Absolute Maximum Ratings

V_{DD} to GND .....-0.3V to +3.97V
SCL, SDA, $\overline{RST}$.....-0.3V to +3.63V
All Other Pins to GND
except REG18 and REG285 .....-0.3V to (V_{DD} + 0.5V)*
Continuous Sink Current ..... 20mA per pin, 50mA total
Continuous Source Current..... 20mA per pin, 50mA total
Operating Temperature Range..... -40°C to +85°C
Storage Temperature Range..... -55°C to +125°C
Lead Temperature (soldering, 10s) .....+300°C
Soldering Temperature (reflow) .....+260°C

**Subject to not exceeding +3.97V.*

Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated in the operational sections of the specifications is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

Package Information

40TQFN

Package Code	T4055+2
Outline Number	21-0140
Land Pattern Number	90-0016

For the latest package outline information and land patterns (footprints), go to www.maximintegrated.com/packages. Note that a "+", "#", or "-" in the package code indicates RoHS status only. Package drawings may show a different suffix character, but the drawing pertains to the package regardless of RoHS status.

Recommended Operating Conditions

(T_A = -40°C to +85°C, unless otherwise noted.)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
V_{DD} Operating Voltage	V_{DD}	(Note 1)	2.97		3.63	V
Input Logic-High	V_{IH}		0.7 x V_{DD}		V_{DD} + 0.3	V
Input Logic-Low	V_{IL}		-0.3		0.3 x V_{DD}	V

DC Electrical Characteristics

(V_{DD} = 2.97V to 3.63V, T_A = -40°C to +85°C, unless otherwise noted. Typical values are at V_{DD} = 3.3V, T_A = +25°C.)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
Supply Current	I_{CPU}	CPU mode, all analog disabled (Notes 2, 3)		4.8		mA
	$I_{FASTCOMP}$			2		
	$I_{SAMPLEHOLDS}$	Both sample/hold		1.5		
	I_{ADC}			2.8		
	I_{DACS}	Per channel (Note 4)		0.6		
Brownout Voltage	V_{BO}	Monitors V_{DD} (Note 1)		2.7		V
Brownout Hysteresis	V_{BOH}	Monitors V_{DD} (Note 1)		0.07		V
1.8V Regulator Initial Voltage	V_{REG18}	(Note 1)	1.71	1.8	1.89	V
2.85V Regulator Initial Voltage	V_{REG285}	(Note 1)	2.8	2.85	2.9	V

DC Electrical Characteristics (continued)

(V_{DD} = 2.97V to 3.63V, T_A = -40°C to +85°C, unless otherwise noted. Typical values are at V_{DD} = 3.3V, T_A = +25°C.)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
Clock Frequencies	$f_{OSC-PERIPHERAL}$	T_A = +25°C (Note 5)		20		MHz
	$f_{MOSC-CORE}$	T_A = +25°C (Note 5)		10		
Clock Error	f_{ERR}	T_A = -40°C to +85°C			±8	%
External Clock Input	f_{XCLK}		20		133	MHz
Voltage Range: GP[15:0], SHEN, DACPW[7:0], REFINA, REFINB	V_{RANGE}	(Note 1)	-0.3		V_{DD} + 0.3	V
Output Logic-Low: SCL, SDA, MDIO, MDI, MCL, MCS, REFINA, REFINB, All GPIO Pins	V_{OL1}	I_{OL} = 4mA (Note 1)			0.4	V
Output Logic-High: SDA, MDIO, MDI, MCL, MCS, REFINA, REFINB, All GPIO Pins Not Open Drain	V_{OH1}	I_{OH} = -4mA (Note 1)	V_{DD} - 0.5			V
Pullup Current: MDIO, MDI, MCL, MCS, All GPIO Pins	I_{PU1}	V_{PIN} = 0V	26	55	78	µA
GPIO Drive Strength, Extra Strong Outputs: GP0, GP1, MCS, PW8, PW9	R_{HISt}			9	27.6	Ω
	R_{LOSt}			8	25.2	
GPIO Drive Strength, Strong Outputs: MDI, DACPW3, DACPW6	R_{HIA}			17	32.4	Ω
	R_{LOA}			12	26.4	
GPIO Drive Strength, Excluding Strong GPIO Outputs	R_{HIB}			27	57	Ω
	R_{LOB}			31	63	
DAC						
DAC Resolution	DAC_R		12			Bits
DAC Internal Reference Accuracy	DACREFACC	2.5V internal reference	-1.25		+1.25	%
DAC Internal Reference Power-Up Speed	t_{DACPUP}	99% settled		10		µs
Reference Input Full-Scale Range (REFINA, REFINB)	REFFS		1		2.5	V
DAC Operating Current	I_{DACS}	Per channel	See the *DC Electrical Characteristics*			
DAC Integral Nonlinearity	DACINL	12-bit at 2.5V reference		12		LSB

DC Electrical Characteristics (continued)

(V_{DD} = 2.97V to 3.63V, T_A = -40°C to +85°C, unless otherwise noted. Typical values are at V_{DD} = 3.3V, T_A = +25°C.)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
DAC Differential Nonlinearity	DACDNL	12-bit at 2.5V reference, guaranteed by design, not production tested			1	LSB
DAC Offset	$V_{OFFSET-DAC}$	At code “0”	0		18	mV
DAC Source Load Regulation	$I_{DAC-SOURCE}$	0 to full-scale output			8.6	mV/mA
DAC Sink Capability and Sink Load Regulation	$R_{DAC-SINK}$	0 to 0.5V output, limited by output buffer impedance		500		Ω
	$I_{DAC-SINK}$	0.5V to full-scale output			11.5	mV/mA
DAC Settling Time	t_{DAC}	Output load capacitance between 33pF and 270pF, from 10% to 90%		10		µs
FAST COMPARATOR/QUICK TRIPS						
Fast Comparator Resolution	FC_R		8			Bits
Fast Comparator Internal Reference Accuracy	FCREFACC		-1.25		+1.25	%
Fast Comparator Operating Current	$I_{FASTCOMP}$		See the *DC Electrical Characteristics*			
Fast Comparator Full Scale	$V_{FS-COMP}$		2.38	2.42	2.48	V
Fast Comparator Integral Nonlinearity	INL	Differential mode, 2.2nF capacitor at input, tested at worst-case positions			±2	LSB
Fast Comparator Differential Nonlinearity	DNL	Differential mode, 2.2nF capacitor at input, guaranteed by design			1	LSB
Fast Comparator Offset	$V_{OFFSET-COMP}$				2	LSB
Fast Comparator Input Resistance	$R_{IN-COMP}$	(Note 6)		15		MΩ
Fast Comparator Input Capacitance	$C_{IN-COMP}$			4		pF
Fast Comparator Sample Rate	f_{COMP}			625		ksps
ADC						
ADC Resolution	ADC_R	Default or slow ADC clock setting	13			Bits
ADC Internal Reference Accuracy	ADCREFACC		-0.85		+0.85	%
ADC Operating Current	I_{ADC}		See the *DC Electrical Characteristics*			
ADC Full-Scale 1	$V_{FS-ADC1}$			1.2		V
ADC Full-Scale 2	$V_{FS-ADC2}$			0.6		V
ADC Full-Scale 3	$V_{FS-ADC3}$			2.4		V
ADC Full-Scale 4	$V_{FS-ADC4}$			4.8		V
ADC Integral Nonlinearity	ADCINL	Computed using end points best fit: 13-bit, +25°C, V_{DD} = 3.3V, $V_{FS-ADC3}$		10		LSB

DC Electrical Characteristics (continued)

(V_{DD} = 2.97V to 3.63V, T_A = -40°C to +85°C, unless otherwise noted. Typical values are at V_{DD} = 3.3V, T_A = +25°C.)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
ADC Differential Nonlinearity	ADCDNL	V_{FS} = 1.2V		±0.5		LSB
ADC Sample-Sample Deviation		ADC full-scale set to $V_{FS-ADC3}$		5		LSB
ADC Offset	$V_{OFFSET-ADC}$	13-bit, V_{FS} = 1.2V	-8	+1	+8	LSB
GP[15:0] Input Resistance	R_{IN-ADC}			15		MΩ
ADC Sample Rate	f_{SAMPLE}	(Note 7)	8			ksps
ADC Temperature Conversion Time	t_{TEMP}			4.2		ms
Internal Temperature Measurement Error	$TINT_{ERR}$	(Note 8)		±2		°C
Remote Temperature Measurement Error (DS4830 Error Only)	$TREM_{ERR}$	(Note 8)		±2		°C
SAMPLE/HOLD						
Sample/Hold Input Range	V_{SHP}	ADC-SHN[1:0] = GND	0		1	V
Sample/Hold Capacitance	C_{SH}	ADC-SHP[1:0] to ADC-SHN[1:0]		5		pF
Sample Input Leakage	I_{SHLKG}	ADC-SHP[1:0] and ADC-SHN[1:0] connected to GND			1.2	µA
Sample Time	t_s	ADC-SHP[1:0] and ADC-SHN[1:0] connected to 50Ω voltage source	300			ns
Sample Conversion Complete	t_h	Time from valid sample to ADC data available			320	µs
Sample Offset	V_{SH-OFF}	Measured at 10mV	-10	-1.6	+7	mV
Sample Error	ERR_{SH}	$V_{ADC-SHP}$ to $V_{ADC-SHN}$ = 300mV, t_s = 300ns, driven with 50Ω voltage source	-4		+4	%
Sample Discharge Strength	R_{DIS}	ADC-SHP[1:0] or ADC-SHN[1:0] to GND		900		Ω
FLASH MEMORY						
Flash Erase Time	t_{ME}	Mass erase	22.9	24.14	25.35	ms
	t_{PE}	Page erase	22.9	24.14	25.35	
Flash Programming Time per Word	t_{PROG}	(Note 9)	69	74	79	µs
Flash Programming Temperature	T_{FLASH}		-40		+85	°C
Flash Endurance	n_{FLASH}	T_A = +50°C, guaranteed by design	20,000			Write Cycles
Data Retention	t_{RET}	T_A = +50°C, guaranteed by design	100			Years

AC Electrical Characteristics

(V_{DD} = 2.97V to 3.63V, T_A = -40°C to +85°C, unless otherwise noted.)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
I²C-COMPATIBLE INTERFACE (See Figure 1)						
SCL/MSCL Clock Frequency	f_{SCL}	Timeout not enabled			400	kHz
SCL Bootloader Clock Frequency	$f_{SCL:BOOT}$				100	kHz
Bus Free Time Between a STOP and START Condition	t_{BUF}		1.3			µs
Hold Time (Repeated) START Condition	$t_{HD:STA}$	(Note 10)	0.6			µs
Low Period of SCL/MSCL Clock	t_{LOW}		1.3			µs
High Period of SCL/MSCL Clock	t_{HIGH}		0.6			µs
Setup Time for a (Repeated) START Condition	$t_{SU:STA}$		0.6			µs
Data Hold Time (Note 11)	$t_{HD:DAT}$	Receive	0			ns
		Transmit	300			
Data Setup Time	$t_{SU:DAT}$	(Note 12)	100			ns
SCL/MSCL, SDA/MSDA Capacitive Loading	C_B	(Note 12)			400	pF
Rise Time of Both SDA and SCL Signals	t_R	(Note 12)	$20 + 0.1C_B$		300	ns
Fall Time of Both SDA and SCL Signals	t_F	(Note 12)	$20 + 0.1C_B$		300	ns
Setup Time for STOP Condition	$t_{SU:STO}$		0.6			µs
Spike Pulse Width That Can Be Suppressed by Input Filter	t_{SP}	(Note 13)		50		ns
SCL/MSCL and SDA/MSDA Input Capacitance	C_{BIN}			5		pF
SMBus Timeout	t_{SMBUS}			30		ms
3-WIRE DIGITAL INTERFACE (See Figure 2)						
MCL Clock Frequency	f_{SCLOUT}			1000		kHz
MCL Duty Cycle	t_{3WDC}			50		%
MDIO Setup Time	t_{DS}			100		ns
MDIO Hold Time	t_{DH}			100		ns
MCS Pulse-Width Low	t_{CSW}			500		ns
MCS Leading Time Before the First MCL Edge	t_L			500		ns
MCS Trailing Time After the Last MCL Edge	t_T			500		ns
MDIO, MCL Load	C_{B3W}	Total bus capacitance on one line		10		pF

AC Electrical Characteristics (continued)

(V_{DD} = 2.97V to 3.63V, T_A = -40°C to +85°C, unless otherwise noted.)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
SPI DIGITAL INTERFACE (See Figure 3 and Figure 4)						
SPI Master Operating Frequency	$1/t_{MSPICK}$				$f_{SYS}/2$	MHz
SPI Slave Operating Frequency	$1/t_{SSPICK}$				$f_{SYS}/4$	MHz
SPI I/O Rise/Fall Time	t_{SPI_RF}	C_L = 15pF, pullup = 560Ω			25	ns
MSPICK Output Pulse-Width High/Low	t_{MCH}, t_{MCL}		$t_{MSPICK}/2 - t_{SPI\_RF}$			ns
MSPIDO Output Hold After MSPICK Sample Edge	t_{MOH}		$t_{MSPICK}/2 - t_{SPI\_RF}$			ns
MSPIDO Output Valid to MSPICK Sample Edge (MSPIDO Setup)	t_{MOV}		$t_{MSPICK}/2 - t_{SPI\_RF}$			ns
MSPIDI Input Valid to MSPICK Sample Edge (MSPIDI Setup)	t_{MIS}		$2t_{SPI\_RF}$			ns
MSPIDI Input to MSPICK Sample Edge Rise/Fall Hold	t_{MIH}		0			ns
MSPICK Inactive to MSPIDO Inactive	t_{MLH}		$t_{MSPICK}/2 - t_{SPI\_RF}$			ns
SSPICK Input Pulse-Width High/Low	t_{SCH}, t_{SCL}		$t_{SSPICK}/2$			ns
$\overline{SSPICS}$ Active to First Shift Edge	t_{SSE}		t_{SPI_RF}			ns
SSPIDI Input to SSPICK Sample Edge Rise/Fall Setup	t_{SIS}		t_{SPI_RF}			ns
SSPIDI Input from SSPICK Sample Edge Transition Hold	t_{SIH}		t_{SPI_RF}			ns
SSPIDO Output Valid After SSPICK Shift Edge Transition	t_{SOV}				$2t_{SPI\_RF}$	ns
$\overline{SSPICS}$ Inactive	t_{SSH}		$t_{SSPICK} + t_{SPI\_RF}$			ns
SSPICK Inactive to $\overline{SSPICS}$ Rising	t_{SD}		t_{SPI_RF}			ns
SSPIDO Output Disabled After $\overline{SSPICS}$ Edge Rise	t_{SLH}				$2t_{SSPICK} + 2t_{SPI\_RF}$	ns

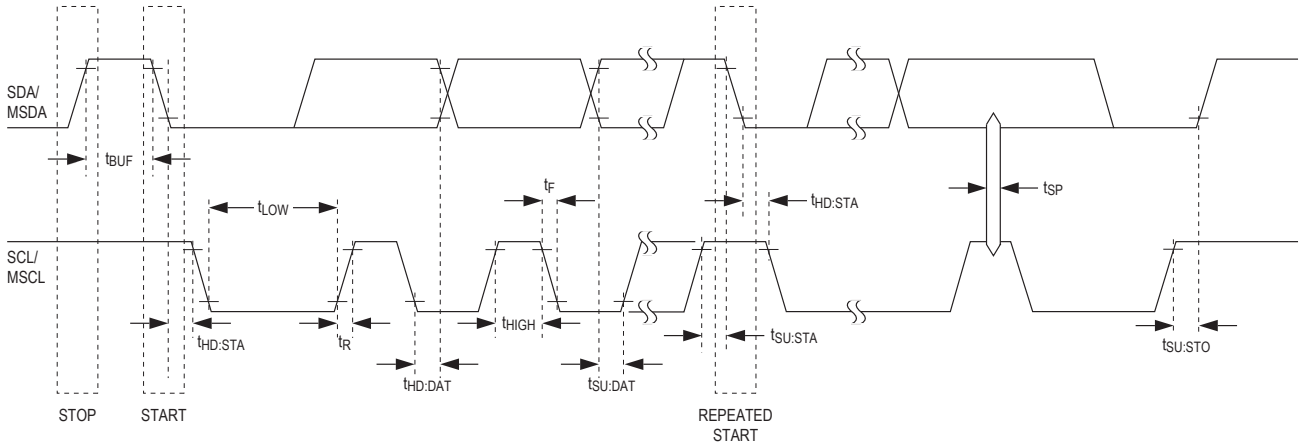
AC Electrical Characteristics (continued)

(V_{DD} = 2.97V to 3.63V, T_A = -40°C to +85°C, unless otherwise noted.)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
JTAG DIGITAL INTERFACE (See Figure 5)						
JTAG Logic Reference	V_{REF}			$V_{DD}/2$		V
TCK High Time	t_{TH}			0.5		µs
TCK Low Time	t_{TL}			0.5		µs
TCK Low to TDO Output	t_{TLQ}			0.125		µs
TMS, TDI Input Setup to TCK High	t_{DVTH}			0.25		µs
TMS, TDI Input Hold After TCK High	t_{THDX}			0.25		µs

Note 1: All voltages are referenced to GND. Currents entering the IC are specified as positive, and currents exiting the IC are specified as negative.
Note 2: Maximum current assuming 100% CPU duty cycle.
Note 3: This value does not include current in GPIO, SCL, SDA, MDIO, MDI, MCL, REFINA, and REFINB.
Note 4: Using internal reference.
Note 5: There is one internal oscillator. The oscillator (peripheral clock) goes through a 2:1 divider to create the core clock.
Note 6: Guaranteed by design.
Note 7: ADC conversions are delayed up to 1.6µs if the fast comparator is sampling the selected ADC channel. This can cause a slight decrease in the ADC sampling rate.
Note 8: Temperature readings average 64 times.
Note 9: Programming time does not include overhead associated with the utility ROM interface.
Note 10: f_{SCL} must meet the minimum clock low time plus the rise/fall times.
Note 11: This device internally provides a hold time of at least 75ns for the SDA signal (referred to the $V_{IH:MIN}$ of the SCL signal) to bridge the undefined region of the falling edge of SCL.
Note 12: C_B—Total capacitance of one bus line in pF.
Note 13: Filters on SDA and SCL suppress noise spikes at the input buffers and delay the sampling instant.

Timing Diagrams



NOTE: TIMING IS REFERENCED TO V_{ILMAX} AND V_{IHMIN}.

Figure 1. I²C Timing Diagram

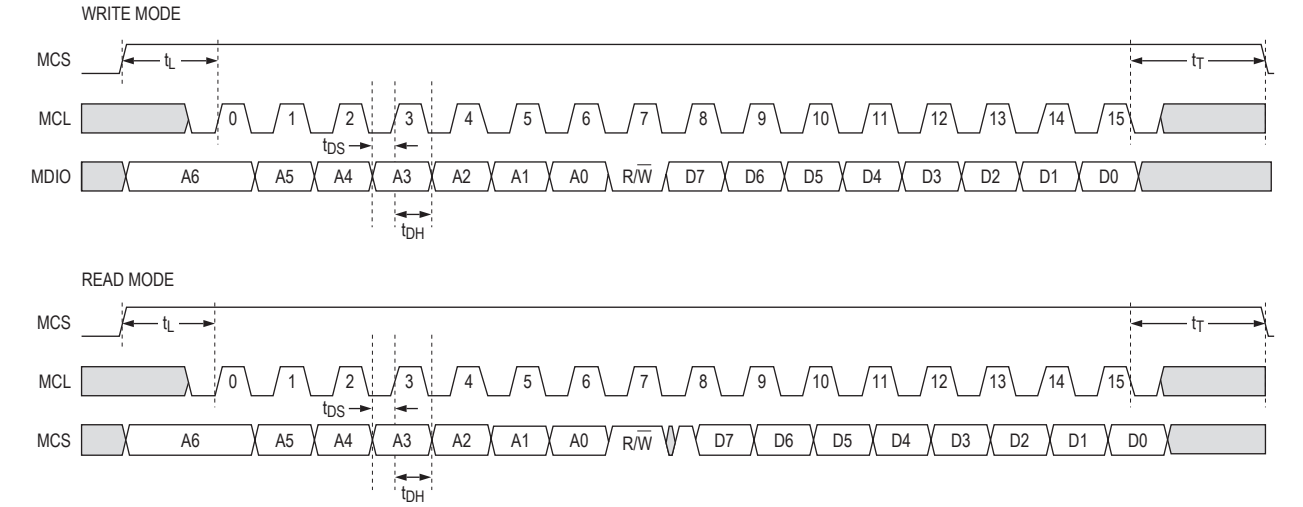


Figure 2. 3-Wire Timing Diagram

Timing Diagrams (continued)

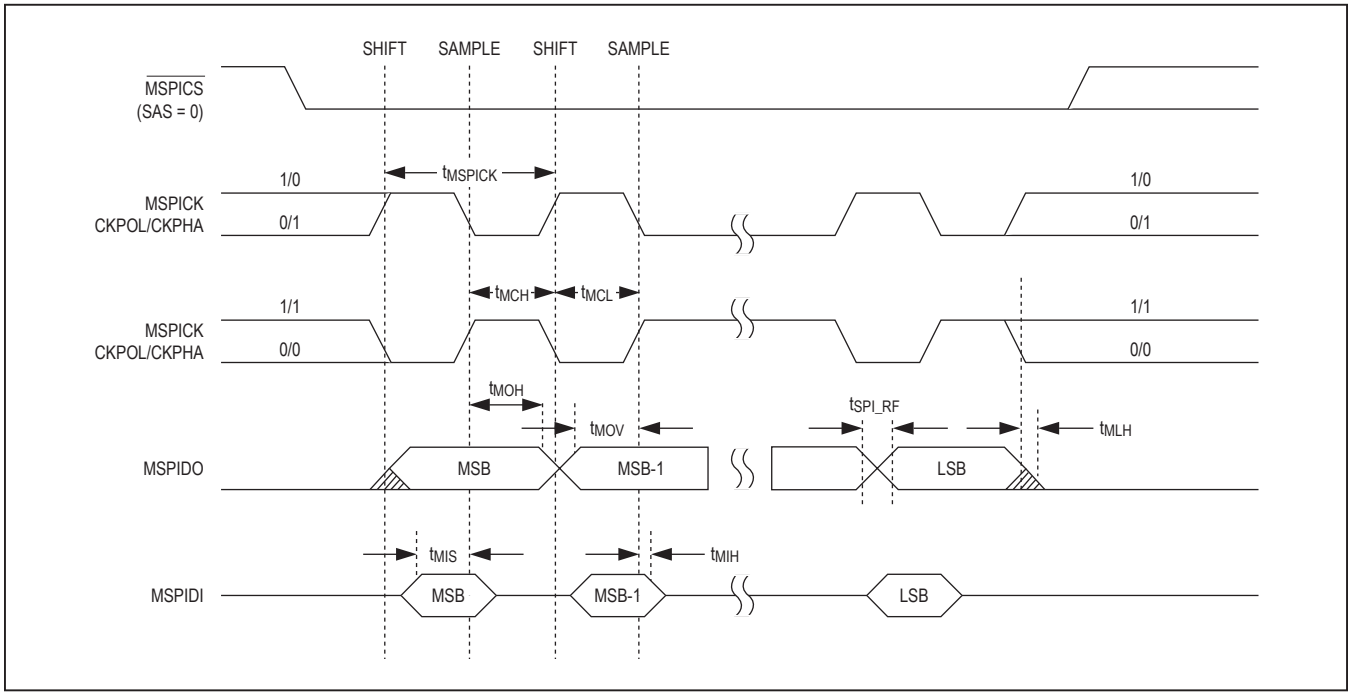


Figure 3. SPI Master Communications Timing Diagram

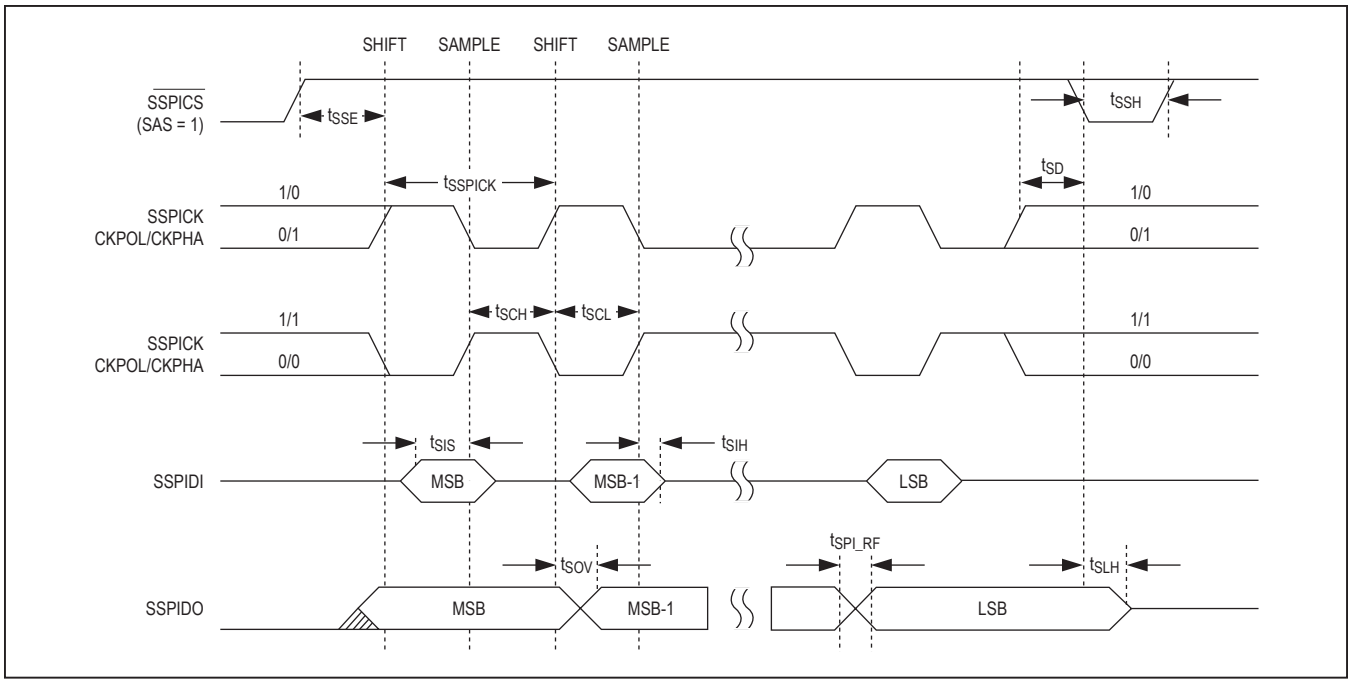


Figure 4. SPI Slave Communications Timing Diagram

Timing Diagrams (continued)

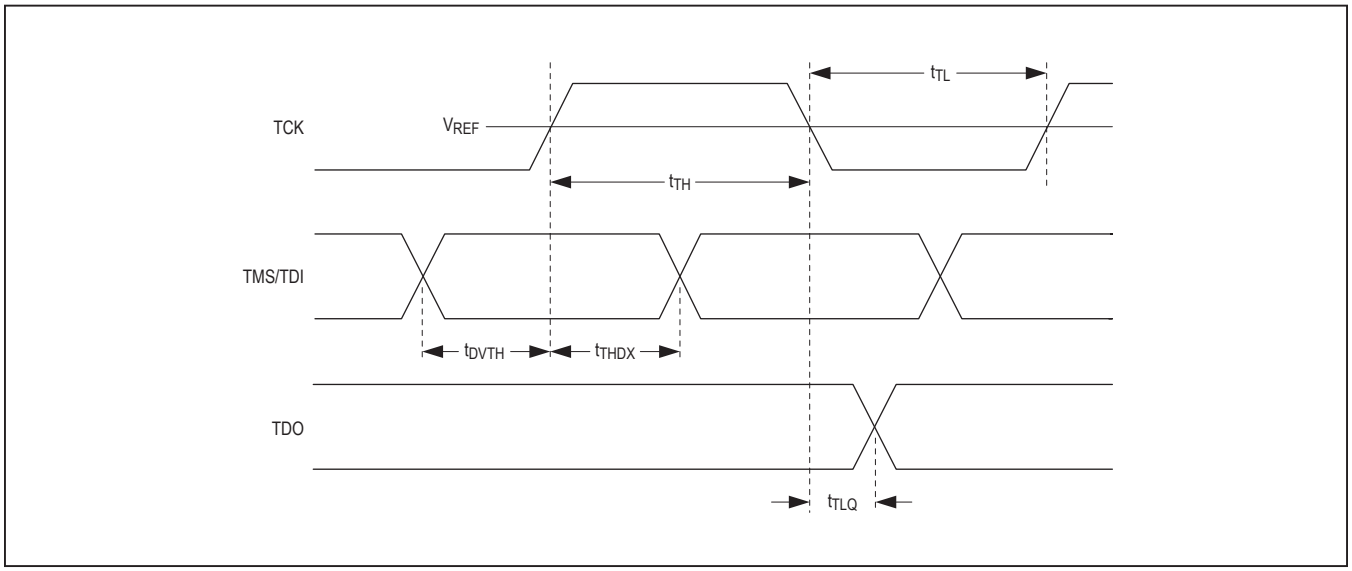
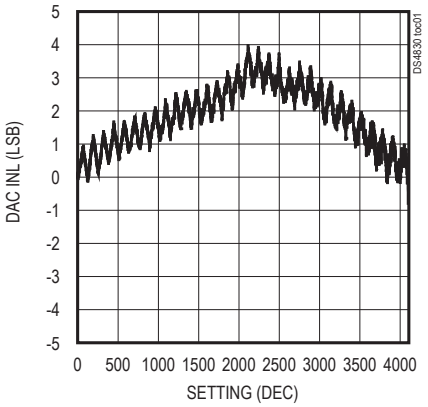


Figure 5. JTAG Timing Diagram

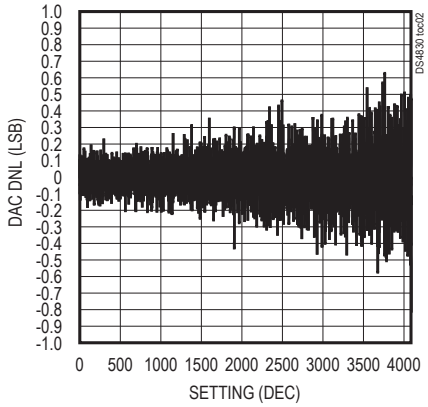
Typical Operating Characteristics

(T_A = +25°C, unless otherwise noted.)

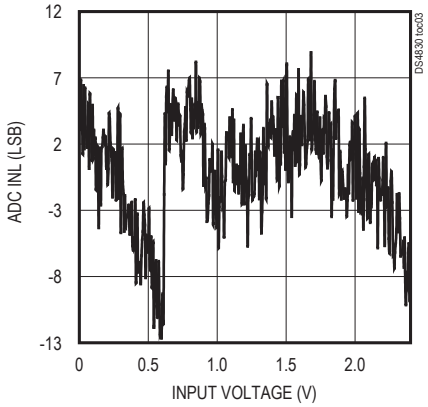
DAC INL (LSB)



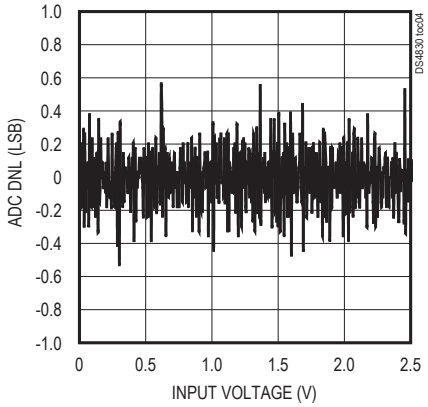
DAC DNL (LSB)



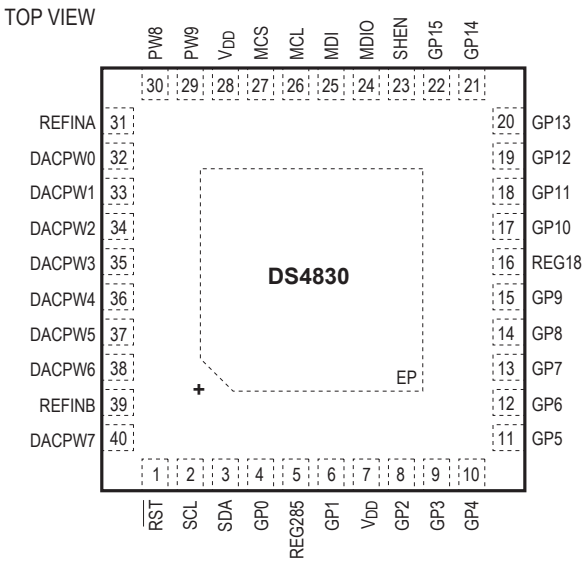
ADC INL (LSB)



ADC DNL (LSB)



Pin Configuration



TQFN
(5mm x 5mm)

Pin Description

PIN	NAME	INPUT STRUCTURE(S)	OUTPUT STRUCTURE	POWER-ON STATE	SELECTABLE FUNCTIONS (FIRST COLUMN IS DEFAULT FUNCTION)				PORT
1	$\overline{RST}$	Digital	Open Drain	High Impedance	$\overline{RST}$	—	—	—	—
2	SCL	Digital	Open Drain	High Impedance	I²C Slave Clock SCL	SPI SSPICK	—	—	—
3	SDA	Digital	Open Drain	High Impedance	I²C Slave Data SDA	SPI SSPIDI	—	—	—
4	GP0	ADC/Digital Input	Push-Pull, Extra Strong	55µA Pullup	ADC-S0	ADC-D0P	PW0	—	P2.0
5	REG285	V_{REG}	None	2.85V	Only function is for bypass capacitor for 2.85V internal regulator				—
6	GP1	ADC/Digital Input	Push-Pull, Extra Strong	55µA Pullup	ADC-S1	ADC-D0N	PW1	—	P2.1
7	V_{DD}	Voltage Supply, ADC Input	None	V_{DD}	ADC-VDD	—	—	—	—
8	GP2	SH Input, ADC Input	None	High Impedance	ADC-S2	ADC-SHP0	ADC-D1P	—	—

Pin Description (continued)

PIN	NAME	INPUT STRUCTURE(S)	OUTPUT STRUCTURE	POWER-ON STATE	SELECTABLE FUNCTIONS (FIRST COLUMN IS DEFAULT FUNCTION)				PORT
9	GP3	SH input, ADC Input	None	High Impedance	ADC-S3	ADC-SHN0	ADC-D1N	—	—
10	GP4	ADC/Digital Input	Push-Pull	55µA Pullup	JTAG TCK	ADC-S4	ADC-D2P	—	P6.0
11	GP5	ADC/Digital Input	Push-Pull	55µA Pullup	JTAG TDI	ADC-S5	ADC-D2N	—	P6.1
12	GP6	ADC/Digital Input	Push-Pull	55µA Pullup	ADC-S6	ADC-D3P	PW2	SPI SSPIDO	P2.2
13	GP7	ADC/Digital Input	Push-Pull	55µA Pullup	ADC-S7	ADC-D3N	PW3	SPI $\overline{\text{SSPICS}}$	P2.3
14	GP8	ADC/Digital I/P, External Temp A+ I/P (ADC-TEXT_A)	Push-Pull	55µA Pullup	ADC-S8	ADC-D4P	—	—	P2.4
15	GP9	ADC/Digital I/P, External Temp A- I/P (ADC-TEXT_A)	Push-Pull	55µA Pullup	ADC-S9	ADC-D4N	—	—	P2.5
16	REG18	V_{REG}	None	1.8V	Pin for 1.8V regulator bypass capacitor				—
17	GP10	ADC/Digital I/P, External Temp A+ I/P (ADC-TEXT_B)	Push-Pull	55µA Pullup	JTAG TMS	ADC-S10	ADC-D5P	—	P6.2
18	GP11	ADC/Digital I/P, External Temp A+ I/P (ADC-TEXT_B)	Push-Pull	55µA Pullup	JTAG TDO	ADC-S11	ADC-D5N	—	P6.3
19	GP12	SH Input, ADC/Digital Input	Push-Pull	55µA Pullup	ADC-S12	ADC-SHP1	ADC-D6P	—	P0.0
20	GP13	SH Input, ADC/Digital Input	Push-Pull	55µA Pullup	ADC-S13	ADC-SHN1	ADC-D6N	—	P0.1
21	GP14	ADC/Digital Input	Push-Pull	55µA Pullup	ADC-S14	ADC-D7P	SHEN1	—	P0.2
22	GP15	ADC/Digital Input	Push-Pull	55µA Pullup	ADC-S15	ADC-D7N	—	—	P0.3
23	SHEN	Digital	Push-Pull	55µA Pullup	SHEN0	—	—	—	P6.4
24	MDIO	Digital	Push-Pull	55µA Pullup	3-Wire Data MDIO	I^2C MSDA	SPI MSPIDO	PW4	P1.0
25	MDI	Digital	Push-Pull, Strong	55µA Pullup	—	—	SPI MSPIDI	PW5	P1.3
26	MCL	Digital	Push-Pull	55µA Pullup	3-Wire Clock MCL	I^2C MSCL	SPI MSPICK	PW6	P1.1

Pin Description (continued)

PIN	NAME	INPUT STRUCTURE(S)	OUTPUT STRUCTURE	POWER-ON STATE	SELECTABLE FUNCTIONS (FIRST COLUMN IS DEFAULT FUNCTION)				PORT
27	MCS	Digital	Push-Pull, Extra Strong	55µA Pullup	3-Wire Chip Select MCS	—	SPI $\overline{\text{MSPICS}}$	PW7	P1.2
28	V_{DD}	Voltage Supply	None	V_{DD}	ADC-VDD	—	—	—	—
29	PW9	Digital	Push-Pull, Extra Strong	55µA Pullup	PW9	—	—	—	P0.7
30	PW8	Digital	Push-Pull, Extra Strong	55µA Pullup	PW8	—	—	—	P0.6
31	REFINA	Reference, ADC/Digital Input (ADC_REFA)	Push-Pull	55µA Pullup	ADC-REFINA	—	—	—	P2.6
32	DACPW0	Digital	Push-Pull	55µA Pullup	DAC0, FS = REFINA or Internal Reference	PW0	—	—	P0.4
33	DACPW1	Digital	Push-Pull	55µA Pullup	DAC1, FS = REFINA or Internal Reference	PW1	—	—	P0.5
34	DACPW2	Digital	Push-Pull	55µA Pullup	DAC2, FS = REFINA or Internal Reference	PW2	CLKIN	—	P6.5
35	DACPW3	Digital	Push-Pull, Strong	55µA Pullup	DAC3, FS = REFINA or Internal Reference	PW3	—	—	P1.5
36	DACPW4	Digital	Push-Pull	55µA Pullup	DAC4, FS = REFINB or Internal Reference	PW4	—	—	P1.6
37	DACPW5	Digital	Push-Pull	55µA Pullup	DAC5, FS = REFINB or Internal Reference	PW5	—	—	P1.7
38	DACPW6	Digital	Push-Pull, Strong	55µA Pullup	DAC6, FS = REFINB or Internal Reference	PW6	—	—	P6.6
39	REFINB	Reference, ADC/ Digital Input	Push-Pull	55µA Pullup	ADC-REFINB	—	—	—	P1.4

Pin Description (continued)

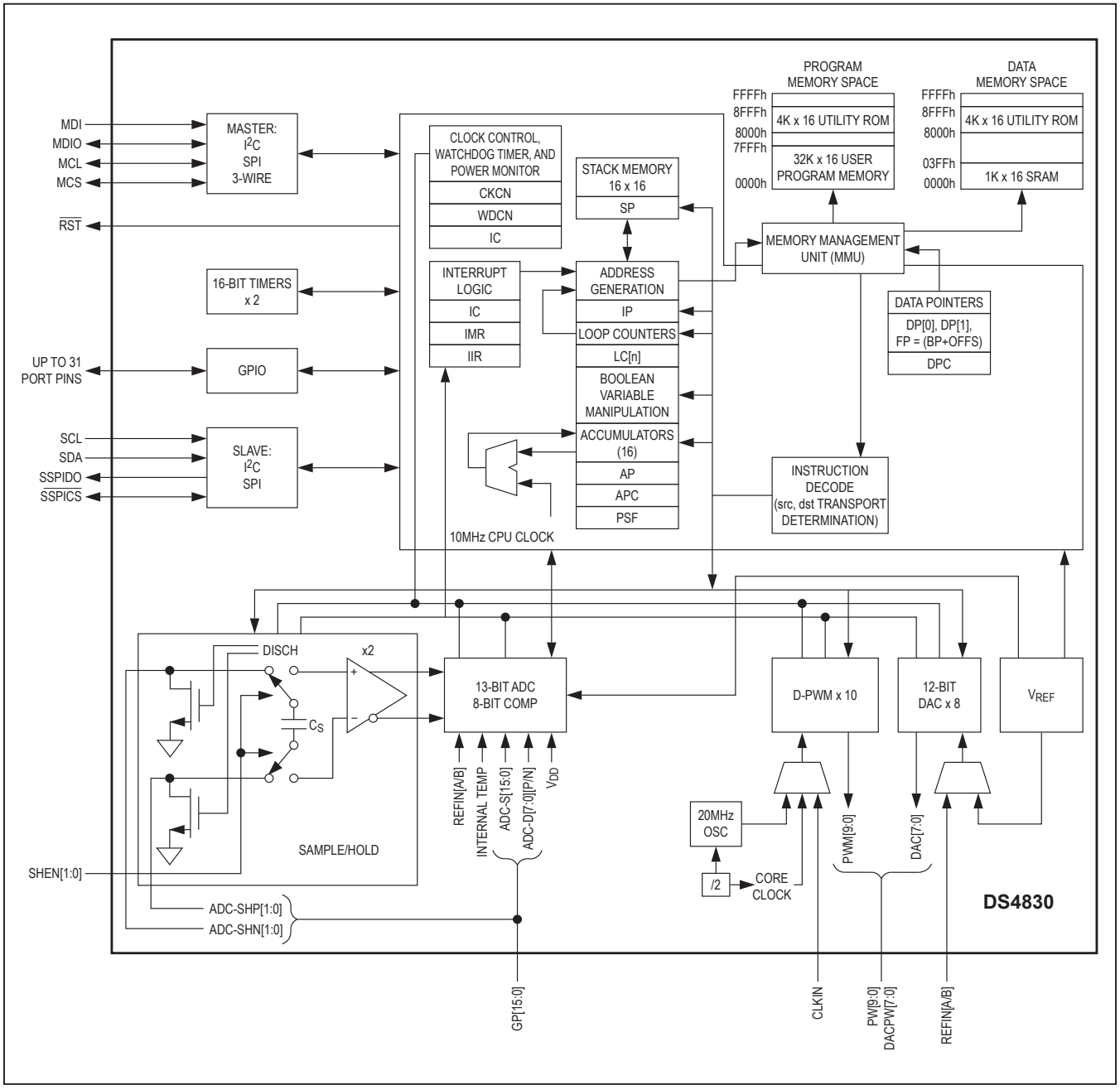
PIN	NAME	INPUT STRUCTURE(S)	OUTPUT STRUCTURE	POWER-ON STATE	SELECTABLE FUNCTIONS (FIRST COLUMN IS DEFAULT FUNCTION)				PORT
40	DACPW7	Digital	Push-Pull	55µA Pullup	DAC7, FS = REFINB or Internal Reference	PW7	—	—	P2.7
—	EP	Exposed Pad (Connect to GND)	—	GND	—	—	—	—	—

Note: *Bypass V_{DD}, REG285, and REG18 each with a 1µF X5R and 10nF capacitors to ground. All input-only pins and open-drain outputs are high impedance after V_{DD} exceeds V_{BO} and prior to code execution. Pins configured as GPIO have a weak internal pullup. See the Selectable Functions table for more information.*

Selectable Functions

FUNCTION NAME	DESCRIPTION
ADC-D[7:0][P/N]	Differential Inputs to ADC. Also used for external temperature sensors.
ADC-REFIN[A/B]	REFINA and REFINB Monitor Inputs to ADC
ADC-S[15:0]	Single-Ended Inputs to ADC
ADC-SH[P/N][1:0]	Sample/Hold Inputs 1 and 0
ADC-VDD	V_{DD} Monitor Input to ADC
DAC[7:0]	Voltage DAC Outputs
MCL, MCS, MDIO	Maxim Proprietary 3-Wire Interface, MCL (Clock), MCS (Chip Select), MDIO (Data). Used to control the MAX3798 family of high-speed laser drivers.
MSCL, MSDA	I^2C Master Interface: MSCL (I^2C Master Slave), MSDA (I^2C Master Data)
MSPICK, $\overline{\text{MSPICS}}$, MSPIDI, MSPIDO	SPI Master Interface: MSPICK (Clock), $\overline{\text{MSPICS}}$ (Active-Low Chip Select), MSPIDI (Data In), MSPIDO (Data Out)
P0.n, P1.n, P2.n, P6.n	General-Purpose Inputs/Outputs. Can also function as interrupts.
PW[9:0]	PWM Outputs
$\overline{\text{RST}}$	Used by JTAG and as Active-Low Reset for Device
SCL, SDA	I^2C Slave Interface: SCL (I^2C Slave Clock), SDA (I^2C Slave Data). These also function as a password-protected programming interface.
SHEN[1:0]	Sample/Hold Enable Inputs. Can also function as interrupts.
SSPICK, $\overline{\text{SSPICS}}$, SSPIDI, SSPIDO	SPI Slave Interface: SSPICK (Clock), $\overline{\text{SSPICS}}$ (Active-Low Chip Select), SSPIDI (Data In), SSPIDO (Data Out). In SPI slave mode, the I^2C slave interface is disabled.
TCK, TDI, TDO, TMS	JTAG Interface Pins. Also includes $\overline{\text{RST}}$.

Block Diagram



Detailed Description

The following is an introduction to the primary features of the DS4830 optical microcontroller. More detailed descriptions of the device features can be found in the DS4830 User's Guide.

Microcontroller Core Architecture

The device employs a low-power, low-cost, high-performance, 16-bit RISC microcontroller with on-chip flash memory. It is structured on an advanced, 16 accumulator- based, 16-bit RISC architecture. Fetch and execution operations are completed in one cycle without pipelining, since the instruction contains both the op code and data. The highly efficient core is supported by 16 accumulators and a 16-level hardware stack, enabling fast subroutine calling and task switching. Data can be quickly and efficiently manipulated with three internal data pointers. Multiple data pointers allow more than one function to access data memory without having to save and restore data pointers each time. The data pointers can automatically increment or decrement following an operation, eliminating the need for software intervention.

Module Information

Top-level instruction decoding is extremely simple and based on transfers to and from registers. The registers are organized into functional modules, which are in turn divided into the system register and peripheral register groups.

Peripherals and other features are accessed through peripheral registers. These registers reside in modules 0 to 5. The following provides information about the specific module in which each peripheral resides:

- Module 0: Timer 1, GPIO Ports 0, 1, and 2
- Module 1: I2C Master, GPIO Port 6, SPI Slave, SVM
- Module 2: I2C Slave, Analog-to-Digital Converter (ADC), Sample/Hold, Temperature, 3-Wire Master
- Module 3: Timer 2, MAC-Related Registers
- Module 4: Digital-to-Analog Converter (DAC)
- Module 5: Quick Trips, SPI Master, PWM

Instruction Set

The instruction set is composed of fixed-length, 16-bit instructions that operate on registers and memory locations. The instruction set is highly orthogonal, allowing arithmetic and logical operations to use any register along with the accumulator. Special-function registers control the peripherals and are subdivided into register modules.

Memory Organization

The device incorporates several memory areas:

- 32 kWords of flash memory for application program storage
- 1 kWords of SRAM for storage of temporary variables
- 4 kWords of utility ROM contain a debugger and program loader
- 16-level stack memory for storage of program return addresses and general-purpose use

The memory is implemented with separate address spaces for program memory, data memory, and register space. ROM, application code, and data memory can be placed into a single contiguous memory map. The device allows data memory to be mapped into program space, permitting code execution from data memory. In addition, program memory can be mapped into data space, permitting code constants to be accessed as data memory. Figure 6 shows the DS4830's memory map when executing from program memory space. Refer to the DS4830 User's Guide for memory map information when executing from data or ROM space.

The incorporation of flash memory allows field upgrade of the firmware. Flash memory can be password protected with a 16-word key, denying access to program memory by unauthorized individuals.

Utility ROM

The utility ROM is a 4 kWord block of internal ROM memory that defaults to a starting address of 8000h. The utility ROM consists of subroutines that can be called from application software, which includes the following:

- In-system programming (bootstrap loader) over JTAG or I2C-compatible interfaces
- In-circuit debug routines
- Internal self-test routines
- Callable routines for in-application flash programming

Following any reset, execution begins in the utility ROM. The ROM software determines whether the program execution should immediately jump to location 0000h, the start of application code, or to one of the special routines mentioned. Routines within the utility ROM are firmware-accessible and can be called as subroutines by the application software. More information on the utility ROM contents is contained in the DS4830 User's Guide.

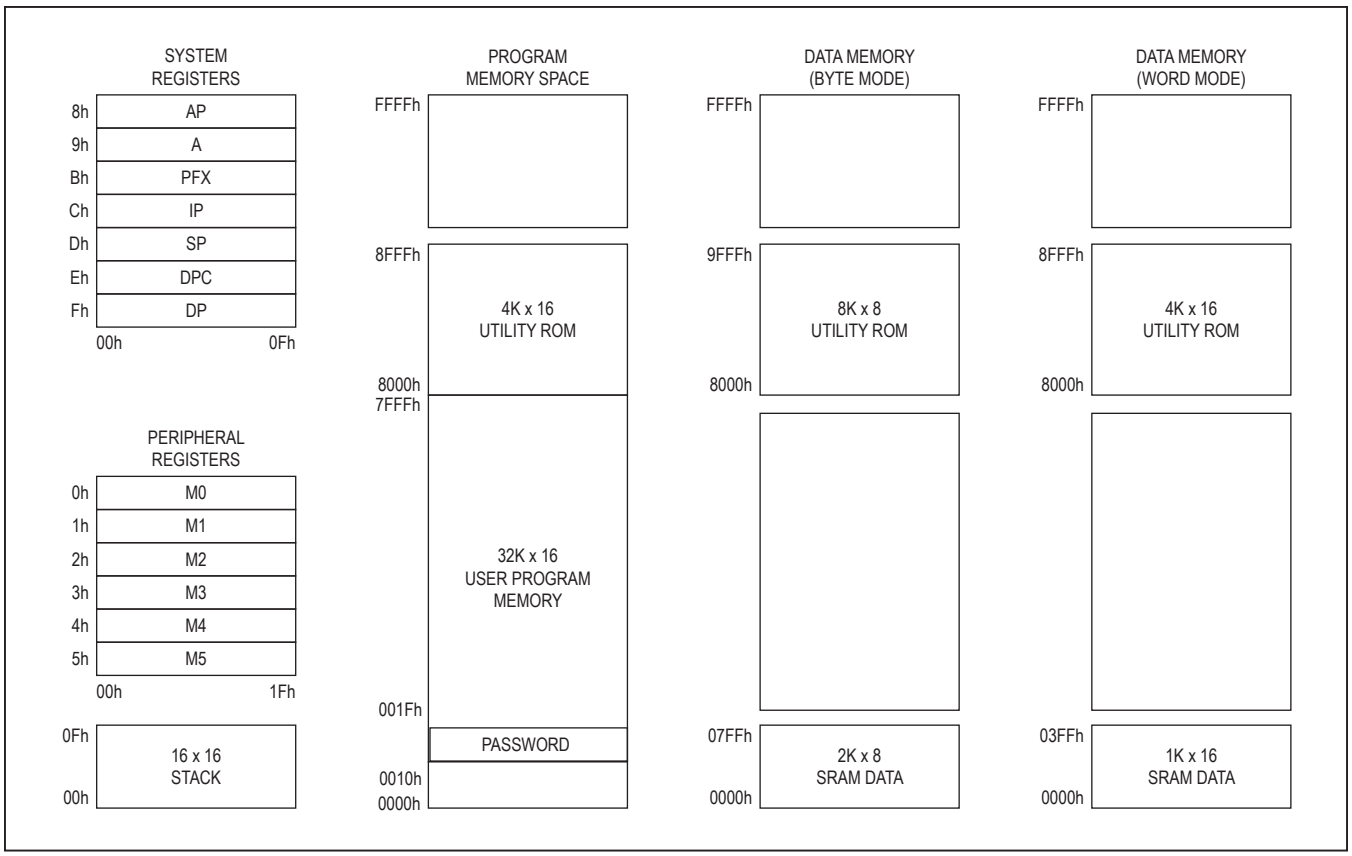


Figure 6. Memory Map When Program is Executing from Flash Memory

Password

Some applications require protection against unauthorized viewing of program code memory. For these applications, access to in-system programming, in-application programming, or in-circuit debugging functions is prohibited until a password has been supplied. The password is defined as the 16 words of physical program memory at addresses 0010h–001Fh.

A single password lock (PWL) bit is implemented in the device. When the PWL is set to 1 (power-on-reset default) and the contents of the memory at addresses 0010h–001Fh are any value other than all FFh or 00h, the password is required to access the utility ROM, including in-circuit debug and in-system programming routines that allow reading or writing of internal memory. When PWL is cleared to 0, these utilities are fully accessible without the password. The password is automatically set to all ones following a mass erase.

Detailed information regarding the password can be found in the DS4830 User's Guide.

Stack Memory

A 16-bit, 16-level internal stack provides storage for program return addresses. The stack is used automatically by the processor when the CALL, RET, and RETI instructions are executed and interrupts serviced. The stack can also be used explicitly to store and retrieve data by using the PUSH, POP, and POPI instructions.

On reset, the stack pointer, SP, initializes to the top of the stack (0Fh). The CALL, PUSH, and interrupt-vectoring operations increment SP, then store a value at the location pointed to by SP. The RET, RETI, POP, and POPI operations retrieve the value at SP and then decrement SP.

Programming

The microcontroller's flash memory can be programmed by one of two methods: in-system programming or in application programming. These provide great flexibility in system design as well as reduce the life-cycle cost of the embedded system. Programming can be password protected to prevent unauthorized access to code memory.

In-System Programming

An internal bootstrap loader allows the device to be programmed over the JTAG or I²C compatible interfaces. As a result, system software can be upgraded in-system, eliminating the need for a costly hardware retrofit when software updates are required.

The programming source select (PSS) bits in the ICDF register determine which interface is used for boot loading operation. The device supports JTAG and I²C as an interface corresponding to the 00 and 01 bits of PSS, respectively. See Figure 7.

In-Application Programming

The in-application programming feature allows the microcontroller to modify its own flash program memory. This allows on-the-fly software updates in mission-critical applications that cannot afford downtime. Alternatively, it allows the application to develop custom loader software that can operate under the control of the application software. The utility ROM contains firmware-accessible flash programming functions that erase and program flash memory. These functions are described in detail in the DS4830 User's Guide.

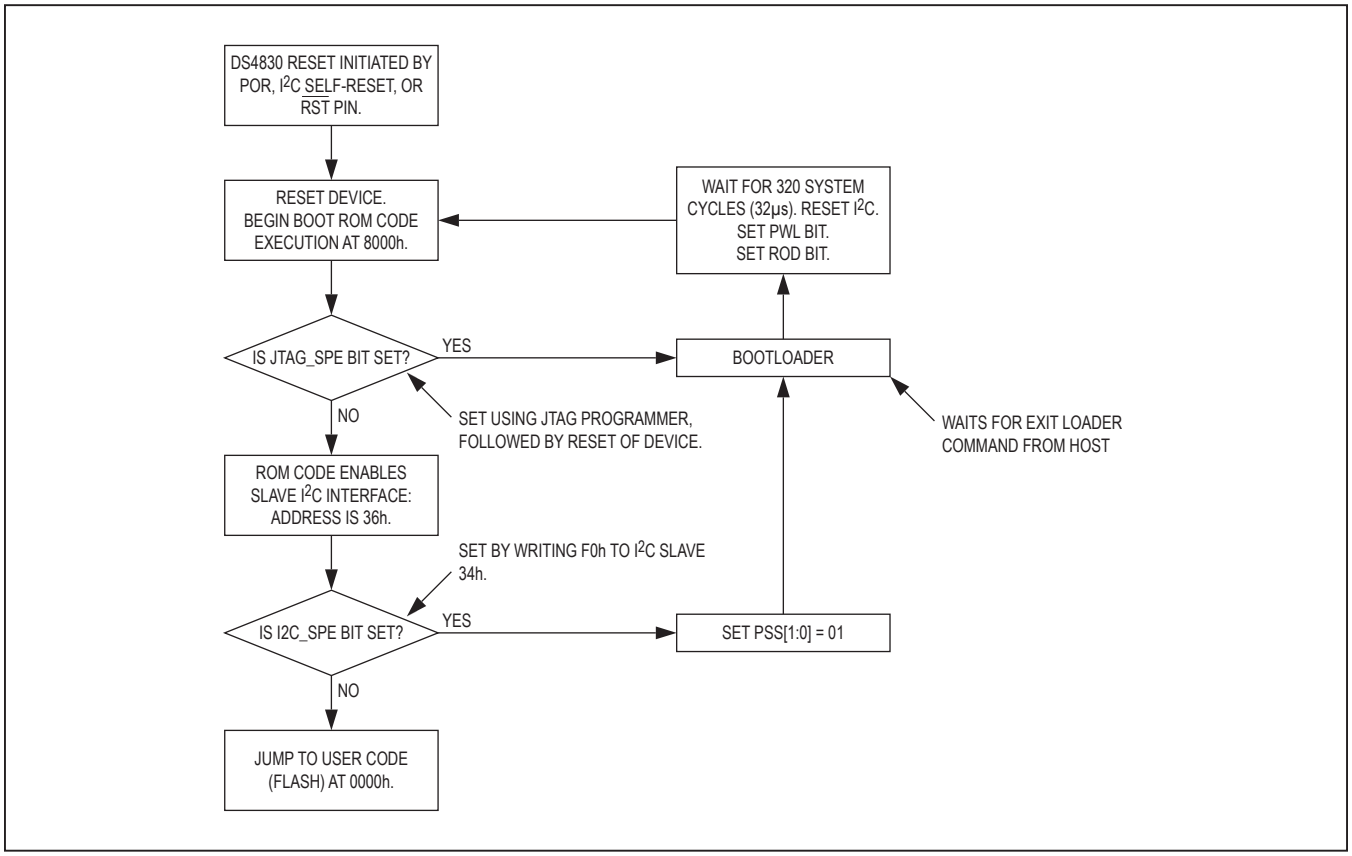


Figure 7. In-System Programming

Register Set

Sets of registers control most device functions. These registers provide a working space for memory operations as well as configuring and addressing peripheral registers on the device. Registers are divided into two major types: system registers (special-purpose registers, or SPRs) and peripheral registers (special-function registers, or SFRs). The common register set, also known as the system registers, includes the ALU, accumulator registers, data pointers, interrupt vectors and control, and stack pointer. The peripheral registers define additional functionality, and the functionality is broken up into discrete modules. Both the system registers and the peripheral registers are described in detail in the DS4830 User's Guide.

System Timing

The device generates its 10MHz instruction clock (MOSC) internally. On power-up, the oscillator's output (which cannot be accessed externally) is disabled until V_{DD} rises above V_{BO}. Once this threshold is reached, the output is enabled after approximately 1ms, clocking the device. See Figure 8.

System Reset

The device features several sources that can be used to reset the DS4830.

Power-On Reset

An internal power-on-reset (POR) circuit is used to enhance system reliability. This circuit forces the device to perform a POR whenever a rising voltage on V_{DD} climbs above V_{BO}. When this happens the following events occur:

- All registers and circuits enter their reset state.
- The POR flag (WDCN.7) is set to indicate the source of the reset.
- Code execution begins at location 8000h when the reset condition is released.

Brownout Detect/Reset

The device features a brownout detect/reset function. Whenever the power monitor detects a brownout condition (when $V_{DD} < V_{BO}$), it immediately issues a reset and stays in that state as long as V_{DD} remains below V_{BO}. Once V_{DD} voltage rises above V_{BO}, the device waits for $t_{SU:MOSC}$ before returning to normal operation, also referred to as CPU state. If a brownout occurs during $t_{SU:MOSC}$, the device again goes back to the brownout state. Otherwise, it enters into CPU state. In CPU state, the brownout detector is also enabled.

On power-up, the device always enters brownout state first and then follows the above sequence. The reset issued by brownout is the same as POR. Any action performed after POR also happens on brownout reset.

All the registers that are cleared on POR are also cleared on brownout reset.

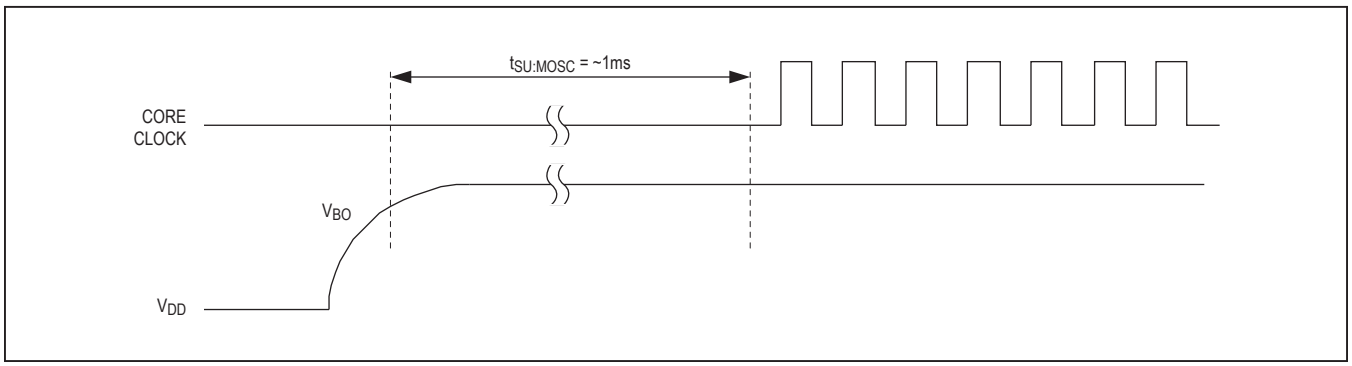


Figure 8. System Timing

Watchdog Timer Reset

The watchdog timer provides a mechanism to reset the processor in the case of undesirable code execution. The watchdog timer is a hardware timer designed to be periodically reset by the application software. If the software operates correctly, the timer is reset before it reaches its maximum count. However, if undesirable code execution prevents a reset of the watchdog timer, the timer reaches its maximum count and resets the processor.

The watchdog timer is controlled through two bits in the WDCN register (WDCN[5:4]: WD[1:0]). Its timeout period can be set to one of the four programmable intervals ranging from 2^{12} to 2^{21} system clock (MOSC) periods (0.409ms to 0.210s). The watchdog interrupt occurs at the end of this timeout period, which is 512 MOSC clock periods, or approximately 50µs, before the reset. The reset generated by the watchdog timer lasts for four system clock cycles, which is 0.4µs. Software can determine if a reset is caused by a watchdog timeout by checking the watchdog timer reset flag (WTRF) in the WDCN register. Execution resumes at location 8000h following a watchdog timer reset.

External Reset

Asserting $\overline{RST}$ low causes the device to enter the reset state. The external reset function is described in the DS4830 User's Guide. Execution resumes at location 8000h after $\overline{RST}$ is released. The DAC and PWM outputs are unchanged during execution of external reset.

Internal System Reset

The host can issue an I^2C command (BBh) to reset the communicating device. This reset has the same effect as the external reset as far as the reset values of all registers are concerned. Also, an internal system reset can occur when the in-system programming is done (ROD = 1). The DAC and PWM outputs are unchanged during execution of an internal reset.

Further details are available in the DS4830 User's Guide.

Programmable Timer

The device features two general-purpose programmable timers. Various timing loops can be implemented using the timers. Each general-purpose timer uses three SFRs. GTCN is the general control register, GTV is the timer value register, and GTC is the timer compare register.The timer can be used in two modes: free-running mode and compare mode with interrupts. Both are described in detail in the DS4830 User's Guide.

The functionality of the timers can be accessed through three SFRs for each of the general-purpose timers. The timer SFRs are accessed in module 0 and module 3. Detailed information regarding the timer block can be found in the DS4830 User's Guide.

Hardware Multiplier

The hardware multiplier (multiply-accumulate, or MAC module) is a very powerful tool, especially for applications that require heavy calculations. This multiplier can execute the multiply or multiply-negate, or multiply-accumulate or multiply-subtract operation for signed or unsigned operands. The MAC module uses eight SFRs, mapped as register 0h–05h and 08h–09h in module M3.

System Interrupts

Multiple interrupt sources are available to respond to internal and external events. The microcontroller architecture uses a single interrupt vector (IV) and single interrupt-service routine (ISR) design. For maximum flexibility, interrupts can be enabled globally, individually, or by module. When an interrupt condition occurs, its individual flag is set, even if the interrupt source is disabled at the local, module, or global level. Interrupt flags must be cleared within the firmware-interrupt routine to avoid repeated interrupts from the same source. Application software must ensure a delay between the write to the flag and the RETI instruction to allow time for the interrupt hardware to remove the internal interrupt condition. Asynchronous interrupt flags require a one-instruction delay and synchronous interrupt flags require a two-instruction delay.

When an enabled interrupt is detected, execution jumps to a user-programmable interrupt vector location. The IV register defaults to 0000h on reset or power-up, so if it is not changed to a different address, application firmware must determine whether a jump to 0000h came from a $\overline{RST}$ or interrupt source.

Once control has been transferred to the ISR, the interrupt identification register (IIR) can be used to determine if a system register or peripheral register was the source of the interrupt. In addition to IIR, MIIR registers are implemented to indicate which particular function under a peripheral module has caused the interrupt. The device contains six peripheral modules, M0 to M5. An MIIR register is implemented in modules M1 and M2. The MIIRs are 16-bit read-only registers and all of them default to all zeros on system reset. Once the module that causes the interrupt is singled out, it can then be interrogated for the specific interrupt source and software can take

appropriate action. Interrupts are evaluated by application code allowing the definition of a unique interrupt priority scheme for each application. Interrupt sources are available from the watchdog timer, the ADC (including sample/holds), fast comparators, the programmable timer, SVM, the I2C-compatible master and slave interface, 3-wire, master and slave SPI, and all GPIO pins.

I/O Port

The device allows for most inputs and outputs to function as general-purpose input and/or output pins. There are four ports: P0, P1, P2, and P6. Note that there is no port corresponding to P6.7. The 7th bit of port 6 is nonfunctional in all SFRs. Each pin is multiplexed with at least one special function, such as interrupts, I/O pins, or JTAG pins, etc.

The GPIO pins have Schmitt trigger receivers and full CMOS output drivers and can support alternate functions. The ports can be accessed through SFRs (PO[0,1,2,6], PI[0,1,2,6], PD[0,1,2,6], EIE[0,1,2,6], EIF[0,1,2,6], and EIES[0,1,2,6]) in modules 0 and 1, and each pin can be individually configured. The pin is either high impedance or a weak pullup when defined as an input, dependent on the state of the corresponding bit in the output register. In addition, each pin can function as external interrupt with individual enable, flag and active edge selection, when programmed as input.

The I/O port SFRs are accessed in module 0 and 1. Detailed information regarding the GPIO block can be found in the DS4830 User's Guide.

DAC Outputs

The device provides eight 12-bit DAC outputs with multiple reference options. An internal 2.5V reference is provided. There are also two selectable external references. REFINA can be selected as the full-scale reference for DAC0 to DAC3. REFINB can be selected as the full-scale reference for DAC4 to DAC7. The DAC outputs are voltage buffered. Each DAC can be individually disabled and put into a low-power power-down mode using DACCFG. An external reset does not affect the DAC outputs.

If a DAC output is used during the lifetime of the DS4830, the DAC must always be enabled to guarantee meeting the INL and offset specifications. If a pin is used for a DAC, it should be used only for the DAC function. The pin's function should not be switched between DAC and PWM or switched between DAC and GPIO.

The DAC SFRs are accessed in module 4. Detailed information regarding the DAC block can be found in the DS4830 User's Guide.

PWM Outputs

The device provides 10 independently configurable PWM outputs. The PWM outputs are configured using three SFRs: PWMCN, PWMDATA, and PWNSYNC. Using PWMCN and PWMDATA, individual PWM channels can be programmed for unique duty cycles (DCYCn), configurations (PWMCFGn), and delays (PWMDLYn), where n represents the PWM channel number.

The PWM clock can be obtained from the core clock, peripheral clock, or an external clock, depending on CLK_SEL bits programmed in individual PWMCFGn registers. The PWMCFGn register also enables/disables the corresponding PWM output and selects the PWM polarity. The user can set the duty cycle and the frequency of each PWM output individually by configuring the corresponding DCYCn register and the PWMCFGn register.

The device allows 4-slot or 32-slot pulse spreading options for each PWM channel. The PWM outputs can be configured to be output on an alternate location using the configuration register. PWMDLY is a 12-bit register used for providing starting delay on different PWM channels, and can be used to create multiphase PWM operation.

Different channels can be synchronized using the PWMSYNC register. Doing so effectively brings the channels in phase by restarting the channels that are to be synchronized. An external reset does not affect the PWM outputs.

The PWM SFRs are accessed in module 5. Detailed information regarding the PWM block can be found in the DS4830 User's Guide.

Analog-to-Digital Converter and Sample/Hold

The analog-to-digital converter (ADC) controller is the digital interface block between the CPU and the ADC. It provides all the necessary controls to the ADC and the CPU interface. The ADC uses a set of SFRs for configuring the ADC in desired mode of operation.

The device contains a 13-bit ADC with an input mux (Figure 9). The mux selects the ADC input from 16 single-ended or eight differential inputs. Additionally, the channels can be configured to convert internal and external temperature, V_{DD}, internal reference, or REFINA/B. Two channels can be programmed to be sample/hold inputs. The internal channel is used exclusively to measure the die temperature. The SFR registers control the ADC.

ADC

When used in voltage input mode, the voltage applied on the corresponding channel (differential or single-ended) is converted to a digital readout. The ADC can be set up to continuously poll selected input channels (continuous-sequence mode) or run a short burst of conversions and enter a shutdown mode to conserve power (single-sequence mode).

In voltage mode there are four full-scale values that can be programmed. These values can be trimmed by modifying the associated gain registers (ADCG1, ADCG2, ADCG3, ADCG4). By default these are set to 1.2V, 0.6V, 2.4V, and 4.8V full scale.

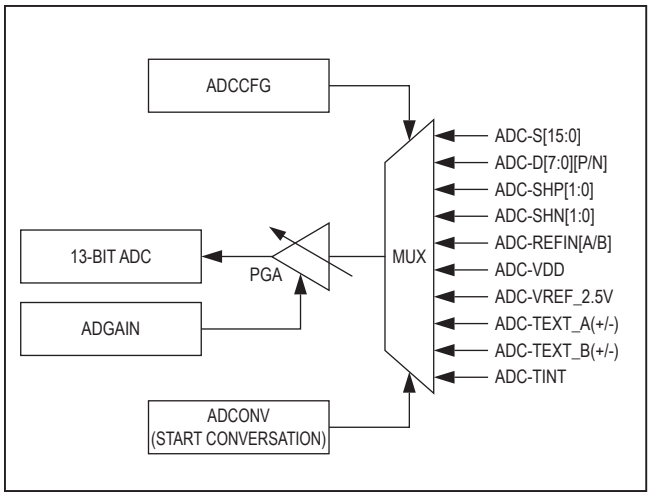


Figure 9. ADC Block Diagram

The ADCCLK is derived from the system clock with division ratio defined by the ADC control register. An A/D conversion takes 15 ADCCLK cycles to complete with additional four core clocks used for data processing. Internally every channel is converted twice and the average of two conversions is written to the data buffer. This gives each conversion result in (30 x ADC Clock Period + 800ns). ADC sampling rate is approximately 40ksps for the fastest ADC clock (core clock/8). In applications where extending the acquisition time is desired, the sample can be acquired over a prolonged period determined by the ADC control register.

Each ADC channel can have its own configuration, such as differential mode select, data alignment select, acquisition extension enable, and ADC gain select, etc. The ADC also has 24 (0 to 23) 16-bit data buffers for conversion result storage. The ADC data available interrupt flag (ADDAI) can be configured to trigger an interrupt following a predetermined number of samples. Once set, ADDAI can be cleared by software or at the start of a conversion process.

Sample/Hold

Pin combinations GP2-GP3 and GP12-GP13 can be used for sample/hold conversions if enabled in the SHCN register. These two can be independently enabled or disabled by writing a 1 or 0 to their corresponding bit locations in SHCN register. A data buffer location is reserved for each channel. When a particular channel is enabled, a sample of the input voltage is taken when a signal is issued on the SHEN pin, converted and stored in the corresponding data buffer.

The two sample/hold channels can sample simultaneously on the same SHEN signal or different SHEN signals depending on the SH_DUAL bit in the SHCN SFR.

The sample/hold data available interrupt flag (SHnDAI) can be configured to trigger an interrupt following sample completion. Once set, SHnDAI can be cleared by software.

Each sample/hold circuit consists of a sampling capacitor, charge injection nulling switches, and a buffer. Also included is a discharge circuit used to discharge parasitic capacitance on the input node and the sample capacitor before sampling begins. The negative input pins can be used to reduce ground offsets and noise.

Temperature Measurement

The device provides an internal temperature sensor for die temperature monitoring and two external remote temperature-sensing channels. In external temperature mode, current is forced into an external diode that is connected between user-specified channel pins (GP8-GP9 or GP10-GP11). The diode temperature is obtained by measuring the diode voltages at multiple bias currents.

These temperature channels can be enabled independently by setting the appropriate bit locations in the TEMPCN register. Whenever a temperature conversion is complete, the corresponding flag (INTDAI for internal conversion, EX0DAI and EX1DAI for external conversion) is set. These can be configured to cause an interrupt, and can be cleared by software. The temperature measurement resolution is 0.0625°C.

The device can use all the three modes explained above simultaneously by using a time-slicing mechanism performed by the internal controller. The ADC-related SFRs are accessed in module 1 and module 2. For details about this and the three blocks, refer to the ADC section of the DS4830 User's Guide.

Fast Comparator/Quick Trips

The device supports 8-bit quick-trip comparison functionality. The quick trips are required to continuously monitor user-defined channels in a round-robin sequence.

The quick- trip controller allows the user control of the list of channels to monitor. Each mode has a corresponding choice of list of channels for the round robin.

In any mode of quick-trip operation, the quick trip (analog) performs two comparisons on any selected channel.

1) Comparison with a high-threshold value.
2) Comparison with a low-threshold value.

Any comparison above the high-threshold value or below the low-threshold value causes a bit to set in the corresponding register. This bit can be used to trigger an interrupt. The threshold values are stored in 32 internal register (16 for low-threshold settings and 16 for high-threshold settings). The quick-trip controller provides the appropriate sequence of clock and threshold values for the quick trips. Because the quick trips and the ADC use the same input pins, the controller ensures that no collision takes place.

The quick-trip-related SFRs are accessed in module 5. Refer to the quick trip section of the DS4830 User's Guidefor more information.

I²C-Compatible Interface Modules

The device provides two independent I²C-compatible interfaces: one is a master and the other is a slave.

I²C-Compatible Master Interface

The device features an internal I²C-compatible master interface for communication with a wide variety of external I²C devices. The I²C-compatible master bus is a bidirectional bus using two bus lines: the serial-data line (MSDA) and the serial-clock line (MSCL). For the I²C-compatible master, the device has ownership of the I²C bus and drives the clock and generates the START and STOP signals. This allows the device to send data to a slave or receive data from a slave.

When the I²C-compatible master interface is disabled, MSDA and MSCL can be used as GPIO pins P1.0 and P1.1, respectively, and accessed through PO1/PI1/PD1.

I²C-Compatible Slave Interface

The device also features an internal I²C-compatible slave interface for communication with a host. Furthermore, the device can be in-system programmed (bootloaded) through the I²C-compatible slave interface. The two interface signals used by the I²C slave interface are SCL and SDA. For the I²C-compatible slave interface, the device relies on an externally generated clock to drive SCL and responds to data and commands only when requested by the I²C master device. The I²C-compatible slave interface is open drain and requires external pullup resistors.

SMBus Timeout

Both the I²C-compatible master and slave interfaces can work in SMBus-compatible mode for communication with other SMBus devices. To achieve this, a 30ms timer has been implemented on the I²C-compatible slave interface to make the interface SMBus compatible. The purpose of this timer is to issue a timeout interrupt and thus the firmware can reset the I²C-compatible slave interface when the SCL is held low for longer than 30ms. The timer only starts when **none** of the following conditions is true:

1) The I²C-compatible slave interface is in the idle state and there is no communications on the bus.
2) The I²C-compatible slave interface is not working in SMBus-compatible mode.
3) The SCL logic level is high.
4) The I²C-compatible slave interface is disabled.

When a timeout occurs, the timeout bit is set and an interrupt is generated, if enabled.

The I2C master-related SFRs are accessed in module 1. The I2C slave-related SFRs are accessed in module 2. Details can be found in the I2C section of the DS4830 User's Guide.

Serial Peripheral Interface Module

The device supports master and slave SPI interfaces. The SPI provides an independent serial communication channel to communicate synchronously with peripheral devices in a multiple master or multiple slave system. The interface allows access to a four-wire, full-duplex serial bus, and can be operated in either master mode or slave mode. Collision detection is provided when two or more masters attempt a data transfer at the same time. The maximum data rate of the SPI is 1/4 the system reference clock frequency for slave mode and 1/2 the system clock frequency for master mode.

The four interface signals used by the SPI are as follows:

- Master In-Slave Out. This signal is an output from a slave device, SSPIDO, and an input to the master device, MSPIDI. It is used to serially transfer data from the slave to the master. Data is transferred most significant bit (MSB) first. The slave device places this pin in an input state with a weak pullup when it is not selected.
- Master Out-Slave In. This signal is an output from a master device, MSPIDO, and an input to the slave devices, SSPIDI. It is used to serially transfer data from the master to the slave. Data is transferred MSB first.
- SPI Clock. This serial clock is an output from the master device, MSPICK, and an input to the slave devices, SSPICK. It is used to synchronize the transfer of data between the master and the slave on the data bus.
- Active-Low Slave Select. The slave-select signal is an input to enable the SPI module in slave mode, SSPICS, by a master device. The SPI module supports configuration of an active SSPICS state through the slave-active select. Normally, this signal has no function in master mode and its port pin can be used as a general-purpose I/O. However, the SSEL can optionally be used as a mode fault detection in master mode.

SPI Master Interface

The master mode is used when the device's SPI controls the data transmission rates and data format. The SPI is placed in master mode by setting the master mode bit (MSTM). Only an SPI master device can initiate a data transfer. Writing a data character to the SPI data buffer (SPIB), when in master mode, starts a data transfer. The SPI master immediately shifts out the data serially on MSPIDO, MSB first, while providing the serial clock on the MSPICK output. New data is simultaneously gated in on MSPIDI into the least significant bit (LSB) of the shift register. At the end of a transfer, the received data is loaded into the data buffer for reading, and the SPI transfer complete flag (SPIC) is set. If SPIC is set, an interrupt request is generated to the interrupt handler, if enabled.

SPI Slave Interface

Slave mode is used when the SPI is controlled by another peripheral device. The SPI is in slave mode when an internal bit (MSTM) is cleared to logic 0. In slave mode the SPI is dependent on the SSPICK sourced from the master to control the data transfer. The SSPICK input frequency should not be greater than the system clock frequency of the slave device divided by 4. The SPI master transfers data to a slave on SSPIDI, MSB first, and the selected slave device simultaneously transfers the contents of its shift register to the master on SSPIDO, also MSB first. Data received from the master replaces data in the slave's shift register at the completion of a transfer. Just like in the master mode, received data is loaded into the read buffer, and the SPI transfer complete flag is set at the end of the transfer. The setting of the transfer complete flag can cause an interrupt if enabled.

The SPI master-related SFRs are accessed in module 5. The SPI slave-related SFRs are accessed in module 1. Details can be found in the SPI section of the DS4830 User's Guide.

3-Wire Interface Module

The DS4830 controls devices like the MAX3798/MAX3799 over a proprietary 3-wire interface. The DS4830 acts as the 3-wire master, initiating communication with and generating the clock for the MAX3798/MAX3799. It is a 3-pin interface consisting of MDIO (a bidirectional data line), an MCL clock signal, and a MCS chip-select output (active high).

The 3-wire master-related SFRs are accessed in module 2. Detailed information regarding the 3-wire interface block can be found in the DS4830 User's Guide.

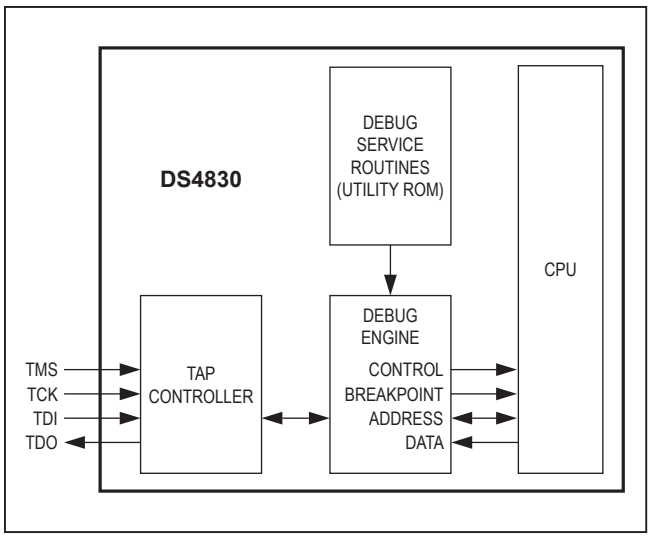


Figure 10. In-Circuit Debugger

In-Circuit Debug

Embedded debugging capability is available through the JTAG-compatible test access port (TAP). Embedded debug hardware and embedded ROM firmware provide in-circuit debugging capability to the user application, eliminating the need for an expensive in-circuit emulator. Figure 10 shows a block diagram of the in-circuit debugger. The in-circuit debug features include the following:

- A hardware debug engine
- A set of registers able to set breakpoints on register, code, or data accesses (ICDA, ICDB, ICDC, ICDD, ICDF, ICDT0, and ICDT1)
- A set of debug service routines stored in the utility ROM

The embedded hardware debug engine is an independent hardware block in the microcontroller. The debug engine can monitor internal activities and interact with selected internal registers while the CPU is executing user code. Collectively, the hardware and software features allow two basic modes of in-circuit debugging:

- Background mode allows the host to configure and set up the in-circuit debugger while the CPU continues to execute the application software at full speed. Debug mode can be invoked from background mode.
- Debug mode allows the debug engine to take control of the CPU, providing read/write access to internal registers and memory, and single-step trace operation.

Applications Information

Power-Supply Decoupling

To achieve the best results when using the DS4830, decouple the V_{DD} power supply with a 0.1µF capacitor. Use a high-quality, ceramic, surface-mount capacitor if possible. Surface-mount components minimize lead inductance, which improves performance, and ceramic capacitors tend to have adequate high-frequency response for decoupling applications. Decouple the REG285 and REG18 pins using 1µF X5R and 10nF capacitors (one each/per output). **Note: Do not use either of these pins for external circuitry.**

Additional Documentation

Designers must have three documents to fully use all the features of this device. This data sheet contains pin descriptions, feature overviews, and electrical specifications. Errata sheets contain deviations from published specifications. User guides offer detailed information about device features and operation. The following documents can be downloaded from **www.maximintegrated.com/DS4830**.

- This DS4830 data sheet, which contains electrical/timing specifications, package information, and pin descriptions.
- The DS4830 revision-specific errata sheet, if applicable.
- The DS4830 User's Guide, which contains detailed information and programming guidelines for core features and peripherals.

Development and Technical Support

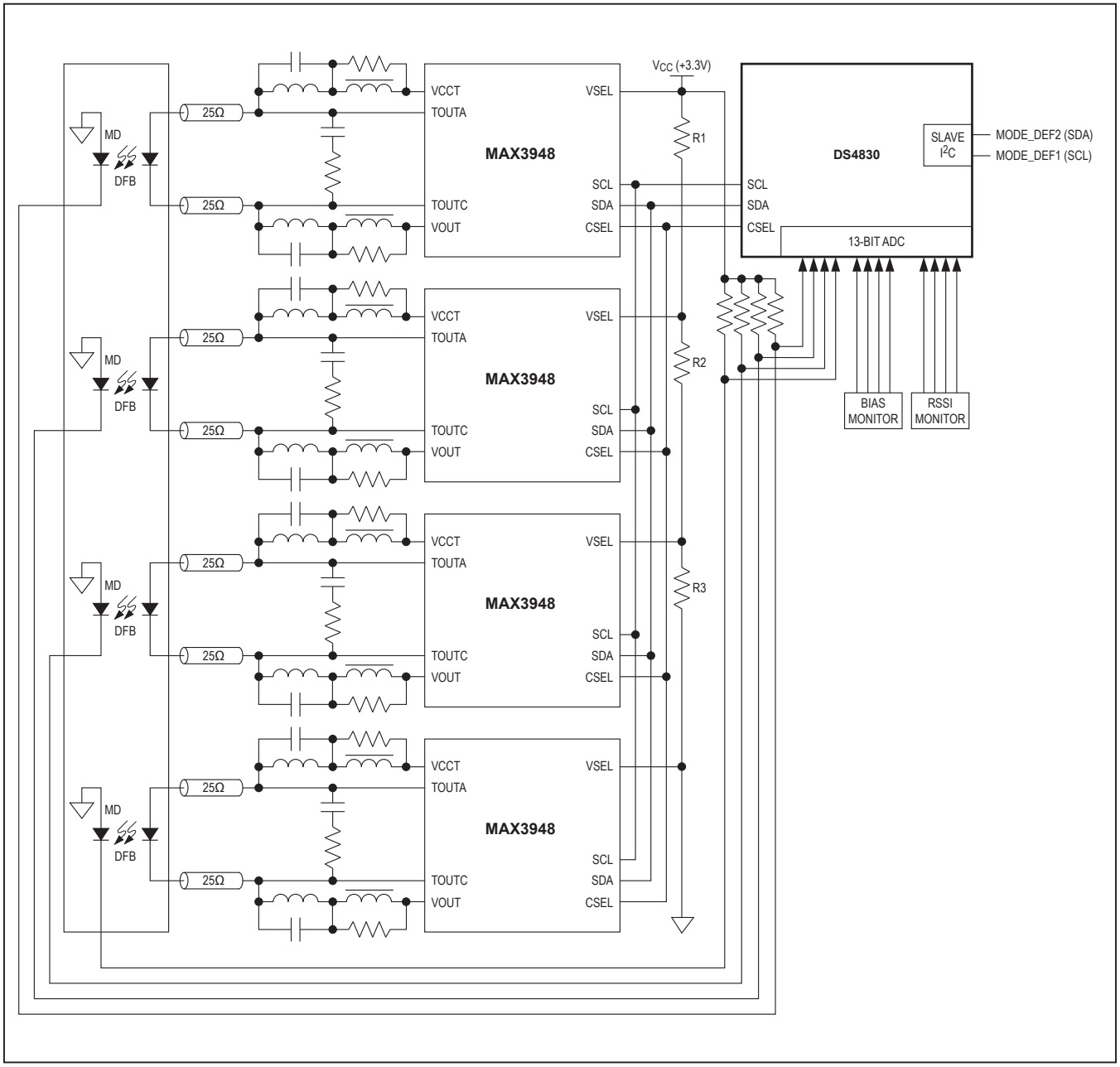
Maxim and third-party suppliers provide a variety of highly versatile, affordably priced development tools for this microcontroller, including the following:

- Compilers (C and assembly)
- In-circuit debugger
- Integrated Development Environments (IDEs)
- Serial-to-JTAG converters for programming and debugging
- USB-to-JTAG converters for programming and debugging

A partial list of development tool vendors can be found at **www.maximintegrated.com/MAXQ_tools**.

Email **mixedsignal.apps@maximintegrated.com** for technical support.

Typical Application Circuit



Ordering Information

PART	TEMP RANGE	PIN-PACKAGE
DS4830T+T	-40°C to +85°C	40 TQFN-EP*

+Denotes a lead(Pb)-free/RoHS-compliant package.
T = Tape and reel.
**EP = Exposed pad.*

Revision History

REVISION NUMBER	REVISION DATE	DESCRIPTION	PAGES CHANGED
0	6/11	Initial release	—
1	10/11	Corrected the lead temperature from +260°C to +300°C in the *Absolute Maximum Ratings* section; added explanation to the *DAC Outputs* section about the DAC operation to achieve desired INL levels	2, 22
2	6/13	Style edits, rephrasing, edits to all *Electrical Characteristics* tables, values updated, *Typical Operating Characteristics* added, Block Diagram updated, and updated Figures 6, 7, 8	1–8, 12, 13, 15–27
3	1/17	Updated *Package Information* table	2

