

```
        I4 = DM (I4_SAVE);
        RTI;
```

Note that this default restores I4 to its value before the interrupt. The user should replace the RTI with a JUMP to their ISR. The PUT_VECTOR ROM subroutine can be used to replace the RTI with the JUMP.

The PIC block manages the sequencing of the eleven motor control peripheral interrupts. In the case of multiple simultaneous interrupts, the PIC will load the PICVECTOR register with the vector address of the highest priority pending interrupt. The contents of the PICVECTOR register will remain fixed until read by the DSP. This action is performed by the default DSP code at location 0x0004. The PIC block only asserts a new interrupt after the PICVECTOR register has been read. For other settings of MMAP and BMODE the user must correctly configure the vector table.

SYSTEM CONTROLLER

MODECTRL REGISTER

The MODECTRL register controls three important features of the ADMC401. It internally configures the SPORT1 pins for boot loading and UART debugging. Dedicated bits in the MODECTRL register also control the operating mode of the PWM generation unit (single or double update mode) and the operating mode of the auxiliary PWM generation unit (independent or offset mode).

Two bits of the MODECTRL register control the internal configuration of the SPORT1 pins as illustrated in Figure 34. Bit 4 (DR1SEL) selects which of the two external receive pins (DR1A or DR1B) is connected to the internal data receive port of the DSP core. Clearing Bit 4 selects the DR1A pin, whereas setting Bit 4 selects the DR1B pin. Following reset, Bit 4 is cleared so that DR1A is selected.

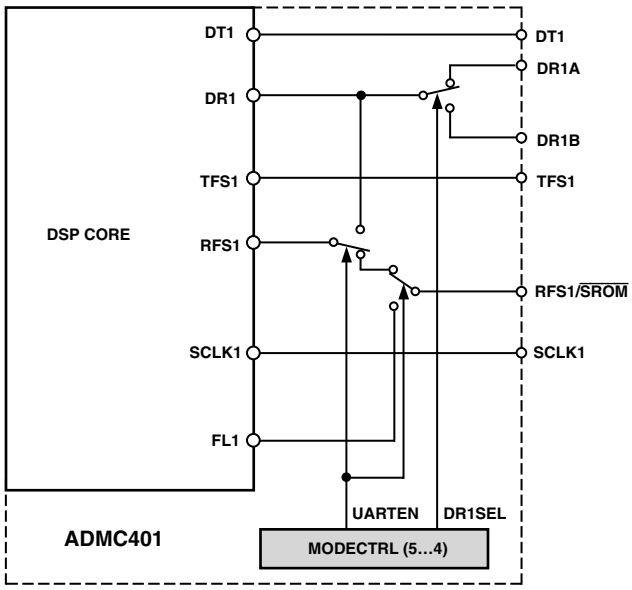


Figure 34. Internal Multiplexing of SPORT1 Pins

Bit 5 (UARTEN) of the MODECTRL register is used to select between UART and SPORT mode of SPORT1. Setting the UARTEN bit connects DR1A to the RFS1 input which allows SPORT1 to be used as a UART port. Additionally, the internal FL1 flag of the DSP core is connected to the RFS1/$\overline{SROM}$ pin of the ADMC401, to be used as a reset for the external serial

Table VIII. Peripheral Register Map of the ADMC401

Address	Name	Type	Bits	Reset Value	Function
0x2000–0x2007					Reserved
0x2008	PWMTM	R/W	[15 . . . 0]	0x0000	PWM Period Register
0x2009	PWMDT	R/W	[9 . . . 0]	0x0000	PWM Deadtime Register
0x200A	PWMPD	R/W	[9 . . . 0]	0x0000	PWM Pulse Deletion Register
0x200B	PWMGATE	R/W	[9 . . . 0]	0x000	PWM Chopping Control
0x200C	PWMCHA	R/W	[15 . . . 0]	0x0000	PWM Channel A Duty Cycle Control
0x200D	PWMCHB	R/W	[15 . . . 0]	0x0000	PWM Channel B Duty Cycle Control
0x200E	PWMCHC	R/W	[15 . . . 0]	0x0000	PWM Channel C Duty Cycle Control
0x200F	PWMSEG	R/W	[8 . . . 0]	0x000	PWM Crossover and Output Enable
0x2010	AUXCH0	R/W	[7 . . . 0]	0x00	Aux. PWM Channel 0 Duty Cycle
0x2011	AUXCH1	R/W	[7 . . . 0]	0x00	Aux. PWM Channel 1 Duty Cycle
0x2012	AUXTM0	R/W	[7 . . . 0]	0xFF	Aux. PWM Channel 0 Period
0x2013	AUXTM1	R/W	[7 . . . 0]	0xFF	Aux. PWM Channel 1 Period
0x2014					Reserved
0x2015	MODECTRL	R/W	[8, 6 . . . 4]	0x000	Mode Control Register
0x2016	SYSSTAT	R	[3 . . . 0]		System Status Register
0x2017					Reserved
0x2018	WDTIMER	R/W	[15 . . . 0]		Watchdog Timer Register
0x2019–0x201B					Reserved
0x201C	PICVECTOR	R	[15 . . . 0]		Peripheral Interrupt Address
0x201D	PICMASK	R/W	[10 . . . 0]	0x000	Peripheral Interrupt Mask Register
0x201E–0x201F					Reserved
0x2020	EIUCNT	R/W	[15 . . . 0]	0x0000	Position Count Value
0x2021	EIUMAXCNT	R/W	[15 . . . 0]	0x0000	Maximum EIUCNT Value
0x2022	EIUSTAT	R	[7 . . . 0]		EIU Status Register
0x2023	EIUCTRL	R/W	[8 . . . 0]	0x000	EIU Control Register
0x2024	EIUPERIOD	R/W	[15 . . . 0]	0x0000	EIU Loop Timer Period Register
0x2025	EIUSCALE	R/W	[7 . . . 0]	0x00	EIU Loop Timer Scale Register
0x2026	EIUTIMER	R/W	[15 . . . 0]	0x0000	EIU Loop Timer Register
0x2027	EETCNT	R	[15 . . . 0]	0x0000	Latched Copy of EIUCNT
0x2028	EIUFILTER	R/W	[5 . . . 0]	0x00	EIU Filter Control Register
0x2029	EIZLATCH	R	[15 . . . 0]		EIZ Latch Register
0x202A	EISLATCH	R	[15 . . . 0]		EIS Latch Register
0x202B–0x202F					Reserved
0x2030	ADC0	R	[15 . . . 0]		ADC0 Data Register
0x2031	ADC1	R	[15 . . . 0]		ADC1 Data Register
0x2032	ADC2	R	[15 . . . 0]		ADC2 Data Register
0x2033	ADC3	R	[15 . . . 0]		ADC3 Data Register
0x2034	ADC4	R	[15 . . . 0]		ADC4 Data Register
0x2035	ADC5	R	[15 . . . 0]		ADC5 Data Register
0x2036	ADC6	R	[15 . . . 0]		ADC6 Data Register
0x2037	ADC7	R	[15 . . . 0]		ADC7 Data Register
0x2038	ADCCTRL	R/W	[4 . . . 3,0]	0x00	ADC Control Register
0x2039	ADCSTAT	R	[4 . . . 0]		ADC Status Register
0x203A					Reserved
0x203B	ADCXTRA	R	[15 . . . 0]		Extra ADC Data Register
0x203C	ADCOTR	R	[7 . . . 0]		ADC Out of Range Register
0x203D–0x203F					Reserved
0x2040	PIOLEVEL	R/W	[11 . . . 0]	0x000	PIO Interrupt Select
0x2041	PIOMODE	R/W	[11 . . . 0]	0x000	PIO Interrupt Edge/Level Select
0x2042	PIOPWM	R/W	[11 . . . 0]	0xFFF	PIO PWMTRIP Enable Register
0x2043					Reserved
0x2044	PIODIR	R/W	[11 . . . 0]	0x000	PIO Direction Control
0x2045	PIODATA	R/W	[11 . . . 0]		PIO Data Register

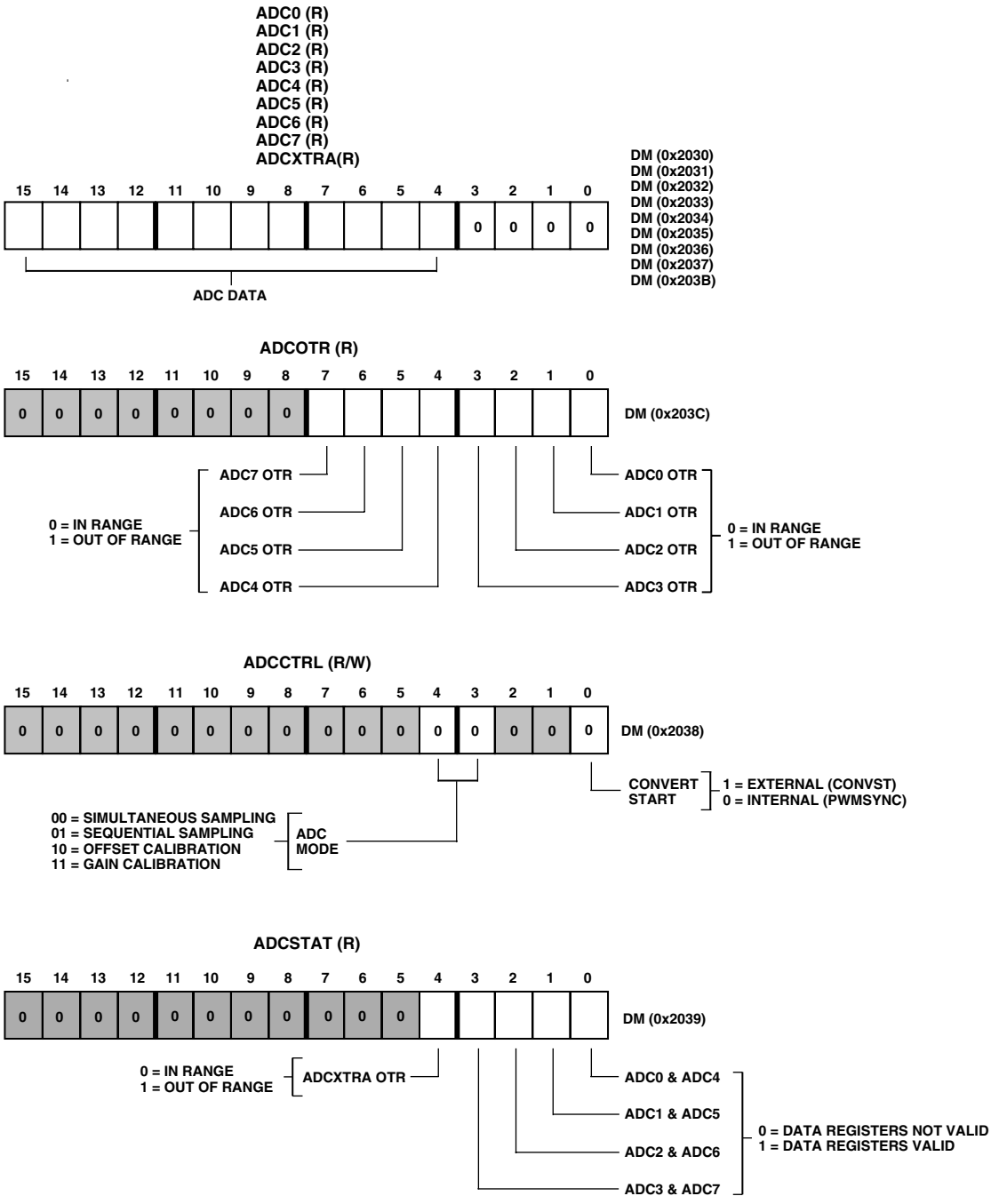


Figure 35. Structure of Registers of the ADMC401

Default bit values are shown; if no value is shown, the bit field is undefined at reset. Reserved bits are shown on a gray field—these bits should always be written as shown.

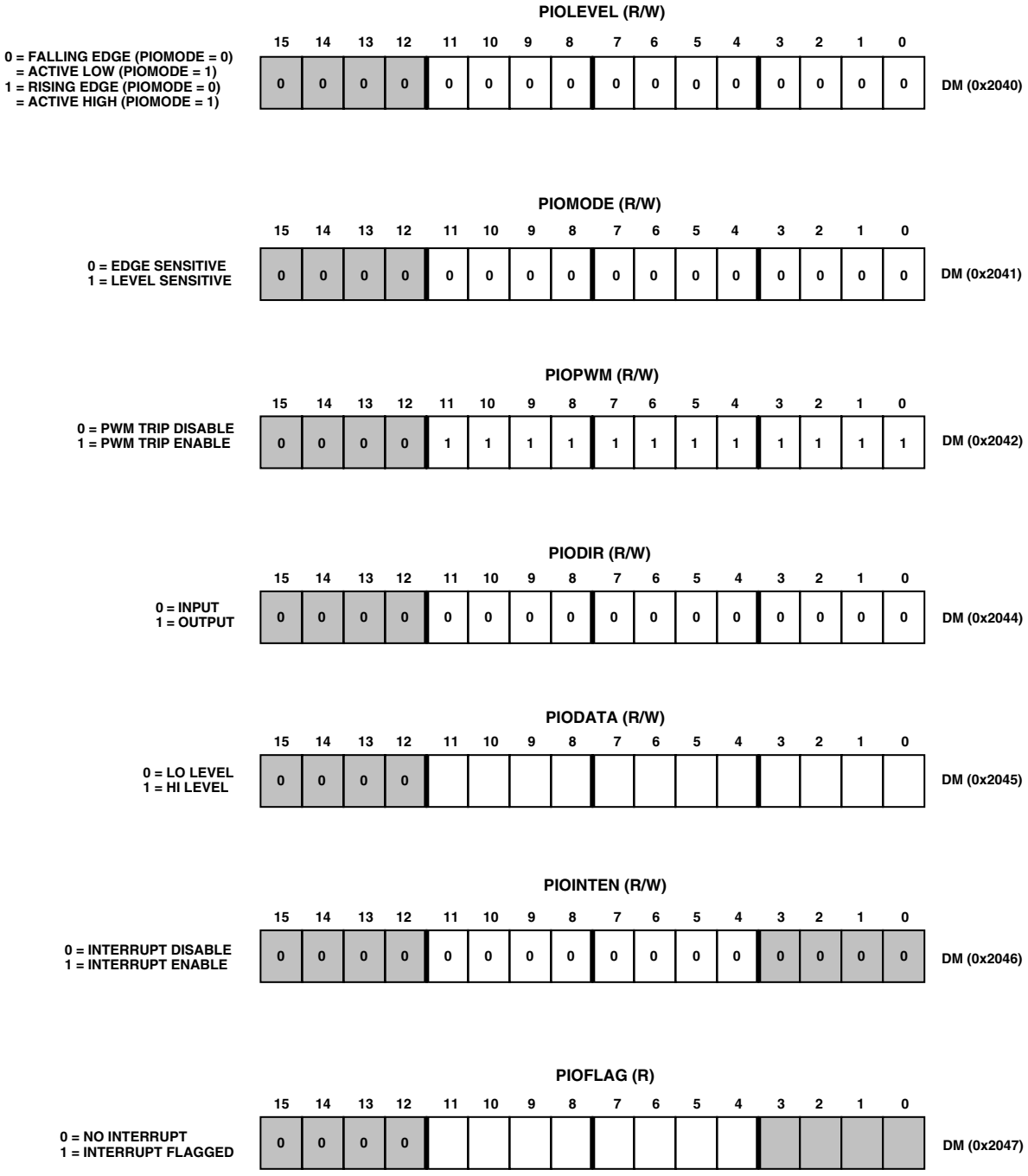


Figure 39. Structure of Registers of ADMC401

Default bit values are shown; if no value is shown, the bit field is undefined at reset. Reserved bits are shown on a gray field—these bits should always be written as shown.

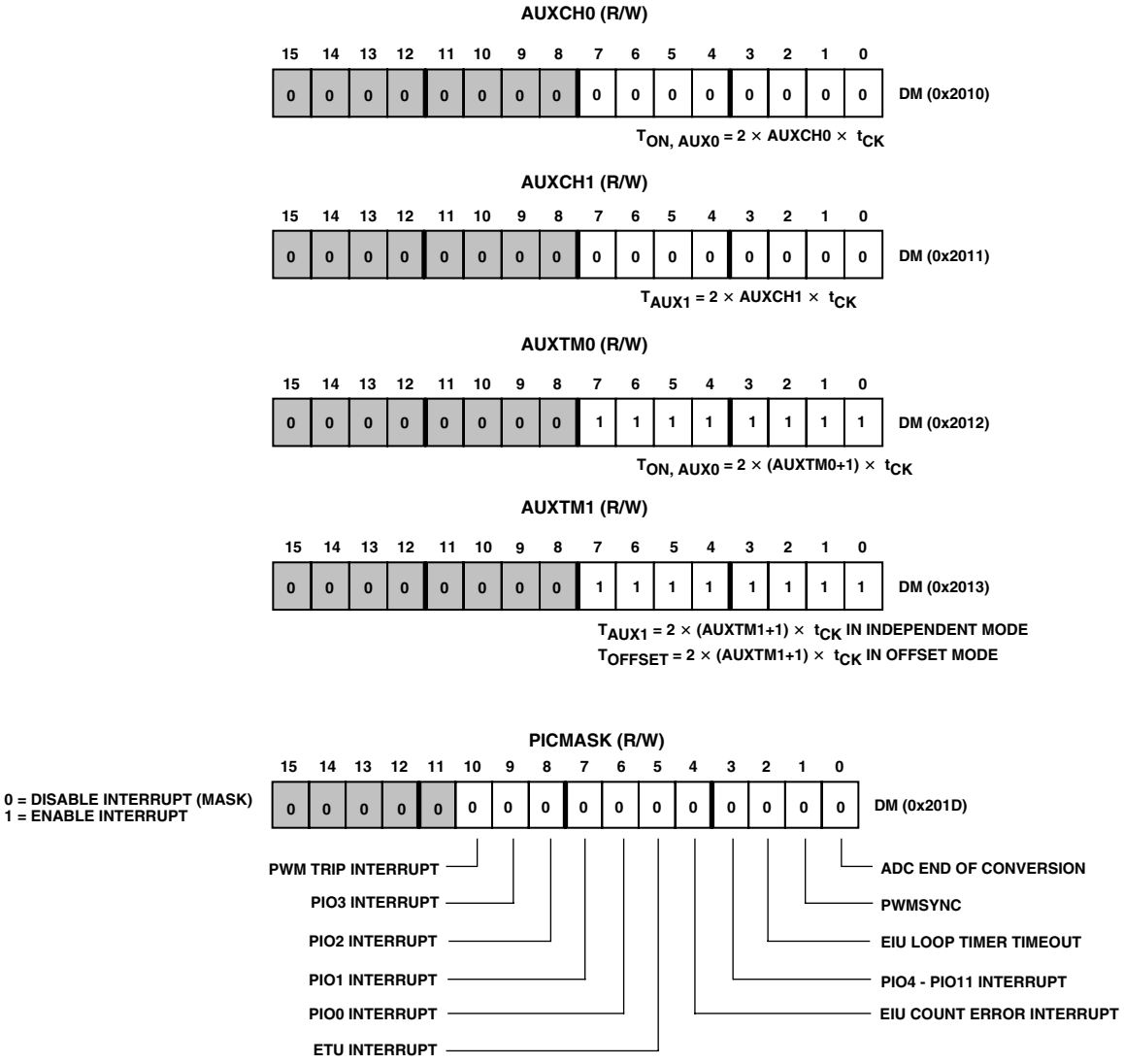


Figure 41. Structure of Registers of ADMC401

Default bit values are shown; if no value is shown, the bit field is undefined at reset. Reserved bits are shown on a gray field—these bits should always be written as shown.

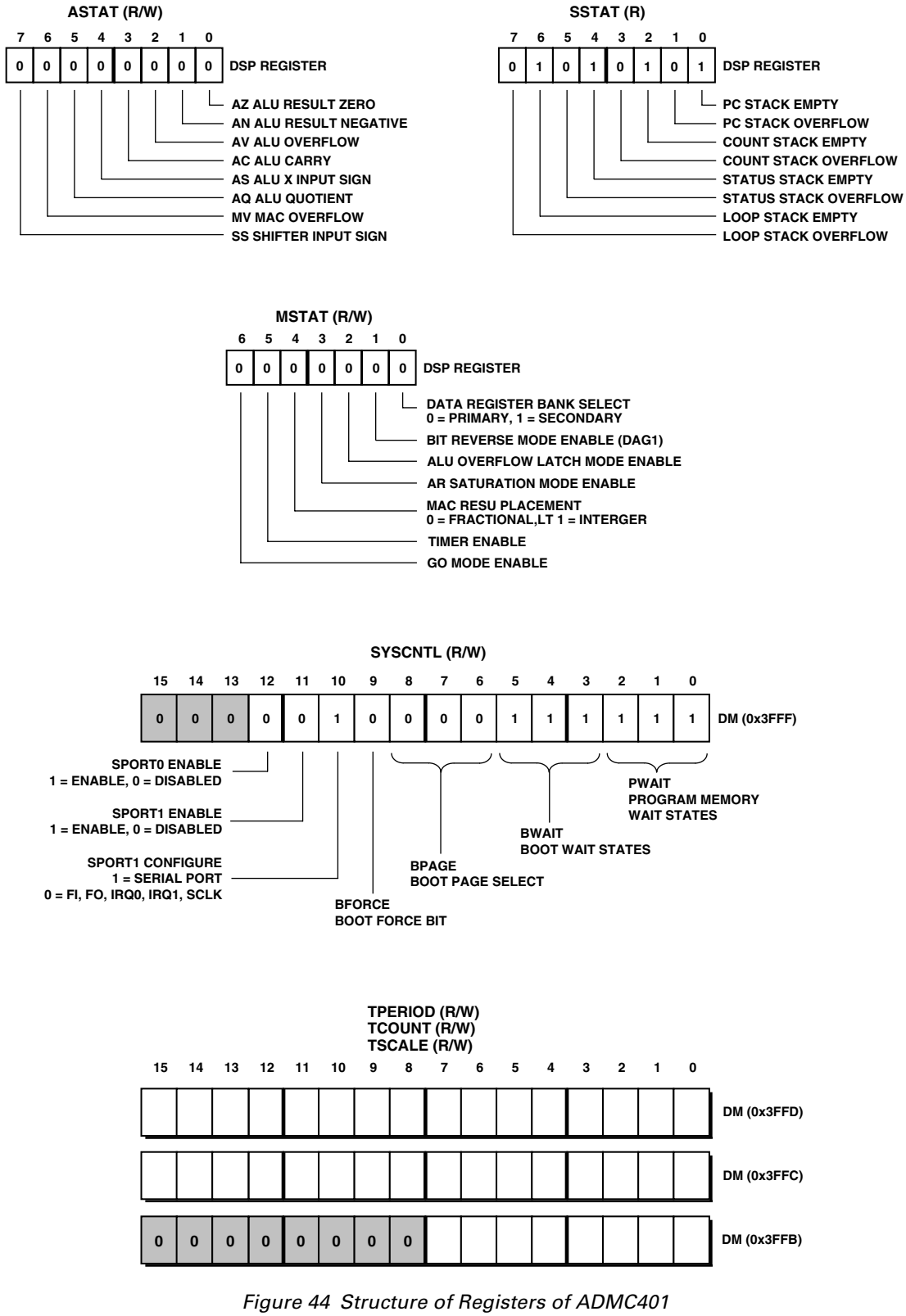


Figure 44 Structure of Registers of ADMC401

Default bit values are shown; if no value is shown, the bit field is undefined at reset. Reserved bits are shown on a gray field—these bits should always be written as shown.

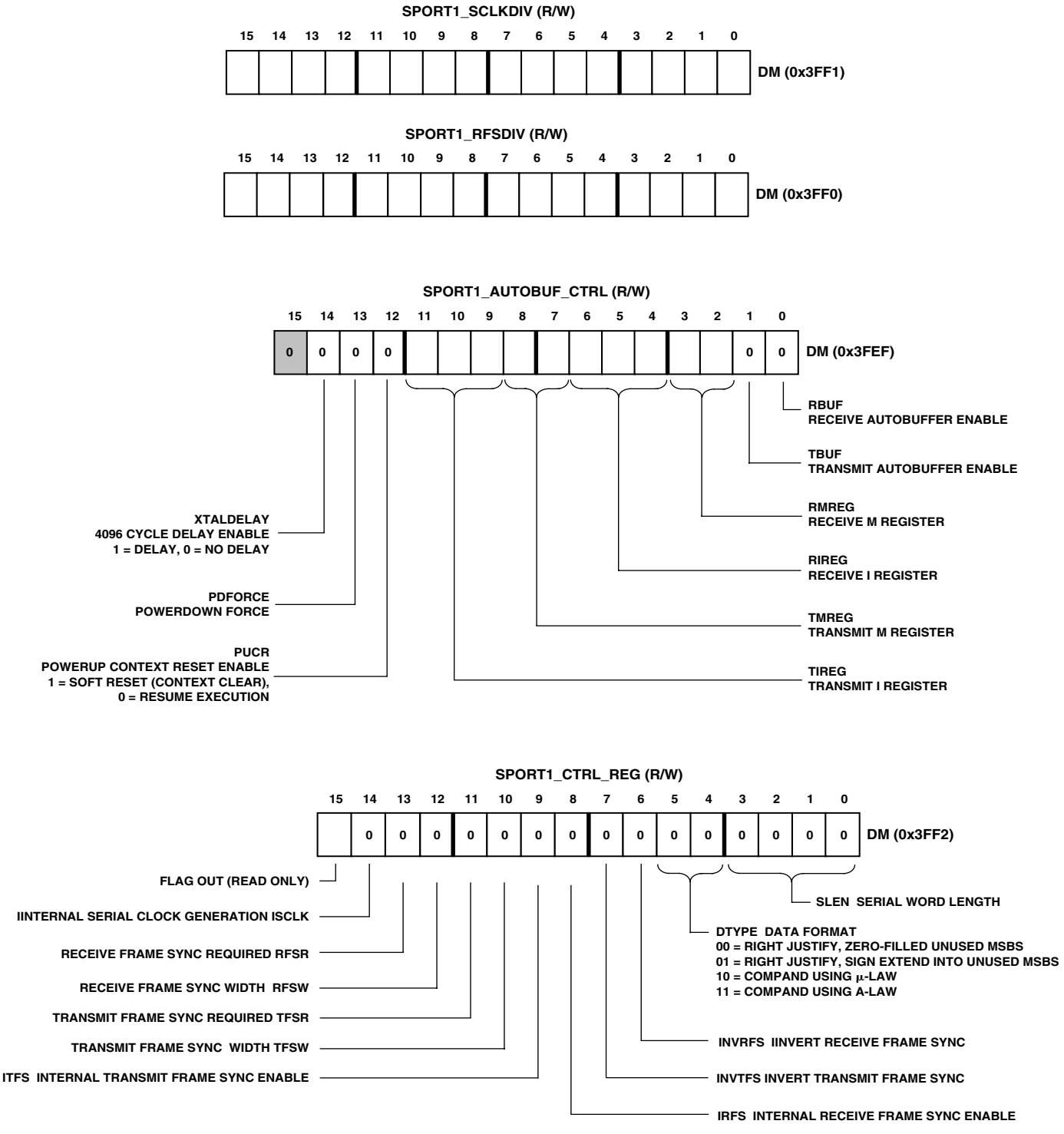


Figure 47. Structure of Registers of ADMC401

Default bit values are shown; if no value is shown, the bit field is undefined at reset. Reserved bits are shown on a gray field—these bits should always be written as shown.