Chipsmall Limited consists of a professional team with an average of over 10 year of expertise in the distribution of electronic components. Based in Hongkong, we have already established firm and mutual-benefit business relationships with customers from,Europe,America and south Asia,supplying obsolete and hard-to-find components to meet their specific needs.

With the principle of “Quality Parts,Customers Priority,Honest Operation,and Considerate Service”,our business mainly focus on the distribution of electronic components. Line cards we deal with include Microchip,ALPS,ROHM,Xilinx,Pulse,ON,Everlight and Freescale. Main products comprise IC,Modules,Potentiometer,IC Socket,Relay,Connector.Our parts cover such applications as commercial,industrial, and automotives areas.

We are looking forward to setting up business relationship with you and hope to provide you with the best service and solution. Let us make a better world for our industry!

Contact us

Tel: +86-755-8981 8866 Fax: +86-755-8427 6832
Email & Skype: info@chipsmall.com Web: www.chipsmall.com
Address: A1208, Overseas Decoration Building, #122 Zhenhua RD., Futian, Shenzhen, China

dsPIC30F
Family Overview

dsPIC® High-Performance 16-bit Digital Signal Controller

Note the following details of the code protection feature on Microchip devices:

- Microchip products meet the specification contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is one of the most secure families of its kind on the market today, when used in the intended manner and under normal conditions.
- There are dishonest and possibly illegal methods used to breach the code protection feature. All of these methods, to our knowledge, require using the Microchip products in a manner outside the operating specifications contained in Microchip's Data Sheets. Most likely, the person doing so is engaged in theft of intellectual property.
- Microchip is willing to work with the customer who is concerned about the integrity of their code.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of their code. Code protection does not mean that we are guaranteeing the product as "unbreakable."

Code protection is constantly evolving. We at Microchip are committed to continuously improving the code protection features of our products. Attempts to break Microchip's code protection feature may be a violation of the Digital Millennium Copyright Act. If such acts allow unauthorized access to your software or other copyrighted work, you may have a right to sue for relief under that Act.

dsPIC® High-Performance 16-bit Digital Signal Controller Family Overview

Operating Range:

- DC – 30 MIPS (30 MIPS @ 4.5-5.5V, -40 to 85°C)
- Wide V_{DD} range: 2.5-5.5V
- Ind. (-40 to 85°C) and Ext. (-40 to 125°C)

High-Performance DSC CPU:

- Modified Harvard architecture
- C compiler optimized instruction set
- 16-bit wide data path
- 24-bit wide instructions
- Linear program memory addressing up to 4M instruction words
- Linear data memory addressing up to 64 Kbytes
- 84 base instructions: mostly 1 word/1 cycle
- 16 16-bit General Purpose Registers (GPR)
- 2 40-bit accumulators:
  - With rounding and saturation options
- Flexible and powerful addressing modes:
  - Indirect, Modulo and Bit-Reversed
- Software stack
- 16 x 16 fractional/integer multiply operations
- 32/16 and 16/16 divide operations
- Single cycle multiply-and-accumulate:
  - Accumulator write back for DSP operations
  - Dual data fetch
- 40-stage barrel shifter

Interrupt Controller:

- 5-cycle latency
- Up to 45 interrupt sources, up to 5 external
- 7 programmable priority levels
- 4 processor exceptions

Digital I/O:

- Up to 54 programmable digital I/O pins
- Wake-up/Interrupt-on-change on up to 24 pins
- 25 mA sink and source on all I/O pins

On-Chip Flash, Data EEPROM and SRAM:

- Flash program memory: up to 144 Kbytes:
  - 10,000 erase/write cycles, min. (-40 to 85°C)
  - 100,000 erase/write cycles, typical
- Data EEPROM: up to 4 Kbytes:
  - 100,000 erase/write cycles, min. (-40 to 85°C)
  - 1M erase/write cycles, typical
  - Data EEPROM retention > 20 years
- Data SRAM: up to 8 Kbytes

System Management:

- Flexible clock options:
  - External, crystal, resonator, internal RC
  - Fully integrated PLL (4X, 8X, 16X)
  - Extremely low jitter PLL
- Programmable Power-up Timer
- Oscillator start-up timer/stabilizer
- Watchdog Timer with its own RC oscillator
- Fail-Safe Clock Monitor
- Reset by multiple sources

Power Management:

- Switch between clock sources in real time
- Programmable low-voltage detect
- Programmable Brown-out Reset
- Idle and Sleep modes with fast wake-up

Timers/Capture/Compare/PWM:

- Timer/counters: up to 5 16-bit timers:
  - Can pair up to make 32-bit timers
  - 1 timer can run as real-time clock with external 32 KHz oscillator
  - Programmable prescaler
- Input capture: up to 8 channels:
  - Capture on up, down or both edges
  - 16-bit Capture input functions
  - 4-deep FIFO on each capture
- Output compare: up to 8 channels:
  - Single or dual 16-bit Compare mode
  - 16-bit glitchless PWM mode

Communication Modules:

- 3-wire SPI™: up to 2 modules:
  - Framing supports I/O interface to simple codecs
  - Supports 8-bit and 16-bit data
  - Supports all serial clock formats and sampling modes
- I^2C™ full multi-master Slave mode support:
  - 7-bit and 10-bit addressing
  - Bus collision detection and arbitration
- UART: up to 2 modules:
  - Interrupt-on-address bit detect
  - Wake-up on Start bit from Sleep mode
  - 4-character TX and RX FIFO buffers
- Data Conversion Interface (DCI) module
  - Codec interface
  - Supports I^2S and AC'97 protocols
  - Up to 16-bit data words, up to 16 words per frame
  - 4-word deep TX and RX buffers
- CAN 2.0B active: up to 2 modules:
  - 3 transmit and 2 receive buffers
  - 6 receive filters and 2 masks
  - Loopback, Listen Only and Listen All Messages modes for diagnostics and bus monitoring
  - Wake-up on CAN message

Motor Control Peripherals:

- Motor Control PWM: up to 8 channels:
  - 4 duty cycle generators
  - Independent or Complementary mode
  - Programmable dead time and output polarity
  - Edge or center aligned
  - Manual output override control
  - Up to 2 Fault inputs
  - Trigger for A/D conversions
  - PWM Frequency for 16-bit resolution (@ 30 MIPS) = 915 Hz for Edge-Aligned mode, 457.5 Hz for Center-Aligned mode
  - PWM Frequency for 11-bit resolution (@ 30 MIPS) = 29.3 KHz for Edge-Aligned mode, 14.65 KHz for Center-Aligned mode
- Quadrature Encoder Interface module:
  - Phase A, Phase B and index pulse input
  - 16-bit up/down position counter
  - Count direction status
  - Position Measurement (x2 and x4) mode
  - Programmable digital noise filters on inputs
  - Alternate 16-bit Timer/Counter mode
  - Interrupt on position counter rollover/ underflow

Analog-to-Digital Converters:

- 10-bit 1 Msps A/D Converter module:
  - 2 or 4 simultaneous samples
  - Up to 16 input channels with auto scanning
  - 16 deep result buffer
  - Conversion start can be manual or synchronized with 1 of 4 trigger sources
  - Conversion possible in Sleep mode
  - ±1 LSB max. integral non-linearity
  - ±1 LSB max. differential non-linearity
- 12-bit 200 ksps A/D Converter module:
  - Up to 16 input channels with auto scanning
  - 16 deep result buffer
  - Conversion start can be manual or synchronized with 1 of 3 trigger sources
  - Conversion possible in Sleep mode
  - ±1 LSB max. integral non-linearity
  - ±1 LSB max. differential non-linearity

CMOS Flash Technology:

- Low-power, high-speed Flash technology
- Fully static design
- Wide operating voltage range (2.5V to 5.5V)
- Industrial and extended temperature ranges
- Low-power consumption

Packaging:

- 80-pin TQFP (12x12x1 mm, 14x14x1 mm)
- 64-pin TQFP (10x10x1 mm, 14x14x1 mm)
- 40-pin DIP, 44-pin TQFP
- 28-pin DIP (300 mil), 28-pin SOIC
- 28-pin QFN
- 18-pin DIP (300 mil), 18-pin SOIC

Note: See Table 1-1, Table 1-2 and Table 1-3 for exact peripheral features per device.

1.0 dsPIC30F PRODUCT FAMILIES

1.1 General Purpose Family

The dsPIC30F General Purpose Family (Table 1-1) is ideal for a wide variety of 16-bit MCU embedded applications. The variants with codec interfaces are well suited for audio applications.

TABLE 1-1: dsPIC30F GENERAL PURPOSE FAMILY VARIANTS

Device	Pins	Program Memory		SRAM Bytes	EEPROM Bytes	Timer 16-bit	Input Capture	Output Compare Std. PWM	Codec Interface	A/D 12-bit 200 ksps	UART	SPI™	I^2C™	CAN	I/O Pins (Max.)(1)	Packages (2)
		Bytes	Instructions													
dsPIC30F3014	40/44	24K	8K	2048	1024	3	2	2	—	13 ch	2	1	1	—	30	PG, PT
dsPIC30F4013	40/44	48K	16K	2048	1024	5	4	4	AC'97, I2S	13 ch	2	1	1	1	30	PG, PT
dsPIC30F5011	64	66K	22K	4096	1024	5	8	8	AC'97, I2S	16 ch	2	2	1	2	52	PT
dsPIC30F6011(3) dsPIC30F6011A	64	132K	44K	6144	2048	5	8	8	—	16 ch	2	2	1	2	52	PF, PT
dsPIC30F6012(3) dsPIC30F6012A	64	144K	48K	8192	4096	5	8	8	AC'97, I2S	16 ch	2	2	1	2	52	PF, PT
dsPIC30F5013	80	66K	22K	4096	1024	5	8	8	AC'97, I2S	16 ch	2	2	1	2	68	PT
dsPIC30F6013(3) dsPIC30F6013A	80	132K	44K	6144	2048	5	8	8	—	16 ch	2	2	1	2	68	PF, PT
dsPIC30F6014(3) dsPIC30F6014A	80	144K	48K	8192	4096	5	8	8	AC'97, I2S	16 ch	2	2	1	2	68	PF, PT

Note 1: Maximum I/O pin count includes pins shared by the peripheral functions.
2: All 28- and 40-pin devices may be offered in ML packages in the future, depending on die size.
3: This device is not recommended for new designs..

1.2 Motor Control and Power Conversion Family

These variants of dsPIC30F controllers (Table 1-2) support a variety of motor control applications such as brushless DC motors, single and 3-phase induction motors, and switched reluctance motors. They are also well suited for Uninterrupted Power Supply (UPS), inverters, switched mode power supplies and power factor correction, and also for controlling the Power Management module in servers, telecommunication equipment and other industrial equipment.

TABLE 1-2: dsPIC30F MOTOR CONTROL AND POWER CONVERSION FAMILY VARIANTS

Device	Pins	Program Memory		SRAM Bytes	EEPROM Bytes	Timer 16-bit	Input Capture	Output Compare/ Std. PWM	Motor Control PWM	A/D 10-bit 1 Msps	Quad Enc.	UART	SPI™	I^2C™	CAN	I/O Pins (Max.)(1)	Packages (2)
		Bytes	Instructions														
dsPIC30F2010	28	12K	4K	512	1024	3	4	2	6 ch	6 ch	1	1	1	1	—	20	SOG, PG, ML
dsPIC30F3010	28	24K	8K	1024	1024	5	4	2	6 ch	6 ch	1	1	1	1	—	20	SOG, PG
dsPIC30F4012	28	48K	16K	2048	1024	5	4	2	6 ch	6 ch	1	1	1	1	1	20	SOG, PG
dsPIC30F3011	40/44	24K	8K	1024	1024	5	4	4	6 ch	9 ch	1	2	1	1	—	30	PG, PT
dsPIC30F4011	40/44	48K	16K	2048	1024	5	4	4	6 ch	9 ch	1	2	1	1	1	30	PG, PT
dsPIC30F5015	64	66K	22K	2048	1024	5	4	4	8 ch	16 ch	1	1	2	1	1	52	PT
dsPIC30F6015	64	144K	48K	8192	4096	5	8	8	8 ch	16 ch	1	2	2	1	2	52	PT
dsPIC30F6010(3) dsPIC30F6010A	80	144K	48K	8192	4096	5	8	8	8 ch	16 ch	1	2	2	1	2	68	PF, PT

Note 1: Maximum I/O pin count includes pins shared by the peripheral functions.
2: All 28- and 40-pin devices may be offered in ML packages in the future, depending on die size.
3: This device is not recommended for new designs.

1.3 Sensor Family

The dsPIC30F Sensor Family products (Table 1-3) have features that support high-performance, low-cost embedded control applications. The 18- and 28-pin packages are designed to fit space-critical applications.

TABLE 1-3: dsPIC30F SENSOR PROCESSOR FAMILY VARIANTS

Device	Pins	Program Memory		SRAM Bytes	EEPROM Bytes	Timer 16-bit	Input Capture	Output Compare Std. PWM	A/D 12-bit 200 Ksps	UART	SPI™	I²C™	I/O Pins (<Max.)[1]	Packages[2]
		Bytes	Instructions											
dsPIC30F2011	18	12K	4K	1024	0	3	2	2	8 ch	1	1	1	12	SOG, PG
dsPIC30F3012	18	24K	8K	2048	1024	3	2	2	8 ch	1	1	1	12	SOG, PG
dsPIC30F2012	28	12K	4K	1024	0	3	2	2	10 ch	1	1	1	20	SOG, PG
dsPIC30F3013	28	24K	8K	2048	1024	3	2	2	10 ch	2	1	1	20	SOG, PG

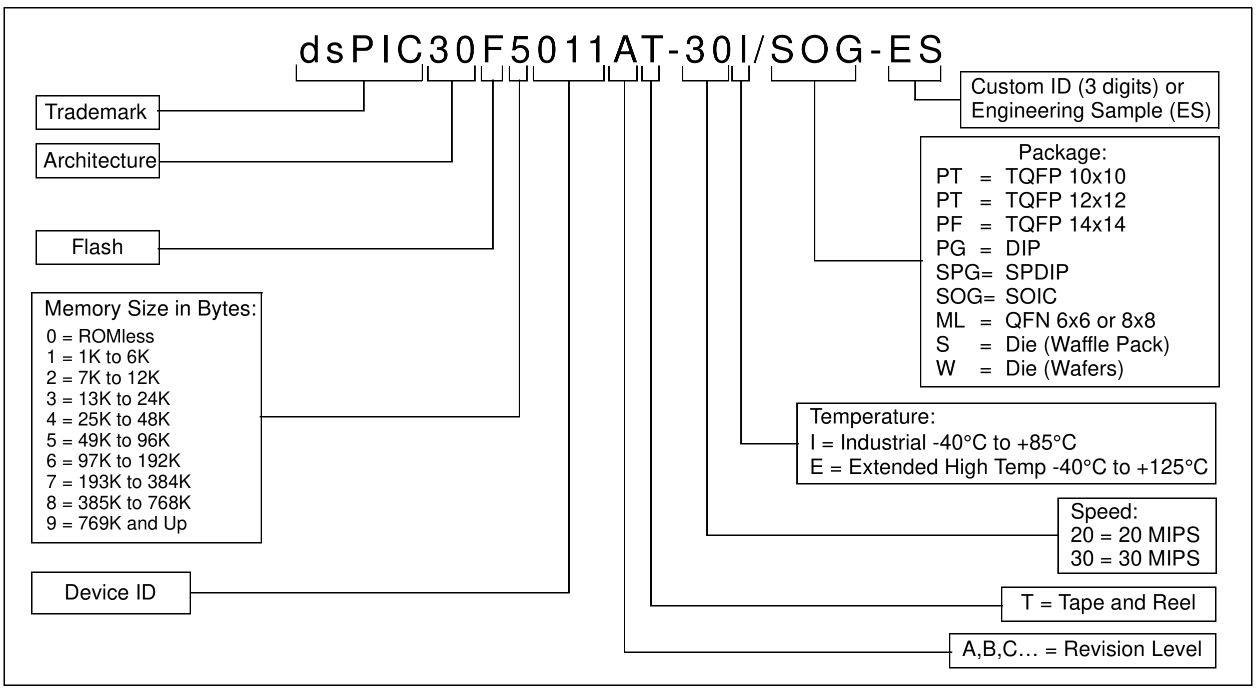
Note 1: Maximum I/O pin count includes pins shared by the peripheral functions.
2: All 28- and 40-pin devices may be offered in ML packages in the future, depending on die size.

1.4 Product Identification System

Figure 1-1 illustrates the part number structure.

To order or obtain information (e.g., on pricing or delivery), refer to the factory or a listed sales office (sales offices and locations are listed in the back of this document).

FIGURE 1-1: PART NUMBER STRUCTURE



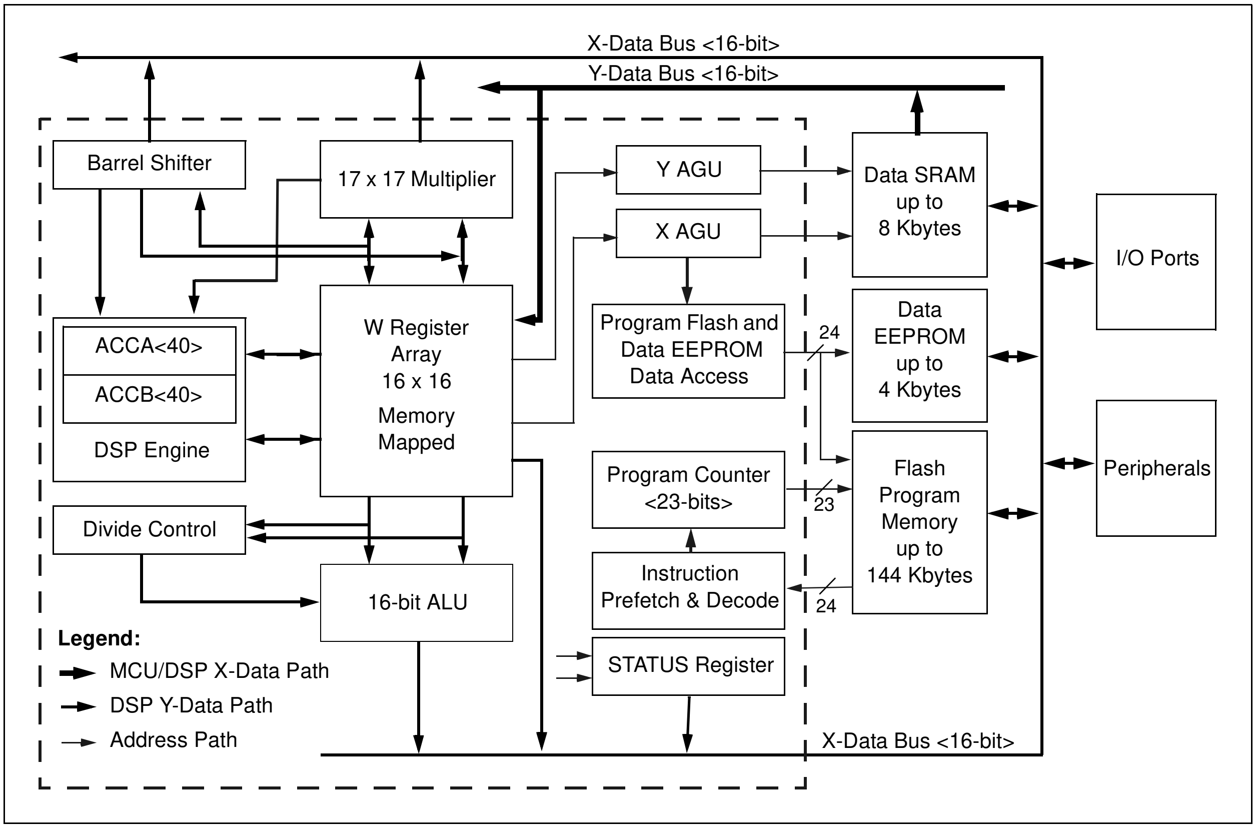
2.0 dsPIC30F DEVICE FAMILY OVERVIEW

The dsPIC30F device family employs a powerful 16-bit architecture that seamlessly integrates the control features of a microprocessor (MCU) with the computational capabilities of a digital signal processor (DSP). The resulting functionality is ideal for applications that rely on high-speed, repetitive computations as well as control.

The DSP engine, dual 40-bit accumulators, hardware support for division operations, barrel shifter, 17 x 17 multiplier, a large array of 16-bit working registers and a wide variety of data addressing modes, together provide the dsPIC30F CPU with extensive mathematical processing capability. Flexible and deterministic interrupt handling, coupled with a powerful array of peripherals, renders the dsPIC30F devices suitable for control applications. Reliable, field programmable Flash program memory and data EEPROM ensure scalability of applications that use dsPIC30F devices.

Figure 2-1 shows a sample device block diagram typical of the dsPIC30F product family.

FIGURE 2-1: dsPIC30F FAMILY BLOCK DIAGRAM

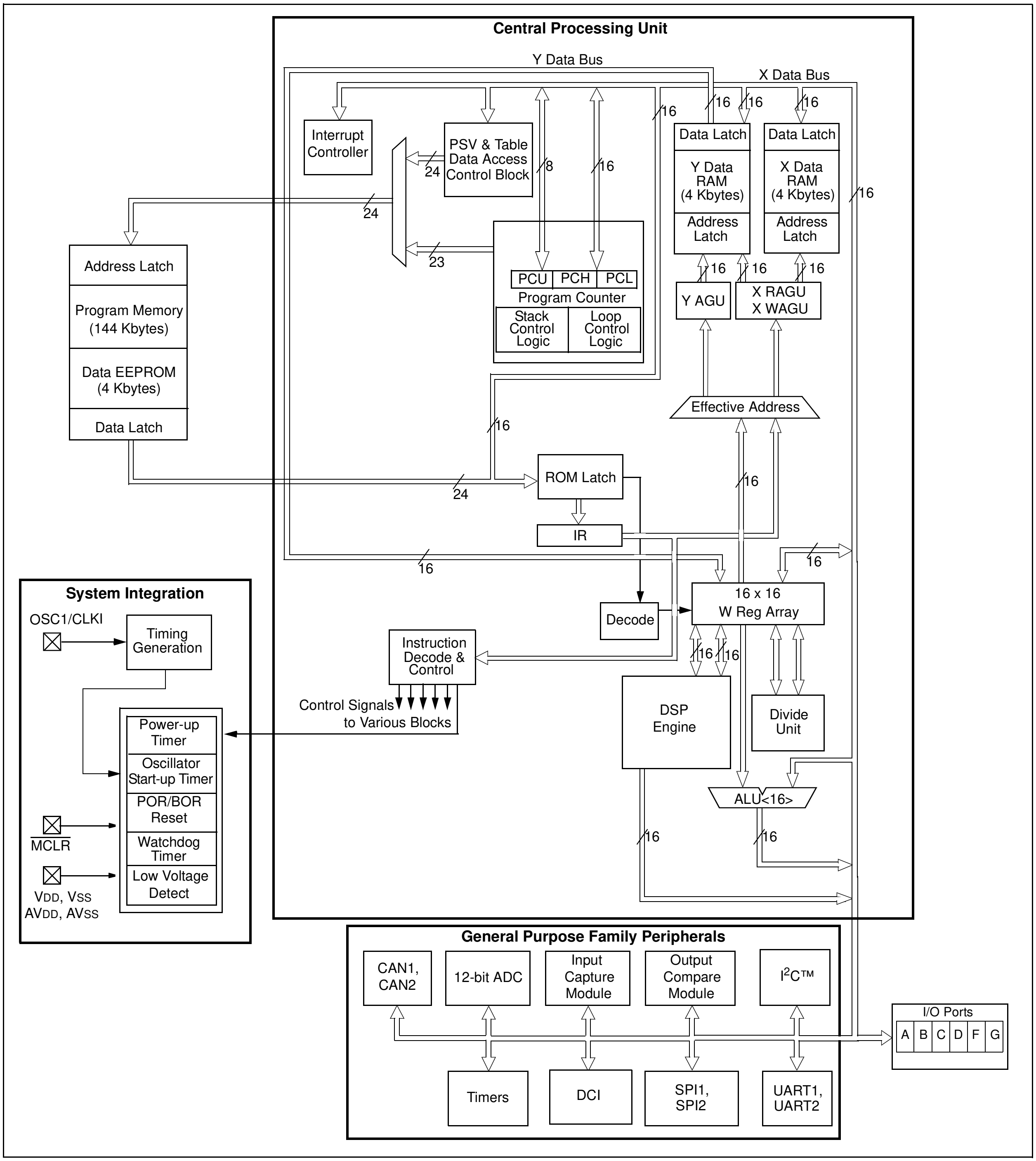


3.0 DEVICE OVERVIEW FOR GENERAL PURPOSE AND SENSOR FAMILIES

Figure 3-1 shows a sample device block diagram typical of the dsPIC30F General Purpose Product Family. Pin functionality and pinouts for this family are shown in Appendix A.

Note:	The device depicted in Figure 3-1 is representative of this family. Other devices of the same family may vary in terms of number of pins and multiplexing of pin functions. Typically, smaller devices in the family contain a subset of the peripherals present in the device(s) shown here.

FIGURE 3-1: dsPIC30F5013/6013A/6014A BLOCK DIAGRAM

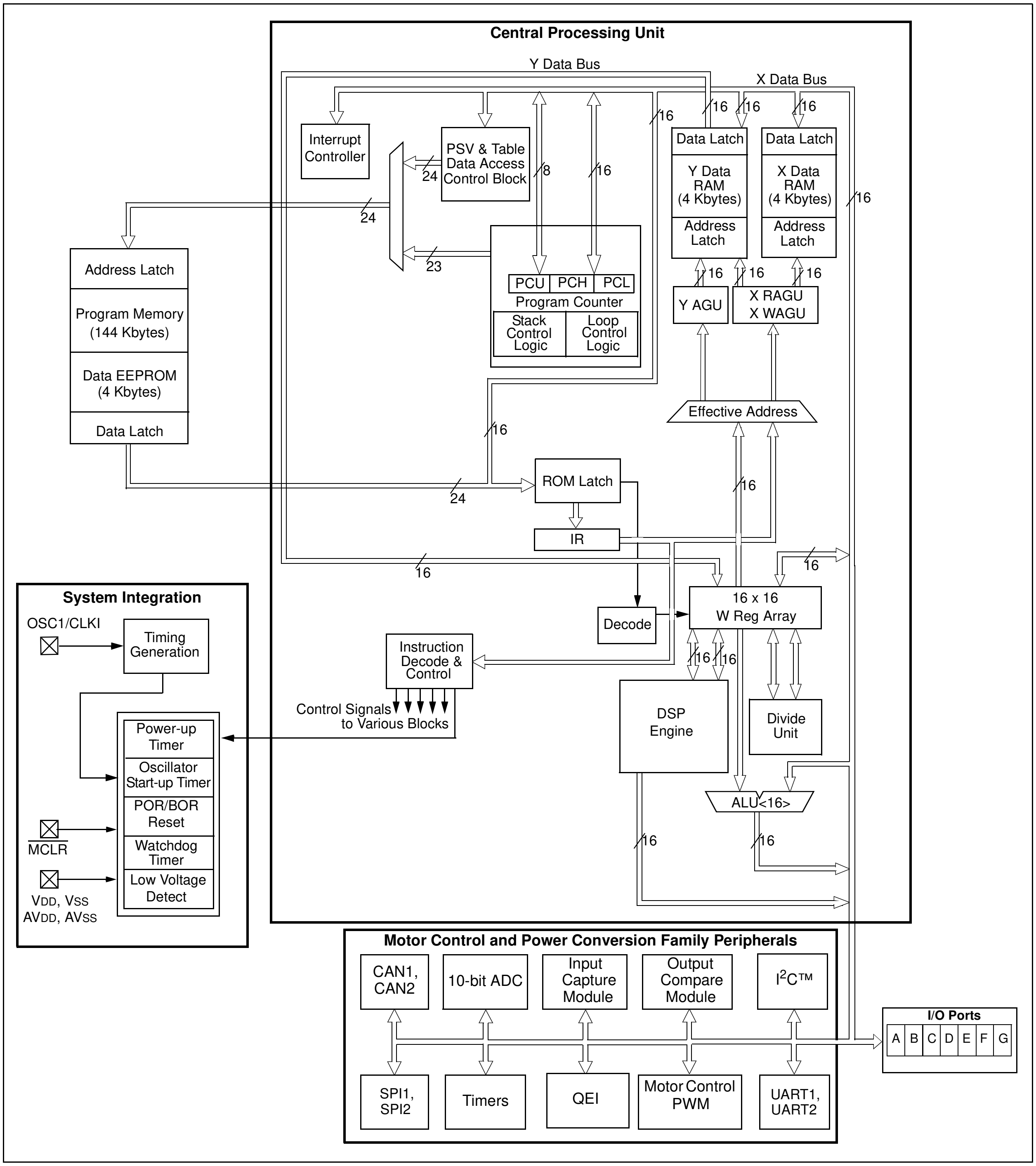


4.0 DEVICE OVERVIEW FOR MOTOR CONTROL AND POWER CONVERSION FAMILY

Figure 4-1 shows a sample device block diagram typical of the dsPIC30F Motor Control Product Family. Pin functionality and pinouts for this family are shown in Appendix B.

Note:	The device depicted in Figure 4-1 is representative of this family. Other devices of the same family may vary in terms of number of pins and multiplexing of pin functions. Typically, smaller devices in the family contain a subset of the peripherals present in the device(s) shown here.

FIGURE 4-1: dsPIC30F6010A BLOCK DIAGRAM



5.0 CPU ARCHITECTURE

5.1 Overview

The dsPIC30F CPU module has a 16-bit (data) modified Harvard architecture with an enhanced instruction set, including significant support for DSP. The CPU has a 24-bit instruction word, with a variable length opcode field. The Program Counter (PC) is 23 bits wide and addresses up to 4M x 24 bits of user program memory space. The actual amount of program memory implemented, as illustrated in Figure 5-1, varies from one device to another. A single-cycle instruction pre-fetch mechanism is used to help maintain throughput and provides predictable execution. All instructions execute in a single cycle, with the exception of instructions that change the program flow, the double-word move (MOV.D) instruction and the table instructions. Overhead free program loop constructs are supported using the DO and REPEAT instructions, both of which are interruptible at any point.

The dsPIC30F devices have sixteen 16-bit working registers in the programmer's model. Each of the working registers can serve as a data, address or address offset register. The 16th working register (W15) operates as a software Stack Pointer for interrupts and calls.

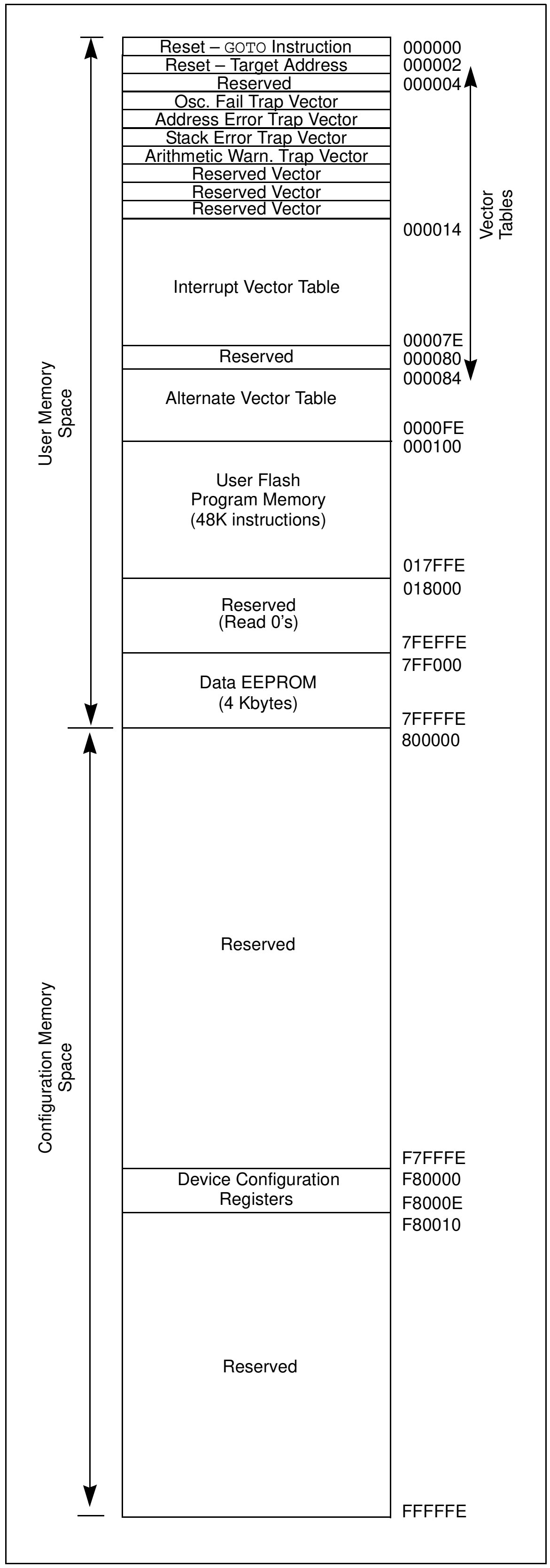
The dsPIC30F instruction set has two classes of instructions: the MCU class of instructions and the DSP class of instructions. These two instruction classes are seamlessly integrated into the architecture and execute from a single execution unit. The instruction set includes many addressing modes and is designed for optimum C compiler efficiency.

5.1.1 DATA MEMORY OVERVIEW

The data space can be addressed as 32 Kwords or 64 Kbytes and is split into two blocks, referred to as X and Y data memory. Each memory block has its own independent Address Generation Unit (AGU). The MCU class of instructions operate solely through the X memory AGU, which accesses the entire memory map as one linear data space. Certain DSP instructions operate through the X and Y AGUs to support dual operand reads, which splits the data address space into two parts. The X and Y data space boundary is device specific.

The upper 32 Kbytes of the data space memory map can optionally be mapped into program space at any 16K program word boundary defined by the 8-bit Program Space Visibility Page (PSVPAG) register. The program-to-data-space mapping feature lets any instruction access program space as if it were data space.

FIGURE 5-1: PROGRAM SPACE MEMORY MAP



5.1.2 ADDRESSING MODES OVERVIEW

Overhead free circular buffers (modulo addressing) are supported in both X and Y address spaces. The modulo addressing removes the software boundary checking overhead for DSP algorithms. Furthermore, the X AGU circular addressing can be used with any of the MCU class of instructions. The X AGU also supports bit-reversed addressing to greatly simplify input or output data reordering for radix-2 FFT algorithms.

The CPU supports Inherent (no operand), Relative, Literal, Memory Direct, Register Direct and Register Indirect Addressing modes. Each instruction is associated with a predefined addressing mode group depending upon its functional requirements. As many as 6 addressing modes are supported for each instruction.

For most instructions, the dsPIC30F is capable of executing a data (or program data) memory read, a working register (data) read, a data memory write and a program (instruction) memory read per instruction cycle. As a result, three parameter instructions can be supported, allowing A + B = C operations to be executed in a single cycle.

5.1.3 DSP ENGINE OVERVIEW

The DSP engine features a high speed, 17-bit by 17-bit multiplier, a 40-bit ALU, two 40-bit saturating accumulators and a 40-bit bidirectional barrel shifter. The barrel shifter is capable of shifting a 40-bit value up to 16 bits right or left, in a single cycle. The DSP instructions operate seamlessly with all other instructions and have been designed for optimal real-time performance. The MAC instruction and other associated instructions can concurrently fetch two data operands from memory while multiplying two W registers and accumulating and optionally saturating the result in the same cycle. This instruction functionality requires that the RAM memory data space be split for these instructions and linear for all others. Data space partitioning is achieved in a transparent and flexible manner through dedicating certain working registers to each address space.

5.1.4 SPECIAL MCU FEATURES

The dsPIC30F features a 17-bit by 17-bit single-cycle multiplier that is shared by both the MCU ALU and DSP engine. The multiplier can perform signed, unsigned and mixed-sign multiplication. Using a 17-bit by 17-bit multiplier for 16-bit by 16-bit multiplication not only allows you to perform mixed-sign multiplication, it also achieves accurate results for special operations such as (-1.0) x (-1.0).

The dsPIC30F supports 16/16 and 32/16 divide operations, both fractional and integer. All divide instructions are iterative operations. They must be executed within a REPEAT loop, resulting in a total execution time of 19 instruction cycles. The divide operation can be interrupted during any of those 19 cycles without loss of data.

A 40-bit barrel shifter is used to perform up to a 16-bit left or right shift, in a single cycle. The barrel shifter can be used by both MCU and DSP instructions.

5.1.5 INTERRUPT OVERVIEW

The dsPIC30F has a vectored exception scheme with up to 8 sources of non-maskable traps and 54 interrupt sources. Each interrupt source can be assigned to one of seven priority levels.

5.1.6 FEATURES TO ENHANCE COMPILER EFFICIENCY

In addition to extensive DSP capability, the CPU architecture possesses several features that lead to a more efficient (code size and speed) C compiler.

1. For most instructions, three-parameter instructions can be supported, allowing A + B = C operations to be executed in a single cycle.
2. Instruction addressing modes are extremely flexible to meet compiler needs.
3. The working register array consists of 16 x 16-bit registers, each of which can act as data, address or offset registers. One working register (W15) operates as the software Stack Pointer for interrupts and calls.
4. Linear indirect access of all data space is possible, plus the memory direct address range is up to 8 Kbytes. This capability, together with the addition of 16-bit direct address `MOV` based instructions, has provided a contiguous linear addressing space.
5. Linear indirect access of 32 Kword (64 Kbyte) pages within program space is possible, using any working register via new table read and write instructions.
6. Part of data space can be mapped into program space, allowing constant data to be accessed as if it were in data space.

5.2 Programmer's Model

The programmer's model, shown in Figure 5-2, consists of 16 x 16-bit working registers (W0 through W15), 2 x 40-bit accumulators (ACCA and ACCB), STATUS Register (SR), Data Table Page register (TBLPAG), Program Space Visibility Page register (PSVPAG), DO and REPEAT registers (DOSTART, DOEND, DCOUNT and RCOUNT) and Program Counter (PC). The working registers can act as data, address or offset registers. All registers are memory mapped. W0 is the W register for all instructions that perform file register addressing.

Some of these registers have a shadow register associated with them (see the legend in Figure 5-2). The shadow register is used as a temporary holding register and can transfer its contents to or from its host register upon some event occurring in a single cycle. None of the shadow registers are accessible directly.

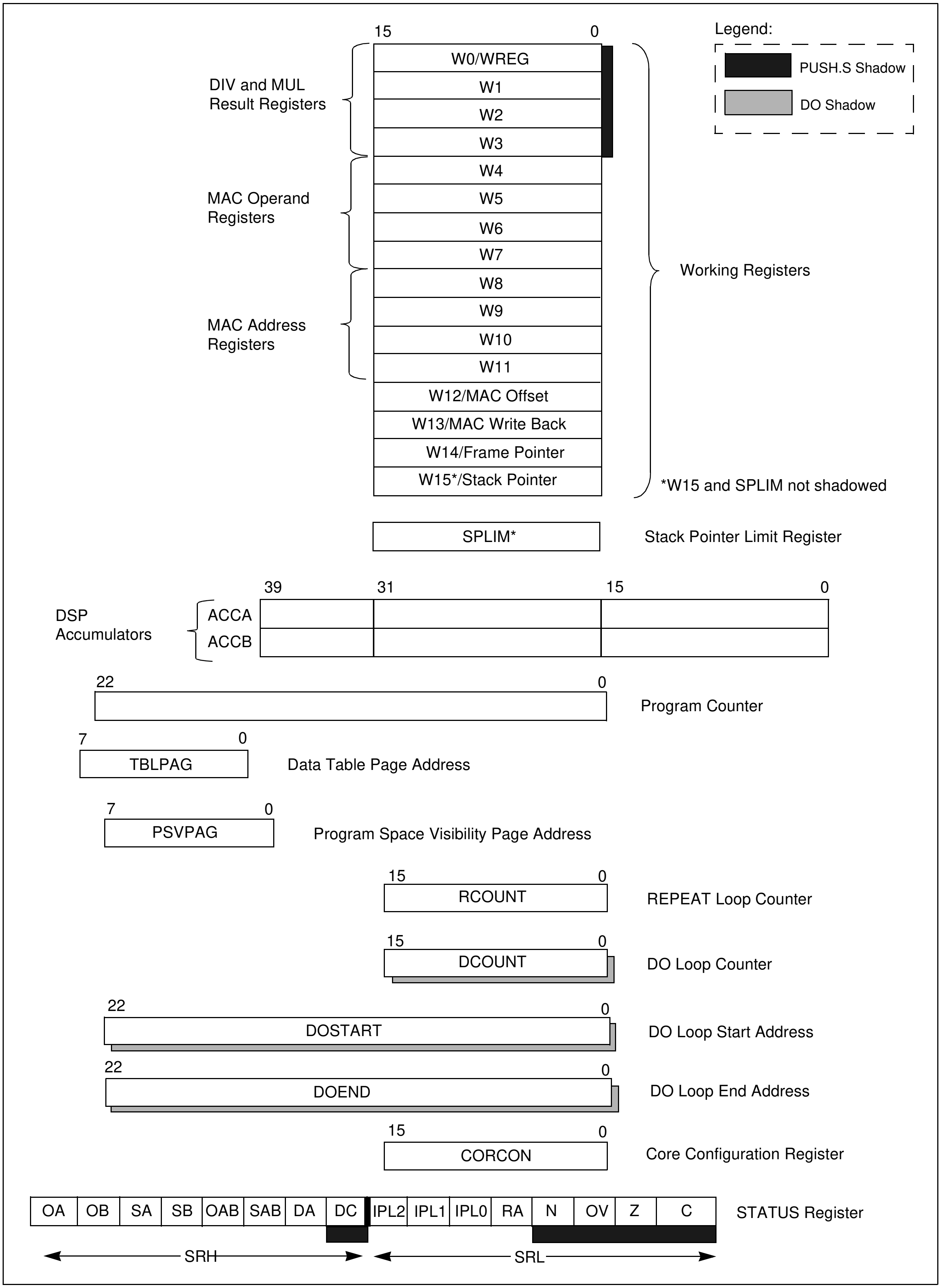
When a byte operation is performed on a working register, only the Least Significant Byte of the target register is affected. However, a benefit of memory mapped working registers is that both the Least and Most Significant Bytes can be manipulated through byte wide data memory space accesses.

W15 is the dedicated software Stack Pointer (SP). It is automatically modified by exception processing and subroutine calls and returns. However, W15 can be referenced by any instruction in the same manner as all other W registers. This simplifies the reading, writing and manipulation of the Stack Pointer (e.g., creating stack frames).

W14 has been dedicated as a Stack Frame Pointer, as defined by the `LNK` and `ULNK` instructions. However, W14 can be referenced by any instruction in the same manner as all other W registers.

The Stack Pointer always points to the first available free word and grows from lower addresses towards higher addresses. It pre-decrements for stack pops (reads) and post-increments for stack pushes (writes).

FIGURE 5-2: PROGRAMMER'S MODEL



5.3 Data Address Space

The core has two data spaces, X and Y. These data spaces can be considered either separate (for some DSP instructions), or as one unified linear address range (for MCU instructions). The data spaces are accessed using two Address Generation Units (AGUs) and separate data paths. This feature allows certain instructions to concurrently fetch two words from RAM, thereby enabling efficient execution of DSP algorithms such as Finite Impulse Response (FIR) filtering and Fast Fourier Transform (FFT).

5.3.1 X AND Y DATA SPACES

The X data space is used by all instructions and supports all addressing modes. There are separate read and write data buses for X data space. The X read data bus is the read data path for all instructions that view data space as combined X and Y address space. It is also the X data prefetch path for the dual operand DSP instructions (`MAC` class).

The Y data space is used in concert with the X data space by the `MAC` class of instructions (`CLR`, `ED`, `EDAC`, `MAC`, `MOVSAC`, `MPY`, `MPY.N` and `MSC`) to provide two concurrent data read paths.

Both the X and Y data spaces support modulo addressing for all instructions, subject to addressing mode restrictions. Bit-reversed addressing is only supported for writes to X data space.

All data memory writes, including in DSP instructions, view data space as combined X and Y address space. The boundary between the X and Y data spaces is device-dependent (an example is shown in Figure 5-3) and is not user programmable.

All effective addresses are 16 bits wide and point to bytes within the data space. Therefore, the data space address range is 64 Kbytes or 32 Kwords, though the implemented memory locations vary from one device to another.

5.3.2 DATA SPACE WIDTH

The core data width is 16-bits. All internal registers are organized as 16-bit wide words. Data space memory is organized in byte addressable, 16-bit wide blocks. Figure 5-3 depicts a sample data space memory map for the dsPIC30F.

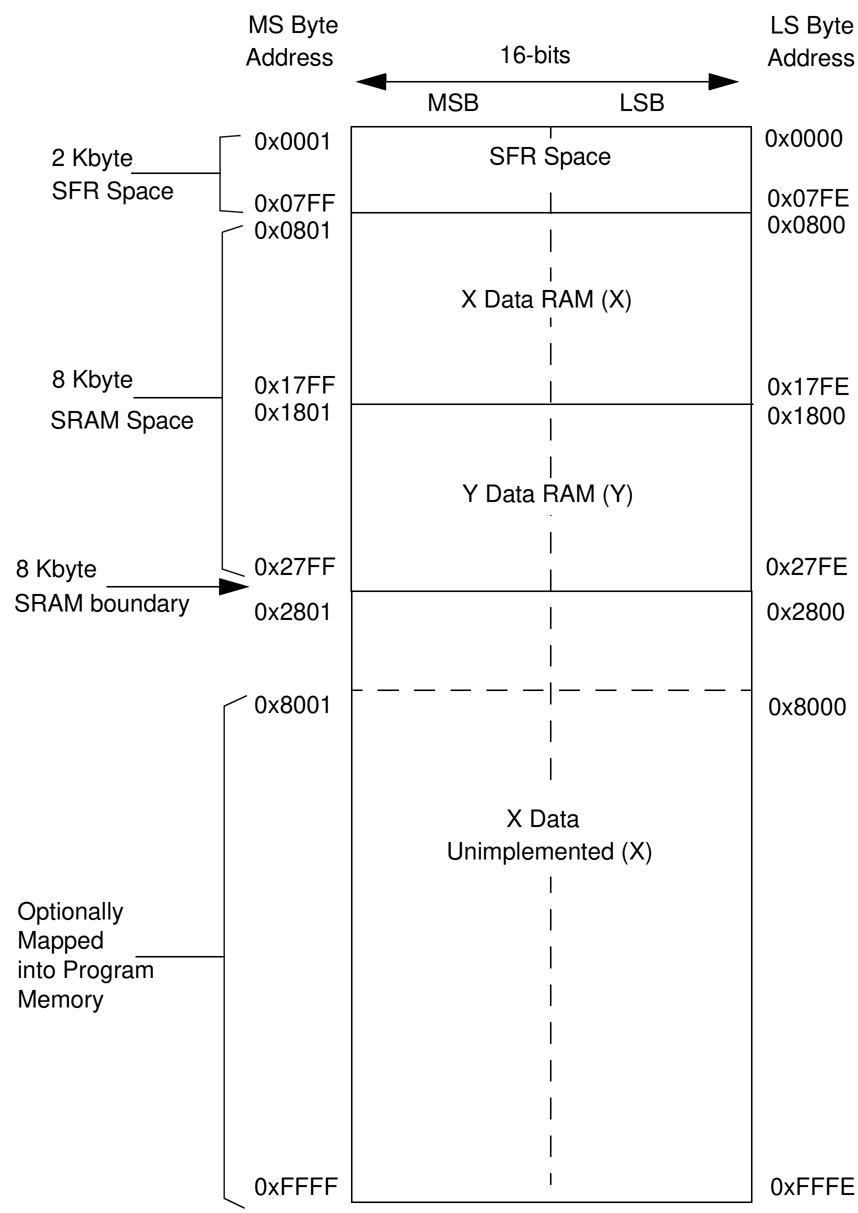
5.3.3 DATA ALIGNMENT

To help maintain backward compatibility with PICmicro® devices and improve data space memory usage efficiency, the dsPIC30F instruction set supports both word and byte operations. Data is aligned in data memory and registers as words, but all data space EAs resolve to bytes. Data byte reads will read the complete word which contains the byte, using the Least Significant bit (LSb) of any EA to determine which byte to select.

As a consequence of this byte accessibility, all effective address calculations are internally scaled. For example, the core would recognize that Post-Modified Register Indirect Addressing mode, [Ws++], will result in a value of Ws+1 for byte operations and Ws+2 for word operations.

All word accesses must be aligned to an even address. Misaligned word data fetches are not supported. Should a misaligned read or write be attempted, a trap will then be executed, allowing the system and/or user to examine the machine state prior to execution of the address fault.

FIGURE 5-3: SAMPLE DATA SPACE MEMORY MAP



5.4 DSP Engine

The DSP engine consists of a high-speed, single-cycle, 17-bit x 17-bit multiplier, a barrel shifter and a 40-bit adder/subtractor with two target accumulators, round and saturation logic, all of which enable efficient execution of computationally intensive DSP algorithms. The 17-bit x 17-bit multiplier is also utilized for MCU-based multiply instructions.

The DSP engine also has the capability to perform inherent accumulator-to-accumulator operations, which require no additional data. These instructions are `ADD`, `SUB` and `NEG`. This feature greatly simplifies basic arithmetic operations on 32-bit or 40-bit data.

A block diagram of the DSP engine is shown in Figure 5-4.

5.4.1 17X17-BIT MULTIPLIER

The 17 x 17-bit multiplier is capable of signed or unsigned operation. It can suitably scale its output to support either 1.31 fractional (Q31) or 32-bit integer results, thereby diminishing the need to manually post-process multiplication results for fractional data.

5.4.2 40-BIT ACCUMULATORS

The data accumulators have a 40-bit adder/subtractor with automatic sign-extension logic. It can select one of two accumulators (A or B) as its pre-accumulation source and post-accumulation destination. For the `ADD` and `LAC` instructions, the data to be accumulated or loaded can be optionally scaled via the barrel shifter prior to accumulation.

The adder/subtractor generates overflow Status bits SA/SB and OA/OB, which are latched and reflected in the STATUS Register and can also optionally generate an Arithmetic Error Trap:

- Overflow from bit 39. This is a catastrophic overflow in which the sign of the accumulator is destroyed.
- Overflow into guard bits 32 through 39. This is a recoverable overflow. This bit (OA/OB) is set whenever all the guard bits are not identical to each other.

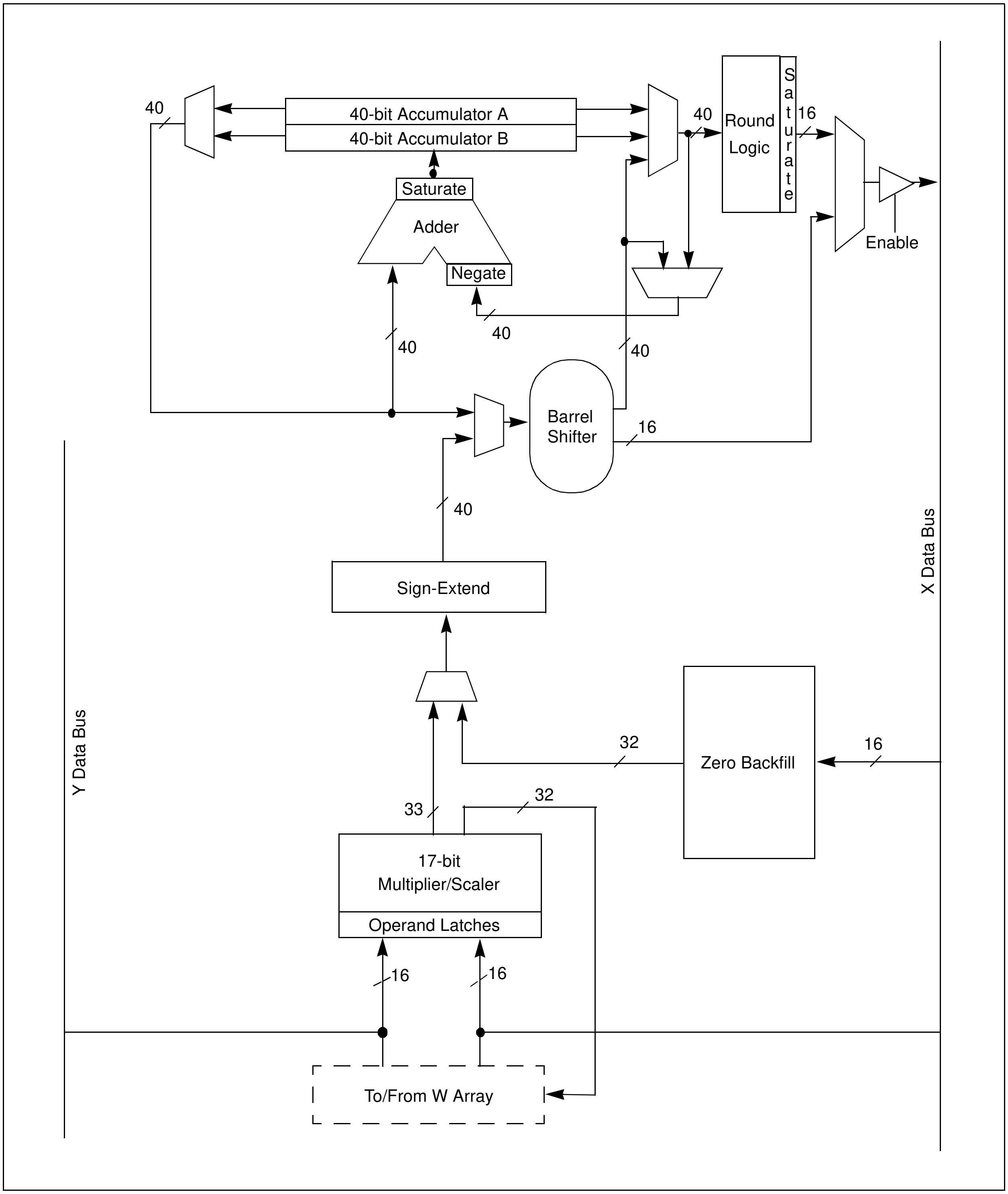
5.4.3 SATURATION AND OVERFLOW

The adder has an additional saturation block that controls accumulator data saturation, if selected. It uses the result of the adder, the overflow Status bits described above, and the user-configured control bits to determine when to saturate and to what value to saturate (a 40-bit or a 32-bit value).

In addition to adder/subtractor saturation, writes to data space can also be saturated, but without affecting the contents of the source accumulator.

The rounding logic performs a conventional (biased) or convergent (unbiased) data rounding function during an accumulator write (store). The Round mode is user-selectable. Rounding generates a 16-bit, 1.15 data value, which is passed to the data space write saturation logic. Data space write saturation ensures that the data in the accumulator is written back accurately even when rounding is performed. If rounding is not indicated by the instruction, a truncated 1.15 data value is stored and the least significant word (lsb) is simply discarded.

FIGURE 5-4: DSP ENGINE BLOCK DIAGRAM



6.0 EXCEPTION PROCESSING

The dsPIC30F has four processor exceptions (traps) and up to 45 sources of interrupts, which must be arbitrated based on a priority scheme.

The processor core is responsible for reading the Interrupt Vector Table (IVT) and transferring the address contained in the interrupt vector to the program counter.

The Interrupt Vector Table (IVT) and Alternate Interrupt Vector Table (AIVT) are placed near the beginning of program memory (0x000004) for ease of debugging.

The interrupt controller hardware pre-processes the interrupts before they are presented to the CPU. The interrupts and traps are enabled, prioritized and controlled using centralized Special Function Registers.

Each individual interrupt source has its own vector address and can be individually enabled and prioritized in user software. Each interrupt source also has its own status flag. This independent control and monitoring of the interrupt eliminates the need to poll various status flags to determine the interrupt source

Table 6-1 contains information about the interrupt vector.

Certain interrupts have specialized control bits for features like edge or level triggered interrupts, interrupt-on-change, etc. Control of these features remains within the Peripheral module, which generates the interrupt.

The special `DISI` instruction can be used to disable the processing of interrupts of priorities 6 and lower for a certain number of instruction cycles, during which the DISI bit remains set.

TABLE 6-1: INTERRUPT VECTORS

Vector Number	IVT Address	AIVT Address	Interrupt Source
8	0x000014	0x000094	INT0 – External Interrupt 0
9	0x000016	0x000096	IC1 – Input Compare 1
10	0x000018	0x000098	OC1 – Output Compare 1
11	0x00001A	0x00009A	T1 – Timer1
12	0x00001C	0x00009C	IC2 – Input Capture 2
13	0x00001E	0x00009E	OC2 – Output Compare 2
14	0x000020	0x0000A0	T2 – Timer2
15	0x000022	0x0000A2	T3 – Timer3
16	0x000024	0x0000A4	SPI1
17	0x000026	0x0000A6	U1RX – UART1 Receiver
18	0x000028	0x0000A8	U1TX – UART1 Transmitter
19	0x00002A	0x0000AA	ADC – ADC Convert Done
20	0x00002C	0x0000AC	NVM – NVM Write Complete
21	0x00002E	0x0000AE	I2C Slave Operation – Message Detect
22	0x000030	0x0000B0	I2C Master Operation – Message Event Complete
23	0x000032	0x0000B2	Change Notice Interrupt
24	0x000034	0x0000B4	INT1 – External Interrupt 1
25	0x000036	0x0000B6	IC7 – Input Capture 7
26	0x000038	0x0000B8	IC8 – Input Capture 8
27	0x00003A	0x0000BA	OC3 – Output Compare 3
28	0x00003C	0x0000BC	OC4 – Output Compare 4
29	0x00003E	0x0000BE	T4 – Timer4
30	0x000040	0x0000C0	T5 – Timer5
31	0x000042	0x0000C2	INT2 – External Interrupt 2
32	0x000044	0x0000C4	U2RX – UART2 Receiver
33	0x000046	0x0000C6	U2TX – UART2 Transmitter
34	0x000048	0x0000C8	SPI2
35	0x00004A	0x0000CA	CAN1

TABLE 6-1: INTERRUPT VECTORS (CONTINUED)

Vector Number	IVT Address	AIVT Address	Interrupt Source
36	0x00004C	0x0000CC	IC3 – Input Capture 3
37	0x00004E	0x0000CE	IC4 – Input Capture 4
38	0x000050	0x0000D0	IC5 – Input Capture 5
39	0x000052	0x0000D2	IC6 – Input Capture 6
40	0x000054	0x0000D4	OC5 – Output Compare 5
41	0x000056	0x0000D6	OC6 – Output Compare 6
42	0x000058	0x0000D8	OC7 – Output Compare 7
43	0x00005A	0x0000DA	OC8 – Output Compare 8
44	0x00005C	0x0000DC	INT3 – External Interrupt 3
45	0x00005E	0x0000DE	INT4 – External Interrupt 4
46	0x000060	0x0000E0	CAN2
47	0x000062	0x0000E2	PWM – PWM Period Match
48	0x000064	0x0000E4	QEI – Position Counter Compare
49	0x000066	0x0000E6	DCI – Codec Transfer Done
50	0x000068	0x0000E8	LVD – Low Voltage Detect
51	0x00006A	0x0000EA	FLTA – MCPWM FAULT A
52	0x00006C	0x0000EC	FLTB – MCPWM FAULT B
53-61	0x00006E-0x00007E	0x00006E-0x00007E	Reserved

6.1 Interrupt Priority

Each interrupt source can be user assigned to one of 8 priority levels, 1 through 7. Levels 7 and 1 represent the highest and lowest maskable priorities, respectively. A priority level of 0 disables the interrupt.

Since more than one interrupt request source may be assigned to a user specified priority level, a means is provided to assign priority within a given level. This method is called “Natural Order Priority.”

The Natural Order Priority of an interrupt is numerically identical to its vector number. The natural order priority scheme has 0 as the highest priority and 53 as the lowest priority.

The ability for the user to assign every interrupt to one of eight priority levels implies that the user can assign a very high overall priority level to an interrupt with a low natural order priority, thereby providing much flexibility in designing applications that use a large number of peripherals.

6.2 Interrupt Nesting

Interrupts, by default, are nestable. Any ISR that is in progress may be interrupted by another source of interrupt with a higher user assigned priority level. Interrupt nesting may be optionally disabled by setting the NSTDIS control bit (INTCON1<15>). When the NSTDIS control bit is set, all interrupts in progress will force the CPU priority to level 7 by setting IPL<2:0> = `111`. This action will effectively mask all other sources of interrupt until a `RETFIE` instruction is executed. When interrupt nesting is disabled, the user assigned interrupt priority levels will have no effect, except to resolve conflicts between simultaneous pending interrupts.

The IPL<2:0> bits become read-only when interrupt nesting is disabled. This prevents the user software from setting IPL<2:0> to a lower value, which would effectively re-enable interrupt nesting.

6.3 Traps

Traps can be considered as non-maskable, nestable interrupts that adhere to a fixed priority structure. Traps are intended to provide the user a means to correct erroneous operation during debug and when operating within the application. If the user does not intend to take corrective action in the event of a trap error condition, these vectors must be loaded with the address of a software routine that will reset the device. Otherwise, the trap vector is programmed with the address of a service routine that will correct the trap condition.

The dsPIC30F has four implemented sources of non-maskable traps:

- Oscillator Failure Trap
- Address Error Trap
- Stack Error Trap
- Arithmetic Error Trap

Many of these trap conditions can only be detected when they happen. Consequently, the instruction that caused the trap is allowed to complete before exception processing begins. Therefore, the user may have to correct the action of the instruction that caused the trap.

Each trap source has a fixed priority as defined by its position in the IVT. An oscillator failure trap has the highest priority, while an arithmetic error trap has the lowest priority.

Table 6-2 contains information about the trap vector.

6.4 Generating a Software Interrupt

Any available interrupt can be manually generated by user software (even if the corresponding peripheral is disabled), simply by enabling the interrupt and then setting the interrupt flag bit when required.

TABLE 6-2: TRAP VECTORS

Vector Number	IVT Address	AIVT Address	Trap Source
0	0x000004	0x000084	Reserved
1	0x000006	0x000086	Oscillator Failure
2	0x000008	0x000088	Address Error
3	0x00000A	0x00008A	Stack Error
4	0x00000C	0x00008C	Arithmetic Error
5	0x00000E	0x00008E	Reserved
6	0x000010	0x000090	Reserved
7	0x000012	0x000092	Reserved

7.0 SYSTEM INTEGRATION

System management services provided by the dsPIC30F device family include:

- Control of clock options and oscillators
- Power-on Reset
- Programmable Brown-out Reset
- Program control of Power-up Timer
- Oscillator start-up timer/stabilizer
- Watchdog Timer with RC oscillator
- Fail-Safe Clock Monitor
- Reset by multiple sources

7.1 Clock Options and Oscillators

There are three primary clock oscillators: XTL, XT and HS. The XTL oscillator is designed for crystals or ceramic resonators in the range of 200 kHz to 4 MHz. The XT oscillator is designed for crystals and ceramic resonators in the range of 4 to 10 MHz. The HS (High-Speed) oscillator is for crystals in the 10 to 25 MHz range. These oscillators use the OSC1 and OSC2 pins.

The secondary (LP) oscillator is designed for low power and uses a 32 kHz crystal or ceramic resonator. The LP oscillator uses the SOSC1 and SOSC2 pins.

The FRC (Fast RC) internal oscillator runs at a nominal 7.37 MHz ±2%. The user software can tune the FRC frequency. The LPRC (Low Power RC) internal osclllator is connected to the Watchdog Timer, and it runs at a nominal 512 kHz. The External RC (ERC) oscillator uses an external resistor and capacitor connected to the OSC1 pin. Frequency of operation is up to 4 MHz.

The OSC1 pin can also be used as an input from an external clock source; this mode is called “EC”.

The dsPIC30F oscillator system provides:

- Various external and internal oscillator options as clock sources
- An on-chip PLL to boost internal operating frequency 4, 8 or 16 times
- In some devices, the FRC oscillator can also be used with the PLL
- Clock switching between various clock sources
- Programmable clock postscaler for system power savings
- A Fail-Safe Clock Monitor (FSCM) that detects clock failure and takes fail-safe measures
- A Clock Control register (OSCCON)
- Nonvolatile Configuration bits for main oscillator selection.

A simplified block diagram of the oscillator system is shown in Figure 7-1.

7.2 Power-On Reset

When a supply voltage is applied to the device, a Power-on Reset is generated. A new Power-on Reset event is generated if the supply voltage falls below the device threshold voltage (V_{POR}). An internal POR pulse is generated when the rising supply voltage crosses the POR circuit threshold voltage.

7.3 Programmable Brown-out Reset

The BOR (Brown-out Reset) module is based on an internal voltage reference circuit. The main purpose of the BOR module is to generate a device Reset when a brown-out condition occurs. Brown-out conditions are generally caused by glitches on the AC mains (i.e., missing portions of the AC cycle waveform due to bad power transmission lines or voltage sags due to excessive current draw when a large inductive load is turned on).

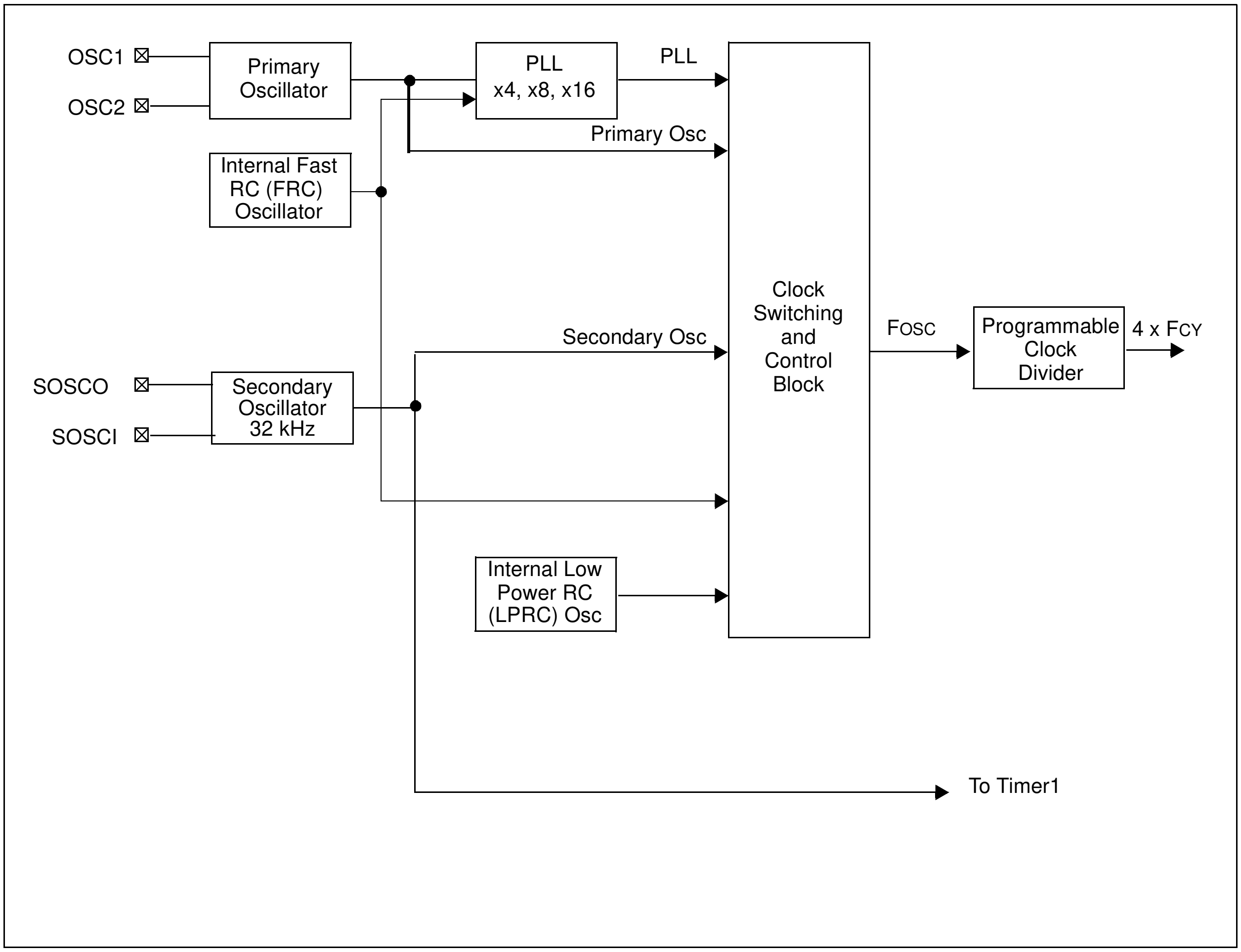
The BOR module allows selection of one of the following voltage trip points:

- 2.0V
- 2.7V
- 4.2V
- 4.5V

Note:	The BOR voltage trip points indicated here are nominal values provided for design guidance only. Refer to the Electrical Specifications in the specific device data sheet for BOR voltage limit specifications.

A BOR generates a Reset pulse, which resets the device.

FIGURE 7-1: OSCILLATOR SYSTEM BLOCK DIAGRAM



7.4 Programmable Power-up Timer (PWRT)

There are two internal timers that offer necessary delays on power-up. One is the Power-up Timer (PWRT), which provides a delay on power-up only. The PWRT keeps the part in Reset while the power supply stabilizes. The other is the Oscillator Start-up Timer (OST), intended to keep the chip in Reset until the crystal oscillator is stable. With these two timers on-chip, most applications need no external Reset circuitry.

7.5 Oscillator Start-up Timer/Stabilizer (OST)

An Oscillator Start-up Timer (OST) is included to ensure that a crystal oscillator (or ceramic resonator) has started and stabilized. The OST is a simple 10-bit counter that counts 1024 TOSC cycles before releasing the oscillator clock to the rest of the system. The time-out period is designated as TOST. The TOST time is involved every time the oscillator has to restart (i.e., on Power-On Reset (POR), Brown-Out Reset (BOR) and wake-up from Sleep). The oscillator start-up timer is applied to the LP oscillator, XT, XTL and HS modes (upon wake-up from Sleep, POR and BOR) for the primary oscillator.

7.6 Watchdog Timer (WDT)

The primary function of the Watchdog Timer (WDT) is to reset the processor in the event of a software malfunction. The WDT is a free running timer that runs off an on-chip RC oscillator, requiring no external component. The WDT Timer continues to operate even if the main processor clock (e.g., the crystal oscillator) fails.

The Watchdog Timer can be “Enabled” or “Disabled” either through a Configuration bit (FWDTEN) in the Configuration register or through an SFR bit (SWDTEN).

Any device programmer capable of programming dsPIC devices (such as Microchip’s PRO MATE® II programmer) allows programming of this and other Configuration bits to the desired state. If enabled, the WDT increments until it overflows or “times out”. A WDT time-out forces a device Reset (except during Sleep).

7.7 Fail-Safe Clock Monitor (FSCM)

The Fail-Safe Clock Monitor (FSCM) allows the device to continue to operate even in the event of an oscillator failure. The FSCM function is enabled by programming. If the FSCM function is enabled, the LPRC internal oscillator runs at all times (except during Sleep mode) and is not subject to control by the watchdog timer.

In the event of an oscillator failure, the FSCM generates a Clock Failure Trap event and switches the system clock over to the FRC oscillator. The application program then can either attempt to restart the oscillator, or execute a controlled shutdown. The Trap can be treated as a warm Reset by simply loading the Reset address into the oscillator fail trap vector.

7.8 Reset System

The Reset system combines all Reset sources and controls the device Master Reset signal.

Device Reset sources include:

- POR: Power-on Reset
- SWR: `RESET` instruction
- EXTR: $\overline{\text{MCLR}}$ Reset
- WDTR: Watchdog Timer Time-out Reset
- BOR: Brown-out Reset
- TRAPR: Trap Conflict Reset
- IOPUWR: Attempted execution of an Illegal Opcode, or Indirect Addressing using an Uninitialized W Register

8.0 DEVICE POWER MANAGEMENT

Power management services provided by the dsPIC30F device include:

- Real-time Clock Source Switching
- Programmable low-voltage detection
- Idle and Sleep modes with fast wake-up

8.1 Real-Time Clock Source Switching

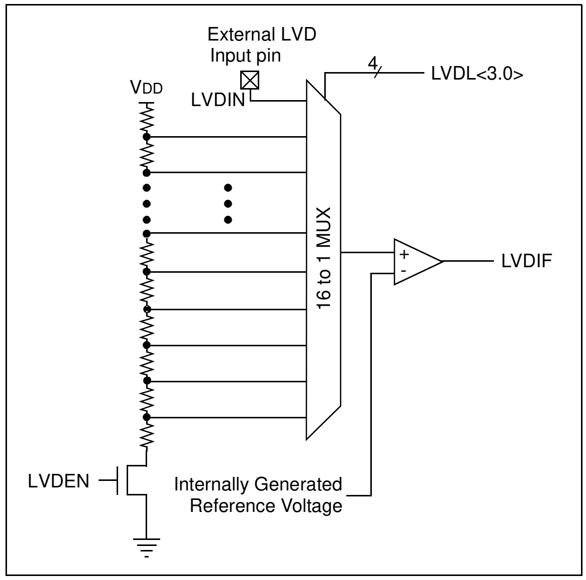
Configuration bits determine the clock source upon Power-on Reset (POR) and Brown-out Reset (BOR). Thereafter, the clock source can be changed between permissible clock sources. The OSCCON register controls the clock switching and reflects system clock related Status bits. To reduce power consumption, the user can switch to a slower clock source.

8.2 Low Voltage Detect (LVD)

The LVD module is used with battery operated applications to detect when the battery voltage (the VDD of the device) drops below a threshold, which is near the end of the battery life for the application. The LVD allows the application to gracefully shut down its operation.

This feature is only available on some devices. Figure 8-1 is a block diagram of the LVD module. A comparator uses an internally generated reference voltage as the set point. When the selected tap output of the device voltage is lower than the reference voltage, the LVD interrupt flag is set. Each node of the resistor divider represents a “trip point” voltage. The voltage is software programmable to any of 16 values, or can be obtained from an external pin (LVDIN).

FIGURE 8-1: LVD MODULE BLOCK DIAGRAM



8.3 Power-Saving Modes

The dsPIC30F devices have two reduced power modes that can be entered through execution of the `PWRSAV` instruction.

- Sleep mode: The CPU, system clock source and any peripherals that operate on the system clock source are disabled. This is the lowest power mode of the device.
- Idle mode: The CPU is disabled, but the system clock source continues to operate. Peripherals continue to operate, but can optionally be disabled.

These modes provide an effective way to reduce power consumption during periods when the CPU is not is use.

8.3.1 SLEEP MODE

When the device enters Sleep mode:

- System clock source is shut down. If an on-chip oscillator is used, it is turned off.
- Device current consumption is at minimum, provided that no I/O pin is sourcing current.
- Fail-Safe Clock Monitor (FSCM) does not operate during Sleep mode because the system clock source is disabled.
- LPRC clock continues to run in Sleep mode if the WDT is enabled.
- Low Voltage Detect circuit, if enabled, remains operative during Sleep mode.
- BOR circuit, if enabled, remains operative during Sleep mode.
- WDT, if enabled, is automatically cleared prior to entering Sleep mode.
- Some peripherals may continue to operate in Sleep mode. These peripherals include I/O pins that detect a change in the input signal, or peripherals that use an external clock input. Any peripheral that is operating on the system clock source is disabled in Sleep mode.

The processor exits (wakes up) from Sleep on one of these events:

- Any interrupt source that is individually enabled.
- Any form of device Reset.
- A WDT time-out.

8.3.2 IDLE MODE

When the device enters Idle mode:

- CPU stops executing instructions.
- WDT is automatically cleared.
- System clock source remains active.
- Peripheral modules, by default, continue to operate normally from the system clock source.
- Peripherals, optionally, can be shut down in Idle mode using their ‘stop-in-idle’ control bit.
- If the WDT or FSCM is enabled, the LPRC also remains active.

The processor wakes from Idle mode on these events:

- Any interrupt that is individually enabled.
- Any source of device Reset.
- A WDT Time-out.

Upon wake up from Idle, the clock is re-applied to the CPU and instruction execution begins immediately starting with the instruction following the `PWRSAV` instruction, or the first instruction in the Interrupt Service Routine (ISR).