



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



20-Pin Flash, 8-Bit Microcontrollers with XLP Technology

High-Performance RISC CPU:

- C Compiler Optimized Architecture
- Only 49 Instructions
- Operating Speed:
 - DC – 20 MHz clock input
 - DC – 200 ns instruction cycle
- Interrupt Capability with Automatic Context Saving
- 16-Level Deep Hardware Stack with Optional Overflow/Underflow Reset
- Direct, Indirect and Relative Addressing modes:
 - Two full 16-bit File Select Registers (FSRs)
 - FSRs can read program and data memory

Flexible Oscillator Structure:

- 16 MHz Internal Oscillator Block:
 - Factory calibrated to $\pm 1\%$, typical
 - Software selectable frequency range from 16 MHz to 31 kHz
- 31 kHz Low-Power Internal Oscillator
- Three External Clock modes up to 20 MHz

Special Microcontroller Features:

- Operating Voltage Range:
 - 1.8V to 3.6V (PIC16LF1508/9)
 - 2.3V to 5.5V (PIC16F1508/9)
- Self-Programmable under Software Control
- Power-on Reset (POR)
- Power-up Timer (PWRT)
- Programmable Low-Power Brown-out Reset (LPBOR)
- Extended Watchdog Timer (WDT):
 - Programmable period from 1 ms to 256s
- Programmable Code Protection
- In-Circuit Serial Programming™ (ICSP™) via Two Pins
- Enhanced Low-Voltage Programming (LVP)
- In-Circuit Debug (ICD) via Two Pins
- Power-Saving Sleep mode:
 - Low-Power Sleep mode
 - Low-Power BOR (LPBOR)
- Integrated Temperature Indicator
- 128 Bytes High-Endurance Flash
 - 100,000 write Flash endurance (minimum)

Memory:

- Up to 8 Kwords Linear Program Memory Addressing
- Up to 512 bytes Linear Data Memory Addressing
- High-Endurance Flash Data Memory (HEF)
 - 128 bytes if nonvolatile data storage
 - 100k erase/write cycles

eXtreme Low-Power (XLP)

Features(PIC16LF1508/9):

- Sleep Current:
 - 20 nA @ 1.8V, typical
- Watchdog Timer Current:
 - 260 nA @ 1.8V, typical
- Operating Current:
 - 30 μ A/MHz @ 1.8V, typical
- Secondary Oscillator Current:
 - 700 nA @ 32 kHz, 1.8V, typical

Peripheral Features:

- Analog-to-Digital Converter (ADC):
 - 10-bit resolution
 - 12 external channels
 - Three internal channels:
 - Fixed Voltage Reference
 - Digital-to-Analog Converter (DAC)
 - Temperature Indicator channel
 - Auto acquisition capability
 - Conversion available during Sleep
- 5-Bit Digital-to-Analog Converter (DAC):
 - Output available externally
 - Positive reference selection
 - Internal connections to comparators and ADC
- Two Comparators:
 - Rail-to-rail inputs
 - Power mode control
 - Software controllable hysteresis
- Voltage Reference:
 - 1.024V Fixed Voltage Reference (FVR) with 1x, 2x and 4x Gain output levels
- 18 I/O Pins (1 Input-only Pin):
 - High current sink/source 25 mA/25 mA
 - Individually programmable weak pull-ups
 - Individually programmable Interrupt-on-Change (IOC) pins
- Timer0: 8-Bit Timer/Counter with 8-Bit Programmable Prescaler
- Enhanced Timer1:
 - 16-bit timer/counter with prescaler
 - External Gate Input mode
- Timer2: 8-Bit Timer/Counter with 8-Bit Period Register, Prescaler and Postscaler
- Four 10-bit PWM modules
- Master Synchronous Serial Port (MSSP) with SPI and I²C with:
 - 7-bit address masking
 - SMBus/PMBus™ compatibility

PIC16(L)F1508/9

Peripheral Features (Continued):

- Enhanced Universal Synchronous Asynchronous Receiver Transmitter (EUSART)
 - RS-232, RS-485 and LIN compatible
 - Auto-Baud Detect
 - Auto-wake-up on Start
- Four Configurable Logic Cell (CLC) modules:
 - 16 selectable input source signals
 - Four inputs per module
 - Software control of combinational/sequential logic/state/clock functions
 - AND/OR/XOR/D Flop/D Latch/SR/JK
 - Inputs from external and internal sources
 - Output available to pins and peripherals
 - Operation while in Sleep
- Numerically Controlled Oscillator (NCO):
 - 20-bit accumulator
 - 16-bit increment
 - True linear frequency control
 - High-speed clock input
 - Selectable Output modes
 - Fixed Duty Cycle (FDC) mode
 - Pulse Frequency (PF) mode
- Complementary Waveform Generator (CWG):
 - Eight selectable signal sources
 - Selectable falling and rising edge dead-band control
 - Polarity control
 - Four auto-shutdown sources
 - Multiple input sources: PWM, CLC, NCO

PIC12(L)F1501/PIC16(L)F150X FAMILY TYPES

Device	Data Sheet Index	Program Memory Flash (words)	Data SRAM (bytes)	I/O's ⁽²⁾	10-bit ADC (ch)	Comparators	DAC	Timers (8/16-bit)	PWM	EUSART	MSSP (I ² C/SPI)	CWG	CLC	NCO	Debug ⁽¹⁾	XLP
PIC12(L)F1501	(1)	1024	64	6	4	1	1	2/1	4	—	—	1	2	1	H	—
PIC16(L)F1503	(2)	2048	128	12	8	2	1	2/1	4	—	1	1	2	1	H	—
PIC16(L)F1507	(3)	2048	128	18	12	—	—	2/1	4	—	—	1	2	1	H	—
PIC16(L)F1508	(4)	4096	256	18	12	2	1	2/1	4	1	1	1	4	1	I/H	Y
PIC16(L)F1509	(4)	8192	512	18	12	2	1	2/1	4	1	1	1	4	1	I/H	Y

Note 1: Debugging Methods: (I) - Integrated on Chip; (H) - using Debug Header; (E) - using Emulation Header.
2: One pin is input-only.

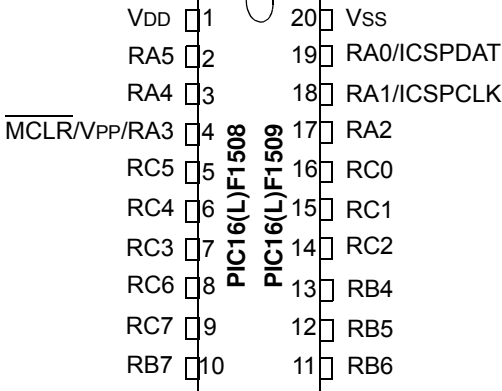
Data Sheet Index: (Unshaded devices are described in this document.)

- 1: DS40001615 [PIC12\(L\)F1501 Data Sheet, 8-Pin Flash, 8-bit Microcontrollers.](#)
- 2: DS40001607 [PIC16\(L\)F1503 Data Sheet, 14-Pin Flash, 8-bit Microcontrollers.](#)
- 3: DS40001586 [PIC16\(L\)F1507 Data Sheet, 20-Pin Flash, 8-bit Microcontrollers.](#)
- 4: DS40001609 [PIC16\(L\)F1508/9 Data Sheet, 20-Pin Flash, 8-bit Microcontrollers.](#)

Note: For other small form-factor package availability and marking information, please visit <http://www.microchip.com/packaging> or contact your local sales office.

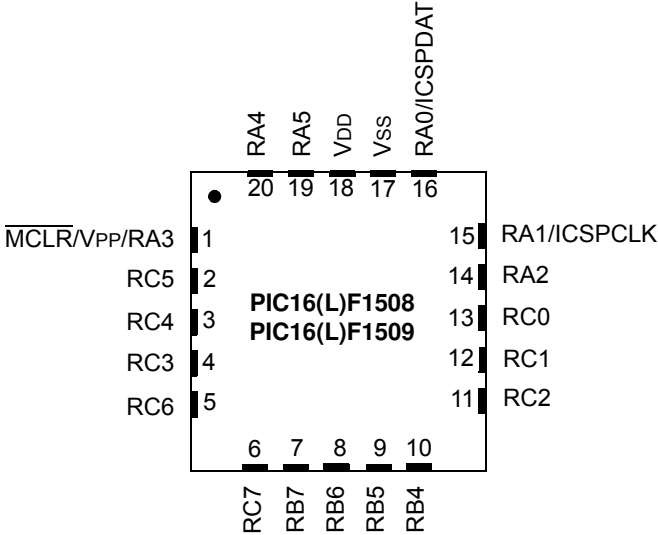
PIN DIAGRAMS

20-pin PDIP, SOIC, SSOP



Note: See [Table 1](#) for location of all peripheral functions.

20-pin QFN, UQFN



Note 1: See [Table 1](#) for location of all peripheral functions.
Note 2: It is recommended that the exposed bottom pad be connected to Vss.

PIC16(L)F1508/9

PIN ALLOCATION TABLE

TABLE 1: 20-PIN ALLOCATION TABLE (PIC16(L)F1508/9)

I/O	20-Pin PDIP/SOIC/SSOP	20-Pin QFN/UQFN	ADC	Reference	Comparator	Timers	EUSART	MSSP	CWG	NCO	CLC	PWM	Interrupt	Pull-up	Basic
RA0	19	16	AN0	DAC1OUT1	C1IN+	—	—	—	—	—	—	—	IOC	Y	ICSPDAT ICDDAT
RA1	18	15	AN1	VREF+	C1IN0- C2IN0-	—	—	—	—	—	CLC4IN1	—	IOC	Y	ICSPCLK ICDCLK
RA2	17	14	AN2	DAC1OUT2	C1OUT	TOCKI	—	—	CWG1FLT	—	CLC1	PWM3	INT/ IOC	Y	—
RA3	4	1	—	—	—	T1G ⁽¹⁾	—	SS ⁽¹⁾	—	—	CLC1IN0	—	IOC	Y	MCLR VPP
RA4	3	20	AN3	—	—	SOSCO T1G	—	—	—	—	—	—	IOC	Y	CLKOUT OSC2
RA5	2	19	—	—	—	SOSCI T1CKI	—	—	—	NCO1CLK	—	—	IOC	Y	CLKIN OSC1
RB4	13	10	AN10	—	—	—	—	SDA/SDI	—	—	CLC3IN0	—	IOC	Y	—
RB5	12	9	AN11	—	—	—	RX/DT	—	—	—	CLC4IN0	—	IOC	Y	—
RB6	11	8	—	—	—	—	—	SCL/SCK	—	—	—	—	IOC	Y	—
RB7	10	7	—	—	—	—	TX/CK	—	—	—	CLC3	—	IOC	Y	—
RC0	16	13	AN4	—	C2IN+	—	—	—	—	—	CLC2	—	—	—	—
RC1	15	12	AN5	—	C1IN1- C2IN1-	—	—	—	—	NCO1	—	PWM4	—	—	—
RC2	14	11	AN6	—	C1IN2- C2IN2-	—	—	—	—	—	—	—	—	—	—
RC3	7	4	AN7	—	C1IN3- C2IN3-	—	—	—	—	—	CLC2IN0	PWM2	—	—	—
RC4	6	3	—	—	C2OUT	—	—	—	CWG1B	—	CLC4 CLC2IN1	—	—	—	—
RC5	5	2	—	—	—	—	—	—	CWG1A	—	CLC1 ⁽¹⁾	PWM1	—	—	—
RC6	8	5	AN8	—	—	—	—	SS	—	NCO1 ⁽¹⁾	CLC3IN1	—	—	—	—
RC7	9	6	AN9	—	—	—	—	SDO	—	—	CLC1IN1	—	—	—	—
VDD	1	18	—	—	—	—	—	—	—	—	—	—	—	—	VDD
VSS	20	17	—	—	—	—	—	—	—	—	—	—	—	—	VSS

Note 1: Alternate pin function selected with the APFCON (Register 11-1) register.

TABLE OF CONTENTS

1.0	Device Overview	8
2.0	Enhanced Mid-Range CPU	13
3.0	Memory Organization	15
4.0	Device Configuration	40
5.0	Oscillator Module (With Fail-Safe Clock Monitor).....	46
6.0	Resets	62
7.0	Interrupts	70
8.0	Power-Down Mode (Sleep)	83
9.0	Watchdog Timer (WDT)	86
10.0	Flash Program Memory Control	90
11.0	I/O Ports	106
12.0	Interrupt-On-Change	119
13.0	Fixed Voltage Reference (FVR)	124
14.0	Temperature Indicator Module	126
15.0	Analog-to-Digital Converter (ADC) Module	128
16.0	5-Bit Digital-to-Analog Converter (DAC) Module.....	142
17.0	Comparator Module.....	145
18.0	Timer0 Module	152
19.0	Timer1 Module with Gate Control.....	155
20.0	Timer2 Module	166
21.0	Master Synchronous Serial Port (MSSP) Module	169
22.0	Enhanced Universal Synchronous Asynchronous Receiver Transmitter (EUSART)	223
23.0	Pulse-Width Modulation (PWM) Module	251
24.0	Configurable Logic Cell (CLC).....	257
25.0	Numerically Controlled Oscillator (NCO) Module	273
26.0	Complementary Waveform Generator (CWG) Module	280
27.0	In-Circuit Serial Programming™ (ICSP™)	292
28.0	Instruction Set Summary	294
29.0	Electrical Specifications.....	309
30.0	DC and AC Characteristics Graphs and Charts	339
31.0	Development Support.....	380
32.0	Packaging Information.....	384
	Appendix A: Data Sheet Revision History.....	397
	The Microchip Website	398
	Customer Change Notification Service	398
	Customer Support.....	398
	Product Identification System	399

TO OUR VALUED CUSTOMERS

It is our intention to provide our valued customers with the best documentation possible to ensure successful use of your Microchip products. To this end, we will continue to improve our publications to better suit your needs. Our publications will be refined and enhanced as new volumes and updates are introduced.

If you have any questions or comments regarding this publication, please contact the Marketing Communications Department via E-mail at docerrors@microchip.com or fax the **Reader Response Form** in the back of this data sheet to (480) 792-4150. We welcome your feedback.

Most Current Data Sheet

To obtain the most up-to-date version of this data sheet, please register at our Worldwide Website at:

<http://www.microchip.com>

You can determine the version of a data sheet by examining its literature number found on the bottom outside corner of any page. The last character of the literature number is the version number, (e.g., DS30000A is version A of document DS30000).

Errata

An errata sheet, describing minor operational differences from the data sheet and recommended workarounds, may exist for current devices. As device/documentation issues become known to us, we will publish an errata sheet. The errata will specify the revision of silicon and revision of document to which it applies.

To determine if an errata sheet exists for a particular device, please check with one of the following:

- Microchip's Worldwide Website; <http://www.microchip.com>
- Your local Microchip sales office (see last page)

When contacting a sales office, please specify which device, revision of silicon and data sheet (include literature number) you are using.

Customer Notification System

Register on our website at www.microchip.com to receive the most current information on all of our products.

PIC16(L)F1508/9

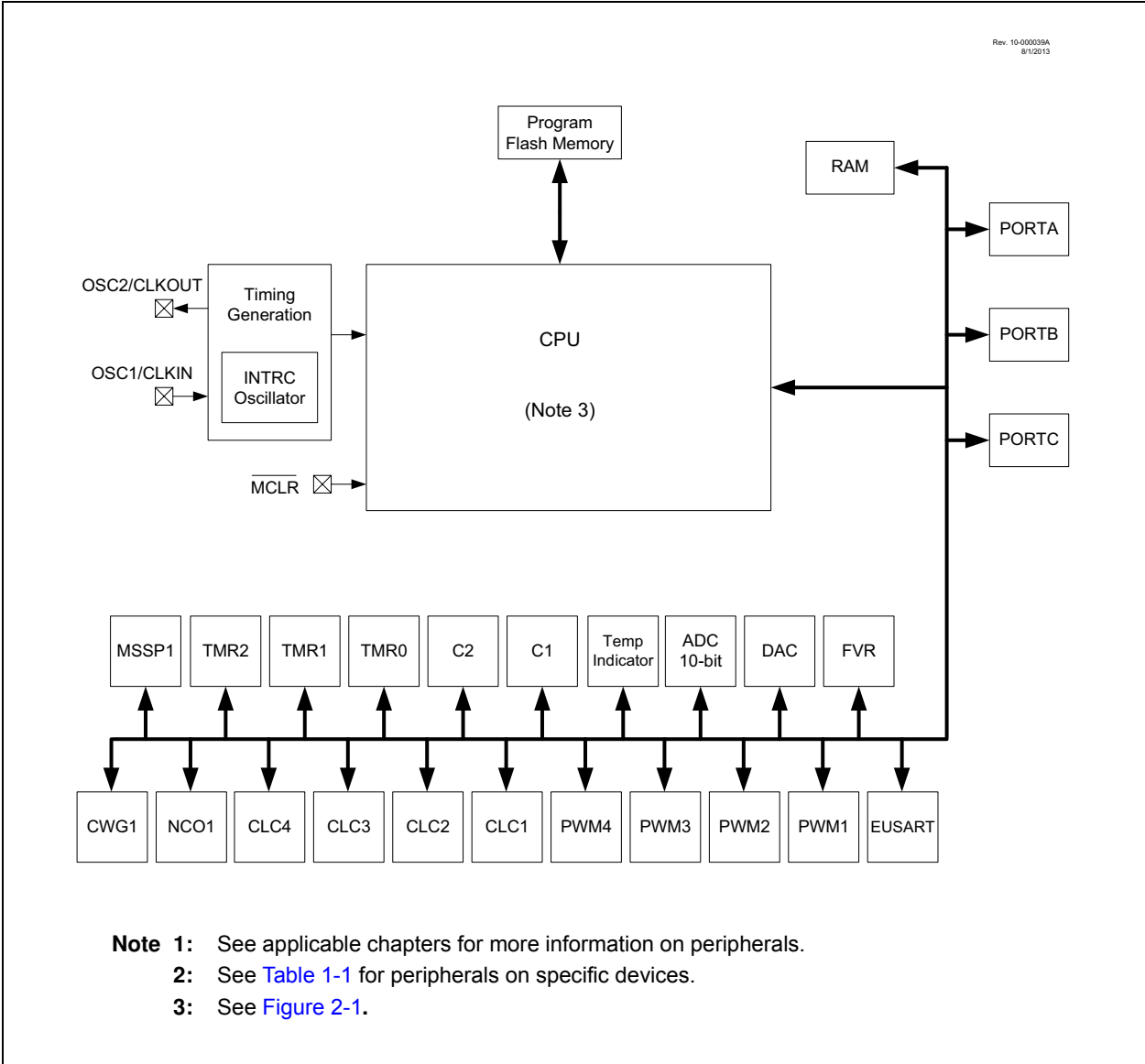
1.0 DEVICE OVERVIEW

The block diagram of these devices are shown in [Figure 1-1](#), the available peripherals are shown in [Table 1-1](#), and the pinout descriptions are shown in [Table 1-2](#).

TABLE 1-1: DEVICE PERIPHERAL SUMMARY

Peripheral		PIC12(L)F1501	PIC16(L)F1503	PIC16(L)F1507	PIC16(L)F1508	PIC16(L)F1509
Analog-to-Digital Converter (ADC)		•	•	•	•	•
Complementary Wave Generator (CWG)		•	•	•	•	•
Digital-to-Analog Converter (DAC)		•	•		•	•
Enhanced Universal Synchronous/Asynchronous Receiver/Transmitter (EUSART)					•	•
Fixed Voltage Reference (FVR)		•	•	•	•	•
Numerically Controlled Oscillator (NCO)		•	•	•	•	•
Temperature Indicator		•	•	•	•	•
Comparators						
	C1	•	•		•	•
	C2		•		•	•
Configurable Logic Cell (CLC)						
	CLC1	•	•	•	•	•
	CLC2	•	•	•	•	•
	CLC3				•	•
	CLC4				•	•
Master Synchronous Serial Ports						
	MSSP1		•		•	•
PWM Modules						
	PWM1	•	•	•	•	•
	PWM2	•	•	•	•	•
	PWM3	•	•	•	•	•
	PWM4	•	•	•	•	•
Timers						
	Timer0	•	•	•	•	•
	Timer1	•	•	•	•	•
	Timer2	•	•	•	•	•

FIGURE 1-1: PIC16(L)F1508/9 BLOCK DIAGRAM



- Note 1:** See applicable chapters for more information on peripherals.
- 2:** See [Table 1-1](#) for peripherals on specific devices.
- 3:** See [Figure 2-1](#).

PIC16(L)F1508/9

TABLE 1-2: PIC16(L)F1508/9 PINOUT DESCRIPTION

Name	Function	Input Type	Output Type	Description
RA0/AN0/C1IN+/DAC1OUT1/ ICSPDAT/ICDDAT	RA0	TTL	CMOS	General purpose I/O.
	AN0	AN	—	ADC Channel input.
	C1IN+	AN	—	Comparator positive input.
	DAC1OUT1	—	AN	Digital-to-Analog Converter output.
	ICSPDAT	ST	CMOS	ICSP™ Data I/O.
	ICDDAT	ST	CMOS	In-Circuit Debug data.
RA1/AN1/CLC4IN1/VREF+/ C1IN0-/C2IN0-/ICSPCLK/ ICDCLK	RA1	TTL	CMOS	General purpose I/O.
	AN1	AN	—	ADC Channel input.
	CLC4IN1	ST	—	Configurable Logic Cell source input.
	VREF+	AN	—	ADC Positive Voltage Reference input.
	C1IN0-	AN	—	Comparator negative input.
	C2IN0-	AN	—	Comparator negative input.
	ICSPCLK	ST	—	ICSP Programming Clock.
	ICDCLK	ST	—	In-Circuit Debug Clock.
RA2/AN2/C1OUT/DAC1OUT2/ T0CKI/INT/PWM3/CLC1/ CWG1FLT	RA2	ST	CMOS	General purpose I/O.
	AN2	AN	—	ADC Channel input.
	C1OUT	—	CMOS	Comparator output.
	DAC1OUT2	—	AN	Digital-to-Analog Converter output.
	T0CKI	ST	—	Timer0 clock input.
	INT	ST	—	External interrupt.
	PWM3	—	CMOS	PWM output.
	CLC1	—	CMOS	Configurable Logic Cell source output.
	CWG1FLT	ST	—	Complementary Waveform Generator Fault input.
RA3/CLC1IN0/VPP/T1G ⁽¹⁾ /SS ⁽¹⁾ / MCLR	RA3	TTL	—	General purpose input with IOC and WPU.
	CLC1IN0	ST	—	Configurable Logic Cell source input.
	VPP	HV	—	Programming voltage.
	T1G	ST	—	Timer1 Gate input.
	SS	ST	—	Slave Select input.
	MCLR	ST	—	Master Clear with internal pull-up.
RA4/AN3/SOSCO/ CLKOUT/T1G	RA4	TTL	CMOS	General purpose I/O.
	AN3	AN	—	ADC Channel input.
	SOSCO	XTAL	XTAL	Secondary Oscillator Connection.
	CLKOUT	—	CMOS	Fosc/4 output.
	T1G	ST	—	Timer1 Gate input.
RA5/CLKIN/T1CKI/NCO1CLK/ SOSCI	RA5	TTL	CMOS	General purpose I/O.
	CLKIN	CMOS	—	External clock input (EC mode).
	T1CKI	ST	—	Timer1 clock input.
	NCO1CLK	ST	—	Numerically Controlled Oscillator Clock source input.
	SOSCI	XTAL	XTAL	Secondary Oscillator Connection.

Legend: AN = Analog input or output CMOS = CMOS compatible input or output OD = Open-Drain
TTL = TTL compatible input ST = Schmitt Trigger input with CMOS levels I²C = Schmitt Trigger input with I²C levels
HV = High Voltage XTAL = Crystal

Note 1: Alternate pin function selected with the APFCON (Register 11-1) register.

TABLE 1-2: PIC16(L)F1508/9 PINOUT DESCRIPTION (CONTINUED)

Name	Function	Input Type	Output Type	Description
RB4/AN10/CLC3IN0/SDA/SDI	RB4	TTL	CMOS	General purpose I/O.
	AN10	AN	—	ADC Channel input.
	CLC3IN0	ST	—	Configurable Logic Cell source input.
	SDA	I ² C	OD	I ² C data input/output.
	SDI	CMOS	—	SPI data input.
RB5/AN11/CLC4IN0/RX/DT	RB5	TTL	CMOS	General purpose I/O.
	AN11	AN	—	ADC Channel input.
	CLC4IN0	ST	—	Configurable Logic Cell source input.
	RX	ST	—	USART asynchronous input.
	DT	ST	CMOS	USART synchronous data.
RB6/SCL/SCK	RB6	TTL	CMOS	General purpose I/O.
	SCL	I ² C	OD	I ² C clock.
	SCK	ST	CMOS	SPI clock.
RB7/CLC3/TX/CK	RB7	TTL	CMOS	General purpose I/O.
	CLC3	—	CMOS	Configurable Logic Cell source output.
	TX	—	CMOS	USART asynchronous transmit.
	CK	ST	CMOS	USART synchronous clock.
RC0/AN4/CLC2/C2IN+	RC0	TTL	CMOS	General purpose I/O.
	AN4	AN	—	ADC Channel input.
	CLC2	—	CMOS	Configurable Logic Cell source output.
	C2IN+	AN	—	Comparator positive input.
RC1/AN5/C1IN1-/C2IN1-/PWM4/NCO1	RC1	TTL	CMOS	General purpose I/O.
	AN5	AN	—	ADC Channel input.
	C1IN1-	AN	—	Comparator negative input.
	C2IN1-	AN	—	Comparator negative input.
	PWM4	—	CMOS	PWM output.
	NCO1	—	CMOS	Numerically Controlled Oscillator is source output.
RC2/AN6/C1IN2-/C2IN2-	RC2	TTL	CMOS	General purpose I/O.
	AN6	AN	—	ADC Channel input.
	C1IN2-	AN	—	Comparator negative input.
	C2IN2-	AN	—	Comparator negative input.
RC3/AN7/C1IN3-/C2IN3-/PWM2/CLC2IN0	RC3	TTL	CMOS	General purpose I/O.
	AN7	AN	—	ADC Channel input.
	C1IN3-	AN	—	Comparator negative input.
	C2IN3-	AN	—	Comparator negative input.
	PWM2	—	CMOS	PWM output.
	CLC2IN0	ST	—	Configurable Logic Cell source input.
RC4/C2OUT/CLC2IN1/CLC4/CWG1B	RC4	TTL	CMOS	General purpose I/O.
	C2OUT	—	CMOS	Comparator output.
	CLC2IN1	ST	—	Configurable Logic Cell source input.
	CLC4	—	CMOS	Configurable Logic Cell source output.
	CWG1B	—	CMOS	CWG complementary output.

Legend: AN = Analog input or output CMOS = CMOS compatible input or output OD = Open-Drain
TTL = TTL compatible input ST = Schmitt Trigger input with CMOS levels I²C = Schmitt Trigger input with I²C levels
HV = High Voltage XTAL = Crystal

Note 1: Alternate pin function selected with the APFCON (Register 11-1) register.

PIC16(L)F1508/9

TABLE 1-2: PIC16(L)F1508/9 PINOUT DESCRIPTION (CONTINUED)

Name	Function	Input Type	Output Type	Description
RC5/PWM1/CLC1 ⁽¹⁾ / CWG1A	RC5	TTL	CMOS	General purpose I/O.
	PWM1	—	CMOS	PWM output.
	CLC1	—	CMOS	Configurable Logic Cell source output.
	CWG1A	—	CMOS	CWG primary output.
RC6/AN8/NCO1 ⁽¹⁾ /CLC3IN1/ SS	RC6	TTL	CMOS	General purpose I/O.
	AN8	AN	—	ADC Channel input.
	NCO1	—	CMOS	Numerically Controlled Oscillator source output.
	CLC3IN1	ST	—	Configurable Logic Cell source input.
RC7/AN9/CLC1IN1/SDO	RC7	TTL	CMOS	General purpose I/O.
	AN9	AN	—	ADC Channel input.
	CLC1IN1	ST	—	Configurable Logic Cell source input.
	SDO	—	CMOS	SPI data output.
VDD	VDD	Power	—	Positive supply.
Vss	Vss	Power	—	Ground reference.

Legend: AN = Analog input or output CMOS = CMOS compatible input or output OD = Open-Drain
TTL = TTL compatible input ST = Schmitt Trigger input with CMOS levels I²C = Schmitt Trigger input with I²C levels
HV = High Voltage XTAL = Crystal

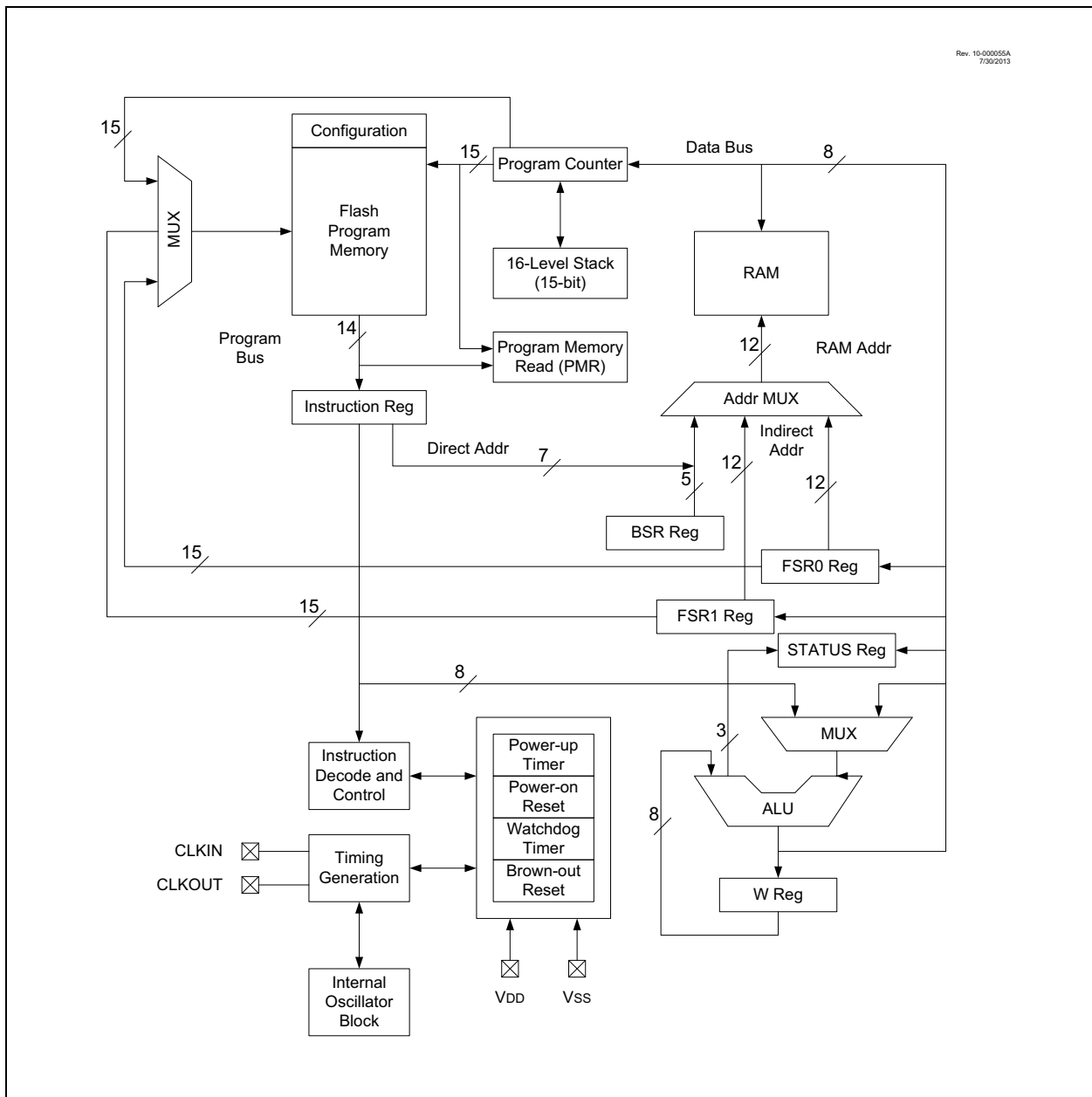
Note 1: Alternate pin function selected with the APFCON (Register 11-1) register.

2.0 ENHANCED MID-RANGE CPU

This family of devices contain an enhanced mid-range 8-bit CPU core. The CPU has 49 instructions. Interrupt capability includes automatic context saving. The hardware stack is 16 levels deep and has Overflow and Underflow Reset capability. Direct, Indirect, and Relative addressing modes are available. Two File Select Registers (FSRs) provide the ability to read program and data memory.

- Automatic Interrupt Context Saving
- 16-level Stack with Overflow and Underflow
- File Select Registers
- Instruction Set

FIGURE 2-1: CORE BLOCK DIAGRAM



PIC16(L)F1508/9

2.1 Automatic Interrupt Context Saving

During interrupts, certain registers are automatically saved in shadow registers and restored when returning from the interrupt. This saves stack space and user code. See [Section 7.5 “Automatic Context Saving”](#), for more information.

2.2 16-Level Stack with Overflow and Underflow

These devices have a hardware stack memory 15 bits wide and 16 words deep. A Stack Overflow or Underflow will set the appropriate bit (STKOVF or STKUNF) in the PCON register, and if enabled, will cause a software Reset. See [Section 3.5 “Stack”](#) for more details.

2.3 File Select Registers

There are two 16-bit File Select Registers (FSR). FSRs can access all file registers and program memory, which allows one Data Pointer for all memory. When an FSR points to program memory, there is one additional instruction cycle in instructions using INDF to allow the data to be fetched. General purpose memory can now also be addressed linearly, providing the ability to access contiguous data larger than 80 bytes. There are also new instructions to support the FSRs. See [Section 3.6 “Indirect Addressing”](#) for more details.

2.4 Instruction Set

There are 49 instructions for the enhanced mid-range CPU to support the features of the CPU. See [Section 28.0 “Instruction Set Summary”](#) for more details.

3.0 MEMORY ORGANIZATION

These devices contain the following types of memory:

- Program Memory
 - Configuration Words
 - Device ID
 - User ID
 - Flash Program Memory
- Data Memory
 - Core Registers
 - Special Function Registers
 - General Purpose RAM
 - Common RAM

The following features are associated with access and control of program memory and data memory:

- PCL and PCLATH
- Stack
- Indirect Addressing

3.1 Program Memory Organization

The enhanced mid-range core has a 15-bit program counter capable of addressing a 32K x 14 program memory space. [Table 3-1](#) shows the memory sizes implemented. Accessing a location above these boundaries will cause a wrap-around within the implemented memory space. The Reset vector is at 0000h and the interrupt vector is at 0004h (See [Figure 3-1](#)).

3.2 High-Endurance Flash

This device has a 128 byte section of high-endurance program Flash memory (PFM) in lieu of data EEPROM. This area is especially well suited for nonvolatile data storage that is expected to be updated frequently over the life of the end product. See [Section 10.2 “Flash Program Memory Overview”](#) for more information on writing data to PFM. See [Section 3.2.1.2 “Indirect Read with FSR”](#) for more information about using the FSR registers to read byte data stored in PFM.

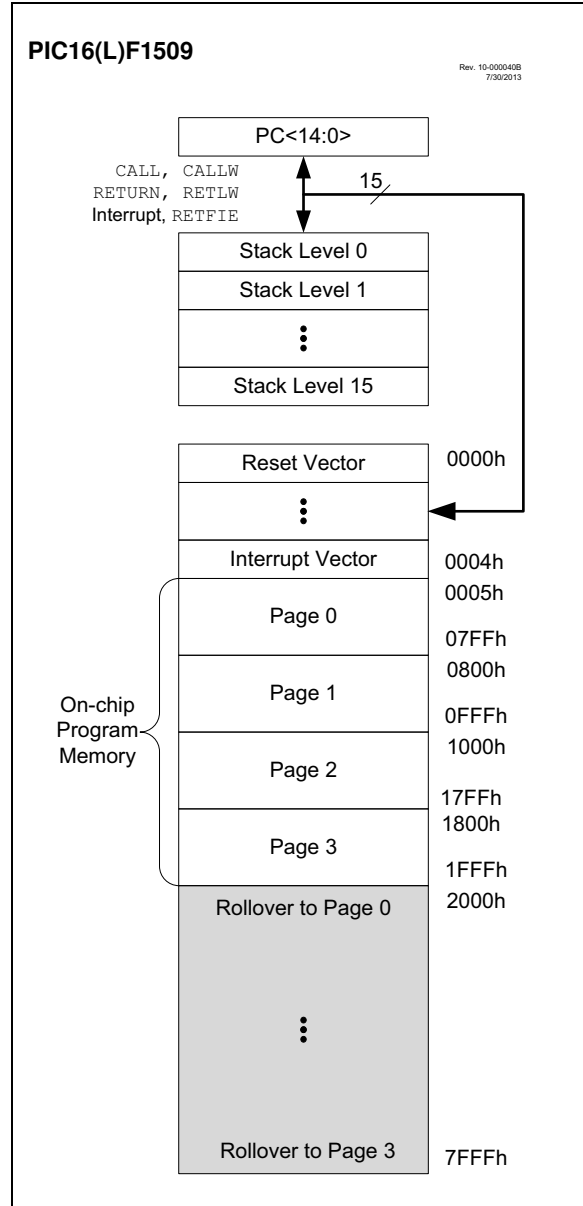
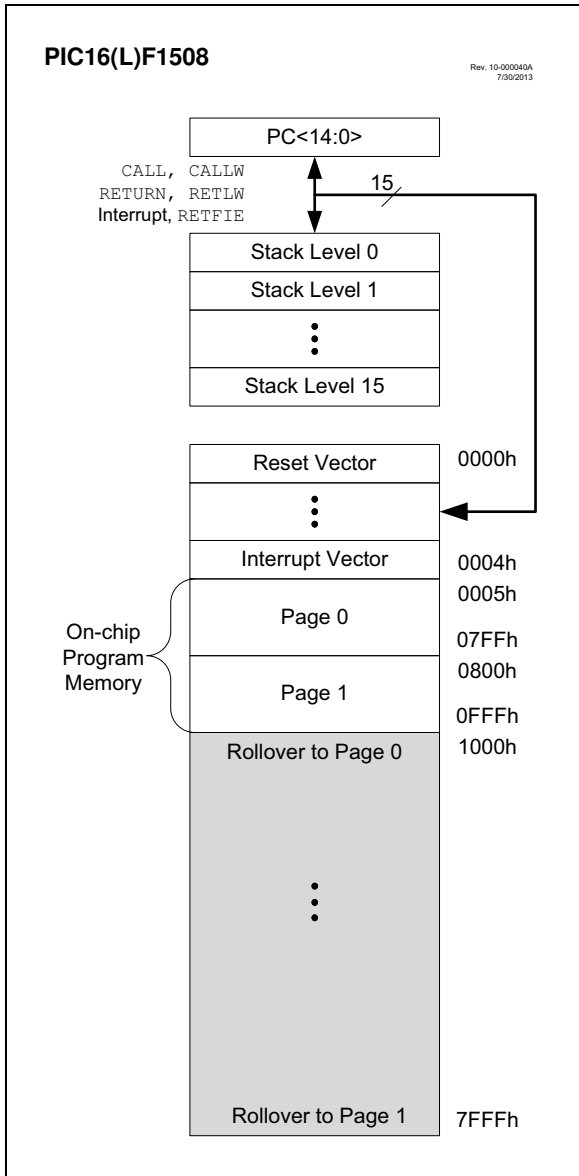
TABLE 3-1: DEVICE SIZES AND ADDRESSES

Device	Program Memory Space (Words)	Last Program Memory Address	High-Endurance Flash Memory Address Range ⁽¹⁾
PIC16LF1508 PIC16F1508	4,096	0FFFh	0F80h-0FFFh
PIC16LF1509 PIC16F1509	8,192	1FFFh	1F80h-1FFFh

Note 1: High-endurance Flash applies to low byte of each address in the range.

PIC16(L)F1508/9

FIGURE 3-1: PROGRAM MEMORY MAP AND STACK FOR PIC16(L)F1508



3.2.1 READING PROGRAM MEMORY AS DATA

There are two methods of accessing constants in program memory. The first method is to use tables of RETLW instructions. The second method is to set an FSR to point to the program memory.

3.2.1.1 RETLW Instruction

The RETLW instruction can be used to provide access to tables of constants. The recommended way to create such a table is shown in [Example 3-1](#).

EXAMPLE 3-1: RETLW INSTRUCTION

```
constants
    BRW                ;Add Index in W to
                      ;program counter to
                      ;select data
    RETLW DATA0       ;Index0 data
    RETLW DATA1       ;Index1 data
    RETLW DATA2
    RETLW DATA3

my_function
    ;... LOTS OF CODE...
    MOVLW DATA_INDEX
    call constants
    ;... THE CONSTANT IS IN W
```

The BRW instruction makes this type of table very simple to implement. If your code must remain portable with previous generations of microcontrollers, then the BRW instruction is not available so the older table read method must be used.

3.2.1.2 Indirect Read with FSR

The program memory can be accessed as data by setting bit 7 of the FSRxH register and reading the matching INDFx register. The MOVIW instruction will place the lower eight bits of the addressed word in the W register. Writes to the program memory cannot be performed via the INDF registers. Instructions that access the program memory via the FSR require one extra instruction cycle to complete. [Example 3-2](#) demonstrates accessing the program memory via an FSR.

The HIGH operator will set bit<7> if a label points to a location in program memory.

EXAMPLE 3-2: ACCESSING PROGRAM MEMORY VIA FSR

```
constants
    DW DATA0          ;First constant
    DW DATA1          ;Second constant
    DW DATA2
    DW DATA3

my_function
    ;... LOTS OF CODE...
    MOVLW DATA_INDEX
    ADDLW LOW constants
    MOVWF FSR1L
    MOVLW HIGH constants;MSb sets
                          automatically

    MOVWF FSR1H
    BTFSC STATUS, C      ;carry from ADDLW?
    INCF FSR1h, f        ;yes
    MOVIW 0[FSR1]
    ;THE PROGRAM MEMORY IS IN W
```

PIC16(L)F1508/9

3.3 Data Memory Organization

The data memory is partitioned in 32 memory banks with 128 bytes in a bank. Each bank consists of (Figure 3-2):

- 12 core registers
- 20 Special Function Registers (SFR)
- Up to 80 bytes of General Purpose RAM (GPR)
- 16 bytes of common RAM

The active bank is selected by writing the bank number into the Bank Select Register (BSR). Unimplemented memory will read as '0'. All data memory can be accessed either directly (via instructions that use the file registers) or indirectly via the two File Select Registers (FSR). See [Section 3.6 "Indirect Addressing"](#) for more information.

Data memory uses a 12-bit address. The upper five bits of the address define the Bank address and the lower seven bits select the registers/RAM in that bank.

3.3.1 CORE REGISTERS

The core registers contain the registers that directly affect the basic operation. The core registers occupy the first 12 addresses of every data memory bank (addresses x00h/x08h through x0Bh/x8Bh). These registers are listed below in [Table 3-2](#). For detailed information, see [Table 3-8](#).

TABLE 3-2: CORE REGISTERS

Addresses	BANKx
x00h or x80h	INDF0
x01h or x81h	INDF1
x02h or x82h	PCL
x03h or x83h	STATUS
x04h or x84h	FSR0L
x05h or x85h	FSR0H
x06h or x86h	FSR1L
x07h or x87h	FSR1H
x08h or x88h	BSR
x09h or x89h	WREG
x0Ah or x8Ah	PCLATH
x0Bh or x8Bh	INTCON

3.3.1.1 STATUS Register

The STATUS register, shown in [Register 3-1](#), contains:

- the arithmetic status of the ALU
- the Reset status

The STATUS register can be the destination for any instruction, like any other register. If the STATUS register is the destination for an instruction that affects the Z, DC or C bits, then the write to these three bits is disabled. These bits are set or cleared according to the device logic. Furthermore, the TO and PD bits are not writable. Therefore, the result of an instruction with the STATUS register as destination may be different than intended.

For example, `CLRF STATUS` will clear the upper three bits and set the Z bit. This leaves the STATUS register as '000u u1uu' (where u = unchanged).

It is recommended, therefore, that only `BCF`, `BSE`, `SWAPF` and `MOVWF` instructions are used to alter the STATUS register, because these instructions do not affect any Status bits. For other instructions not affecting any Status bits (Refer to [Section 28.0 "Instruction Set Summary"](#)).

Note 1: The C and DC bits operate as Borrow and Digit Borrow out bits, respectively, in subtraction.

REGISTER 3-1: STATUS: STATUS REGISTER

U-0	U-0	U-0	R-1/q	R-1/q	R/W-0/u	R/W-0/u	R/W-0/u
—	—	—	<u>TO</u>	<u>PD</u>	Z	<u>DC</u> ⁽¹⁾	<u>C</u> ⁽¹⁾
bit 7							bit 0

Legend:

R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	q = Value depends on condition

bit 7-5 **Unimplemented:** Read as '0'

bit 4 **TO:** Time-Out bit
 1 = After power-up, `CLRWDI` instruction or `SLEEP` instruction
 0 = A WDT time-out occurred

bit 3 **PD:** Power-Down bit
 1 = After power-up or by the `CLRWDI` instruction
 0 = By execution of the `SLEEP` instruction

bit 2 **Z:** Zero bit
 1 = The result of an arithmetic or logic operation is zero
 0 = The result of an arithmetic or logic operation is not zero

bit 1 **DC:** Digit Carry/Digit Borrow bit (`ADDWF`, `ADDLW`, `SUBLW`, `SUBWF` instructions)⁽¹⁾
 1 = A carry-out from the 4th low-order bit of the result occurred
 0 = No carry-out from the 4th low-order bit of the result

bit 0 **C:** Carry/Borrow bit⁽¹⁾ (`ADDWF`, `ADDLW`, `SUBLW`, `SUBWF` instructions)⁽¹⁾
 1 = A carry-out from the Most Significant bit of the result occurred
 0 = No carry-out from the Most Significant bit of the result occurred

Note 1: For Borrow, the polarity is reversed. A subtraction is executed by adding the two's complement of the second operand. For rotate (`RRF`, `RLF`) instructions, this bit is loaded with either the high-order or low-order bit of the source register.

PIC16(L)F1508/9

3.3.2 SPECIAL FUNCTION REGISTER

The Special Function Registers are registers used by the application to control the desired operation of peripheral functions in the device. The Special Function Registers occupy the 20 bytes after the core registers of every data memory bank (addresses x0Ch/x8Ch through x1Fh/x9Fh). The registers associated with the operation of the peripherals are described in the appropriate peripheral chapter of this data sheet.

3.3.3 GENERAL PURPOSE RAM

There are up to 80 bytes of GPR in each data memory bank. The Special Function Registers occupy the 20 bytes after the core registers of every data memory bank (addresses x0Ch/x8Ch through x1Fh/x9Fh).

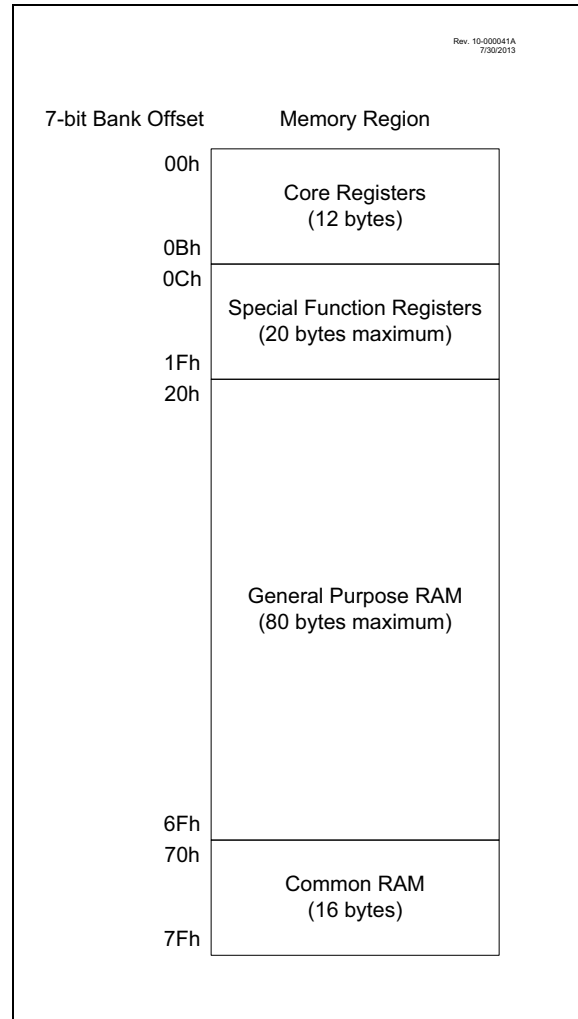
3.3.3.1 Linear Access to GPR

The general purpose RAM can be accessed in a non-banked method via the FSRs. This can simplify access to large memory structures. See [Section 3.6.2 “Linear Data Memory”](#) for more information.

3.3.4 COMMON RAM

There are 16 bytes of common RAM accessible from all banks.

FIGURE 3-2: BANKED MEMORY PARTITIONING



3.3.5 DEVICE MEMORY MAPS

The memory maps for Bank 0 through Bank 31 are shown in the tables in this section.

TABLE 3-3: PIC16(L)F1508 MEMORY MAP, BANK 0-7

BANK 0		BANK 1		BANK 2		BANK 3		BANK 4		BANK 5		BANK 6		BANK 7	
000h	Core Registers (Table 3-2)	080h	Core Registers (Table 3-2)	100h	Core Registers (Table 3-2)	180h	Core Registers (Table 3-2)	200h	Core Registers (Table 3-2)	280h	Core Registers (Table 3-2)	300h	Core Registers (Table 3-2)	380h	Core Registers (Table 3-2)
00Bh	—	08Bh	—	10Bh	—	18Bh	—	20Bh	—	28Bh	—	30Bh	—	38Bh	—
00Ch	PORTA	08Ch	TRISA	10Ch	LATA	18Ch	ANSELA	20Ch	WPUA	28Ch	—	30Ch	—	38Ch	—
00Dh	PORTB	08Dh	TRISB	10Dh	LATB	18Dh	ANSELB	20Dh	WPUB	28Dh	—	30Dh	—	38Dh	—
00Eh	PORTC	08Eh	TRISC	10Eh	LATC	18Eh	ANSELC	20Eh	—	28Eh	—	30Eh	—	38Eh	—
00Fh	—	08Fh	—	10Fh	—	18Fh	—	20Fh	—	28Fh	—	30Fh	—	38Fh	—
010h	—	090h	—	110h	—	190h	—	210h	—	290h	—	310h	—	390h	—
011h	PIR1	091h	PIE1	111h	CM1CON0	191h	PMADRL	211h	SSP1BUF	291h	—	311h	—	391h	IOCAP
012h	PIR2	092h	PIE2	112h	CM1CON1	192h	PMADRH	212h	SSP1ADD	292h	—	312h	—	392h	IOCAN
013h	PIR3	093h	PIE3	113h	CM2CON0	193h	PMDATL	213h	SSP1MSK	293h	—	313h	—	393h	IOCAF
014h	—	094h	—	114h	CM2CON1	194h	PMDATH	214h	SSP1STAT	294h	—	314h	—	394h	IOCBP
015h	TMR0	095h	OPTION_REG	115h	CMOUT	195h	PMCON1	215h	SSP1CON1	295h	—	315h	—	395h	IOCBN
016h	TMR1L	096h	PCON	116h	BORCON	196h	PMCON2	216h	SSP1CON2	296h	—	316h	—	396h	IOCBF
017h	TMR1H	097h	WDTCON	117h	FVRCON	197h	VREGCON	217h	SSP1CON3	297h	—	317h	—	397h	—
018h	T1CON	098h	—	118h	DAC1CON0	198h	—	218h	—	298h	—	318h	—	398h	—
019h	T1GCON	099h	OSCCON	119h	DAC1CON1	199h	RCREG	219h	—	299h	—	319h	—	399h	—
01Ah	TMR2	09Ah	OSCSTAT	11Ah	—	19Ah	TXREG	21Ah	—	29Ah	—	31Ah	—	39Ah	—
01Bh	PR2	09Bh	ADRESL	11Bh	—	19Bh	SPBRG	21Bh	—	29Bh	—	31Bh	—	39Bh	—
01Ch	T2CON	09Ch	ADRESH	11Ch	—	19Ch	SPBRGH	21Ch	—	29Ch	—	31Ch	—	39Ch	—
01Dh	—	09Dh	ADCON0	11Dh	APFCON	19Dh	RCSTA	21Dh	—	29Dh	—	31Dh	—	39Dh	—
01Eh	—	09Eh	ADCON1	11Eh	—	19Eh	TXSTA	21Eh	—	29Eh	—	31Eh	—	39Eh	—
01Fh	—	09Fh	ADCON2	11Fh	—	19Fh	BAUDCON	21Fh	—	29Fh	—	31Fh	—	39Fh	—
020h	General Purpose Register 80 Bytes	0A0h	General Purpose Register 80 Bytes	120h	General Purpose Register 80 Bytes	1A0h	Unimplemented Read as '0'	220h	Unimplemented Read as '0'	2A0h	Unimplemented Read as '0'	320h	Unimplemented Read as '0'	3A0h	Unimplemented Read as '0'
06Fh	—	0EFh	—	16Fh	—	1EFh	—	26Fh	—	2EFh	—	36Fh	—	3EFh	—
070h	Common RAM	0F0h	Common RAM (Accesses 70h – 7Fh)	170h	Common RAM (Accesses 70h – 7Fh)	1F0h	Common RAM (Accesses 70h – 7Fh)	270h	Common RAM (Accesses 70h – 7Fh)	2F0h	Common RAM (Accesses 70h – 7Fh)	370h	Common RAM (Accesses 70h – 7Fh)	3F0h	Common RAM (Accesses 70h – 7Fh)
07Fh	—	0FFh	—	17Fh	—	1FFh	—	27Fh	—	2FFh	—	37Fh	—	3FFh	—

Legend: ■ = Unimplemented data memory locations, read as '0'.

TABLE 3-4: PIC16(L)F1509 MEMORY MAP, BANK 0-7

BANK 0		BANK 1		BANK 2		BANK 3		BANK 4		BANK 5		BANK 6		BANK 7	
00h	Core Registers (Table 3-2)	080h	Core Registers (Table 3-2)	100h	Core Registers (Table 3-2)	180h	Core Registers (Table 3-2)	200h	Core Registers (Table 3-2)	280h	Core Registers (Table 3-2)	300h	Core Registers (Table 3-2)	380h	Core Registers (Table 3-2)
00Bh		08Bh		10Bh		18Bh		20Bh		28Bh		30Bh		38Bh	
00Ch	PORTA	08Ch	TRISA	10Ch	LATA	18Ch	ANSELA	20Ch	WPUA	28Ch	—	30Ch	—	38Ch	—
00Dh	PORTB	08Dh	TRISB	10Dh	LATB	18Dh	ANSELB	20Dh	WPUB	28Dh	—	30Dh	—	38Dh	—
00Eh	PORTC	08Eh	TRISC	10Eh	LATC	18Eh	ANSELC	20Eh	—	28Eh	—	30Eh	—	38Eh	—
00Fh	—	08Fh	—	10Fh	—	18Fh	—	20Fh	—	28Fh	—	30Fh	—	38Fh	—
010h	—	090h	—	110h	—	190h	—	210h	—	290h	—	310h	—	390h	—
011h	PIR1	091h	PIE1	111h	CM1CON0	191h	PMADRL	211h	SSP1BUF	291h	—	311h	—	391h	IOCAP
012h	PIR2	092h	PIE2	112h	CM1CON1	192h	PMADRH	212h	SSP1ADD	292h	—	312h	—	392h	IOCAN
013h	PIR3	093h	PIE3	113h	CM2CON0	193h	PMDATL	213h	SSP1MSK	293h	—	313h	—	393h	IOCAF
014h	—	094h	—	114h	CM2CON1	194h	PMDATH	214h	SSP1STAT	294h	—	314h	—	394h	IOCBP
015h	TMR0	095h	OPTION_REG	115h	CMOUT	195h	PMCON1	215h	SSP1CON1	295h	—	315h	—	395h	IOCBN
016h	TMR1L	096h	PCON	116h	BORCON	196h	PMCON2	216h	SSP1CON2	296h	—	316h	—	396h	IOCBF
017h	TMR1H	097h	WDTCON	117h	FVRCON	197h	VREGCON	217h	SSP1CON3	297h	—	317h	—	397h	—
018h	T1CON	098h	—	118h	DAC1CON0	198h	—	218h	—	298h	—	318h	—	398h	—
019h	T1GCON	099h	OSCCON	119h	DAC1CON1	199h	RCREG	219h	—	299h	—	319h	—	399h	—
01Ah	TMR2	09Ah	OSCSTAT	11Ah	—	19Ah	TXREG	21Ah	—	29Ah	—	31Ah	—	39Ah	—
01Bh	PR2	09Bh	ADRESL	11Bh	—	19Bh	SPBRG	21Bh	—	29Bh	—	31Bh	—	39Bh	—
01Ch	T2CON	09Ch	ADRESH	11Ch	—	19Ch	SPBRGH	21Ch	—	29Ch	—	31Ch	—	39Ch	—
01Dh	—	09Dh	ADCON0	11Dh	APFCON	19Dh	RCSTA	21Dh	—	29Dh	—	31Dh	—	39Dh	—
01Eh	—	09Eh	ADCON1	11Eh	—	19Eh	TXSTA	21Eh	—	29Eh	—	31Eh	—	39Eh	—
01Fh	—	09Fh	ADCON2	11Fh	—	19Fh	BAUDCON	21Fh	—	29Fh	—	31Fh	—	39Fh	—
020h	General Purpose Register 80 Bytes	0A0h	General Purpose Register 80 Bytes	120h	General Purpose Register 80 Bytes	1A0h	General Purpose Register 80 Bytes	220h	General Purpose Register 80 Bytes	2A0h	General Purpose Register 80 Bytes	320h	General Purpose Register 16Bytes	3A0h	Unimplemented Read as '0'
06Fh	Common RAM	0EFh	Accesses 70h – 7Fh	16Fh	Accesses 70h – 7Fh	1EFh	Accesses 70h – 7Fh	26Fh	Accesses 70h – 7Fh	2EFh	Accesses 70h – 7Fh	36Fh	Accesses 70h – 7Fh	3EFh	Accesses 70h – 7Fh
070h		0F0h		170h		1F0h		270h		2F0h		370h		3F0h	
07Fh		0FFh		17Fh		1FFh		27Fh		2FFh		37Fh		3FFh	

Legend: = Unimplemented data memory locations, read as '0'.

TABLE 3-5: PIC16(L)F1508/9 MEMORY MAP, BANK 8-23

BANK 8		BANK 9		BANK 10		BANK 11		BANK 12		BANK 13		BANK 14		BANK 15	
400h	Core Registers (Table 3-2)	480h	Core Registers (Table 3-2)	500h	Core Registers (Table 3-2)	580h	Core Registers (Table 3-2)	600h	Core Registers (Table 3-2)	680h	Core Registers (Table 3-2)	700h	Core Registers (Table 3-2)	780h	Core Registers (Table 3-2)
40Bh	—	48Bh	—	50Bh	—	58Bh	—	60Bh	—	68Bh	—	70Bh	—	78Bh	—
40Ch	—	48Ch	—	50Ch	—	58Ch	—	60Ch	—	68Ch	—	70Ch	—	78Ch	—
40Dh	—	48Dh	—	50Dh	—	58Dh	—	60Dh	—	68Dh	—	70Dh	—	78Dh	—
40Eh	—	48Eh	—	50Eh	—	58Eh	—	60Eh	—	68Eh	—	70Eh	—	78Eh	—
40Fh	—	48Fh	—	50Fh	—	58Fh	—	60Fh	—	68Fh	—	70Fh	—	78Fh	—
410h	—	490h	—	510h	—	590h	—	610h	—	690h	—	710h	—	790h	—
411h	—	491h	—	511h	—	591h	—	611h	PWM1DCL	691h	CWG1DBR	711h	—	791h	—
412h	—	492h	—	512h	—	592h	—	612h	PWM1DCH	692h	CWG1DBF	712h	—	792h	—
413h	—	493h	—	513h	—	593h	—	613h	PWM1CON	693h	CWG1CON0	713h	—	793h	—
414h	—	494h	—	514h	—	594h	—	614h	PWM2DCL	694h	CWG1CON1	714h	—	794h	—
415h	—	495h	—	515h	—	595h	—	615h	PWM2DCH	695h	CWG1CON2	715h	—	795h	—
416h	—	496h	—	516h	—	596h	—	616h	PWM2CON	696h	—	716h	—	796h	—
417h	—	497h	—	517h	—	597h	—	617h	PWM3DCL	697h	—	717h	—	797h	—
418h	—	498h	NCO1ACCL	518h	—	598h	—	618h	PWM3DCH	698h	—	718h	—	798h	—
419h	—	499h	NCO1ACCH	519h	—	599h	—	619h	PWM3CON	699h	—	719h	—	799h	—
41Ah	—	49Ah	NCO1ACCU	51Ah	—	59Ah	—	61Ah	PWM4DCL	69Ah	—	71Ah	—	79Ah	—
41Bh	—	49Bh	NCO1INCL	51Bh	—	59Bh	—	61Bh	PWM4DCH	69Bh	—	71Bh	—	79Bh	—
41Ch	—	49Ch	NCO1INCH	51Ch	—	59Ch	—	61Ch	PWM4CON	69Ch	—	71Ch	—	79Ch	—
41Dh	—	49Dh	—	51Dh	—	59Dh	—	61Dh	—	69Dh	—	71Dh	—	79Dh	—
41Eh	—	49Eh	NCO1CON	51Eh	—	59Eh	—	61Eh	—	69Eh	—	71Eh	—	79Eh	—
41Fh	—	49Fh	NCO1CLK	51Fh	—	59Fh	—	61Fh	—	69Fh	—	71Fh	—	79Fh	—
420h	Unimplemented Read as '0'	4A0h	Unimplemented Read as '0'	520h	Unimplemented Read as '0'	5A0h	Unimplemented Read as '0'	620h	Unimplemented Read as '0'	6A0h	Unimplemented Read as '0'	720h	Unimplemented Read as '0'	7A0h	Unimplemented Read as '0'
46Fh	Accesses 70h – 7Fh	4EFh	Accesses 70h – 7Fh	56Fh	Accesses 70h – 7Fh	5EFh	Accesses 70h – 7Fh	66Fh	Accesses 70h – 7Fh	6EFh	Accesses 70h – 7Fh	76Fh	Accesses 70h – 7Fh	7EFh	Accesses 70h – 7Fh
470h	—	4F0h	—	570h	—	5F0h	—	670h	—	6F0h	—	770h	—	7F0h	—
47Fh	—	4FFh	—	57Fh	—	5FFh	—	67Fh	—	6FFh	—	77Fh	—	7FFh	—
BANK 16		BANK 17		BANK 18		BANK 19		BANK 20		BANK 21		BANK 22		BANK 23	
800h	Core Registers (Table 3-2)	880h	Core Registers (Table 3-2)	900h	Core Registers (Table 3-2)	980h	Core Registers (Table 3-2)	A00h	Core Registers (Table 3-2)	A80h	Core Registers (Table 3-2)	B00h	Core Registers (Table 3-2)	B80h	Core Registers (Table 3-2)
80Bh	—	88Bh	—	90Bh	—	98Bh	—	A0Bh	—	A8Bh	—	B0Bh	—	B8Bh	—
80Ch	Unimplemented Read as '0'	88Ch	Unimplemented Read as '0'	90Ch	Unimplemented Read as '0'	98Ch	Unimplemented Read as '0'	A0Ch	Unimplemented Read as '0'	A8Ch	Unimplemented Read as '0'	B0Ch	Unimplemented Read as '0'	B8Ch	Unimplemented Read as '0'
86Fh	Accesses 70h – 7Fh	8EFh	Accesses 70h – 7Fh	96Fh	Accesses 70h – 7Fh	9EFh	Accesses 70h – 7Fh	A6Fh	Accesses 70h – 7Fh	A6Fh	Accesses 70h – 7Fh	B6Fh	Accesses 70h – 7Fh	BEFh	Accesses 70h – 7Fh
870h	—	8F0h	—	970h	—	9F0h	—	A70h	—	A70h	—	B70h	—	BF0h	—
87Fh	—	8FFh	—	97Fh	—	9FFh	—	A7Fh	—	A7Fh	—	B7Fh	—	BFh	—

Legend: ■ = Unimplemented data memory locations, read as '0'.

TABLE 3-6: PIC16(L)F1508/9 MEMORY MAP, BANK 24-31

BANK 24		BANK 25		BANK 26		BANK 27		BANK 28		BANK 29		BANK 30		BANK 31	
C00h	Core Registers (Table 3-2)	C80h	Core Registers (Table 3-2)	D00h	Core Registers (Table 3-2)	D80h	Core Registers (Table 3-2)	E00h	Core Registers (Table 3-2)	E80h	Core Registers (Table 3-2)	F00h	Core Registers (Table 3-2)	F80h	Core Registers (Table 3-2)
C0Bh	—	C8Bh	—	D0Bh	—	D8Bh	—	E0Bh	—	E8Bh	—	F0Bh	—	F8Bh	—
C0Ch	—	C8Ch	—	D0Ch	—	D8Ch	—	E0Ch	—	E8Ch	—	F0Ch	—	F8Ch	—
C0Dh	—	C8Dh	—	D0Dh	—	D8Dh	—	E0Dh	—	E8Dh	—	F0Dh	—	F8Dh	—
C0Eh	—	C8Eh	—	D0Eh	—	D8Eh	—	E0Eh	—	E8Eh	—	F0Eh	—	F8Eh	—
C0Fh	—	C8Fh	—	D0Fh	—	D8Fh	—	E0Fh	—	E8Fh	—	F0Fh	—	F8Fh	—
C10h	—	C90h	—	D10h	—	D90h	—	E10h	—	E90h	—	F10h	—	F90h	—
C11h	—	C91h	—	D11h	—	D91h	—	E11h	—	E91h	—	F11h	—	F91h	—
C12h	—	C92h	—	D12h	—	D92h	—	E12h	—	E92h	—	F12h	—	F92h	—
C13h	—	C93h	—	D13h	—	D93h	—	E13h	—	E93h	—	F13h	—	F93h	—
C14h	—	C94h	—	D14h	—	D94h	—	E14h	—	E94h	—	F14h	—	F94h	—
C15h	—	C95h	—	D15h	—	D95h	—	E15h	—	E95h	—	F15h	—	F95h	—
C16h	—	C96h	—	D16h	—	D96h	—	E16h	—	E96h	—	F16h	—	F96h	—
C17h	—	C97h	—	D17h	—	D97h	—	E17h	—	E97h	—	F17h	—	F97h	—
C18h	—	C98h	—	D18h	—	D98h	—	E18h	—	E98h	—	F18h	—	F98h	—
C19h	—	C99h	—	D19h	—	D99h	—	E19h	—	E99h	—	F19h	—	F99h	—
C1Ah	—	C9Ah	—	D1Ah	—	D9Ah	—	E1Ah	—	E9Ah	—	F1Ah	—	F9Ah	—
C1Bh	—	C9Bh	—	D1Bh	—	D9Bh	—	E1Bh	—	E9Bh	—	F1Bh	—	F9Bh	—
C1Ch	—	C9Ch	—	D1Ch	—	D9Ch	—	E1Ch	—	E9Ch	—	F1Ch	—	F9Ch	—
C1Dh	—	C9Dh	—	D1Dh	—	D9Dh	—	E1Dh	—	E9Dh	—	F1Dh	—	F9Dh	—
C1Eh	—	C9Eh	—	D1Eh	—	D9Eh	—	E1Eh	—	E9Eh	—	F1Eh	—	F9Eh	—
C1Fh	—	C9Fh	—	D1Fh	—	D9Fh	—	E1Fh	—	E9Fh	—	F1Fh	—	F9Fh	—
C20h	Unimplemented Read as '0'	CA0h	Unimplemented Read as '0'	D20h	Unimplemented Read as '0'	DA0h	Unimplemented Read as '0'	E20h	Unimplemented Read as '0'	EA0h	Unimplemented Read as '0'	F20h	—	FA0h	—
C6Fh	—	CEFh	—	D6Fh	—	DEFh	—	E6Fh	—	EEFh	—	F6Fh	—	FEFh	—
C70h	Accesses 70h – 7Fh	CF0h	Accesses 70h – 7Fh	D70h	Accesses 70h – 7Fh	DF0h	Accesses 70h – 7Fh	E70h	Accesses 70h – 7Fh	EF0h	Accesses 70h – 7Fh	F70h	Accesses 70h – 7Fh	FF0h	Accesses 70h – 7Fh
CFh	—	CFh	—	D7Fh	—	DFh	—	E7Fh	—	EFh	—	F7Fh	—	FFh	—

Legend: ■ = Unimplemented data memory locations, read as '0'.

TABLE 3-7: PIC16(L)F1508/9 MEMORY MAP, BANK 30-31

Bank 30		Bank 31		
F0Ch	—	F8Ch	Unimplemented Read as '0'	
F0Dh	—			
F0Eh	—			
F0Fh	CLCDATA			
F10h	CLC1CON			
F11h	CLC1POL	FE3h		
F12h	CLC1SEL0	FE4h		STATUS_SHAD
F13h	CLC1SEL1	FE5h		WREG_SHAD
F14h	CLC1GLS0	FE6h		BSR_SHAD
F15h	CLC1GLS1	FE7h		PCLATH_SHAD
F16h	CLC1GLS2	FE8h		FSR0L_SHAD
F17h	CLC1GLS3	FE9h		FSR0H_SHAD
F18h	CLC2CON	FEAh	FSR1L_SHAD	
F19h	CLC2POL	FEBh	FSR1H_SHAD	
F1Ah	CLC2SEL0	FECh	—	
F1Bh	CLC2SEL1	FEDh	STKPTR	
F1Ch	CLC2GLS0	FEEh	TOSL	
F1Dh	CLC2GLS1	FEFh	TOSH	
F1Eh	CLC2GLS2			
F1Fh	CLC2GLS3			
F20h	CLC3CON			
F21h	CLC3POL			
F22h	CLC3SEL0			
F23h	CLC3SEL1			
F24h	CLC3GLS0			
F25h	CLC3GLS1			
F26h	CLC3GLS2			
F27h	CLC3GLS3			
F28h	CLC4CON			
F29h	CLC4POL			
F2Ah	CLC4SEL0			
F2Bh	CLC4SEL1			
F2Ch	CLC4GLS0			
F2Dh	CLC4GLS1			
F2Eh	CLC4GLS2			
F2Fh	CLC4GLS3			
F30h	Unimplemented Read as '0'			
F6Fh				

Legend: = Unimplemented data memory locations, read as '0'.

PIC16(L)F1508/9

3.3.6 CORE FUNCTION REGISTERS SUMMARY

The Core Function registers listed in [Table 3-8](#) can be addressed from any Bank.

TABLE 3-8: CORE FUNCTION REGISTERS SUMMARY

Addr	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other Resets	
Bank 0-31												
x00h or x80h	INDF0	Addressing this location uses contents of FSR0H/FSR0L to address data memory (not a physical register)								xxxx xxxx	uuuu uuuu	
x01h or x81h	INDF1	Addressing this location uses contents of FSR1H/FSR1L to address data memory (not a physical register)								xxxx xxxx	uuuu uuuu	
x02h or x82h	PCL	Program Counter (PC) Least Significant Byte								0000 0000	0000 0000	
x03h or x83h	STATUS	—	—	—	\overline{TO}	\overline{PD}	Z	DC	C	---1 1000	---q quuu	
x04h or x84h	FSR0L	Indirect Data Memory Address 0 Low Pointer								0000 0000	uuuu uuuu	
x05h or x85h	FSR0H	Indirect Data Memory Address 0 High Pointer								0000 0000	0000 0000	
x06h or x86h	FSR1L	Indirect Data Memory Address 1 Low Pointer								0000 0000	uuuu uuuu	
x07h or x87h	FSR1H	Indirect Data Memory Address 1 High Pointer								0000 0000	0000 0000	
x08h or x88h	BSR	—	—	—	BSR<4:0>				---	0 0000	---	0 0000
x09h or x89h	WREG	Working Register								0000 0000	uuuu uuuu	
x0Ah or x8Ah	PCLATH	—	Write Buffer for the upper 7 bits of the Program Counter								-000 0000	-000 0000
x0Bh or x8Bh	INTCON	GIE	PEIE	TMR0IE	INTE	IOCF	TMR0IF	INTF	IOCF	0000 0000	0000 0000	

Legend: x = unknown, u = unchanged, q = value depends on condition, - = unimplemented, read as '0', r = reserved. Shaded locations are unimplemented, read as '0'.