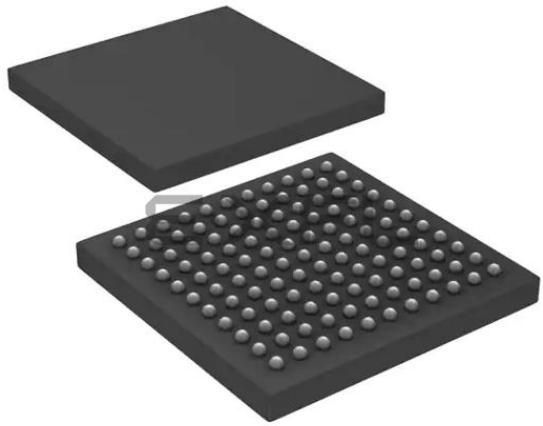
Microchip Technology - PIC24FJ512GB610T-I/BG Datasheet



Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Active
Core Processor	PIC
Core Size	16-Bit
Speed	32MHz
Connectivity	I²C, IrDA, LINbus, PMP, SPI, UART/USART, USB OTG
Peripherals	Brown-out Detect/Reset, DMA, HLVD, POR, PWM, WDT
Number of I/O	85
Program Memory Size	512KB (170K x 24)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	32K x 8
Voltage - Supply (Vcc/Vdd)	2V ~ 3.6V
Data Converters	A/D 24x10/12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	121-TFBGA
Supplier Device Package	121-TFBGA (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic24fj512gb610t-i-bg

4.2 Data Memory Space

Note: This data sheet summarizes the features of this group of PIC24F devices. It is not intended to be a comprehensive reference source. For more information, refer to the *"dsPIC33/PIC24 Family Reference Manual"*, **"Data Memory with Extended Data Space (EDS)"** (DS39733). The information in this data sheet supersedes the information in the FRM.

The PIC24F core has a 16-bit wide data memory space, addressable as a single linear range. The Data Space is accessed using two Address Generation Units (AGUs), one each for read and write operations. The Data Space memory map is shown in Figure 4-3.

The 16-bit wide data addresses in the data memory space point to bytes within the Data Space (DS). This gives a DS address range of 32 Kbytes or 16K words. The lower half (0000h to 7FFFh) is used for implemented (on-chip) memory addresses.

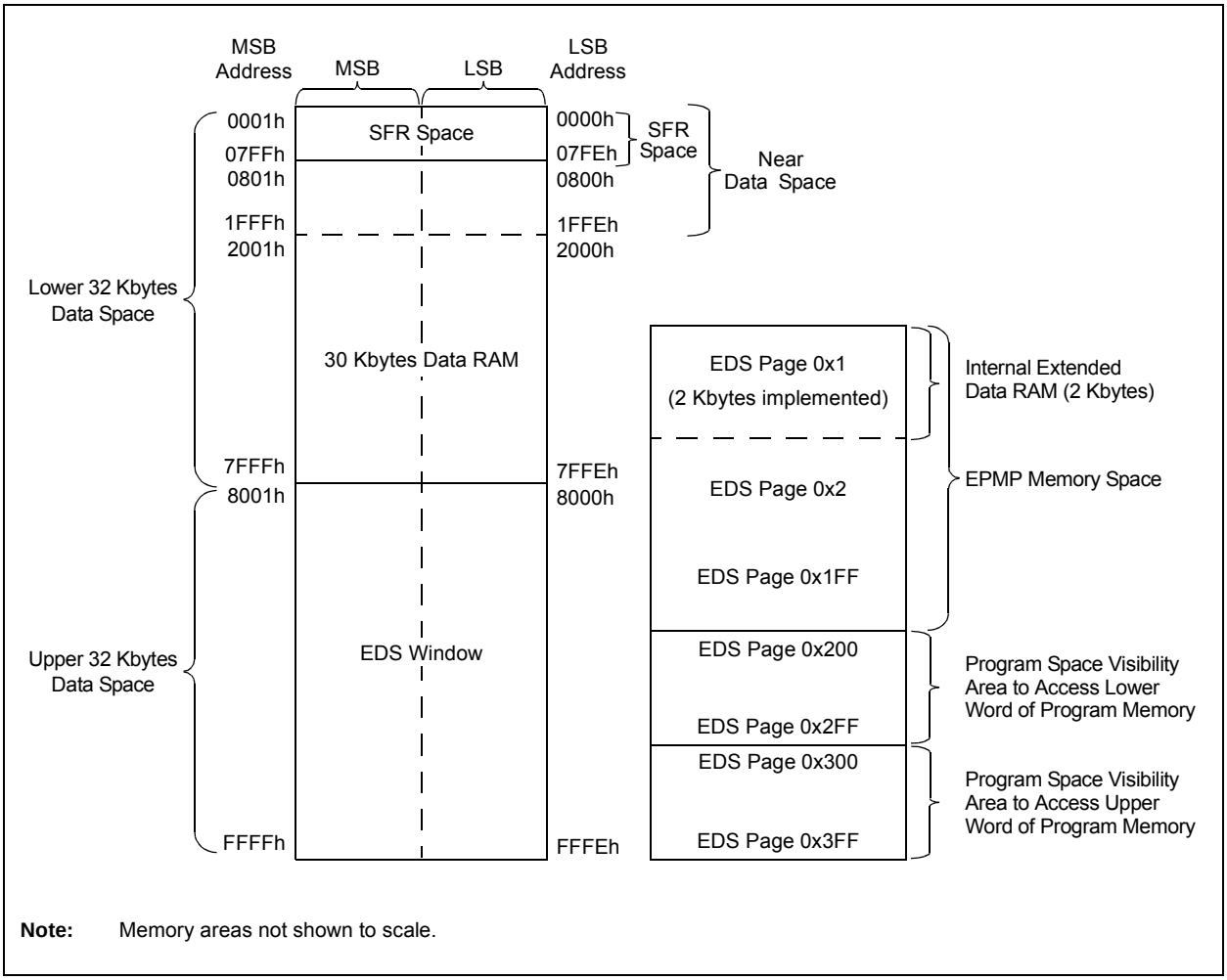
The upper half of data memory address space (8000h to FFFFh) is used as a window into the Extended Data Space (EDS). This allows the microcontroller to directly access a greater range of data beyond the standard 16-bit address range. EDS is discussed in detail in **Section 4.2.5 "Extended Data Space (EDS)"**.

The lower half of DS is compatible with previous PIC24F microcontrollers without EDS. All PIC24FJ1024GA610/GB610 family devices implement 30 Kbytes of data RAM in the lower half of DS, from 0800h to 7FFF.

4.2.1 DATA SPACE WIDTH

The data memory space is organized in byte-addressable, 16-bit wide blocks. Data is aligned in data memory and registers as 16-bit words, but all Data Space EAs resolve to bytes. The Least Significant Bytes (LSBs) of each word have even addresses, while the Most Significant Bytes (MSBs) have odd addresses.

FIGURE 4-3: DATA SPACE MEMORY MAP FOR PIC24FJ1024GA610/GB610 FAMILY DEVICES



Note: Memory areas not shown to scale.

6.2 RTSP Operation

The PIC24F Flash program memory array is organized into rows of 128 instructions or 384 bytes. RTSP allows the user to erase blocks of eight rows (1024 instructions) at a time and to program one row at a time. It is also possible to program two instruction word blocks.

The 8-row erase blocks and single row write blocks are edge-aligned, from the beginning of program memory, on boundaries of 3072 bytes and 384 bytes, respectively.

When data is written to program memory using TBLWT instructions, the data is not written directly to memory. Instead, data written using Table Writes is stored in holding latches until the programming sequence is executed.

Any number of TBLWT instructions can be executed and a write will be successfully performed. However, 128 TBLWT instructions are required to write the full row of memory.

To ensure that no data is corrupted during a write, any unused address should be programmed with FFFFFFh. This is because the holding latches reset to an unknown state, so if the addresses are left in the Reset state, they may overwrite the locations on rows which were not rewritten.

The basic sequence for RTSP programming is to set the Table Pointer to point to the programming latches, do a series of TBLWT instructions to load the buffers and set the NVMADRU/NVMADR registers to point to the destination. Programming is performed by setting the control bits in the NVMCON register.

Data can be loaded in any order and the holding registers can be written to multiple times before performing a write operation. Subsequent writes, however, will wipe out any previous writes.

Note:	Writing to a location multiple times without erasing is *not* recommended.

All of the Table Write operations are single-word writes (2 instruction cycles), because only the buffers are written. A programming cycle is required for programming each row.

6.3 JTAG Operation

The PIC24F family supports JTAG boundary scan. Boundary scan can improve the manufacturing process by verifying pin to PCB connectivity.

6.4 Enhanced In-Circuit Serial Programming

Enhanced In-Circuit Serial Programming uses an on-board bootloader, known as the Program Executive (PE), to manage the programming process. Using an SPI data frame format, the Program Executive can erase, program and verify program memory. For more information on Enhanced ICSP, see the device programming specification.

6.5 Control Registers

There are four SFRs used to read and write the program Flash memory: NVMCON, NVMADRU, NVMADR and NVMKEY.

The NVMCON register (Register 6-1) controls which blocks are to be erased, which memory type is to be programmed and when the programming cycle starts.

NVMKEY is a write-only register that is used for write protection. To start a programming or erase sequence, the user must consecutively write 55h and AAh to the NVMKEY register. Refer to **Section 6.6 "Programming Operations"** for further details.

The NVMADRU/NVMADR registers contain the upper byte and lower word of the destination of the NVM write or erase operation. Some operations (chip erase, Inactive Partition erase) operate on fixed locations and do not require an address value.

6.6 Programming Operations

A complete programming sequence is necessary for programming or erasing the internal Flash in RTSP mode. During a programming or erase operation, the processor stalls (waits) until the operation is finished. Setting the WR bit (NVMCON<15>) starts the operation and the WR bit is automatically cleared when the operation is finished.

In Dual Partition mode, programming or erasing the Inactive Partition will not stall the processor; the code in the Active Partition will still execute during the programming operation.

It is important to mask interrupts for a minimum of 5 instruction cycles during Flash programming. This can be done in Assembly using the DISI instruction (see Example 6-1).

TABLE 7-3: RESET DELAY TIMES FOR VARIOUS DEVICE RESETS

Reset Type	Clock Source	$\overline{\text{SYSRST}}$ Delay	System Clock Delay	Notes
POR	EC	TPOR + TSTARTUP + TRST	—	**1, 2, 3**
	ECPLL	TPOR + TSTARTUP + TRST	TLOCK	**1, 2, 3, 5**
	XT, HS, SOSC	TPOR + TSTARTUP + TRST	TOST	**1, 2, 3, 4**
	XTPLL, HSPLL	TPOR + TSTARTUP + TRST	TOST + TLOCK	**1, 2, 3, 4, 5**
	FRC, OSCFDIV	TPOR + TSTARTUP + TRST	TFRC	**1, 2, 3, 6, 7**
	FRCPLL	TPOR + TSTARTUP + TRST	TFRC + TLOCK	**1, 2, 3, 5, 6**
	LPRC	TPOR + TSTARTUP + TRST	TLPRC	**1, 2, 3, 6**
	DCO	TPOR + TSTARTUP + TRST	TDCO	**1, 2, 3, 8**
BOR	EC	TSTARTUP + TRST	—	**2, 3**
	ECPLL	TSTARTUP + TRST	TLOCK	**2, 3, 5**
	XT, HS, SOSC	TSTARTUP + TRST	TOST	**2, 3, 4**
	XTPLL, HSPLL	TSTARTUP + TRST	TOST + TLOCK	**2, 3, 4, 5**
	FRC, OSCFDIV	TSTARTUP + TRST	TFRC	**2, 3, 6, 7**
	FRCPLL	TSTARTUP + TRST	TFRC + TLOCK	**2, 3, 5, 6**
	LPRC	TSTARTUP + TRST	TLPRC	**2, 3, 6**
	DCO	TPOR + TSTARTUP + TRST	TDCO	**1, 2, 3, 8**
$\overline{\text{MCLR}}$	Any Clock	TRST	—	**3**
WDT	Any Clock	TRST	—	**3**
Software	Any clock	TRST	—	**3**
Illegal Opcode	Any Clock	TRST	—	**3**
Uninitialized W	Any Clock	TRST	—	**3**
Trap Conflict	Any Clock	TRST	—	**3**

Note 1: TPOR = Power-on Reset Delay (10 μs nominal).

2: TSTARTUP = TVREG.

3: TRST = Internal State Reset Time (2 μs nominal).

4: TOST = Oscillator Start-up Timer (OST). A 10-bit counter counts 1024 oscillator periods before releasing the oscillator clock to the system.

5: TLOCK = PLL Lock Time.

6: TFRC and TLPRC = RC Oscillator Start-up Times.

7: If Two-Speed Start-up is enabled, regardless of the Primary Oscillator selected, the device starts with FRC so the system clock delay is just TFRC, and in such cases, FRC start-up time is valid; it switches to the Primary Oscillator after its respective clock delay.

8: TDCO = DCO Start-up and Stabilization Times.

TABLE 9-1: CONFIGURATION BIT VALUES FOR CLOCK SELECTION

Oscillator Mode	Oscillator Source	FNOSC<2:0>	Notes
Oscillator with Frequency Division (OSCFDIV)	Internal/External	111	**1, 2, 3**
Digitally Controlled Oscillator (DCO)	Internal	110	**3**
Low-Power RC Oscillator (LPRC)	Internal	101	**3**
Secondary (Timer1) Oscillator (SOSC)	Secondary	100	**3**
Primary Oscillator (XT, HS or EC) with PLL Module	Primary	011	**4**
Primary Oscillator (XT, HS or EC)	Primary	010	**4**
Fast RC Oscillator with PLL Module (FRCPLL)	Internal	001	**3**
Fast RC Oscillator (FRC)	Internal	000	**3**

Note 1: The input oscillator to the OSCFDIV Clock mode is determined by the RCDIV<2:0> (CLKDIV<10:8) bits. At POR, the default value selects the FRC module.

2: This is the default oscillator mode for an unprogrammed (erased) device.

3: OSCO pin function is determined by the OSCIOFCN Configuration bit.

4: The POSCMD<1:0> Configuration bits select the oscillator driver mode (XT, HS or EC).

9.3 Control Registers

The operation of the oscillator is controlled by five Special Function Registers:

- OSCCON
- CLKDIV
- OSCTUN
- OSCDIV
- OSCFDIV

In addition, two registers are used to control the DCO:

- DCOCON
- DCOTUN

The OSCCON register (Register 9-1) is the main control register for the oscillator. It controls clock source switching and allows the monitoring of clock sources. OSCCON is protected by a write lock to prevent inadvertent clock switches. See **Section 9.4 “Clock Switching Operation”** for more information.

The CLKDIV register (Register 9-2) controls the features associated with Doze mode, as well as the postscalers for the OSCFDIV Clock mode and the PLL module.

The OSCTUN register (Register 9-3) allows the user to fine-tune the FRC Oscillator over a range of approximately ±1.5%. It also controls the FRC self-tuning features described in **Section 9.5 “FRC Active Clock Tuning”**.

The OSCDIV and OSCFDIV registers provide control for the system Oscillator Frequency Divider.

9.3.1 DCO OVERVIEW

The DCO (Digitally Controlled Oscillator) is a low-power alternative to the FRC. It can generate a wider selection of operating frequencies and can be trimmed to correct process variations if an exact frequency is required. However, the DCO is not designed for use with USB applications and cannot meet USB timing restrictions.

10.0 POWER-SAVING FEATURES

Note: This data sheet summarizes the features of this group of PIC24F devices. It is not intended to be a comprehensive reference source. For more information, refer to the *"dsPIC33/PIC24 Family Reference Manual"*, **"Power-Saving Features"** (DS39698), which is available from the Microchip web site (www.microchip.com). The information in this data sheet supersedes the information in the FRM.

The PIC24FJ1024GA610/GB610 family of devices provides the ability to manage power consumption by selectively managing clocking to the CPU and the peripherals. In general, a lower clock frequency and a reduction in the number of circuits being clocked constitutes lower consumed power. All PIC24F devices manage power consumption in four different ways:

- Clock Frequency
- Instruction-Based Sleep and Idle modes
- Software-Controlled Doze mode
- Selective Peripheral Control in Software

Combinations of these methods can be used to selectively tailor an application's power consumption, while still maintaining critical application features, such as timing-sensitive communications.

10.1 Clock Frequency and Clock Switching

PIC24F devices allow for a wide range of clock frequencies to be selected under application control. If the system clock configuration is not locked, users can choose low-power or high-precision oscillators by simply changing the NOSC<2:0> bits. The process of changing a system clock during operation, as well as limitations to the process, are discussed in more detail in **Section 9.0 "Oscillator Configuration"**.

10.2 Instruction-Based Power-Saving Modes

PIC24F devices have two special power-saving modes that are entered through the execution of a special PWRSAV instruction. Sleep mode stops clock operation and halts all code execution; Idle mode halts the CPU and code execution, but allows peripheral modules to continue operation. The assembly syntax of the PWRSAV instruction is shown in Example 10-1.

The XC16 C compiler offers "built-in" functions for the power-saving modes as follows:

```
Idle();          // places part in Idle
Sleep();         // places part in Sleep
```

Sleep and Idle modes can be exited as a result of an enabled interrupt, WDT time-out or a device Reset. When the device exits these modes, it is said to "wake-up".

Note: SLEEP_MODE and IDLE_MODE are constants defined in the assembler include file for the selected device.

10.2.1 SLEEP MODE

Sleep mode has these features:

- The system clock source is shut down. If an on-chip oscillator is used, it is turned off.
- The device current consumption will be reduced to a minimum provided that no I/O pin is sourcing current.
- The Fail-Safe Clock Monitor does not operate during Sleep mode since the system clock source is disabled.
- The LPRC clock will continue to run in Sleep mode if the WDT is enabled.
- The WDT, if enabled, is automatically cleared prior to entering Sleep mode.
- Some device features or peripherals may continue to operate in Sleep mode. This includes items, such as the Input Change Notification (ICN) on the I/O ports or peripherals that use an external clock input. Any peripheral that requires the system clock source for its operation will be disabled in Sleep mode.

The device will wake-up from Sleep mode on any of the these events:

- On any interrupt source that is individually enabled
- On any form of device Reset
- On a WDT time-out

On wake-up from Sleep, the processor will restart with the same clock source that was active when Sleep mode was entered.

EXAMPLE 10-1: PWRSAV INSTRUCTION SYNTAX

```
PWRSAV    #SLEEP_MODE            ; Put the device into SLEEP mode
PWRSAV    #IDLE_MODE             ; Put the device into IDLE mode
```

TABLE 10-2: PERIPHERAL MODULE DISABLE REGISTER SUMMARY

Register	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
PMD1	T5MD	T4MD	T3MD	T2MD	T1MD	—	—	—	I2C1MD	U2MD	U1MD	SPI2MD	SPI1MD	—	—	ADCMD	0000
PMD2	—	—	IC6MD	IC5MD	IC4MD	IC3MD	IC2MD	IC1MD	—	—	OC6MD	OC5MD	OC4MD	OC3MD	OC2MD	OC1MD	0000
PMD3	—	—	—	—	—	CMPMD	RTCCMD	PMPMD	CRCMD	—	—	—	U3MD	I2C3MD	I2C2MD	—	0000
PMD4	—	—	—	—	—	—	—	—	—	—	U4MD	—	REFOMD	CTMUMD	LVDMD	USBMD(1)	0000
PMD5	—	—	—	—	—	—	—	—	—	CCP7MD	CCP6MD	CCP5MD	CCP4MD	CCP3MD	CCP2MD	CCP1MD	0000
PMD6	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SPI3MD	0000
PMD7	—	—	—	—	—	—	—	—	—	—	DMA1MD	DMA0MD	—	—	—	—	0000
PMD8	—	—	—	—	—	—	—	—	U6MD	U5MD	CLC4MD	CLC3MD	CLC2MD	CLC1MD	—	—	0000

Legend: — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

Note 1: USB is not present on PIC24FJXXXXGA6XX devices.

REGISTER 11-44: RPOR8: PERIPHERAL PIN SELECT OUTPUT REGISTER 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	RP17R5	RP17R4	RP17R3	RP17R2	RP17R1	RP17R0
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	RP16R5	RP16R4	RP16R3	RP16R2	RP16R1	RP16R0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15-14 **Unimplemented:** Read as '0'

bit 13-8 **RP17R<5:0>:** RP17 Output Pin Mapping bits
Peripheral Output Number n is assigned to pin, RP17 (see Table 11-4 for peripheral function numbers).

bit 7-6 **Unimplemented:** Read as '0'

bit 5-0 **RP16R<5:0>:** RP16 Output Pin Mapping bits
Peripheral Output Number n is assigned to pin, RP16 (see Table 11-4 for peripheral function numbers).

REGISTER 11-45: RPOR9: PERIPHERAL PIN SELECT OUTPUT REGISTER 9

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	RP19R5	RP19R4	RP19R3	RP19R2	RP19R1	RP19R0
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	RP18R5	RP18R4	RP18R3	RP18R2	RP18R1	RP18R0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15-14 **Unimplemented:** Read as '0'

bit 13-8 **RP19R<5:0>:** RP19 Output Pin Mapping bits
Peripheral Output Number n is assigned to pin, RP19 (see Table 11-4 for peripheral function numbers).

bit 7-6 **Unimplemented:** Read as '0'

bit 5-0 **RP18R<5:0>:** RP18 Output Pin Mapping bits
Peripheral Output Number n is assigned to pin, RP18 (see Table 11-4 for peripheral function numbers).

TABLE 16-5: SYNCHRONIZATION SOURCES

SYNC<4:0>	Synchronization Source
11111	None; Timer with Rollover on CCPxPR Match or FFFFh
11110	Reserved
11101	Reserved
11100	CTMU Trigger
11011	A/D Start Conversion
11010	CMP3 Trigger
11001	CMP2 Trigger
11000	CMP1 Trigger
10111	Reserved
10110	Reserved
10101	Reserved
10100	Reserved
10011	CLC4 Out
10010	CLC3 Out
10001	CLC2 Out
10000	CLC1 Out
01111	Reserved
01110	Reserved
01101	Reserved
01100	Reserved
01011	INT2 Pad
01010	INT1 Pad
01001	INT0 Pad
01000	SCCP7 Sync Out
00111	SCCP6 Sync Out
00110	SCCP5 Sync Out
00101	SCCP4 Sync Out
00100	MCCP3 Sync Out
00011	MCCP2 Sync Out
00010	MCCP1 Sync Out
00001	MCCPx/SCCPx Sync Out(1)
00000	MCCPx/SCCPx Timer Sync Out(1)

Note 1: CCP1 when connected to CCP1, CCP2 when connected to CCP2, etc.

REGISTER 17-11: SPIxURDTL: SPIx UNDERRUN DATA REGISTER LOW

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
URDATA<15:8>							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
URDATA<7:0>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15-0 **URDATA<15:0>:** SPIx Underrun Data bits

These bits are only used when URDTEN = 1. This register holds the data to transmit when a Transmit Underrun condition occurs.

When the MODE<32,16> or WLENGTH<4:0> bits select 16 to 9-bit data, the SPIx only uses URDATA<15:0>. When the MODE<32,16> or WLENGTH<4:0> bits select 8 to 2-bit data, the SPIx only uses URDATA<7:0>.

REGISTER 17-12: SPIxURDTH: SPIx UNDERRUN DATA REGISTER HIGH

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
URDATA<31:24>							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
URDATA<23:16>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15-0 **URDATA<31:16>:** SPIx Underrun Data bits

These bits are only used when URDTEN = 1. This register holds the data to transmit when a Transmit Underrun condition occurs.

When the MODE<32,16> or WLENGTH<4:0> bits select 32 to 25-bit data, the SPIx only uses URDATA<15:0>. When the MODE<32,16> or WLENGTH<4:0> bits select 24 to 17-bit data, the SPIx only uses URDATA<7:0>.

FIGURE 18-1: I2Cx BLOCK DIAGRAM

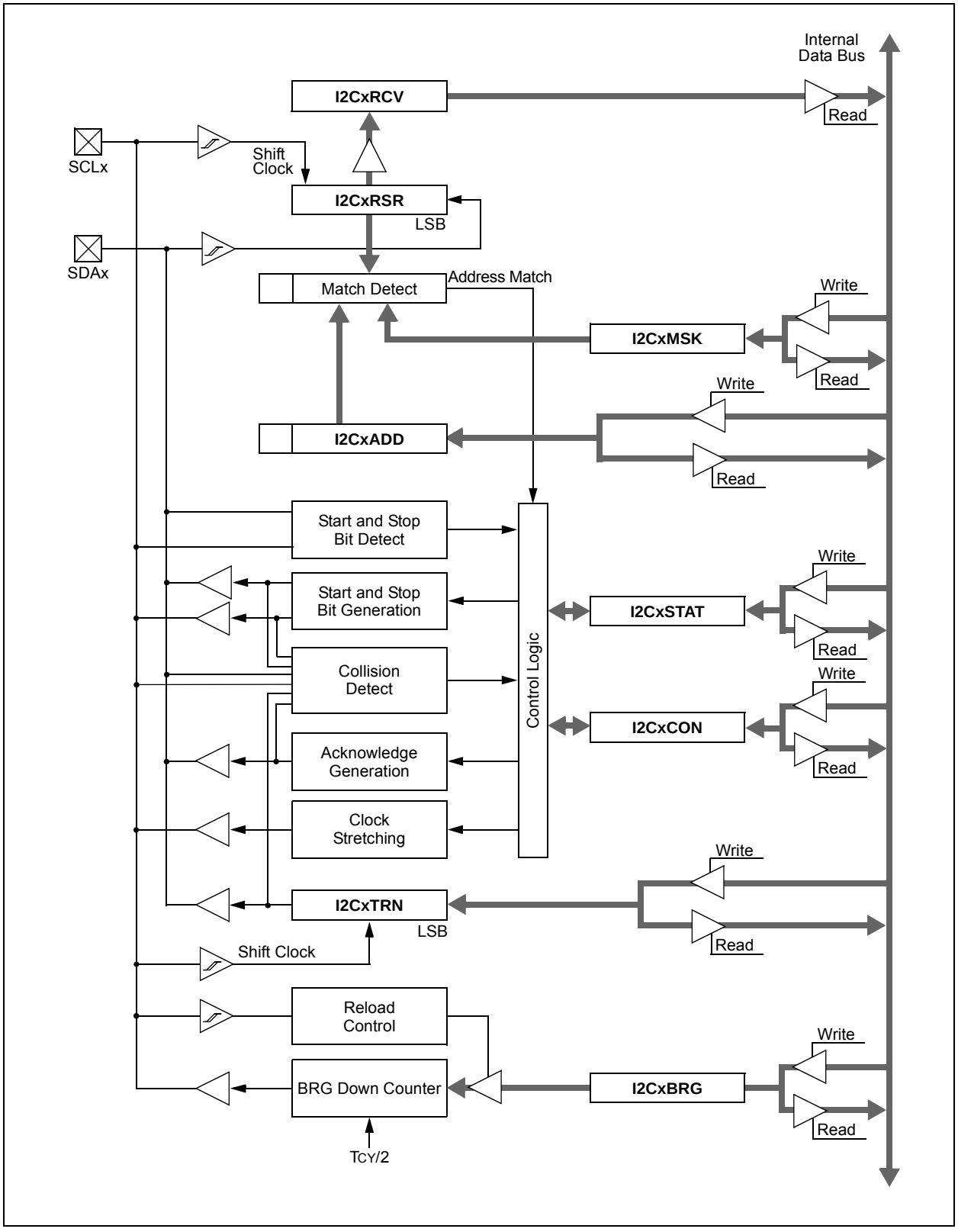
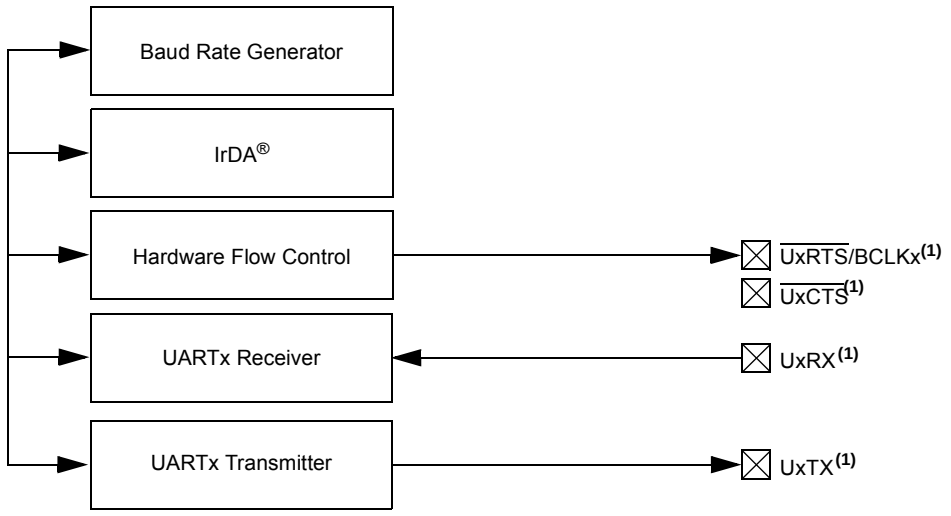


FIGURE 19-1: UARTx SIMPLIFIED BLOCK DIAGRAM



Note 1: The UART1, UART2, UART3 and UART4 inputs and outputs must all be assigned to available RPn/RPIn pins before use. See **Section 11.4 "Peripheral Pin Select (PPS)"** for more information.

REGISTER 20-16: U1IR: USB INTERRUPT STATUS REGISTER (DEVICE MODE ONLY)

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

R/K-0, HS	U-0	R/K-0, HS	R/K-0, HS	R/K-0, HS	R/K-0, HS	R/K-0, HS	R/K-0, HS
STALLIF	—	RESUMEIF	IDLEIF	TRNIF	SOFIF	UERRIF	URSTIF
bit 7							bit 0

Legend:	U = Unimplemented bit, read as '0'		
R = Readable bit	K = Write '1' to Clear bit	HS = Hardware Settable bit	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15-8 **Unimplemented:** Read as '0'

bit 7 **STALLIF:** STALL Handshake Interrupt bit
- 1 = A STALL handshake was sent by the peripheral during the handshake phase of the transaction in Device mode
- 0 = A STALL handshake has not been sent

bit 6 **Unimplemented:** Read as '0'

bit 5 **RESUMEIF:** Resume Interrupt bit
- 1 = A K-state is observed on the D+ or D- pin for 2.5 μs (differential '1' for low speed, differential '0' for full speed)
- 0 = No K-state is observed

bit 4 **IDLEIF:** Idle Detect Interrupt bit
- 1 = Idle condition is detected (constant Idle state of 3 ms or more)
- 0 = No Idle condition is detected

bit 3 **TRNIF:** Token Processing Complete Interrupt bit
- 1 = Processing of the current token is complete; read the U1STAT register for endpoint information
- 0 = Processing of the current token is not complete; clear the U1STAT register or load the next token from STAT (clearing this bit causes the STAT FIFO to advance)

bit 2 **SOFIF:** Start-of-Frame Token Interrupt bit
- 1 = A Start-of-Frame token is received by the peripheral or the Start-of-Frame threshold is reached by the host
- 0 = No Start-of-Frame token is received or threshold reached

bit 1 **UERRIF**: USB Error Condition Interrupt bit
- 1 = An unmasked error condition has occurred; only error states enabled in the U1EIE register can set this bit
- 0 = No unmasked error condition has occurred

bit 0 **URSTIF:** USB Reset Interrupt bit
- 1 = Valid USB Reset has occurred for at least 2.5 μs; Reset state must be cleared before this bit can be reasserted
- 0 = No USB Reset has occurred; individual bits can only be cleared by writing a '1' to the bit position as part of a word write operation on the entire register. Using Boolean instructions or bitwise operations to write to a single bit position will cause all set bits, at the moment of the write, to become cleared

Note: Individual bits can only be cleared by writing a '1' to the bit position as part of a word write operation on the entire register. Using Boolean instructions or bitwise operations to write to a single bit position will cause all set bits, at the moment of the write, to become cleared.

NOTES:

REGISTER 21-1: PMCON1: EPMP CONTROL REGISTER 1

R/W-0	U-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0
PMPEN	—	PSIDL	ADRMUX1	ADRMUX0	—	MODE1	MODE0
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0
CSF1	CSF0	ALP	ALMODE	—	BUSKEEP	IRQM1	IRQM0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15 **PMPEN:** Parallel Master Port Enable bit
1 = EPMP is enabled
0 = EPMP is disabled

bit 14 **Unimplemented:** Read as '0'

bit 13 **PSIDL:** Parallel Master Port Stop in Idle Mode bit
1 = Discontinues module operation when device enters Idle mode
0 = Continues module operation in Idle mode

bit 12-11 **ADRMUX<1:0>:** Address/Data Multiplexing Selection bits
11 = Lower address bits are multiplexed with data bits using 3 address phases
10 = Lower address bits are multiplexed with data bits using 2 address phases
01 = Lower address bits are multiplexed with data bits using 1 address phase
00 = Address and data appear on separate pins

bit 10 **Unimplemented:** Read as '0'

bit 9-8 **MODE<1:0>:** Parallel Port Mode Select bits
11 = Master mode
10 = Enhanced PSP; pins used are PMRD, PMWR, PMCS, PMD<7:0> and PMA<1:0>
01 = Buffered PSP; pins used are PMRD, PMWR, PMCS and PMD<7:0>
00 = Legacy Parallel Slave Port; pins used are PMRD, PMWR, PMCS and PMD<7:0>

bit 7-6 **CSF<1:0>:** Chip Select Function bits
11 = Reserved
10 = PMA15 is used for Chip Select 2, PMA14 is used for Chip Select 1
01 = PMA15 is used for Chip Select 2, PMCS1 is used for Chip Select 1
00 = PMCS2 is used for Chip Select 2, PMCS1 is used for Chip Select 1

bit 5 **ALP:** Address Latch Polarity bit
1 = Active-high (PMALL, PMALH and PMALU)
0 = Active-low ($\overline{\text{PMALL}}$, $\overline{\text{PMALH}}$ and $\overline{\text{PMALU}}$)

bit 4 **ALMODE:** Address Latch Strobe Mode bit
1 = Enables "smart" address strobes (each address phase is only present if the current access would cause a different address in the latch than the previous address)
0 = Disables "smart" address strobes

bit 3 **Unimplemented:** Read as '0'

bit 2 **BUSKEEP:** Bus Keeper bit
1 = Data bus keeps its last value when not actively being driven
0 = Data bus is in a high-impedance state when not actively being driven

bit 1-0 **IRQM<1:0>:** Interrupt Request Mode bits
11 = Interrupt is generated when Read Buffer 3 is read or Write Buffer 3 is written (Buffered PSP mode), or on a read or write operation when PMA<1:0> = 11 (Addressable PSP mode only)
10 = Reserved
01 = Interrupt is generated at the end of a read/write cycle
00 = No interrupt is generated

22.3.2 RTCVAL REGISTER MAPPINGS

REGISTER 22-4: RTCCON2H: RTCC CONTROL REGISTER 2 (HIGH)(1)

R/W-0	R/W-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
DIV<15:8>							
bit 15							bit 8

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
DIV<7:0>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15-0 **DIV<15:0>:** Clock Divide bits
Sets the period of the clock divider counter; value should cause a nominal 1/2 second underflow.

Note 1: A write to this register is only allowed when WRLOCK = 1.

REGISTER 22-6: RTCSTATL: RTCC STATUS REGISTER (LOW)

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	R/C-0	U-0	R/C-0	R-0	R-0	R-0
—	—	ALMEVT	—	TSAEVT(1)	SYNC	ALMSYNC	HALFSEC(2)
bit 7							bit 0

Legend:	C = Clearable bit		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15-6 **Unimplemented:** Read as '0'

bit 5 **ALMEVT:** Alarm Event bit
1 = An alarm event has occurred
0 = An alarm event has not occurred

bit 4 **Unimplemented:** Read as '0'

bit 3 **TSAEVT**: Timestamp A Event bit(1)
1 = A timestamp event has occurred
0 = A timestamp event has not occurred

bit 2 **SYNC:** Synchronization Status bit
1 = Time registers may change during software read
0 = Time registers may be read safely

bit 1 **ALMSYNC:** Alarm Synchronization Status bit
1 = Alarm registers (ALMTIME and ALMDATE) and Alarm Mask bits (AMASK<3:0>) should not be modified, and Alarm Control bits (ALRMEN, ALMRPT<7:0>) may change during software read
0 = Alarm registers and Alarm Control bits may be written/modified safely

bit 0 **HALFSEC:** Half Second Status bit(2)
1 = Second half period of a second
0 = First half period of a second

Note 1: User software may write a '1' to this location to initiate a Timestamp A event; timestamp capture is not valid until TSAEVT reads as '1'.

2: This bit is read-only; it is cleared to '0' on a write to the SECONE<3:0> bits.

REGISTER 22-17: TSADATEL: RTCC TIMESTAMP A DATE REGISTER (LOW)(1)

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	DAYTEN1	DAYTEN0	DAYONE3	DAYONE2	DAYONE1	DAYONE0
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
—	—	—	—	—	WDAY2	WDAY1	WDAY0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15-14 **Unimplemented:** Read as '0'

bit 13-12 **DAYTEN<1:0>:** Binary Coded Decimal Value of Days '10' Digit bits
Contains a value from 0 to 3.

bit 11-8 **DAYONE<3:0>:** Binary Coded Decimal Value of Days '1' Digit bits
Contains a value from 0 to 9.

bit 7-3 **Unimplemented:** Read as '0'

bit 2-0 **WDAY<2:0>:** Binary Coded Decimal Value of Weekdays '1' Digit bits
Contains a value from 0 to 6.

Note 1: If TSAEN = 0, bits<15:0> can be used for persistence storage throughout a non-Power-on Reset ($\overline{\text{MCLR}}$, WDT, etc.).

NOTES:

TABLE 33-25: A/D MODULE SPECIFICATIONS

AC CHARACTERISTICS			Standard Operating Conditions: 2.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial				
Param No.	**Symbol**	**Characteristic**	**Min.**	**Typ**	**Max.**	**Units**	**Conditions**
Device Supply							
AD01	AVDD	Module VDD Supply	Greater of: VDD – 0.3 or 2.2	—	Lesser of: VDD + 0.3 or 3.6	V	
AD02	AVSS	Module VSS Supply	VSS – 0.3	—	VSS + 0.3	V	
Reference Inputs							
AD05	VREFH	Reference Voltage High	AVSS + 1.7	—	AVDD	V	
AD06	VREFL	Reference Voltage Low	AVSS	—	AVDD – 1.7	V	
AD07	VREF	Absolute Reference Voltage	AVSS – 0.3	—	AVDD + 0.3	V	
Analog Inputs							
AD10	VINH-VINL	Full-Scale Input Span	VREFL	—	VREFH	V	**(Note 1)**
AD11	VIN	Absolute Input Voltage	AVSS – 0.3	—	AVDD + 0.3	V	
AD12	VINL	Absolute VINL Input Voltage	AVSS – 0.3	—	AVDD/3	V	
AD13		Leakage Current	—	±1.0	±610	nA	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3V, Source Impedance = 2.5 kΩ
AD17	RIN	Recommended Impedance of Analog Voltage Source	—	—	2.5K	Ω	10-bit
A/D Accuracy							
AD20B	Nr	Resolution	—	12	—	bits	
AD21B	INL	Integral Nonlinearity	—	±1	< ±2	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3V
AD22B	DNL	Differential Nonlinearity	—	—	< ±1	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3V
AD23B	GERR	Gain Error	—	±1	±4	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3V
AD24B	EOFF	Offset Error	—	±1	±2	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3V
AD25B		Monotonicity(1)	—	—	—	—	Guaranteed

Note 1: Measurements are taken with the external VREF+ and VREF- used as the A/D voltage reference.

FIGURE 33-7: INL vs. CODE (12-BIT MODE)

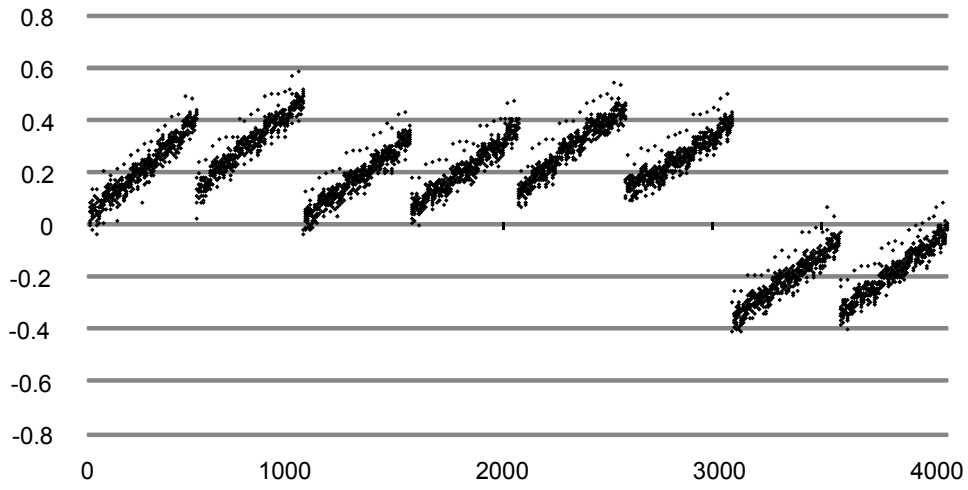
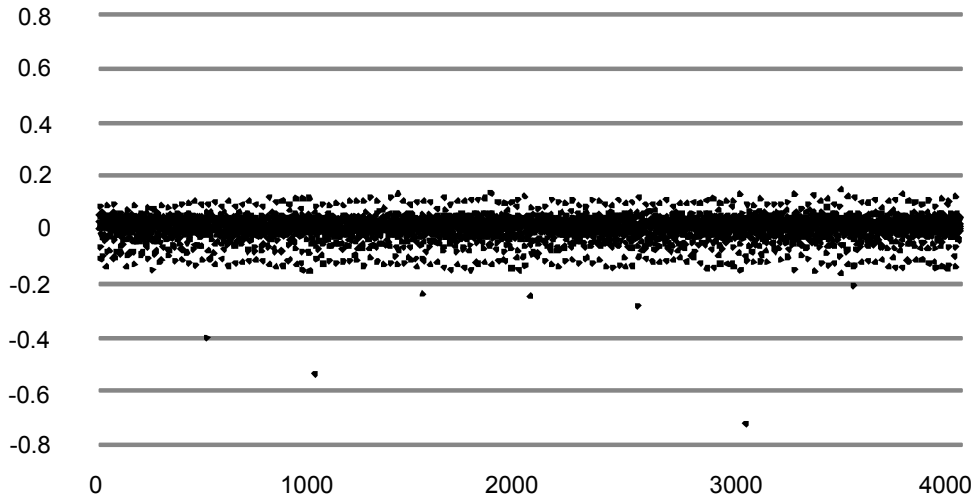


FIGURE 33-8: DNL vs. CODE (12-BIT MODE)(1)



Note 1: The following codes have marginal DNL and may result in a missing code: 1023, 2047, 3070 and 3071.