National Semiconductor

July 1995

LM12434/LM12{L}438 12-Bit + Sign Data Acquisition System with Serial I/O and Self-Calibration

General Description

The LM12434 and LM12{L}438 are highly integrated Data Acquisition Systems. Operating on 3V to 5V, they combine a fully-differential self-calibrating (correcting linearity and zero errors) 13-bit (12-bit + sign) analog-to-digital converter (ADC) and sample-and-hold (S/H) with extensive analog and digital functionality. Up to 32 consecutive conversions, using two's complement format, can be stored in an internal 32-word (16-bit wide) FIFO data buffer. An internal 8-word instruction RAM can store the conversion sequence for up to eight acquisitions through the LM12{L}438's eight-input multiplexer. The LM12434 has a four-channel multiplexer, a differential multiplexer output, and a differential S/H input. The LM12434 and LM12{L}438 can also operate with 8-bit + sign resolution and in a supervisory "watchdog" mode that compares an input signal against two programmable limits.

Acquisition times and conversion rates are programmable through the use of internal clock-driven timers. The differential reference voltage inputs can be externally driven for absolute or ratiometric operation.

All registers, RAM, and FIFO are directly accessible through the high speed and flexible serial I/O interface bus. The serial interface bus is user selectable to interface with the following protocols with zero glue logic: MICROWIRE/PLUS™, Motorola's SPI/QSPI, Hitachi's SCI, 8051 Family's Serial Port (Mode 0), I²C and the TMS320 Family's Serial Port.

An evaluation kit for demonstrating the LM12434 and LM12{L}438 is available.

Key Specifications

f_{CLK} = 8 MHz {L, f_{CLK} = 6 MHz}

- Resolution — 12-bit + sign or 8-bit + sign
- 13-bit conversion time — 5.5 μs {7.3 μs} (max)
- 9-bit conversion time — 2.6 μs {3.5 μs} (max)
- 13-bit Through-put rate — 140k samples/s {105k sample/s} (min)
- Comparison time ("watchdog" mode) — 1.4 μs {1.8 μs} (max)
- Serial Clock — 10 MHz {6 MHz} (max)
- Integral Linearity Error — ±1 LSB (max)
- V_{IN} range — GND to V_A^+
- Power dissipation — 45 mW {20 mW} (max)
- Stand-by mode power dissipation — 25 μW {16.5 μW} (typ)
- Supply voltage LM12L438 — 3.3V ±10%
  LM12434/8 — 5V ±10%

Features

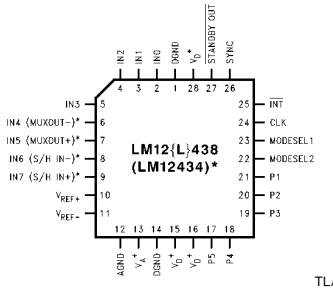
- Three operating modes: 12-bit + sign, 8-bit + sign, and "watchdog" comparison mode
- Single-ended or differential inputs
- Built-in Sample-and-Hold
- Instruction RAM and event sequencer
- 8-channel (LM12{L}438) or 4-channel (LM12434) multiplexer
- 32-word conversion FIFO
- Programmable acquisition times and conversion rates
- Self-calibration and diagnostic mode
- Power down output for system power management
- Read while convert capability for maximum through-put rate

Applications

- Data Logging
- Portable Instrumentation
- Process Control
- Energy Management
- Robotics

Connection Diagrams

28-Pin PLCC Package

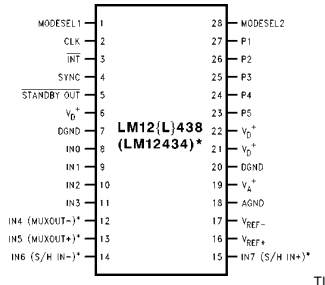


TL/H/11879–1

*Pin names in ( ) apply to the LM12434

Order Number LM12434CIV, LM12438CIV, or LM12L438CIV
See NS Package Number V28A

28-Pin Wide Body SO Package



TL/H/11879–2

Order Number LM12434CIWM, LM12438CIWM, or LM12L438CIWM
See NS Package Number M28B

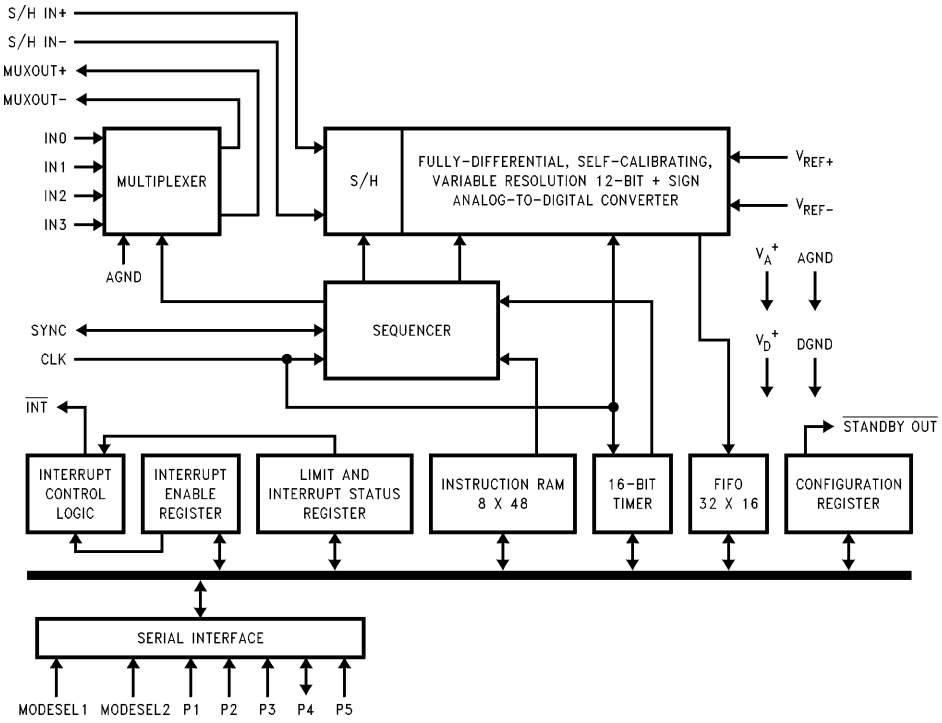
RRD-B30M85/Printed in U. S. A.

LM12434/LM12{L}438 12-Bit + Sign Data Acquisition System with Serial I/O and Self-Calibration

Table of Contents

1.0 Functional Diagrams

LM12434



TL/H/11879–3

INTERFACE	MODESEL1	MODESEL2	P1	P2	P3	P4	P5
Standard	0	1	$\overline{R}$/F	$\overline{CS}$	DI	DO	SCLK
8051	0	0	1*	1*	$\overline{CS}$	RXD	TXD
I²C	1	0	SAD0	SAD1	SAD2	SDA	SCL
TMS320	1	1	FSR	FSX	DX	DR	SCLK

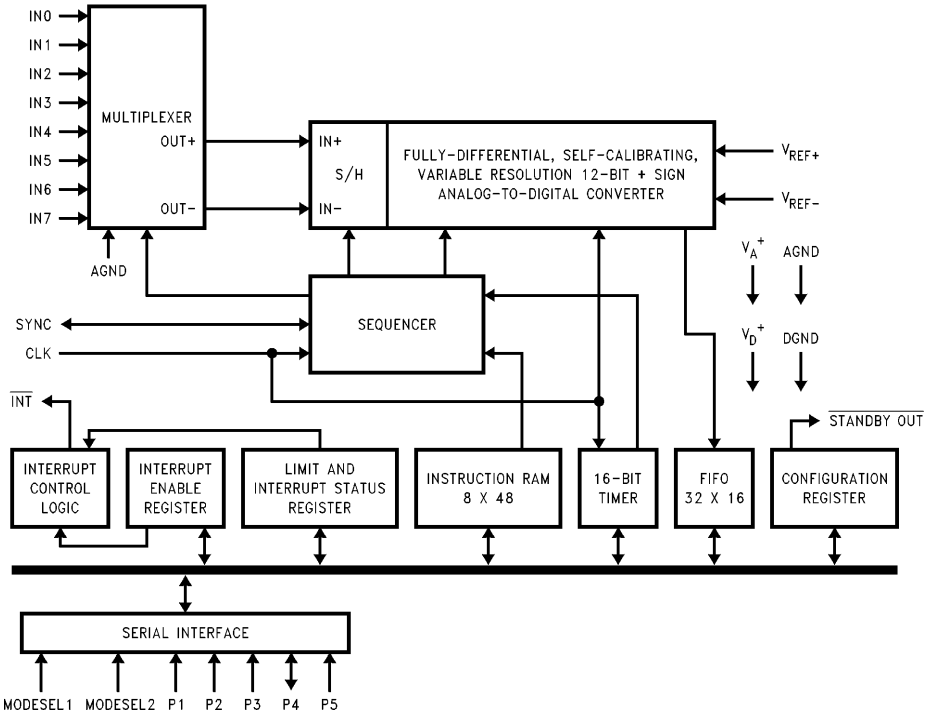
*Internal pull-up

Ordering Information (LM12434)

Part Number	Package Type	NSC Package Number	Temperature Range
LM12434CIV	28-Pin PLCC	V28A	−40°C to +85°C
LM12434CIWM	28-Pin Wide Body SO	M28B	−40°C to +85°C

1.0 Functional Diagrams (Continued)

LM12{L}438



TL/H/11879–4

INTERFACE	MODESEL1	MODESEL2	P1	P2	P3	P4	P5
Standard	0	1	$\overline{R}$/F	$\overline{CS}$	DI	DO	SCLK
8051	0	0	1*	1*	$\overline{CS}$	RXD	TXD
I²C	1	0	SAD0	SAD1	SAD2	SDA	SCL
TMS320	1	1	FSR	FSX	DX	DR	SCLK

*Internal pull-up

Ordering Information (LM12{L}438)

Part Number	Package Type	NSC Package Number	Temperature Range
LM12438CIV LM12L438CIV	28-Pin PLCC	V28A	−40°C to +85°C
LM12438CIWM LM12L438CIWM	28-Pin Wide Body SO	M28B	−40°C to +85°C
LM12438 Eval	Evaluation Board and Windows® based software		

2.0 Electrical Specifications

2.1 RATINGS

2.1.1 Absolute Maximum Ratings (Notes 1 & 2)

If Military/Aerospace specified devices are required, please contact the National Semiconductor Sales Office/Distributors for availability and specifications.

Supply Voltage (V_A^+ and V_D^+) 6.0V

Voltage at Input and Output Pins
except IN0−IN3 (LM12434) and IN0−IN7 (LM12{L}438) $-0.3V$ to $V^+ + 0.3V$

Voltage at Analog Inputs IN0−IN3 (LM12434) and IN0−IN7 (LM12{L}438) GND − 5V to $V^+ + 5V$

$|V_A^+ - V_D^+|$ 300 mV

$|AGND - DGND|$ 300 mV

Input Current at Any Pin (Note 3) ±5 mA

Package Input Current (Note 3) ±20 mA

Power Dissipation ($T_A = 25°C$) (Note 4)
V Package
WM Package

Storage Temperature −65°C to +150°C

Soldering Information, Lead Temperature (Note 19)
V Package, Vapor Phase (60 seconds)
Infrared (15 seconds)
WM Package, Vapor Phase (60 seconds)
Infrared (15 seconds)

ESD Susceptibility (Note 5) 1.5 kV

2.1.2 Operating Ratings (Notes 1 & 2)

Temperature Range ($T_{min} \le T_A \le T_{max}$)
LM12434CIV/LM12{L}438CIV $-40°C \le T_A \le 85°C$
LM12434CIWM, LM12{L}438CIWM $-40°C \le T_A \le 85°C$

Supply Voltage
V_A^+, V_D^+ 3.0V to 5.5V
$|V_A^+ - V_D^+|$ ≤100 mV
$|AGDND - DGND|$ ≤100 mV

Analog Inputs Range $GND \le V_{IN+} \le V_A^+$

V_{REF+} Input Voltage $1V \le V_{REF+} \le V_A^+$

V_{REF-} Input Voltage $0V \le V_{REF-} \le V_{REF+} - 1V$

$V_{REF+} - V_{REF-}$ $1V \le V_{REF} \le V_A^+$

V_{REF} Common Mode Range (Note 16) $0.1\ V_A^+ \le V_{REFCM} \le 0.6\ V_A^+$

2.2 PERFORMANCE CHARACTERISTICS All specifications apply to the LM12434, LM12438, and LM12L438 unless otherwise noted. Specifications in braces { } apply only to the LM12L438.

2.2.1 Converter Static Characteristics The following specifications apply to the LM12434 and LM12{L}438 for $V_A^+ = V_D^+ = 5V$ {3.3V}, AGND = DGND = 0V, $V_{REF+} = 4.096V$ {2.5V}, $V_{REF-} = 0V$, 12-bit + sign conversion mode, $f_{CLK} = 8.0$ MHz {6 MHz}, $R_S = 25\Omega$, source impedance for V_{REF+} and $V_{REF-} \le 25\Omega$, fully-differential input with fixed 2.048V {1.25V} common-mode voltage, and minimum acquisition time unless otherwise specified. **Boldface limits apply for $T_A = T_J = T_{MIN}$ to T_{MAX}**; all other limits $T_A = T_J = 25°C$. (Notes 6, 7, 8 and 9)

Symbol	Parameter	Conditions	Typical (Note 10)	Limits (Note 11)	Units (Limit)
ILE	Positive and Negative Integral Linearity Error	After Auto-Cal (Notes 12, 17)	±0.35	**±1**	LSB (max)
TUE	Total Unadjusted Error	After Auto-Cal (Note 12)	±1		LSB
	Resolution with No Missing Codes	After Auto-Cal (Note 12)		**13**	Bits
DNL	Differential Non-Linearity	After Auto-Cal	±0.2	**±1**	LSB (max)
	Zero Error	After Auto-Cal (Notes 13, 17)	±0.2	**±1**	LSB (max)
	Positive Full-Scale Error	After Auto-Cal (Notes 12, 17)	±0.2	**±2**	LSB (max)
	Negative Full-Scale Error	After Auto-Cal (Notes 12, 17)	±0.2	**±2**	LSB (max)
	DC Common Mode Error	(Note 14)	±2	**±3.5** {±4.0}	LSB (max)
ILE	8-Bit + Sign and "Watchdog" Mode Positive and Negative Integral Linearity Error	(Note 12)	±0.15	**±1/2**	LSB (max)
TUE	8-Bit + Sign and "Watchdog" Mode Total Unadjusted Error	After Auto-Zero	±1/2	**±1/2**	LSB (max)
	8-Bit + Sign and "Watchdog" Mode Resolution with No Missing Codes			**9**	Bits (max)

2.0 Electrical Specifications (Continued)

2.2.1 Converter Static Characteristics The following specifications apply to the LM12434 and LM12{L}438 for $V_A^+ = V_D^+ = 5V$ {3.3V}, AGND = DGND = 0V, $V_{REF+} = 4.096V$ {2.5V}, $V_{REF-} = 0V$, 12-bit + sign conversion mode, $f_{CLK} = 8.0$ MHz {6 MHz}, $R_S = 25\Omega$, source impedance for V_{REF+} and $V_{REF-} \leq 25\Omega$, fully-differential input with fixed 2.048V {1.25V} common-mode voltage, and minimum acquisition time unless otherwise specified. **Boldface limits apply for $T_A = T_J = T_{MIN}$ to T_{MAX}**; all other limits $T_A = T_J = 25°C$. (Notes 6, 7, 8 and 9) (Continued)

Symbol	Parameter		Conditions	Typical (Note 10)	Limits (Note 11)	Units (Limit)
DNL	8-Bit + Sign and "Watchdog" Mode Differential Non-Linearity			±0.15	**±1/2**	LSB (max)
	8-Bit + Sign and "Watchdog" Mode Zero Error		After Auto-Zero	±0.05	**±1/2**	LSB (max)
	8-Bit + Sign and "Watchdog" Positive and Negative Full-Scale Error			±0.1	**±1/2**	LSB (max)
	8-Bit + Sign and "Watchdog" Mode DC Common Mode Error			±1/8		LSB
	Multiplexer Channel-to-Channel Matching			±0.05		LSB
V_{IN+}	Non-Inverting Input Range				**GND** **V_A^+**	V (min) V (max)
V_{IN-}	Inverting Input Range				**GND** **V_A^+**	V (min) V (max)
$V_{IN+} - V_{IN-}$	Differential Input Voltage Range				**$-V_A^+$** **V_A^+**	V (min) V (max)
$\frac{V_{IN+} - V_{IN-}}{2}$	Common Mode Input Voltage Range				**GND** **V_A^+**	V (min) V (max)
PSS	Power Supply Sensitivity (Note 15)	Zero Error Full-Scale Error Linearity Error	$V_A^+ = V_D^+ = 5V$ ±10%, $V_{REF+} = 4.096V$, V_{REF-} = GND	±0.05 ±0.25 ±0.2	**±1.0** **±1.5**	LSB (max) LSB (max) LSB
C_{REF}	V_{REF+}/V_{REF-} Input Capacitance			85		pF
C_{IN}	Selected Multiplexer Channel Input Capacitance			75		pF

2.2.2 Converter Dynamic Characteristics The following specifications apply only to the LM12434 and LM12438 for $V_A^+ = V_D^+ = 5V$, AGND = DGND = 0V, $V_{REF+} = 4.096V$, $V_{REF-} = 0V$, 12-bit + sign conversion mode, $f_{CLK} = 8.0$ MHz, throughput rate = 133.3 kHz, $R_S = 25\Omega$, source impedance for V_{REF+} and $V_{REF-} \leq 25\Omega$, fully-differential input with fixed 2.048V {1.25V} common-mode voltage, and minimum acquisition time unless otherwise specified. **Boldface limits apply for $T_A = T_J = T_{MIN}$ to T_{MAX}**; all other limits $T_A = T_J = 25°C$. (Notes 6, 7, 8 and 9)

Symbol	Parameter	Conditions	Typical (Note 10)	Limits (Note 11)	Units (Limit)
	CLK Duty Cycle		50	 **40** **60**	% % (min) % (max)
t_C	Conversion Time	13-Bit Resolution, Sequencer State S5 *(Figure 10)*	44 (t_{CLK})	**44 (t_{CLK}) + 50 ns**	(max)
		9-Bit Resolution, Sequencer State S5 *(Figure 10)*	21 (t_{CLK})	**21 (t_{CLK}) + 50 ns**	(max)
t_A	Acquisition Time (Programmable)	Sequencer State S7 *(Figure 10)* Minimum for 13-Bits Maximum for 13-Bits (D = 15)	 9 (t_{CLK}) 39 (t_{CLK})	 **9 (t_{CLK}) + 50 ns** **39 (t_{CLK}) + 50 ns**	t_{CLK} = CLK Period (max) (max)
		Minimum for 9-Bits *(Figure 10)* Maximum for 9-Bits (D = 15)	2 (t_{CLK}) 2 (t_{CLK})	**2 (t_{CLK}) + 50 ns** **32 (t_{CLK}) + 50 ns**	(max) (max)

2.0 Electrical Specifications (Continued)

2.2.2 Converter Dynamic Characteristics The following specifications apply only to the LM12434 and LM12438 for $V_A^+ = V_D^+ = 5V$, AGND = DGND = 0V, $V_{REF+} = 4.096V$, $V_{REF-} = 0V$, 12-bit + sign conversion mode, $f_{CLK} = 8.0$ MHz, throughput rate = 133.3 kHz, $R_S = 25\Omega$, source impedance for V_{REF+} and $V_{REF-} \leq 25\Omega$, fully-differential input with fixed 2.048V common-mode voltage, and minimum acquisition time unless otherwise specified. **Boldface limits apply for $T_A = T_J = T_{MIN}$ to T_{MAX}**; all other limits $T_A = T_J = 25°C$. (Notes 6, 7, 8 and 9) (Continued)

Symbol	Parameter	Conditions	Typical (Note 10)	Limits (Note 11)	Units (Limit)
t_Z	Auto-Zero Time	Sequencer State S2 *(Figure 10)*	76 (t_{CLK})	**76 (t_{CLK}) + 50 ns**	(max)
t_{CAL}	Full Calibration Time	Sequencer State S2 *(Figure 10)*	4944 (t_{CLK})	**4944 (t_{CLK}) + 50 ns**	(max)
	Throughput Rate	(Note 18)	142	**140**	kHz (min)
t_{WD}	"Watchdog" Mode Comparison Time	Sequencer States S6, S4, and S5 *(Figure 10)*	11 (t_{CLK})	**11 (t_{CLK}) + 50 ns**	(max)
SNR	Signal-to-Noise Ratio, Differential Input	$V_{IN} = \pm 4.096V$ (Note 20) $f_{IN} = 1$ kHz $f_{IN} = 10$ kHz $f_{IN} = 62$ kHz	 79 79 70		 dB dB dB
SNR	Signal-to-Noise Ratio, Single-Ended Input	$V_{IN} = 4.096\ V_{p-p}$ $f_{IN} = 1$ kHz $f_{IN} = 10$ kHz $f_{IN} = 62$ kHz	 71 71 67		 dB dB dB
SINAD	Signal-to-Noise + Distortion Ratio, Differential Input	$V_{IN} = \pm 4.096V$ (Note 20) $f_{IN} = 1$ kHz $f_{IN} = 10$ kHz $f_{IN} = 62$ kHz	 79 78 67		 dB dB dB
SINAD	Signal-to-Noise + Distortion Ratio, Single-Ended Input	$V_{IN} = 4.096\ V_{p-p}$ $f_{IN} = 1$ kHz $f_{IN} = 10$ kHz $f_{IN} = 62$ kHz	 71 70 64		 dB dB dB
THD	Total Harmonic Distortion, Differential Input	$V_{IN} = \pm 4.096V$ (Note 20) $f_{IN} = 1$ kHz $f_{IN} = 10$ kHz $f_{IN} = 62$ kHz	 −90 −85 −71		 dBc dBc dBc
THD	Total Harmonic Distortion, Distortion, Single-Ended Input	$V_{IN} = 4.096\ V_{p-p}$ $f_{IN} = 1$ kHz $f_{IN} = 10$ kHz $f_{IN} = 62$ kHz	 −88 −82 −67		 dBc dBc dBc
ENOB	Effective Number of Bits, Differential Input	$V_{IN} = \pm 4.096V$ (Note 20) $f_{IN} = 1$ kHz $f_{IN} = 10$ kHz $f_{IN} = 62$ kHz	 12.6 12.2 12.1		 Bits Bits Bits
ENOB	Effective Number of Bits, Single-Ended Input	$V_{IN} = 4.096\ V_{p-p}$ $f_{IN} = 1$ kHz $f_{IN} = 10$ kHz $f_{IN} = 62$ kHz	 11.3 11.2 10.8		 Bits Bits Bits
SFDR	Spurious Free Dynamic Range, Differential Input	$V_{IN} = \pm 4.096V$ (Note 20) $f_{IN} = 1$ kHz $f_{IN} = 10$ kHz $f_{IN} = 62$ kHz	 90 86 76		 dBc dBc dBc
SFDR	Spurious Free Dynamic Range, Single-Ended Input	$V_{IN} = 4.096V\ V_{p-p}$ $f_{IN} = 1$ kHz $f_{IN} = 10$ kHz $f_{IN} = 62$ kHz	 90 85 72		 dBc dBc dBc

2.0 Electrical Specifications (Continued)

2.2.2 Converter Dynamic Characteristics The following specifications apply only to the LM12434 and LM12438 for $V_A^+ = V_D^+ = 5V$, AGND = DGND = 0V, $V_{REF+} = 4.096V$, $V_{REF-} = 0V$, 12-bit + sign conversion mode, $f_{CLK} = 8.0$ MHz, throughput rate = 133.3 kHz, $R_S = 25\Omega$, source impedance for V_{REF+} and $V_{REF-} \leq 25\Omega$, fully-differential input with fixed 2.048V common-mode voltage, and minimum acquisition time unless otherwise specified. **Boldface limits apply for $T_A = T_J = T_{MIN}$ to T_{MAX}**; all other limits $T_A = T_J = 25°C$. (Notes 6, 7, 8 and 9) (Continued)

Symbol	Parameter	Conditions	Typical (Note 10)	Limits (Note 11)	Units (Limit)
IMD	Two Tone Intermodulation Distortion Differential Input	$V_{IN} = \pm 4.096V$ (Note 20) $f_1 = 19.190$ kHz $f_2 = 19.482$ kHz	−82		dBc
IMD	Two Tone Intermodulation Distortion Single Ended Input	$V_{IN} = 4.096\ V_{pp}$ $f_1 = 19.190$ kHz $f_2 = 19.482$ kHz	−80		dBc
	Multiplexer Channel-to-Channel Crosstalk	$V_{IN} = 4.096\ V_{PP}$ $f_{IN} = 5$ kHz $f_{CROSSTALK} = 40$ kHz LM12434 MUXOUT Only and LM12438 MUX plus Converter (Note 21)	−90		dBc
t_{PU}	Power-Up Time		10		ms
t_{WU}	Wake-Up Time	(Note 22)	2		ms

2.2.3 DC Characteristics The following specifications apply to the LM12434 and LM12{L}438 for $V_A^+ = V_D^+ = 5V$ {3.3V], AGND = DGND = 0V, $V_{REF+} = 4.096V$ {2.5V}, $V_{REF-} = 0V$, $f_{CLK} = 8.0$ MHz {6 MHz}and minimum acquisition time unless otherwise specified. **Boldface limits apply for $T_A = T_J = T_{MIN}$ to T_{MAX}**; all other limits $T_A = T_J = 25°C$. (Notes 6, 7 and 8)

Symbol	Parameter	Conditions	Typical (Note 10)	Limits (Note 11)	Units (Limit)
I_D^+	V_D^+ Supply Current	$f_{CLK} = 8$ MHz {6 MHz} f_{SCLK} = Stopped $f_{SCLK} = 10$ MHz {8 MHz}	 2.0 {1.4} 4.0 {2.0}	 5.0 {2.5}	 mA (max) mA (max)
I_A^+	V_A^+ Supply Current	$f_{CLK} = 8$ MHz {6 MHz}	2.8 {2.2}	4.0 {3.5}	mA (max)
I_{ST}	Stand-By Supply Current ($I_D^+ + I_A^+$)	Stand-By Mode Selected f_{SCLK} = Stopped f_{CLK} = Stopped $f_{CLK} = 8$ MHz {6 MHz} $f_{SCLK} = 10$ MHz {8 MHz} f_{CLK} = Stopped $f_{CLK} = 8$ MHz {6 MHz}	 5{5} 120 {50} 1.4 {0.8} 1.4 {0.8}	 	 μA (max) μA (max) mA (max) mA (max)
	Multiplexer ON-Channel Leakage Current	$V_A^+ = 5.5V$ ON-Channel = 5.5V OFF-Channel = 0V ON-Channel = 0V OFF-Channel = 5.5V	 0.1	 **1.0** {**3.0**} **1.0** {**3.0**}	 μA (max) μA (max)
	Multiplexer OFF-Channel Leakage Current	$V_A^+ = 5.5V$ {3.3V} ON-Channel = 5.5V {3.3V} OFF-Channel = 0V ON-Channel = 0V OFF-Channel = 5.5V {3.3V}	 0.1	 **1.0** {**3.0**} **1.0** {**3.0**}	 μA (max) μA (max)

2.0 Electrical Specifications (Continued)

2.2.3 DC Characteristics The following specifications apply to the LM12434 and LM12{L}438 for $V_A^+ = V_D^+ = 5V$ {3.3V], AGND = DGND = 0V, $V_{REF+} = 4.096V$ {2.5V}, $V_{REF-} = 0V$, $f_{CLK} = 8.0$ MHz {6 MHz}and minimum acquisition time unless otherwise specified. **Boldface limits apply for $T_A = T_J = T_{MIN}$ to T_{MAX}**; all other limits $T_A = T_J = 25°C$. (Notes 6, 7 and 8) (Continued)

Symbol	Parameter	Conditions	Typical (Note 10)	Limits (Note 11)	Units (Limit)
R_{ON}	Multiplexer ON-Resistance	LM12434			
		$V_{IN} = 5V$	650	**1000**	Ω(max)
		$V_{IN} = 2.5V$	700	**1000**	Ω(max)
		$V_{IN} = 0V$	630	**1000**	Ω(max)
	Multiplexer Channel-to-Channel R_{ON} matching	LM12434			
		$V_{IN} = 5V$	±1.0%	±**3.0%**	(max)
		$V_{IN} = 2.5V$	±1.0%	±**3.0%**	(max)
		$V_{IN} = 0V$	±1.0%	±**3.0%**	(max)

2.2.4 Digital DC Characteristics The following specifications apply to the LM12434 and LM12{L}438 for $V_A^+ = V_D^+ = 5V$ {3.3V}, AGND = DGND = 0V, unless otherwise specified. **Boldface limits apply for $T_A = T_J = T_{MIN}$ to T_{MAX}**; all other limits $T_A = T_J = 25°C$. (Notes 6, 7 and 8)

Symbol	Parameter	Conditions	Typical (Note 10)	Limits (Note 11)	Units (Limit)
$V_{IN(1)}$	Logical "1" Input Voltage	$V_A^+ = V_D^+ = 5.5V$ {3.6V}		**2.0**	V (min)
$V_{IN(0)}$	Logical "0" Input Voltage	$V_A^+ = V_D^+ = 4.5V$ {3.0V}		**0.8**	V (max)
$I_{IN(1)}$	Logical "1" Input Current	$V_{IN} = 5V$ {3.3V}	0.005	**1.0**	μA (max)
$I_{IN(0)}$	Logical "0" Input Current	$V_{IN} = 0V$	−0.005	−**1.0**	μA (max)
C_{IN}	All Digital Inputs		6		pF
$V_{OUT(1)}$	Logical "1" Output Voltage	$V_A^+ = V_D^+ = 4.5V$ {3.0V}			
		$I_{OUT} = -360\ \mu A$		**2.4**	V (min)
		$I_{OUT} = -10\ \mu A$		**4.25** {**2.9**}	V (min)
$V_{OUT(0)}$	Logical "0" Output Voltage	$V_A^+ = V_D^+ = 4.5V$ {3.0V}			
		$I_{OUT} = 1.6$ mA		**0.4**	V (max)
I_{OUT}	TRI-STATE® Output Leakage Current	$V_{OUT} = 0V$	−0.05	−**3.0**	μA (max)
		$V_{OUT} = 5V$ {3.3V}	0.05	**3.0**	μA (max)

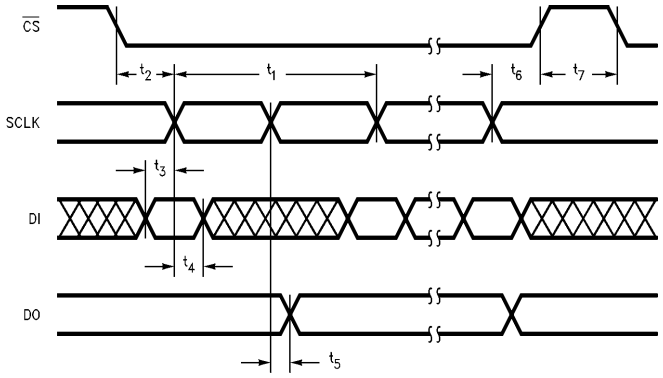
2.0 Electrical Specifications (Continued)

2.3 DIGITAL SWITCHING CHARACTERISTICS The following specifications apply to the LM12434 and LM12{L}438 for $V_A+ = V_D+ = 5V$ {3.3V}, AGND = DGND = 0V, C_L (load capacitance) on output lines = 80 pF unless otherwise specified. **Boldface limits apply for $T_A = T_J = T_{MIN}$ to T_{MAX},** all other limits for $T_A = T_J = 25°C$. (Notes 6, 7, and 9)

2.3.1 Standard Mode Interface (MICROWIRE/PLUS™, SCI and SPI/QSPI)

Symbol (See *Figure* Below)	Parameter	Conditions	Typical (Note 10)	Limits (Note 11)	Units (Limit)
t_1	SCLK (Serial Clock) Period			**100** {**125**}	ns (min)
t_2	CS Set-Up Time to First Clock Transition			**25** {**30**}	ns (min)
t_3	DI Valid Set-Up Time to Data Capture Transition of SCLK			**0**	ns (min)
t_4	DI Valid Hold Time to Data Capture Transition of SCLK			**40**	ns (min)
t_5	DO Hold Time from Data Shift Transition of SCLK			**70** {**120**}	ns (max)
t_6	CS Hold Time from Last SCLK Transition in a Read or Write Cycle (Excluding Burst Read Cycle)			**25**	ns (min)
t_7	CS Inactive to CS Active Again			**3**	CLK Cycle (min)*
t_8	SCLK Idle Time between the End of the Command Byte Transfer and the Start of the Data Transfer in Read Cycles			**3**	CLK Cycle (min)*

*CLK is the main clock input to the device, pin number 24 in PLCC package or pin number 2 in SO package.



TL/H/11879–18

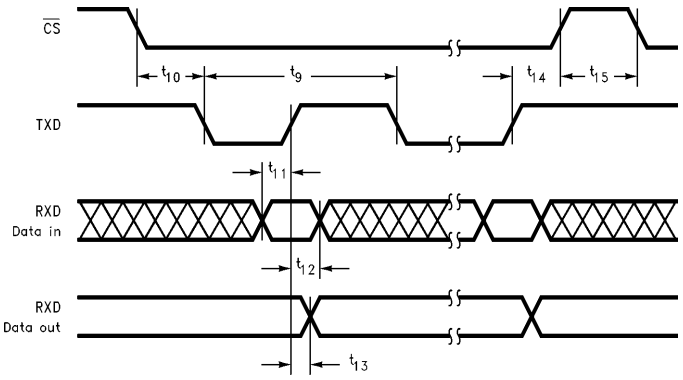
2.0 Electrical Specifications (Continued)

2.3 DIGITAL SWITCHING CHARACTERISTICS The following specifications apply to the LM12434 and LM12{L}438 for $V_A+ = V_D+ = 5V$ {3.3V}, AGND = DGND = 0V, C_L (load capacitance) on output lines = 80 pF unless otherwise specified. **Boldface limits apply for $T_A = T_J = T_{MIN}$ to T_{MAX},** all other limits for $T_A = T_J = 25°C$. (Notes 6, 7, and 9) (Continued)

2.3.2 8051 Interface Mode

Symbol (See *Figure* Below)	Parameter	Conditions	Typical (Note 10)	Limits (Note 11)	Units (Limit)
t_9	TXD (Serial Clock Period)			**125** {**250**}	ns (min)
t_{10}	CS Set-Up Time to First Clock Transition			**25** {**40**}	ns (min)
t_{11}	Data in Valid Set-Up Time to TXD Clock High			**40**	ns (min)
t_{12}	Data in Valid Hold Time from TXD Clock High			**40** {**90**}	ns (min)
t_{13}	Data Out Hold Time from TXD Clock High			**70** {**120**}	ns (max)
t_{14}	CS Hold Time from Last TXD High in a Read or Write Cycle (Excluding Burst Read Cycle)			**25** {**50**}	ns (min)
t_{15}	CS Inactive to CS Active Again			**3**	CLK Cycle (min)*
t_{16}	SCLK Idle Time between the End of the Command Byte Transfer and the Start of the Data Transfer in Read Cycles			**3**	CLK Cycle (min)*

*CLK is the main clock input to the device, pin number 24 in PLCC package or pin number 2 in SO package.



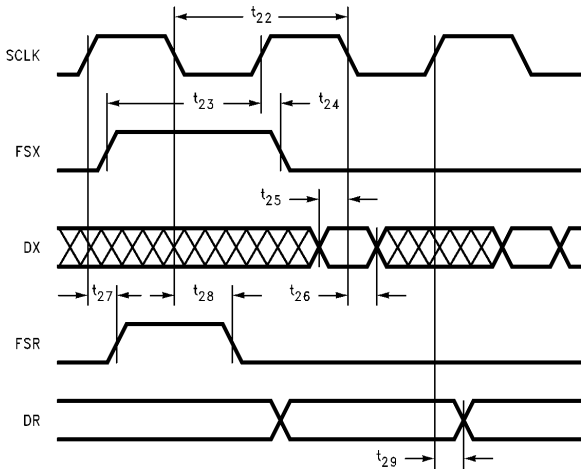
TL/H/11879–21

2.0 Electrical Specifications (Continued)

2.3 DIGITAL SWITCHING CHARACTERISTICS The following specifications apply to the LM12434 and LM12{L}438 for $V_A+ = V_D+ = 5V$ {3.3V}, AGND = DGND = 0V, C_L (load capacitance) on output lines = 80 pF unless otherwise specified. **Boldface limits apply for $T_A = T_J = T_{MIN}$ to T_{MAX},** all other limits for $T_A = T_J = 25°C$. (Notes 6, 7, and 9) (Continued)

2.3.3 TMS320 Interface Mode

Symbol (See *Figure* Below)	Parameter	Conditions	Typical (Note 10)	Limits (Note 11)	Units (Limit)
t_{22}	SCLK (Serial Clock) Period			**125** {**167**}	ns (min)
t_{23}	FSX Set-Up Time to SCLK High			**30** {**50**}	ns (min)
t_{24}	FSX Hold Time from SCLK High			**10**	ns (min)
t_{25}	Data in (DX) Set-Up Time to SCLK Low			**0**	ns (min)
t_{26}	Data in DX Hold Time from SCLK Low			**30** {**120**}	ns (min)
t_{27}	FSR High from SCLK High			**80** {**100**}	ns (max)
t_{28}	FSR Low from SCLK Low			**120**	ns (max)
t_{29}	SCLK High to Data Out (DR) Change			**90**	ns (max)



TL/H/11879–23

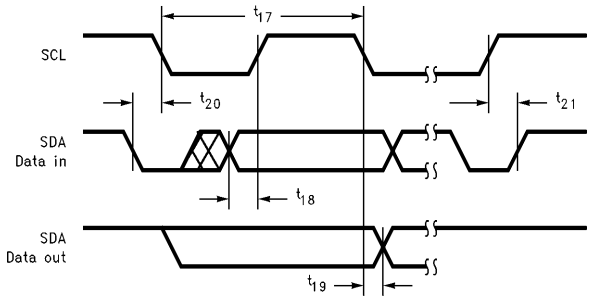
2.0 Electrical Specifications (Continued)

2.3 DIGITAL SWITCHING CHARACTERISTICS The following specifications apply to the LM12434 and LM12{L}438 for $V_A+ = V_D+ = 5V$ {3.3V}, AGND = DGND = 0V, C_L (load capacitance) on output lines = 80 pF unless otherwise specified. **Boldface limits apply for $T_A = T_J = T_{MIN}$ to T_{MAX},** all other limits for $T_A = T_J = 25°C$. (Notes 6, 7, and 9) (Continued)

2.3.4 I²C Bus Interface

The switching characteristics of the LM12434/8 for I²C bus interface fully meets or exceeds the published specifications of the I²C bus. The following parameters given here are the timing relationships between SCL and SDA signals related to the LM12434/8. They are not the I²C bus specifications.

Symbol (See *Figure* Below)	Parameter	Conditions	Typical (Note 10)	Limits (Note 11)	Units (Limit)
t_{17}	SCL (Clock) Period			**2500** {**10000**}	ns (min)
t_{18}	Data in Set-Up Time to SCL High			**30**	ns (min)
t_{19}	Data Out Stable after SCL Low			**900** {**1400**}	ns (max)
t_{20}	SDA Low Set-Up Time to SCL Low (Start Condition)			**40**	ns (min)
t_{21}	SDA High Hold Time after SCL High (Stop Condition)			**40**	ns (min)



TL/H/11879–22

2.0 Electrical Specifications (Continued)

2.4 NOTES ON SPECIFICATIONS

Note 1: Absolute Maximum Ratings indicate limits beyond which damage to the device may occur. Operating Ratings indicate conditions for which the device is functional, but do not guarantee specific performance limits. For guaranteed specifications and test conditions, see the Electrical Characteristics. The guaranteed specifications apply only for the test conditions listed. Some performance characteristics may degrade when the device is not operated under the listed test conditions.

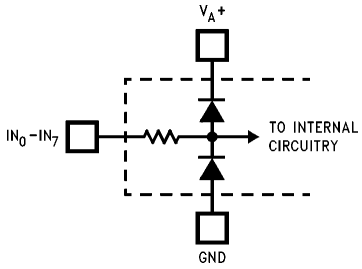
Note 2: All voltages are measured with respect to GND, unless otherwise specified. GND specifies either AGND and/or DGND and V^+ specifies either V_A^+ and/or V_D^+.

Note 3: When the input voltage (V_{IN}) at any pin exceeds the power supply rails ($V_{IN} < GND$ or $V_{IN} > (V_A^+$ or $V_D^+)$), the current at that pin should be limited to 5 mA. The 20 mA maximum package input current rating allows the voltage at any four pins, with an input current of 5 mA, to simultaneously exceed the power supply voltages.

Note 4: The maximum power dissipation must be derated at elevated temperatures and is dictated by T_{Jmax} (maximum junction temperature), Θ_{JA} (package junction to ambient thermal resistance), and T_A (ambient temperature). The maximum allowable power dissipation at any temperature is $PD_{max} = (T_{Jmax} - T_A)/\Theta_{JA}$ or the number given in the Absolute Maximum Ratings, whichever is lower. For this device, $T_{Jmax} = 150°C$, and the typical thermal resistance (Θ_{JA}) of the V package, when board mounted, is 70°C/W and in the WM package, when board mounted, is 60°C/W.

Note 5: Human body model, 100 pF discharged through a 1.5 kΩ resistor.

Note 6: Two on-chip diodes are tied to each analog input through a series resistor, as shown below. Input voltage magnitude up to 5V above V_A^+ or 5V below GND will not damage the part. However, errors in the A/D conversion can occur if these diodes are forward biased by more than 100 mV. As an example, if V_A^+ is 4.5 V_{DC}, the full-scale input voltage must be $\leq$4.6 V_{DC} to ensure accurate conversions.



TL/H/11879–5

Note 7: V_A^+ and V_D^+ must be connected together to the same power supply voltage and bypassed with separate capacitors at each V^+ pin to assure conversion/comparison accuracy. Refer to Section 8.0 for a detailed discussion on grounding the DAS.

Note 8: Accuracy is guaranteed when operating the LM12434/LM12{L}438 at $f_{CLK} = 8$ MHz {6 MHz}.

Note 9: With the test condition for V_{REF} ($V_{REF+} - V_{REF-}$) given as +4.096V, the 12-bit LSB is 1 mV and the 8-bit/"Watchdog" LSB is 19 mV.

Note 10: Typicals are at $T_A = 25°C$ and represent most likely parametric norm.

Note 11: Limits are guaranteed to National's AOQL (Average Output Quality Level).

Note 12: Positive integral linearity error is defined as the deviation of the analog value, expressed in LSBs, from the straight line that passes through positive full-scale and zero. For negative integral linearity error the straight line passes through negative full-scale and zero. (See *Figures 5b* and *5c*).

Note 13: Zero error is a measure of the deviation from the mid-scale voltage (a code of zero), expressed in LSB. It is the average value of the code transitions between −1 to 0 and 0 to +1 (see *Figure 6*).

Note 14: The DC common-mode error is measured with both the inverted and non-inverted inputs shorted together and driven from 0V to 5V {3.3V}. The measured value is referred to the resulting output value when the inputs are driven with a 2.5V {1.65V}signal.

Note 15: Power Supply Sensitivity is measured after Auto-Zero and/or Auto-Calibration cycle has been completed with V_A^+ and V_D^+ at the specified extremes.

Note 16: V_{REFCM} (Reference Voltage Common Mode Range) is defined as $(V_{REF+} + V_{REF-})/2$. See *Figures 3* and *4*.

Note 17: The device self-calibration technique ensures linearity and offset errors as specified, but noise inherent in the self-calibration process will result in a repeatability uncertainty of ±0.10 LSB.

Note 18: The Throughput Rate is for a single instruction repeated continuously while reading data during conversions with a serial clock frequency $f_{SCLK} = 10$ MHz {8 MHz}. Sequencer states 0 (1 clock cycle), 1 (1 clock cycle), 7 (9 clock cycles) and 5 (44 clock cycles) are used (see *Figure 10*) for a total of 56 clock cycles per conversion. The Throughput Rate is f_{CLK} (MHz)/N, where N is the number of clock cycles/conversion.

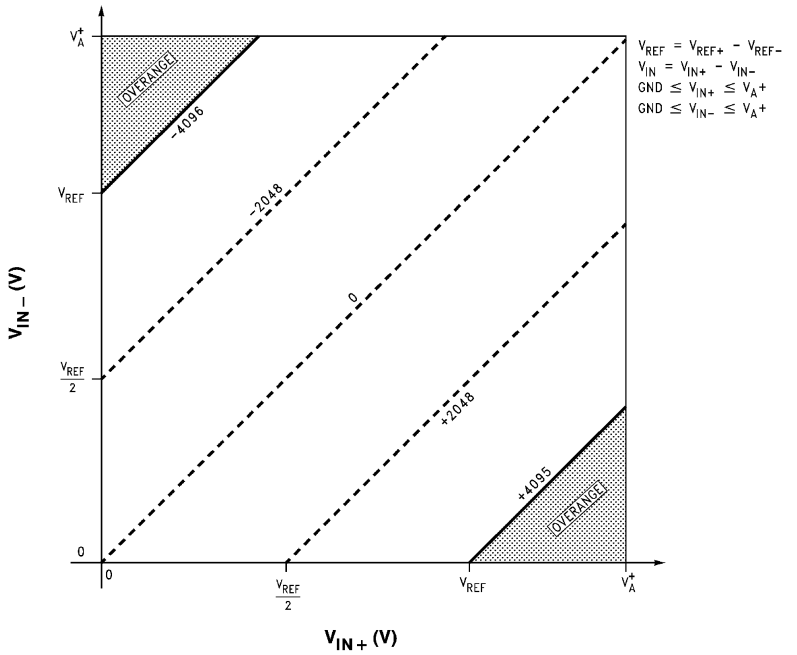
Note 19: See AN-450 "Surface Mounting Methods and their Effect on Product Reliability" for other methods of soldering surface mount devices.

Note 20: Each input referenced to the other input sees a ±4.096V (8.192 V_{p-p}) sine wave. However the voltage at each input stays within the supply rails. This is done by applying two sine waves with 180° phase shift and 4.096 V_{p-p} (between GND and V_A^+) to the inputs.

Note 21: Multiplexer channel-to-channel crosstalk is measured by placing a sinewave with a frequency of $f_{IN} = 5$ kHz on one channel and another sinewave with a frequency of $f_{CROSSTALK} = 40$ kHz on the remaining channels. 8192 conversions are performed on the channel with the 5 kHz signal. A special response is generated by doing a FFT on these samples. The crosstalk is then calculated by subtracting the amplitude of the frequency component at 40 kHz from the amplitude of the fundamental frequency at 5 kHz.

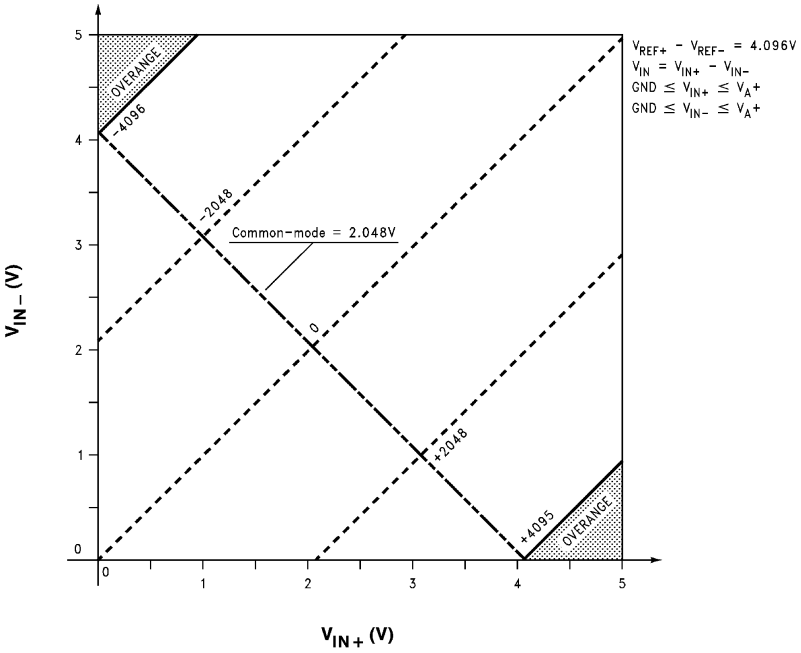
Note 22: Interrupt 7 is set to return an out-of-standby flag 10 ms (typ) after the device is requested to come out of standby mode. However, characterization has shown the devices will perform to their rated specifications in 2 ms.

3.0 Electrical Characteristics



TL/H/11879–6

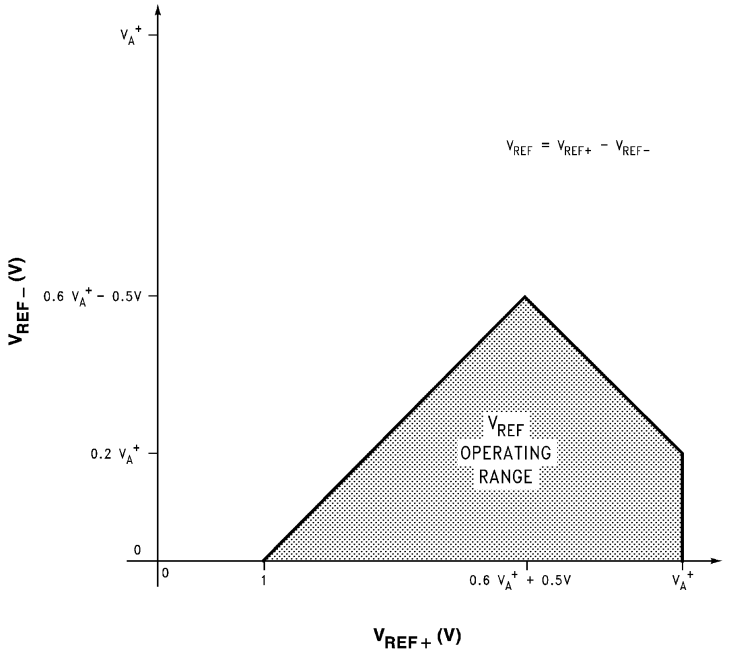
FIGURE 1. Output Digital Code vs the Operating Input Voltage Range (General Case)



TL/H/11879–7

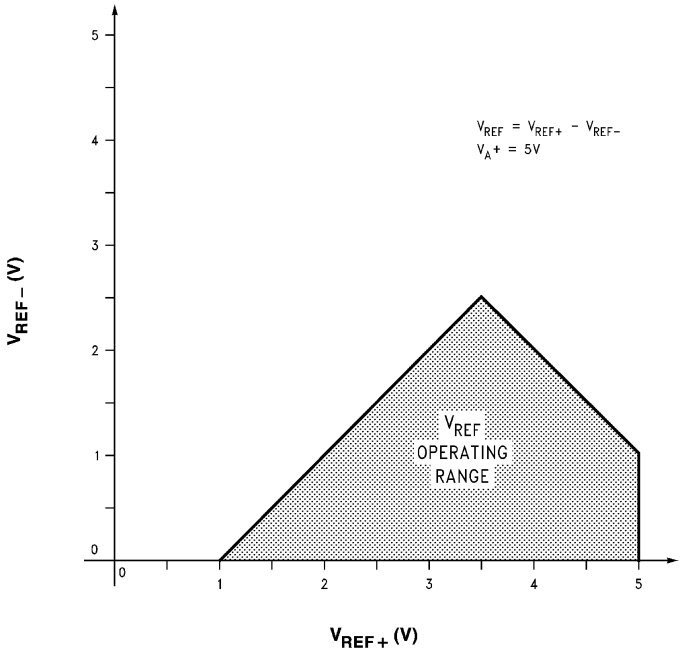
FIGURE 2. Output Digital Code vs the Operating Input Voltage Range for $V_{REF} = 4.096V$

3.0 Electrical Characteristics (Continued)



TL/H/11879–8

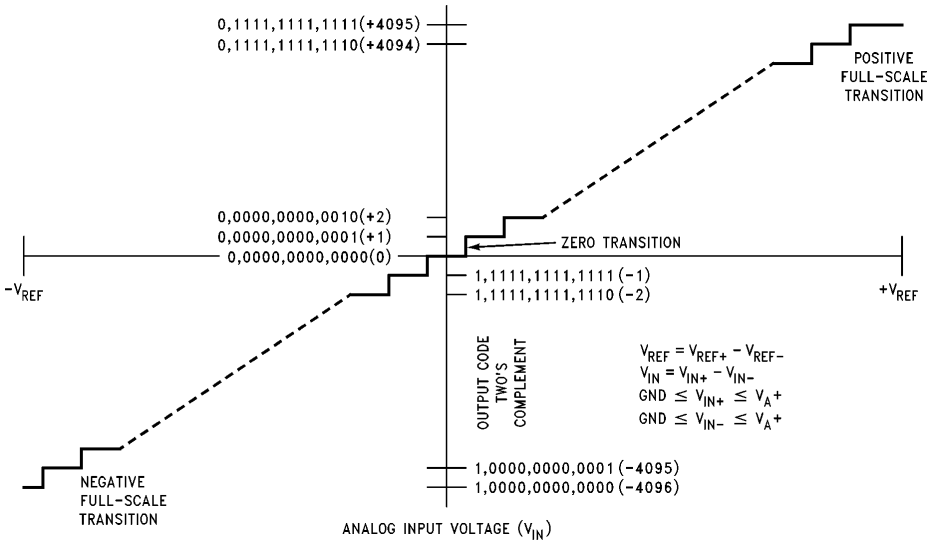
FIGURE 3. V_{REF} Operating Range (General Case)



TL/H/11879–9

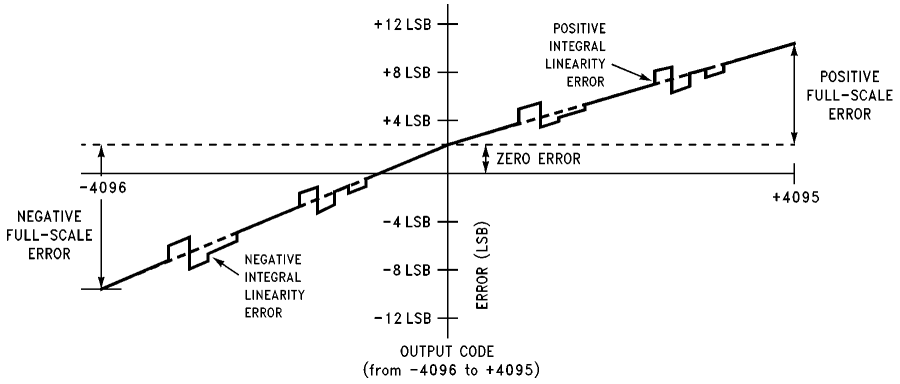
FIGURE 4. V_{REF} Operating Range for $V_A^+ = 5V$

3.0 Electrical Characteristics (Continued)



TL/H/11879–10

FIGURE 5a. Transfer Characteristic



TL/H/11879–11

FIGURE 5b. Simplified Error Curve vs Output Code without Auto-Calibration or Auto-Zero Cycles

3.0 Electrical Characteristics (Continued)

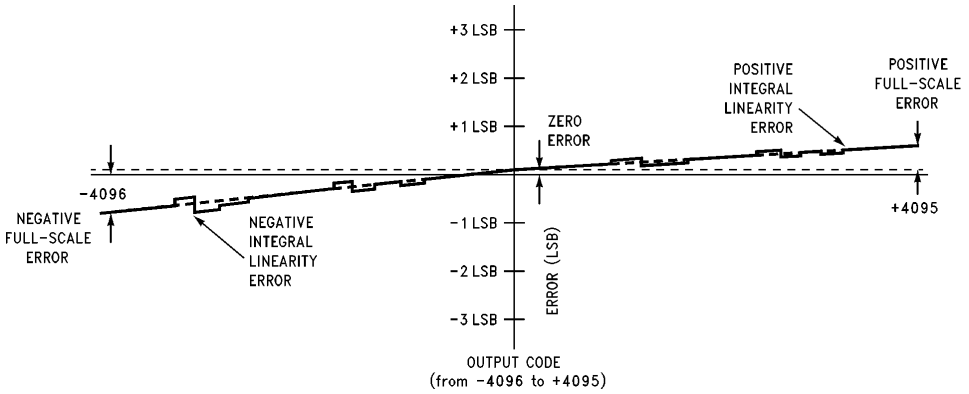


FIGURE 5c. Simplified Error Curve vs Output Code after Auto-Calibration Cycle

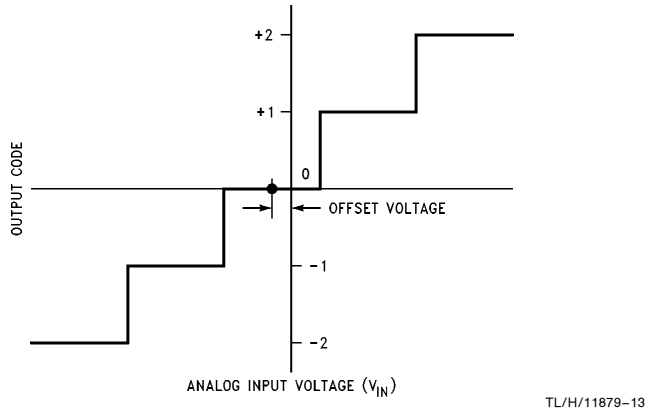
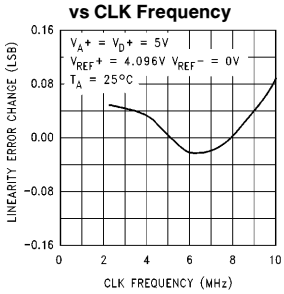


FIGURE 6. Offset or Zero Error Voltage

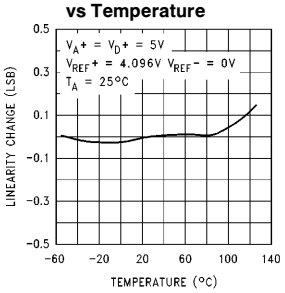
4.0 Typical Performance Characteristics

The following curves apply for 12-bit + sign mode after auto-calibration unless otherwise specified. The performance for 8-bit + sign and ''watchdog'' modes is equal to or better than shown. (Note 9)

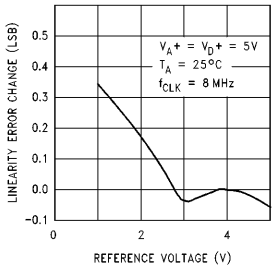
Linearity Error Change vs CLK Frequency



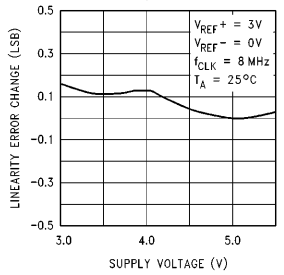
Linearity Error Change vs Temperature



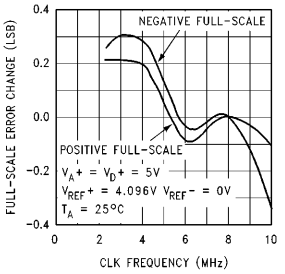
Linearity Error Change vs Reference Voltage



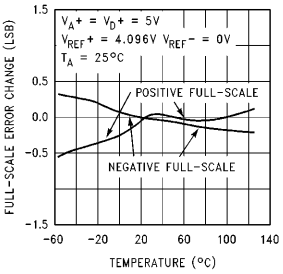
Linearity Error Change vs Supply Voltage



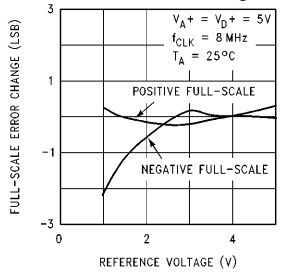
Full-Scale Error Change vs CLK Frequency



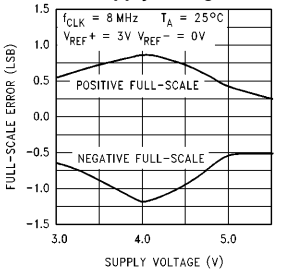
Full-Scale Error Change vs Temperature



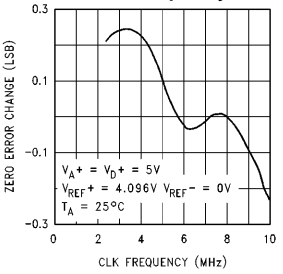
Full-Scale Error Change vs Reference Voltage



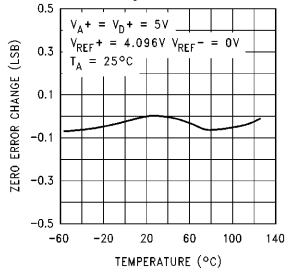
Full-Scale Error vs Supply Voltage



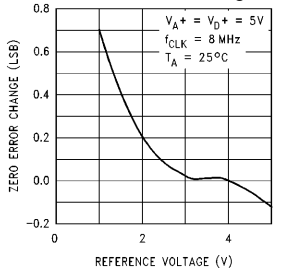
Zero Error Change vs CLK Frequency



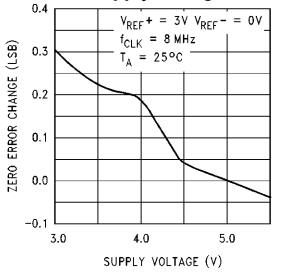
Zero Error Change vs Temperature



Zero Error Change vs Reference Voltage



Zero Error Change vs Supply Voltage

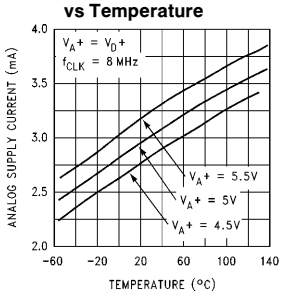


TL/H/11879–14

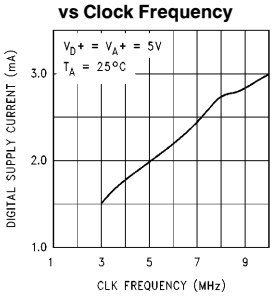
4.0 Typical Performance Characteristics (Continued)

The following curves apply for 12-bit + sign mode after auto-calibration unless otherwise specified. The performance for 8-bit + sign and "watchdog" modes is equal to or better than shown. (Note 9)

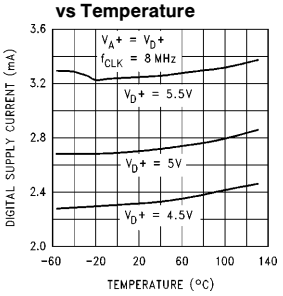
Analog Supply Current vs Temperature



***Digital Supply Current vs Clock Frequency**



***Digital Supply Current vs Temperature**

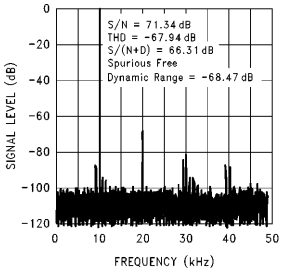


TL/H/11879–15

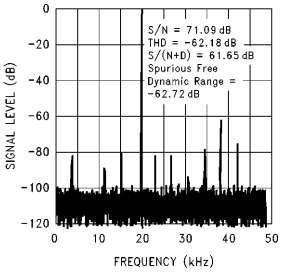
*Free-running conversion and SPI mode data read at 200 ns SCLK period.

The following curves apply to the LM12L438 in 12-bit + sign mode after auto-calibration unless otherwise specified. R_S = 50Ω, T_A = 25°C, $V_A^+ = V_D^+$ = 3.3V, V_{REF} = 2.5V, f_{CLK} = 6 MHz, f_{SCLK} = 8 MHz, V_{IN} = 2.5V ⟶ 0 dB, Sampling Rate = 100 kHz.

Unipolar Spectral Response with 10 kHz Sine Wave at 0 dB



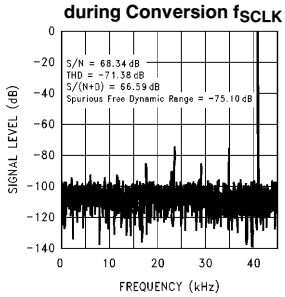
Unipolar Spectral Response with 20 kHz Sine Wave at 0 dB



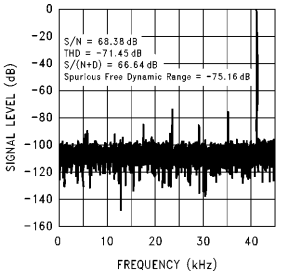
TL/H/11879–84

The following curves apply for 12-bit + sign mode after auto-calibration unless otherwise specified. R_S = 50Ω, T_A = 25°C, $V_A^+ = V_D^+$ = 5V, V_{REF} = 4.096V, f_{CLK} = 8 MHz, f_{SCLK} = 10 MHz, V_{IN} = 4.096V ⟶ 0 dB, Sampling Rate = 100 kHz.

Unipolar Special Response with 41.2 kHz Sine Wave at 0 dB Reading Data during Conversion f_{SCLK} = 10 MHz



Unipolar Special Response with 41.2 kHz Sine Wave at 0 dB Reading Data between Conversions

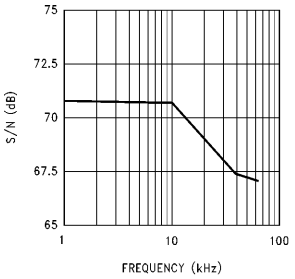


TL/H/11879–55

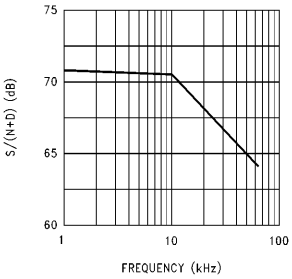
4.0 Typical Performance Characteristics (Continued)

The following curves apply for 12-bit + sign mode after auto-calibration unless otherwise specified. $R_S = 50\Omega$, $T_A = 25°C$, $V_A^+ = V_D^+ = 5V$, $V_{REF} = 4.096V$, $f_{CLK} = 8$ MHz, $f_{SCLK} = 10$ MHz, $V_{IN} = 4.096V \rightarrow 0$ dB, Sampling Rate = 133.3 kHz.

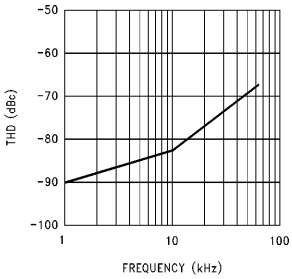
Unipolar Signal-to-Noise Ratio vs Input Frequency



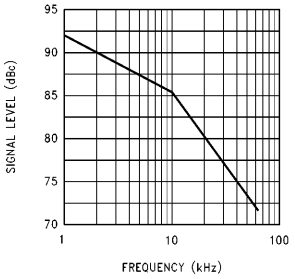
Unipolar Signal-to-Noise + Distortion vs Input Frequency



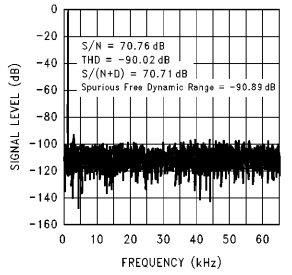
Unipolar Total Harmonic Distortion vs Input Frequency



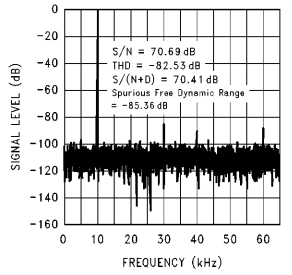
Unipolar Spurious Free Dynamic Range vs Input Frequency



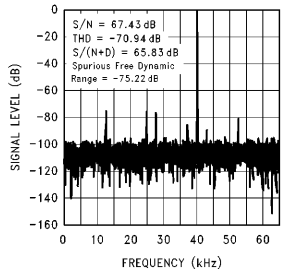
Unipolar Spectral Response with 1.025 kHz Sine Wave at 0 dB



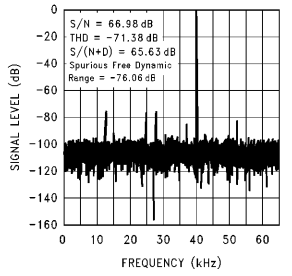
Unipolar Spectral Response with 10.010 kHz Sine Wave at 0 dB



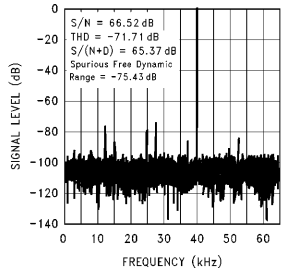
Unipolar Spectral Response with 40.283 kHz Sine Wave at 0 dB



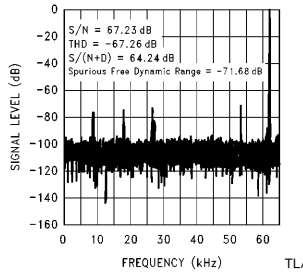
Unipolar Spectral Response with 40.283 kHz Sine Wave at −0.5 dB



Unipolar Spectral Response with 40.283 kHz Sine Wave at −1.0 dB

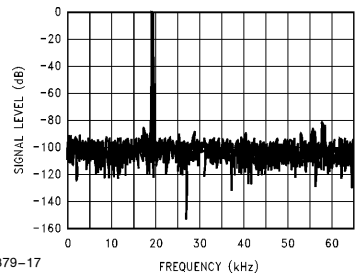


Unipolar Spectral Response with 62.256 kHz Sine Wave at 0 dB



TL/H/11879–17

Unipolar Two Tone Spectral Response with f1 = 19.190 kHz and f2 = 19.482 kHz Sine Wave

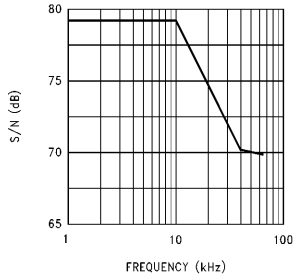


TL/H/11879–24

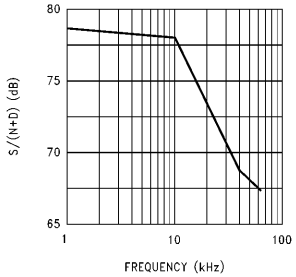
4.0 Typical Performance Characteristics (Continued)

The following curves apply for 12-bit + sign mode after auto-calibration unless otherwise specified.
$R_S = 50\Omega$, $T_A = 25°C$, $V_A^+ = V_D^+ = 5V$, $V_{REF} = 4.096V$, $f_{CLK} = 8$ MHz, $f_{SCLK} = 10$ MHz, $V_{IN} = \pm 4.096V \rightarrow 0$ dB, Sampling Rate = 133.3 kHz.

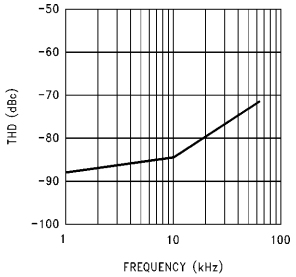
Bipolar Signal-to-Noise Ratio vs Input Frequency



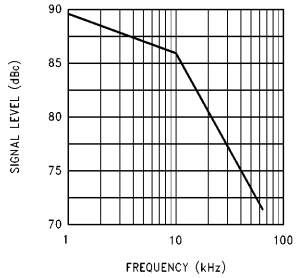
Bipolar Signal-to-Noise + Distortion vs Input Frequency



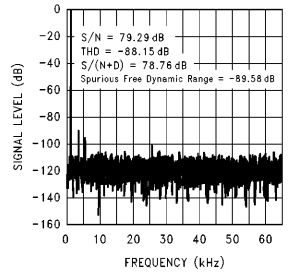
Bipolar Total Harmonic Distortion vs Input Frequency



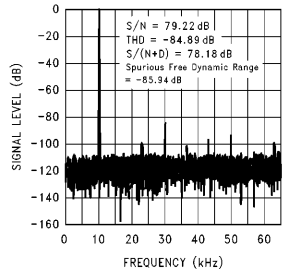
Bipolar Spurious Free Dynamic Range vs Input Frequency



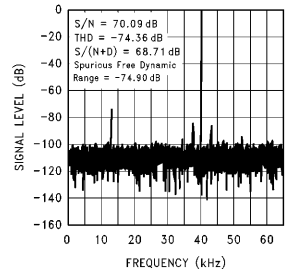
Bipolar Spectral Response with 1.025 kHz Sine Wave at 0 dB



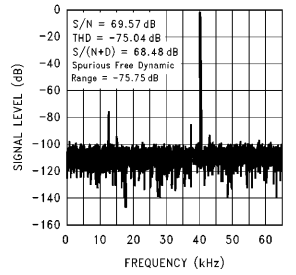
Bipolar Spectral Response with 10.010 kHz Sine Wave at 0 dB



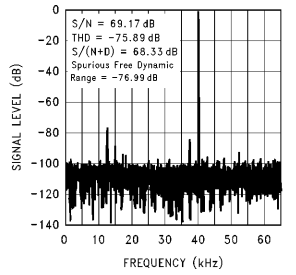
Bipolar Spectral Response with 40.283 kHz Sine Wave at 0 dB



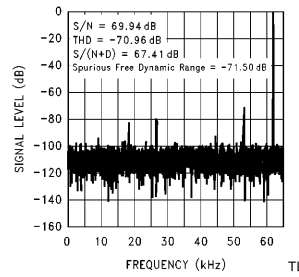
Bipolar Spectral Response with 40.283 kHz Sine Wave at −0.5 dB



Bipolar Spectral Response with 40.283 kHz Sine Wave at −1.0 dB

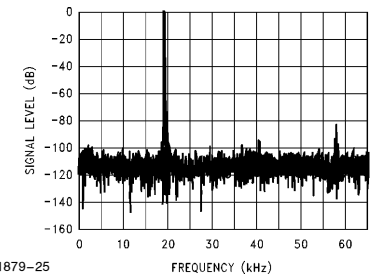


Bipolar Spectral Response with 62.25 kHz Sine Wave at 0 dB



TL/H/11879–25

Bipolar Two Tone Spectral Response with f1 = 19.190 kHz and f2 = 19.482 kHz Sine Waves



TL/H/11879–26

5.0 Pin Descriptions

TABLE I. LM12{L}438 Pin Description

Pin Number		Pin Name	Description
PLCC Pkg.	SO Pkg.		
1	7	DGND	Digital ground. This is the device's digital supply ground connection. It should be connected through a low resistance and low inductance ground return to the system power supply.
2 3 4 5 6 7 8 9	8 9 10 11 12 13 14 15	IN0 IN1 IN2 IN3 IN4 IN5 IN6 IN7	These are the eight analog inputs to the multiplexer. For each conversion to be performed, the active channels are selected according to the instruction RAM programming. Any individual channel can be selected for a single-ended conversion referenced to AGND, or any pair of channels, whether adjacent or non adjacent, can be selected as a fully differential input pairs.
10	16	V_{REF}^+	Positive reference input. The operating voltage range for this input is $1V \leq V_{REF}^+ \leq V_A^+$ (See *Figures 3* and *4*). In order to achieve 12-bit performance this pin should be by passed to AGND at least with a parallel combination of a 10 μF and a 0.1 μF (ceramic) capacitor. The capacitors should be placed as close to the part as possible.
11	17	V_{REF}^-	Negative reference input. The operating voltage range for this input is $0\ V \leq V_{REF}^- \leq V_{REF}^+ - 1V$ (See *Figures 3* and *4*). In order to achieve 12-bit performance, this pin should be bypassed to AGND at least with a parallel combination of a 10 μF and a 0.1 μF (ceramic) capacitor. The capacitors should be placed as close to the part as possible.
12	18	AGND	Analog ground. This is the device's analog supply ground connection. It should be connected through a low resistance and low inductance ground return to the system power supply.
13	19	V_A^+	Analog supply. This is the supply connection for the analog circuitry. The device operating supply voltage range is $+3.0V$ to $+5.5V$. Accuracy is guaranteed only if the V_A^+ and V_D^+ are connected to the same potential. In order to achieve 12-bit performance, this pin should be bypassed to AGND at least with a parallel combination of a 10 μF and a 0.1 μF (ceramic) capacitor. The capacitors should be placed as close to the part as possible.
14	20	DGND	Digital ground. See above definition.
15 16	21 22	V_D^+	Digital supply. This is the supply connection for the analog circuitry. The device operating supply voltage range is $+3.0V$ to $+5.5V$. The device accuracy is guaranteed only if the V_A^+ and V_D^+ are connected to the same potential. In order to achieve 12-bit performance this pin should be by passed to DGND at least with a parallel combination of a 10 μF and a 0.1 μF (ceramic) capacitor. The capacitors should be placed as close to the part as possible.
17	23	P5	P1–P5 are the multi-function serial interface input or output pins that have different assignments depending on the selected mode. Serial interface input: Standard: SCLK 8051: TXD I^2C: SCL TMS320: DR
18	24	P4	Serial interface input/output: Standard: DO 8051: RXD I^2C: SDA TMS320: DR
19	25	P3	Serial interface input: Standard: DI 8051: CS I^2C: SAD2 TMS320: DX

5.0 Pin Descriptions (Continued)

TABLE I. LM12{L}438 Pin Description (Continued)

Pin Number		Pin Name	Description		
PLCC Pkg.	SO Pkg.				
20	26	P2	Serial interface input:	Standard: 8051: I^2C: TMS320:	CS 1 SAD1 FSX
21	27	P1	Serial interface input:	Standard: 8051: I^2C: TMS320:	$\overline{R}$/F (Clock rise/fall) 1 SAD0 FSR
22 23	28 1	MODESEL2 MODESEL1	Serial mode selection inputs. The logic states of these inputs determine the operation of the serial mode as shown below. The standard mode covers the National's MICROWIRE, Motorola's SPI and Hitachi's SCl protocols. MODESEL1, MODESEL2:	01 00 10 11	Standard mode 8051 I^2C TMS320
24	2	CLK	The device main clock input. The operating range of clock frequency is 0.05 MHz to 10.0 MHz. The device accuracy is guaranteed only for the clock frequencies indicated in the specification tables.		
25	3	$\overline{INT}$	Interrupt output. This is an active low output. An interrupt is generated any time a non-masked interrupt condition takes place. There are seven different conditions that can generate an interrupt. (Refer to Section 6.2.4). The interrupt is set high (inactive) by reading the interrupt status register. This output can drive up to 100 pF of capacitive loads. An external buffer should be used for driving higher capacitive loads.		
26	4	SYNC	Synchronization input/output. SYNC is an input if the Configuration Register's SYNC I/O bit is ''0'' and output when the bit is ''1''. When sync is an input, a rising edge on this pin causes the internal S/H to hold the input signal and a conversion cycle or a comparison cycle (depending on the programmed instruction) to be started. (The conversion or comparison actually begins on the rising edge of the CLK immediately following the rising edge of sync.) When output, it goes high at the start of a conversion or a comparison cycle and returns low when the cycle is completed. At power up the SYNC pin is set as an input. When used as an output it can drive up to 100 pF of capacitive loads. An external buffer should be used for driving higher capacitive loads.		
27	5	$\overline{STANDBYOUT}$	Stand-by output. This is an active low output. $\overline{STANDBYOUT}$ will be activated when the LM12{L}438 is put into stand-by mode through the Configuration Register's stand-by bit. It is used to force any other devices in the system (signal conditioning circuitry, for example) to go into power-down mode. This is done by connecting the ''shutdown'', ''powerdown'', ''standby'', etc. pins of the other ICs to $\overline{STANDBYOUT}$. In those cases where the peripheral ICs do not have the power-down inputs, $\overline{STANDBYOUT}$ can be used to turn off their power through an electronic switch. Note that the logic polarity of the $\overline{STANDBYOUT}$ is the opposite to that of the stand-by bit in the Configuration Register.		
28	6	V_D+	Digital supply. See above definition. LM12434 Pin Description. (Same as LM12{L}438 with the exceptions of the following pins.)		
LM12434 Pin Description (Same As LM12{L}438 with the exception of the following pins.)					
6 7	12 13	MUXOUT− MUXOUT+	Multiplexer outputs. These are the LM12434's externally available analog MUX output pins. Analog inputs are directed to these outputs based on the Instruction RAM programming.		
8 9	14 15	S/H IN− S/H IN+	Sample-and-hold inputs. These are the inverting and non-inverting inputs of the sample-and-hold. LM12434 allows external analog signal conditioning circuits to be placed between MUX outputs and S/H inputs.		

6.0 Operational Information

6.1 FUNCTIONAL DESCRIPTION

The LM12434 and LM12{L}438 are multi-functional Data Acquisition Systems that include a fully differential 12-bit-plus-sign self-calibrating analog-to-digital converter (ADC) with a two's-complement output format, an 8-channel (LM12{L}438) or a 4-channel (LM12434) analog multiplexer, a first-in-first-out (FIFO) register that can store 32 conversion results, and an Instruction RAM that can store as many as eight instructions to be sequentially executed. The LM12434 also has a differential multiplexer output and a differential S/H input. All of this circuitry operates on only a single +5V power supply. For simplicity, the DAS (Data Acquisition System) abbreviation is used as a generic name for the members of the LM12434 and LM12{L}438 family thoughout this discussion.

Figure 7 illustrates the functional block diagram or user programming model of the DAS. Note that this diagram is not meant to reflect the actual implementation of the internal building blocks. The model consists of the following blocks:

— A flexible analog multiplexer with differential output at the front end of the device.

— A fully-differential, self-calibrating 12-bit + sign ADC converter with sample and hold.

— A 32-word FIFO register as the output data buffer.

— An 8-word instruction RAM that can be programmed to repeatedly perform a series of conversions and comparisons on selected input channels.

— A series of registers for overall control and configuration of DAS operation and indication of internal operational status.

— Interrupt generation logic to request service from the processor under specified conditions.

— Serial interface logic for input/output operations between the DAS and the processor. All the registers shown in the diagram can be read and most of them can also be written to by the user through the input/output block.

— A controller unit that manages the interactions of the different blocks inside the DAS and controls the conversion, comparison and calibration sequences.

The DAS has 3 different modes of operation:

— 12-bit + sign conversion

— 8-bit + sign conversion

— 8-bit + sign comparison (also called ''watchdog'' mode)

The fully differential 12-bit-plus-sign ADC uses a charge redistribution topology that includes calibration capabilities. Charge re-distribution ADCs use a capacitor ladder in place of a resistor ladder to form an internal DAC. The DAC is used by a successive approximation register to generate intermediate voltages between the voltages applied to V_{REF}^- and V_{REF}^+. These intermediate voltages are compared against the sampled analog input voltage as each bit is charged.

Conversion accuracy is ensured by an internal auto-calibration system. Two different calibration modes are available; one compensates for offset voltage, or zero error, while the other corrects the ADC's linearity and offset errors.

When correcting offset only, the offset error is measured once and a correction coefficient is created. During the full calibration, the offset error is measured eight times, averaged, and a correction coefficient is created. After completion of either calibration mode, the offset correction coefficient is stored in an internal offset correction register.

The LM12434 and LM12{L}438's overall linearity correction is achieved by correcting the internal DAC's capacitor mismatch. Each capacitor is compared eight times against all remaining smaller value capacitors and any errors are averaged. A correction coefficient is then created and stored in one of the thirteen linearity correction registers. A state machine, using patterns stored in 16-bit x 8-bit ROM, executes each calibration algorithm.

Once the converter has been calibrated, an arithmetic logic unit (ALU) uses the offset correction coefficient and the 13 linearity correction coefficients to reduce the conversion's offset error and linearity error, in the background, during the 12-bit + sign conversion. 8-bit + sign conversions and ''watchdog'' comparisons use only the offset coefficient. An 8-bit + sign conversion requires less than half the time needed for a 12-bit + sign conversion.

Diagnostic Mode

A diagnostic mode is available that allows verification of the LM12{L}438's operation. The diagnostic mode is disabled in the LM12434. This mode internally connects the voltages present at the V_{REF}^+ and V_{REF}^- pins to the internal V_{IN}^+ and V_{IN}^- S/H inputs. This mode is activated by setting the Diagnostic bit (Bit 11) in the Configuration register to a ''1''. More information concerning this mode of operation can be found in Section 6.2.2.

Watchdog Mode

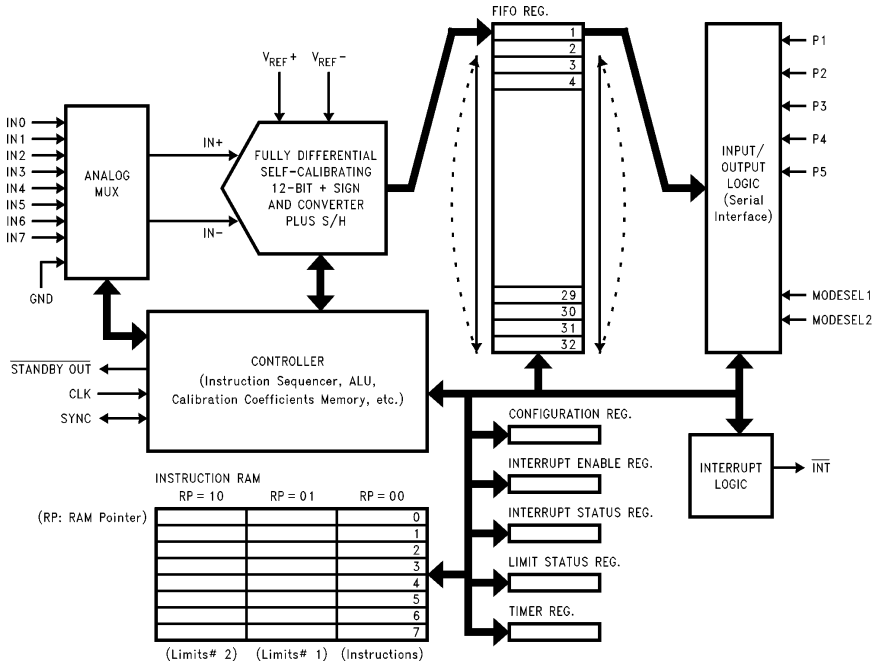
In the watchdog mode no conversion is performed, but the DAS samples an input and compares it with the values of the two limits stored in the Instruction RAM. If the input voltage is above or below the limits (as defined by the user) an interrupt can be generated to indicate a fault condition. The LM12434 and LM{L}438's ''watchdog'' mode is used to monitor a single-ended or differential signal's amplitude and generate an output if the signal's amplitude falls outsidde of a programmable ''window''. Each watchdog instruction includes two limits. An interrupt can be generated if the input signal is above or below either of the two limits. This allows interrupt to be generated when analog voltage inputs are ''outside the window''. After a ''watchdog'' mode interrupt, the processor can then request a conversion on the input signal and read the signal's magnitude.

Analog Input Multiplexer

The analog input multiplexer can be configured for any combination of single-ended or fully differential operation. Each input is referenced to AGND when a multiplexer channel operates in the single-ended mode. Fully differential analog input channels are formed by pairing any two channels together.

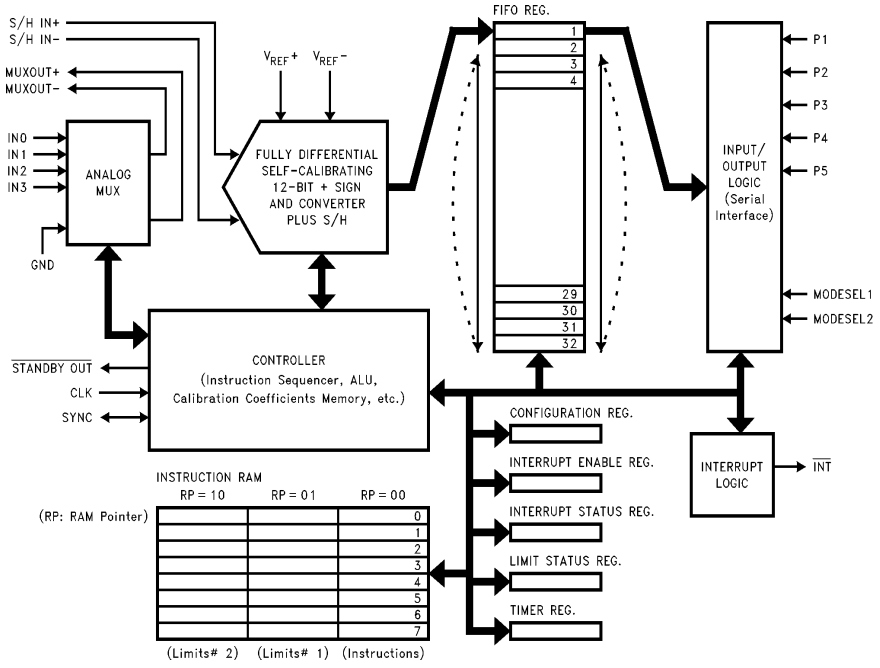
The LM12434's multiplexer outputs and S/H inputs (MUXOUT+, MUXOUT− and S/H IN+, S/H IN−) provide the option for additional analog signal processing after the multiplexer. Fixed-gain amplifiers, programmable-gain amplifiers, filters, and other processing circuits can operate on the multiplexer output signals before they are applied to the ADC's S/H inputs. If external processing is not used, connect MUXOUT+ to S/H IN+ and MUXOUT− to S/H IN−.

6.0 Operational Information (Continued)



TL/H/11879–27

(a) The LM12{L}438



TL/H/11879–28

(b) The LM12434

FIGURE 7. The LM12{L}438 and LM12434 Functional Block Diagram (Programming Model)

6.0 Operational Information (Continued)

Acquisition Time

The LM12434 and LM12{L}438's internal S/H is designed to operate at its minimum acquisition time (1.125 [1.5] μs for a 12-bit + sign conversion) when the source impedance, R_S, is less than or equal to 60 {80} Ω ($f_{CLK} \leq 8$ {6} MHz). When 60 {80} $\Omega < R_S \leq 4.17$ {5.56} kΩ, the internal S/H's acquisition time can be increased to a maximum of 4.88 {6.5} μs (12 + sign bits, $f_{CLK} = 8$ {6} MHz) to provide sufficient time for the sampling capacitor to charge. See Section 6.2.1 (Instruction RAM "00") Bits 12–15 for more information.

Instruction Register

The INSTRUCTION RAM is divided into 8 separate words, each with 48 (3 x 16) bit length. Each word is separated into three 16-bit sections. Each word has a unique address and different sections of the instruction word are selected by the 2-bit RAM pointer (RP) in the configuration register. As shown in *Figure 7*, the Instruction RAM sections are labeled Instructions, Limits #1 and Limits #2. The Instruction section holds operational (12-bit + sign, 8-bit + sign or watchdog) information such as the input channels to be selected, the mode of operation to be performed for each instruction, and the duration of the acquisition period. The other two sections are used in the watchdog mode and the user-defined limits are stored in them. Each watchdog instruction has 2 limits associated with it (usually a low limit and a high limit, but two low limits or two high limits may be programmed instead). The DAS starts executing from instruction 0 and moves through the next instructions up to any user-specified instruction and then "loop back" to instruction 0. It is not necessary to execute all 8 instructions in the instruction loop. The cycle may be repeatedly executed until stopped by the user. The processor should access the Instruction RAM only when the instruction sequencer is stopped.

FIFO Register

The FIFO Register stores the conversion results. This register is "Read only" and all the locations are accessed through a single address. Each time a conversion is performed the result is stored in the FIFO and the FIFO's internal write pointer points to the next location. The pointer rolls back to location 1 after a Write to location 32. The same flow occurs when reading from the FIFO. The internal FIFO Writes and the external FIFO Reads do not affect each other's pointer locations.

Microprocessor overhead is reduced through the use of the internal conversion FIFO. Thirty-two consecutive conversions can be completed and stored in the FIFO without any microprocessor intervention. The microprocessor can, at any time, interrogate the FIFO and retrieve its contents. It can also wait for the LM12434 and LM12{L}438 to issue an interrupt when the FIFO is full or after any number ($\leq$ 32) of conversions have been stored.

Configuration Register

The CONFIGURATION Register is the main "control panel" of the DAS. Writing 1s and 0s to the different bits of the Configuration Register commands the DAS start or stop the sequencer, reset the pointers and flags, go into "standby" mode for low power consumption, calibrate offset and linearity, and select sections of the RAM.

Other Registers

The INTERRUPT ENABLE Register lets the user activate up to 7 sources for interrupt generation (refer to Section 6.2.3). It also holds two user-programmable values. One is the number of conversions to be stored in the FIFO register before the generation of the Data Ready interrupt. The other value is the instruction number that generates an interrupt when the sequencer reaches that instruction.

The INTERRUPT STATUS and LIMIT STATUS Registers are "Read only" registers. They are used as vectors to indicate which conditions have generated the interrupt and what watchdog limit boundaries have been passed. Note that the bits are set in the status registers upon occurrence of their corresponding interrupt conditions, regardless of whether the condition is enabled for external interrupt generation.

The TIMER Register can be programmed to insert a delay before execution of each instruction. A bit in the instruction register enables or disables the insertion of the delay before the execution of an instruction.

Serial I/O

A very flexible serial synchronous interface is provided to facilitate reading from and writing to the LM12434 and LM12{L}438's registers. The communication between the LM12434 and LM12{L}438 and microcontrollers, microprocessors and other circuitry is accomplished through this serial interface. The serial interface is designed to directly communicate with the synchronous serial interfaces of the most popular microprocessors with no extra hardware requirement. The interface has been also designed to simplify software development.

6.0 Operational Information (Continued)

Instruction RAM RP = 10 Limits #2	(Read/Write) RP = 01 Limits #1	RP = 00 Instructions
ADD = 0000	ADD = 0000	ADD = 0000
ADD = 0001	ADD = 0001	ADD = 0001
ADD = 0010	ADD = 0010	ADD = 0010
ADD = 0011	ADD = 0011	ADD = 0011
ADD = 0100	ADD = 0100	ADD = 0100
ADD = 0101	ADD = 0101	ADD = 0101
ADD = 0110	ADD = 0110	ADD = 0110
ADD = 0111	ADD = 0111	ADD = 0111

RP = RAM Pointer
ADD = A3, A2, A1, A0

	(Read/Write)
CONFIGURATION REGISTER	ADD = 1000
	(Read/Write)
INTERRUPT ENABLE REGISTER	ADD = 1001
	(Read Only)
INTERRUPT STATUS REGISTER	ADD = 1010
	(Read/Write)
TIMER REGISTER	ADD = 1011
	(Read Only)
CONVERSION FIFO (32 Locations, 1 address)	ADD = 1100
	(Read Only)
LIMIT STATUS REGISTER	ADD = 1101

FIGURE 8. LM12434 and LM12{L}438 User Accessible Registers

6.0 Operational Information (Continued)

6.2 INTERNAL USER-ACCESSIBLE REGISTERS

Figure 8 shows the LM12434 and LM12{L}438 internal user accessible registers. *Figure 9* shows the bit assignment for each register. All the registers are accessible through the serial interface bus. Following are the descriptions of the registers and their bit assignments.

6.2.1 Instruction RAM

The instruction RAM holds up to eight sequentially executable instructions. Each 48-bit long instruction is divided into three 16-bit sections. READ and WRITE operations can be issued to each 16-bit section using the instruction's address and the 2-bit ''RAM pointer'' in the Configuration register. The eight instructions are located at addresses 0000 through 0111. They can be accessed and programmed in random order.

Read/Write Operations

Any Instruction RAM READ or WRITE can affect the sequencer's operation.

Therefore, the Sequencer should be stopped by setting the RESET bit to a ''1'' or by resetting the START bit in the Configuration Register and waiting for the current instruction to finish execution before any Instruction RAM READ or WRITE is initiated.

A soft RESET should be issued by writing a ''1'' to the Configuration Register's RESET bit after any READ or WRITE to the Instruction RAM.

The three sections in the Instruction RAM are selected by the Configuration Register's 2-bit ''RAM Pointer'', bits D8 and D9. The first 16-bit Instruction RAM section is selected with the RAM Pointer equal to ''00''. This section can be programmed for multiplexer channel selection, conversion resolution, watchdog mode operation, timer or external SYNC use, pause in instruction and loop bit as described later. The second 16-bit section holds ''watchdog'' limit #1, its sign, and a bit that determines whether an interrupt can be generated when the input is greater than or less than limit #1. The third 16-bit section holds ''watchdog'' limit #2, its sign, and the ''greater than/less than'' selection bit.

Instruction RAM, Bank 1, RP = 00

Bit 0 is the LOOP bit. After an instruction with Bit 0 set to a ''1'' is executed, the sequencer will loop back to instruction 0. The next instruction to be executed will be instruction 0.

Bit 1 is the PAUSE bit. When the PAUSE bit is set (''1''), the Sequencer will stop after reading the current instruction. The instruction will not execute at this point, and the START bit in the Configuration register will reset to ''0''. Setting the PAUSE also causes an interrupt to be issued. The Sequencer is restarted by placing a ''1'' in the Configuration register's Bit 0 (Start bit).

After the Instruction RAM has been programmed and the RESET bit is set to ''1'', the Sequencer retrieves Instruction 0, decodes it, and waits for a ''1'' to be placed in the Configuration register's START bit. The START bit value of ''1'' ''overrides'' the action of Instruction 0's PAUSE bit when the Sequencer is started. Once started, the Sequencer executes Instruction 0 and retrieves, decodes, and executes each of the remaining instructions. With the PAUSE bit set to ''1'' in instruction 0, no PAUSE Interrupt (INT 5) is generated the first time the Sequencer executes Instruction 0. When the Sequencer encounters a LOOP bit or completes all eight instructions, Instruction 0 is retrieved and decoded. A set PAUSE bit in Instruction 0 now halts the Sequencer before the instruction is executed. If Pause = 0, the instruction loop continues to execute.

Bits 2–4 select which of the eight input channels (IN0–IN7) will be the non-inverting inputs to the LM12{L}438's ADC. (See Table III.) They select which of the four input channels (for IN0–IN3) will be the non-inverting inputs to the LM12434's ADC. (See Table IV.)

Bits 5–7 select which of the seven input channels (IN1 to IN7) will be the inverting inputs to the LM12{L}438 ADC. (See Table III.) They select which of the three input channels (IN1–IN4) will be the inverting inputs to the LM12434's ADC. (See Table IV.) Fully differential operation is created by selecting two multiplexer channels, one non-inverting and the other inverting. A code of ''000'' selects ground as the inverting input for single ended operation.

Bit 8 is the SYNC bit. Setting Bit 8 to ''1'' causes the Sequencer to hold operation at the internal S/H's acquisition cycle and to wait until a rising edge appears at the SYNC pin. When a rising edge appears, the S/H goes into the ''Hold'' mode and the ADC begins to perform a conversion on the next rising edge of CLK. To make the SYNC pin serve as an input, the Configuration register's ''SYNC I/O'' bit (Bit 7) must be set to a ''0''. With SYNC configured as an input, it is possible to synchronize the start of a conversion to external events. When SYNC pin is defined as an output (SYNC I/O bit = 1) the SYNC bit in the instruction registers must not be set to 1.

When the LM12434 and LM12{L}438 are used in the ''watchdog'' mode with external synchronization, two rising edges on the SYNC input are required to initiate the two comparisons that are performed during a watchdog instruction. The first rising edge initiates the comparison of the selected analog input signal with Limit #1 (found in Instruction RAM ''01'') and the second rising edge initiates the comparison of the same analog input signal with Limit #2 (found in Instruction RAM ''10'').

Bit 9 is the TIMER bit. When Bit 9 is set to ''1'', the Sequencer will halt until the internal 16-bit Timer counts down to zero. During this time interval, no ''watchdog'' comparisons or analog-to-digital conversions will be performed.

Bit 10 selects the ADC conversion resolution. Setting Bit 10 to ''1'' selects 8-bit + sign and resetting to ''0'' selects 12-bit + sign.

Bit 11 is the ''watchdog'' comparison mode enable bit. When operating in the ''watchdog'' comparison mode, the selected analog input signal is compared with the programmable values stored in Limit #1 and Limit #2 (see Instruction RAM ''01'' and Instruction RAM ''10''). Setting Bit 11 to ''1'' causes two comparisons of the selected analog input signal, one with each of the two stored limits. When Bit 11 is reset to ''0'', an 8-bit + sign or 12-bit + sign (depending on the state of Bit 10 of Instruction RAM ''00'') conversion of the input signal can take place.

6.0 Operational Information (Continued)

A4	A3	A2	A1	Purpose	Type	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
0	0	0	0	Instruction RAM (RAM Pointer = 00)	R/W	Acquisition Time				Watch-dog	8/12	Timer	Sync	S/H IN− (MUXIN−)*			S/H IN+ (MUXIN+)*			Pause	Loop
to																					
0	1	1	1																		
0	0	0	0	Instruction RAM (RAM Pointer = 01)	R/W	Don't Care						>/<	Sign	Limit #1							
to																					
0	1	1	1																		
0	0	0	0	Instruction RAM (RAM Pointer = 10)	R/W	Don't Care						>/<	Sign	Limit #2							
to																					
0	1	1	1																		
1	0	0	0	Configuration Register	R/W	Don't Care				DIAG†	Test = 0	RAM Pointer		SYNC I/O	A/Z Each Cycle	I/S	Stand-by	Full CAL	Auto-Zero	Reset	Start
1	0	0	1	Interrupt Enable Register	R/W	Number of Conversion Results in FIFO to Generate Interrupt (INT2)					Instruction Number to Generate Interrupt (INT1)			INT7	X	INT5	INT4	INT3	INT2	INT1	INT0
1	0	1	0	Interrupt Status Register	R	Number of Unread Conversion Results in FIFO					Instruction Number being Executed			INST7	X	INST5	INST4	INST3	INST2	INST1	INST0
1	0	1	1	Timer Register	R/W	Timer Preset High Byte								Timer Preset Low Byte							
1	1	0	0	Conversion FIFO	R	Instruction Number or Extended Sign			Sign	Conversion Data: MSBs				Conversion Data: LSBs							
1	1	0	1	Limit Status Register	R	Limit #2: Status								Limit #1: Status							

*LM12434 (Refer to Table IV).
†LM12{L}438 only. Must be set to "0" for the LM12434.
X No interrupt is associated with this bit. When programming the interrupt Enable Register, bit-6 is a don't care condition.

FIGURE 9. Bit Assignments for LM12434 and LM12{L}438 Internal Registers

6.0 Operational Information (Continued)

CONFIGURATION REGISTER (Read/Write):

D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
Don't Care				Diag.	Test	RAM Pointer		Sync I/O	A/Z Each Cycle	I/S	Stand-by	Full Cal	Auto Zero	Reset	Start

D0: Start: 0 stops the instruction execution. 1 starts the instruction execution.

D1: Reset: When set to 1, resets Start bit; also resets all the bits in status registers and resets the instruction pointer to zero. D1 will then automatically reset itself to zero after 2 clock pulses.

D2: Auto-Zero: When set to 1 a long (8-cycle) auto-zero calibration cycle is performed.

D3: Full Calibration: When set to 1 a full calibration cycle (linearity and auto-zero) is performed.

D4: Standby: When set to 1 the chip goes to low-power standby mode. Resetting the bit will return the chip to active mode after a short delay.

D5: I/S: Instruction # or extended sign. 0 = Bits 13–15 of the conversion result hold the instruction number to which the result belongs; 1 = Bits 13–15 of the result hold the extended sign bit.

D6: A/Z each Cycle: When set to 1 a short auto-zero cycle is performed before each conversion.

D7: Sync I/O: 0 = Sync pin is input: 1 = Sync pin is output.

D9–D8: RAM Pointer: Selects the sections of the instruction RAM, 00 = Instruction, 01 = Limits #1, 10 = Limits #2.

D10: This bit is used for production testing and must be kept zero for normal operation.

D11: Diagnostic: When set to 1, the LM12{L}438 will perform a diagnostic conversion along with a properly selected instruction. This mode is not available on the LM12434.

D15–D12: Don't Care.

INSTRUCTION RAM (Read/Write):

Instruction:

D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
Acquisition Time				Watchdog	8/12	Timer	Sync	MUXIN−			MUXIN+			Pause	Loop

D0: Loop: 0 = Go to next instruction; 1 = Loop back to in instruction #0.

D1: Pause: 0 = No pause; 1 = Pause; don't do the instruction. The start bit in the Configuration register resets to 0 when a pause encountered; a 1 written to the Start bit restarts the instruction execution.

D4–D2: MUXIN+: For the LM12{L}438, these bits select which input channel is connected to the ADC's non-inverting input. For the LM12434, they select which input channel is connected to MUXOUT+.

D7–D5: MUXIN−: For the LM12{L}438, these bits select which input channel is connected to the ADC's inverting input. For the LM12434, they select which input channel is connected to MUXOUT−.

D8: Sync: 0 = Normal operation, internal timing, SYNC is an output. 1 = SYNC is an input; S/H and conversion (comparison) timing are controlled by an external signal applied to SYNC pin.

D9: Timer: 0 = Timer is not used for this instruction; 1 = Instruction execution does not begin until timer counts down to zero.

D10: 8/12: 0 = 12-bit + sign resolution. 1 = 8-bit + sign resolution.

D11: Watchdog: 0 = Conventional conversion (no watchdog comparison); 1 = Instruction performs watchdog comparisons.

D15–D12: Acquisition Time: Determines S/H acquisition time

For 12-bit + sign: (9+2D) clock cycles. For 8-bit + sign: (2 + 2D) clock cycles.
Where D = Contents of D15–D12.
For 12-bit + sign: Choose D for D $\geq$ 0.45 x $R_S[k\Omega]$ $\times$ $f_{CLK}[MHz]$.
For 8-bit + sign: Choose D for D $\geq$ 0.36 x $R_S[k\Omega]$ $\times$ $f_{CLK}[MHz]$.
Where R_S = Input source resistance.

FIGURE 9. Bit Assignments for LM12434 and LM12{L}438 Internal Registers (Continued)

6.0 Operational Information (Continued)

INSTRUCTION RAM (Read/Write): (Continued)

Limits #1 & 2

D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
Don't Care						> / <	Sign	Limit							

D7–D0: Limit: 8-bit limit value.

D8: Sign: Sign of limit value, 0 = Positive; 1 = Negative.

D9: > / <: High Limit/Low limit. 0 = Inputs lower than limit generate interrupt, 1 = Inputs higher than limit generate interrupt.

D15–D10: Don't Care.

INTERRUPT ENABLE REGISTER (Read/Write):

D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
Number of Conversion Results in FIFO to Generate Interrupt (INT2)					Instruction Number to Generate Interrupt (INT1)			INT7	X	INT5	INT4	INT3	INT2	INT1	INT0

Bits # 0 to 7 enable interrupt generation for the following conditions when the bit is set to 1.

D0: INT0: Generates an interrupt when a limit is passed in watchdog mode.

D1: INT1: Generates an interrupt when the sequencer has loaded the instruction number contained in bits D10, D9, and D8 of the Interrupt Enable register.

D2: INT2: Generates an interrupt when the number of conversion results in the FIFO is equal to the programmed value (D15–D11).

D3: INT3: Generates an interrupt when an auto-zero cycle is completed.

D4: INT4: Generates an interrupt when a full calibration cycle is completed.

D5: INT5: Generates an interrupt when a pause condition is encountered.

D6: This bit is a don't care condition. No interrupt is associated with this bit.

D7: INT7: Generates an interrupt when the chip is returned from standby and is ready for operation.

D10–D8: Programmable instruction number used to generate an interrupt when that instruction has been reached.

D15–D11: Programmable number of conversion results in the FIFO to generate an interrupt.

TIMER REGISTER (Read/Write):

D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
N = Timer Preset Value															

The Timer delays the execution of an instruction if the Timer bit is set in that instruction.

The time delay is:

Delay = (32 × N) + 2 [Clock Cycles]

FIGURE 9. Bit Assignments for LM12434 and LM12{L}438 Internal Registers (Continued)

6.0 Operational Information (Continued)

FIFO REGISTER (Read only):

D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
Instruction Number or Extended Sign			Sign	Conversion Result											

D11–D0: Conversion Result:
For 12-bit + sign: 12-bit result value
For 8-bit + sign: D11–D4 = result value, D3–D0 = 1110

D12: Sign: Conversion result sign bit, 0 = Positive, 1 = Negative

D15–D13: Instruction number associated with the conversion result or the extended sign bit for 2's complement arithmetic, selected by bit D5 (Channel Mask) of the Configuration register.

INTERRUPT STATUS REGISTER (Read only):

D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
Number of Unread Results in FIFO					Instruction Number Being Executed			INST7	X	INST5	INST4	INST3	INST2	INST1	INST0

Bits #0 to 7 are interrupt flags (vectors) that will be set to 1 when the following conditions occur. The bits are set to 1 whether the interrupt is enabled or disabled in the Interrupt Enable register. The bits are reset to 0 when the register is read, or by a device reset through the Configuration register.

D0: INST0: Is set to 1 when a limit is passed in watchdog mode.

D1: INST1: Is set to 1 when the sequencer has loaded the instruction number contained in bits D10, D9, and D8 of the Interrupt Enable register.

D2: INST2: Is set to 1 when number of conversion results in FIFO is equal to the programmed value (D15–D11) in the Interrupt Enable Register.

D3: INST3: Is set to 1 when an auto-zero cycle is completed.

D4: INST4: Is set to 1 when a full calibraton cycle is completed.

D5: INST5: Is set to 1 when a pause condition is encountered.

D6: Don't care.

D7: INST7: Is set to 1 when the chip is returned from standby and is ready.

D10–D8: Holds the instruction number presently being executed or will be executed following a Pause or Timer delay.

D15–D11: Holds the number of conversion results that have been put in the FIFO but that have not yet been read by the user.

LIMIT STATUS REGISTER (Read only):

D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
Limit #2: Status								Limit #1: Status							

The bits in this register are limit flags (vectors) that will be set to 1 when a limit is passed. The bits are associated to individual instruction limits as indicated below.

D0: Limit #1 of Instruction #0 is passed.
D1: Limit #1 of Instruction #1 is passed.
D2: Limit #1 of Instruction #2 is passed.
D3: Limit #1 of Instruction #3 is passed.
D4: Limit #1 of Instruction #4 is passed.
D5: Limit #1 of Instruction #5 is passed.
D6: Limit #1 of Instruction #6 is passed.
D7: Limit #1 of Instruction #7 is passed.
D8: Limit #2 of Instruction #0 is passed.
D9: Limit #2 of Instruction #1 is passed.
D10: Limit #2 of Instruction #2 is passed.
D11: Limit #2 of Instruction #3 is passed.
D12: Limit #2 of Instruction #4 is passed.
D13: Limit #2 of Instruction #5 is passed.
D14: Limit #2 of Instruction #6 is passed.
D15: Limit #2 of Instruction #7 is passed.

FIGURE 9. Bit Assignments for LM12434 and LM12{L}438 Internal Registers (Continued)

6.0 Operational Information (Continued)

Bits 12–15 store the user-programmable acquisition time. The Sequencer keeps the internal S/H in the acquisition mode for a fixed number of clock cycles (nine clock cycles, for 12-bit + sign conversions and two clock cycles for 8-bit + sign conversions or ''watchdog'' comparisons) plus a variable number of clock cycles equal to twice the value stored in Bits 12–15. Thus, the S/H's acquisition time is (9 + 2D) clock cycles for 12-bit + sign conversions and (2 + 2D) clock cycles for 8-bit + sign conversions or ''watchdog'' comparisons, where D is the value stored in Bits 12–15. The minimum acquisition time compensates for the typical internal multiplexer series resistance of 2 kΩ, and any additional delay created by Bits 12–15 compensates for source resistances greater than 60Ω {80Ω}. The necessary acquisition time is determined by the source impedance at the multiplexer input. If the source resistance $R_S < 60\Omega$ and the clock frequency is 8 MHz, the value stored in bits 12–15 (D) can be 0000. If $R_S > 60\Omega$, the following equations determine the value that should be stored in bits 12–15.

$$D = 0.45 \times R_S \times f_{CLK}$$

for 12-bits + sign

$$D = 0.36 \times R_S \times f_{CLK}$$

for 8-bits + sign and ''watchdog''

R_S is in kΩ and f_{CLK} is in MHz. Round the result to the next higher integer value. If the value of 0 obtained from the expressions above is greater than 15, it is advisable to lower the source impedance by using an analog buffer between the signal source and the LM12{L}438's multiplexer inputs. The value of D can also be used to compensate for the settling or response time of external processing circuits connected between the LM12434's MUXOUT and S/H IN pins.

Instruction RAM, Bank 2 RP = 01

The second Instruction RAM section is selected by placing ''01'' in Bits 8 and 9 of the Configuration register.

Bits 0–7 hold ''watchdog'' **limit #1**. When Bit 11 of Instruction RAM ''00'' is set to a ''1'', the LM12434 and LM12{L}438 performs a ''watchdog'' comparison of the sampled analog input signal with the limit #1 value first, followed by a comparison of the same sampled analog input signal with the value found in limit #2 (Instruction RAM ''10'').

Bit 8 holds limit #1's sign.

Bit 9's state determines the limit condition that generates a ''watchdog'' interrupt. A ''1'' causes a voltage greater than limit #1 to generate an interrupt, while a ''0'' causes a voltage less than limit #1 to generate an interrupt.

Bits 10–15 are not used.

Instruction RAM, Bank 3, RP = 10

The third Instruction RAM section is selected by placing ''10'' in Bits 8 and 9 of the Configuration register.

Bits 0–7 hold ''watchdog'' **limit #2**. When Bit 11 of Instruction RAM ''00'' is set to a ''1'', the LM12434 and LM12{L}438 performs a ''watchdog'' comparison of the sampled analog input signal with the limit #1 value first (Instruction RAM ''01''), followed by a comparison of the same sampled analog input signal with the value found in limit #2.

Bit 8 holds limit #2's sign.

Bit 9's state determines the limit condition that generates a ''watchdog'' interrupt. A ''1'' causes a voltage greater than limit #2 to generate an interrupt, while a ''0'' causes a voltage less than limit #2 to generate an interrupt.

Bits 10–15 are not used.

TABLE III. LM12{L}438 Operating Mode Input Channel Selection through Input Multiplexer

Normal Operating Mode			
Non-Inverting Input Channel Selection Bits in Instruction Register D4, D3, D2	**Input Channel to Be Connected to A/D Non-Inverting Input (IN+)**	**Inverting Input Channel Selection Bits in Instruction Register D7, D6, D5**	**Input Channel to Be Connected to A/D Inverting Input (IN−)**
000	IN0	000	GND
001	IN1	001	IN1
010	IN2	010	IN2
011	IN3	011	IN3
100	IN4	100	IN4
101	IN5	101	IN5
110	IN6	110	IN6
111	IN7	111	IN7

6.0 Operational Information (Continued)

TABLE IV. LM12434 Input Channel Selection through Input Multiplexer

Normal Operating Mode			
Non-Inverting Input Channel Selection Bits in Instruction Register D4, D3, D2	**Input Channel to Be Connected to MUX Non-Inverting Output (MUXOUT+)**	**Inverting Input Channel Selection Bits in Instruction Register D7, D6, D5**	**Input Channel to Be Connected to MUX Inverting Output (MUXOUT−)**
000	IN0	000	GND
001	IN1	001	IN1
010	IN2	010	IN2
011	IN3	011	IN3
1XX	None	1XX	None

TABLE V. LM12{L}438 Diagnostic Mode Input Channel Selection through Input Multiplexer

Diagnostic Mode			
Non-Inverting Input Channel Selection Bits in Instruction Register D4, D3, D2	**Input Channel to Be Connected to A/D Non-Inverting Input (IN+)**	**Inverting Input Channel Selection Bits in Instruction Register D7, D6, D5**	**Input Channel to Be Connected to A/D Inverting Input (IN−)**
000	None	000	None
001	$V_{REF}+$	001	$V_{REF}-$
010	IN2	010	IN2
011	IN3	011	IN3
100	IN4	100	IN4
101	IN5	101	IN5
110	IN6	110	IN6
111	IN7	111	IN7

6.0 Operational Information (Continued)

6.2.2 Configuration Register

The Configuration register is a 16-bit control register with read/write capability. It acts as the LM12434's and LM12{L}438's ''control panel'' holding global information as well as start/stop, reset, self-calibration, and stand-by commands.

Bit 0 is the START/STOP bit. Reading Bit 0 returns an indication of the Sequencer's status. A ''0'' indicates that the Sequencer is stopped and waiting to execute the next instruction. A ''1'' shows that the Sequencer is running. Writing a ''0'' halts the Sequencer when the current instruction has finished execution. The next instruction to be executed is pointed to by the instruction pointer found in the status register. Writing a ''1'' to Bit 0 restarts the Sequencer with the instruction currently pointed to by the instruction pointer. (See Bits 8–10 in the Interrupt Status register.)

Bit 1 is the DAS' system RESET bit. Writing a ''1'' to Bit 1 stops the Sequencer (resetting the Configuration register's START/STOP bit), resets the Instruction pointer to ''000'' (found in the Interrupt Status register), clears the Conversion FIFO, and resets all interrupt flags. The RESET bit will return to ''0'' after two clock cycles unless it is forced high by writing a ''1'' into the Configuration register's Standby bit. A reset signal is internally generated when power is first applied to the part. No operation should be started until the RESET bit is ''0''.

Bit 2 is the auto-zero bit. Writing a ''1'' to this bit initiates an auto-zero offset voltage calibration. Unlike the eight-sample auto-zero calibration performed during the full calibration procedure, Bit 2 initiates a ''short'' auto-zero by sampling the offset once and creating a correction coefficient (full calibration averages eight samples of the converter offset voltage when creating a correction coefficient). If the Sequencer is running when Bit 2 is set to ''1'', an auto-zero starts immediately after the conclusion of the currently running instruction. Bit 2 is reset automatically to a ''0'' and an interrupt flag (Bit 3, in the Interrupt Status register) is set at the end of the auto-zero (76 clock cycles). After completion of an auto-zero calibration, the Sequencer fetches the next instruction as pointed to by the Instruction RAM's pointer and resumes execution. If the Sequencer is stopped, an auto-zero is performed immediately at the time requested.

Bit 3 is the calibration bit. Writing a ''1'' to this bit initiates a complete calibration process that includes a ''long'' auto-zero offset voltage correction (this calibration averages eight samples of the comparator offset voltage when creating a correction coefficient) followed by an ADC linearity calibration. This complete calibration is started after the currently running instruction is completed if the Sequencer is running when Bit 3 is set to ''1''. Bit 3 is reset automatically to a ''0'' and an interrupt flag (Bit 4, in the Interrupt Status register) will be generated at the end of the calibration procedure (4944 clock cycles). After completion of a full auto-zero and linearity calibration, the Sequencer fetches the next instruction as pointed to by the Instruction RAM's pointer and resumes execution. If the Sequencer is stopped, a full calibration is performed immediately at the time requested.

Bit 4 is the Standby bit. Writing a ''1'' to Bit 4 immediately places the DAS in Standby mode. Normal operation returns when Bit 4 is reset to a ''0''. The Standby command (''1'') disconnects the external clock from the internal circuitry, decreases the LM12434 and LM12{L}438's internal analog circuitry power supply current, and preserves all internal RAM contents. After writing a ''0'' to the Standby bit, the DAS returns to an operating state identical to that caused by exercising the RESET bit. A Standby completion interrupt is issued after a power-up delay to allow the analog circuitry to settle. The Sequencer should be restarted only after the Standby completion interrupt is issued (see Note 22). The Instruction RAM can still be accessed through read and write operations while the LM12434 and LM12{L}438 are in Standby Mode.

Bit 5 is the Channel Address Mask. If Bit 5 is set to a ''1'', Bits 13–15 in the conversion FIFO will be equal to the sign bit (Bit 12) of the conversion data. Resetting Bit 5 to a ''0'' causes conversion data Bits 13 through 15 to hold the instruction pointer value of the instruction to which the conversion data belongs.

Bit 6 selects a ''short'' auto-zero correction for every conversion. The Sequencer automatically inserts an auto-zero before every conversion or ''watchdog'' comparison if Bit 6 is set to ''1''. No automatic correction will be performed if Bit 6 is reset to ''0''.

The DAS' offset voltage, after calibration, has a typical drift of 0.1 LSB over a temperature range of −40°C to +85°C. This small drift is less than the variability of the change in offset that can occur when using the auto-zero correction with each conversion. This variability is the result of using only one sample of the offset voltage to create a correction value. This variability decreases when using the full calibration mode because eight samples of the offset voltage are taken, averaged, and used to create a correction value. Therefore, it is recommended that this mode not be used.

Bit 7 programs the SYNC pin (29) to operate as either an input or an output. The SYNC pin becomes an output when Bit 7 is a ''1'' and an input when Bit 7 is a ''0''. With SYNC programmed as an input, the rising edge of any logic signal applied to pin 29 will start a conversion or ''watchdog'' comparison. Programmed as an output, the logic level at pin 29 will go high at the start of a conversion or ''watchdog'' comparison and remain high until either have finished. See Instruction RAM ''00'', Bit 8.

Bits 8 and **9** form the RAM Pointer that is used to select each of a 48-bit instruction's three 16-bit sections during read or write actions. A ''00'' selects Instruction RAM section one, ''01'' selects section two, and ''10'' selects section three.

Bit 10 activates the Test mode that is used only during production testing. Always write ''0'' in this bit when programming the Instruction Register.

Bit 11 is the Diagnostic bit and is available only in the LM12{L}438. It can be activated by setting it to a ''1''. The Diagnostic mode, along with a properly chosen instruction, allows verification that the LM12{L}438's ADC is performing correctly. When activated, the inverting and non-inverting inputs are connected as shown in Table V. As an example, an instruction with ''001'' for both IN+ and IN− while using the Diagnostic mode typically results in a full-scale output.

6.2.3 Interrupts

The LM12434 and LM12{L}438 have seven possible interrupts, all with the same priority. Any of these interrupts will cause a hardware interrupt to appear on the $\overline{INT}$ pin (31) if

6.0 Operational Information (Continued)

they are not masked (by the Interrupt Enable register). The Interrupt Status register is then read to determine which of the seven interrupts has been issued.

The Interrupt Status register must be cleared by reading it after writing to the Interrupt Enable register. This removes any spurious interrupts on the $\overline{INT}$ pin generated during an Interrupt Enable register access.

Interrupt 0 is generated whenever the analog input voltage on a selected multiplexer channel crosses a limit while the LM12434 and LM12{L}438 are operating in the ''watchdog'' comparison mode. Two sequential comparisons are made when the LM12434 and LM12{L}438 are executing a ''watchdog'' instruction. Depending on the logic state of Bit 9 in the Instruction RAM's second and third sections, an interrupt will be generated either when the input signal's magnitude is greater than or less than the programmable limits. (See the Instruction RAM, Bit 9 description.) The Limit Status register will indicate which preprogrammed limit (#1 or #2) was crossed, and which instruction was executing when the limit was crossed.

Interrupt 1 is generated when the Sequencer reaches the instruction counter value specified in the Interrupt Enable register's bits 8–10. This flag appears before the instruction's execution. Instructions continue to execute as programmed.

Interrupt 2 is activated when the Conversion FIFO holds a number of conversions equal to the programmable value stored in the Interrupt Enable register's Bits 11–15. This value ranges from 00000 to 11111, with 00001 to 11111 representing 1 to 31 conversions stored in the FIFO, and 00000 generating an interrupt after 32 conversions. See Section 6.2.8 for more FIFO information.

The completion of the short, single-sampled auto-zero calibration generates **Interrupt 3**.

The completion of a full auto-zero and linearity self-calibration generates **Interrupt 4**.

Interrupt 5 is generated when the Sequencer encounters an instruction that has its Pause bit (Bit 1 in Instruction RAM ''00'') set to ''1''.

Interrupt 7 is issued after a short delay (10 ms typ) while the DAS returns from Standby mode to active operation using the Configuration register's Bit 4. This short delay allows the internal analog circuitry to settle sufficiently, ensuring accurate conversion results (see Note 22).

6.2.4 Interrupt Enable Register

The Interrupt Enable register at address location 1001 has READ/WRITE capability. An individual interrupt's ability to produce an external interrupt at pin 31 ($\overline{INT}$) is accomplished by placing a ''1'' in the appropriate bit location. Any of the internal interrupt-producing operations will set their corresponding bits to ''1'' in the Interrupt Status register regardless of the state of the associated bit in the Interrupt Enable register. See Section 2.3 for more information about each of the eight internal interrupts.

Bit 0 enables an external interrupt when an internal ''watchdog'' comparison limit interrupt has taken place.

Bit 1 enables an external interrupt when the Sequencer has reached the address stored in Bits 8–10 of the Interrupt Enable register.

Bit 2 enables an external interrupt when the Conversion FIFO's limit, stored in Bits 11–15 of the Interrupt Enable register, has been reached.

Bit 3 enables an external interrupt when the single-sampled auto-zero calibration has been completed.

Bit 4 enables an external interrupt when a full auto-zero and linearity self-calibration has been completed.

Bit 5 enables an external interrupt when an internal Pause interrupt has been generated.

Bit 6 don't care condition.

Bit 7 enables an external interrupt when the LM12434 and LM12{L}438 returns from standby to active mode (see Note 22).

Bits 8–10 form the storage location of the user-programmable value against which the Sequencer's address is compared. When the Sequencer reaches an address that is equal to the value stored in Bits 8–10, an internal interrupt is generated and appears in Bit 1 of the Interrupt Status register. If Bit 1 of the Interrupt Enable register is set to ''1'', an external interrupt will appear at pin 31 ($\overline{INT}$).

The value stored in bits 8–10 ranges from 000 to 111, representing 1 to 8 instructions stored in the Instruction RAM. After the Instruction RAM has been programmed and the RESET bit is set to ''1'', the Sequencer is started by placing a ''1'' in the Configuration register's START bit. Setting the INT 1 trigger value to 000 **does not generate** an INT 1 the **first** time the Sequencer retrieves and decodes Instruction 000. The Sequencer **generates** INT 1 (by placing a ''1'' in the Interrupt Status register's Bit 1) the **second time and every subsequent time that** the Sequencer encounters Instruction 000. It is important to remember that the Sequencer continues to operate even if an Instruction interrupt (INT 1) is internally or externally generated. The only mechanisms that stop the Sequencer are an instruction with the PAUSE bit set to ''1'' (halts before instruction execution), placing a ''0'' in the Configuration register's START bit, or placing a ''1'' in the Configuration register's RESET bit.

Bits 11–15 hold the number of conversions that must be stored in the Conversion FIFO in order to generate an internal interrupt. This internal interrupt appears in Bit 2 of the Interrupt Status register. If Bit 2 of the Interrupt Enable register is set to ''1'', an external interrupt will appear at pin 31 (INT).

6.2.5 Interrupt Status Register

This read-only register is located at address 1010. The corresponding flag in the Interrupt Status register goes high (''1'') any time that an interrupt condition takes place, whether an interrupt is enabled or disabled in the Interrupt Enable register. Any of the active (''1'') Interrupt Status register flags are reset to ''0'' whenever this register is read or a device reset is issued (see Bit 1 in the Configuration Register).

Bit 0 is set to ''1'' when a ''watchdog'' comparison limit interrupt has taken place.

Bit 1 is set to ''1'' when the Sequencer has reached the address stored in Bits 8–10 of the Interrupt Enable register.

Bit 2 is set to ''1'' when the Conversion FIFO's limit, stored in Bits 11–15 of the Interrupt Enable register, has been reached.

Bit 3 is set to ''1'' when the single-sampled auto-zero has been completed.

Bit 4 is set to ''1'' when an auto-zero and full linearity self-calibration has been completed.

Bit 5 is set to ''1'' when a Pause interrupt has been generated.

6.0 Operational Information (Continued)

Bit 6 no interrupt is associated with this bit. Don't care condition.

Bit 7 is set to ''1'' when the DAS returns from standby to active mode (see Note 22).

Bits 8–10 hold the Sequencer's current instruction number while it is running.

Bits 11–15 hold the current number of conversion results stored in FIFO but have not been read by the user. After each conversion, the result will be stored in the FIFO and the contents of these bits incremented by one. Each single read from FIFO decrements the contents of these bits by one. If more than 32 conversion results being stored in FIFO the numbers on these bits roll over from ''11111'' to ''00000'' and continue incrementing. If reads are performed from FIFO more than the number of conversions stored in it, the contents of these bits roll back from ''00000'' to ''11111'' and continue decrementing.

6.2.6 Limit Status Register

This read-only register is located at address 1101. This register is used in tandem with the Limit #1 and Limit #2 registers in the Instruction RAM. Whenever a given instruction's input voltage exceeds the limit set in its corresponding Limit register (#1 or #2) a bit corresponding to the instruction number is set in the Limit Status register. Any of the active (''1'') Limit Status flags are reset to ''0'' whenever this register is read or a device reset is issued (see Bit 1 in the Configuration register). This register holds the status of limits #1 and #2 for each of the eight instructions.

Bits 0–7 show the Limit #1 status. Each bit will be set high (''1'') when the corresponding instruction's input voltage exceeds the threshold stored in the instruction's Limit #1 register. When, for example, instruction 3 is a ''watchdog'' operation (Bit 11 is set high) and the input for instruction 3 meets the magnitude and/or polarity data stored in instruction 3's Limit #1 register, Bit 3 in the Limit Status register will be set to a ''1''.

Bits 8–15 show the Limit #2 status. Each bit will be set high (''1'') when the corresponding instruction's input voltage exceeds the threshold stored in the instruction's Limit #2 register. When, for example, the input to instruction 6 meets the value stored in instruction 6's Limit #2 register, Bit 14 in the Limit Status register will be set to a ''1''.

6.2.7 Timer

The LM12434 and LM12{L}438 have an on-board 16-bit timer that includes a 5-bit pre-scaler. It uses the clock signal applied to pin 23 as its input. It can generate time intervals of 0 through 2^{21} clock cycles in steps of 2^5. This time interval can be used to delay the execution of instructions. It can also be used to slow the conversion rate when converting slowly changing signals. This can reduce the amount of redundant data stored in the FIFO and retrieved by the controller.

The user-defined timing value used by the Timer is stored in the 16-bit READ/WRITE Timer register at location 1011 and is pre-loaded automatically. Bits 0–7 hold the preset value's low byte and Bits 8–15 hold the high byte. The Timer is activated by the Sequencer only if the current instruction's Bit 9 is set (''1''). If the equivalent decimal value ''N'' ($0 \leq N \leq 2^{16} - 1$) is written inside the 16-bit Timer register and the Timer is enabled by setting an instruction's bit 9 to a ''1'', the Sequencer will delay that instruction's execution by halting at state 3 (S3), as shown in *Figure 11,* for $32 \times N + 2$ clock cycles.

6.2.8 FIFO

The result of each conversion is stored in an internal read-only FIFO (First-In, First-Out) register. It is located at address 1100. This register has 32 16-bit wide locations. Each location holds 13 bits of conversion data. Bits 0–3 hold the four LSBs in the 12 bits + sign mode or ''1110'' in the 8 bits + sign mode. Bits 4–11 hold the eight MSBs and Bit 12 holds the sign bit. Bits 13–15 can hold either the sign bit, extending the register's two's complement data format to a full sixteen bits or the instruction address that generated the conversion and the resulting data. These modes are selected according to the logic state of the Configuration register's Bit 5.

The FIFO status should be read in the Interrupt Status register (Bits 11–15) to determine the number of conversion results that are held in the FIFO before retrieving them. This will help prevent conversion data corruption that may take place if the number of reads are greater than the number of conversion results contained in the FIFO. Trying to read the FIFO when it is empty may corrupt new data being written into the FIFO. Writing more than 32 conversion results into the FIFO by the ADC results in loss of the first conversion results. Therefore, to prevent data loss, it is recommended that the LM12434 and LM12{L}438's interrupt capability be used to inform the system controller that the FIFO is full.

Bits 0–12 hold 12-bit + sign conversion data. **Bits 0–3** will be 1110 when using 8-bit plus sign resolution.

Bits 13–15 hold either the instruction responsible for the associated conversion data or the sign bit. Either mode is selected with Bit 5 in the Configuration register.

Using the FIFO's full depth is achieved as follows. Set the value of the Interrupt Enable registers's Bits 11–15 to 00000 and the Interrupt Enable register's Bit 2 to a ''1''. This generates an external interrupt when the 31st conversion is stored in the FIFO. This gives the host processor a chance to send a ''0'' to the LM12434 and LM12{L}438's Start bit (Configuration register) and halt the ADC before it completes the 32nd conversion. The Sequencer halts after the current (32) conversion is completed. The conversion data is then transferred to the FIFO and occupies the 32nd location. FIFO overflow is avoided if the Sequencer is halted before the start of the 32nd conversion by placing a ''0'' in the Start bit (Configuration register). It is important to remember that the Sequencer **continues to operate even if a FIFO interrupt (INT 2) is internally or externally generated**. The only mechanisms that stop the Sequencer are an instruction with the PAUSE bit set to ''1'' (halts before instruction execution), placing a ''0'' in the Configuration register's START bit, or placing a ''1'' in the Configuration register's RESET bit.

6.0 Operational Information (Continued)

6.3 INSTRUCTION SEQUENCER

The Sequencer uses a 3-bit counter (Instruction Pointer, or IP) to retrieve the programmable conversion instructions stored in the Instruction RAM. The counter is reset to 000 during chip reset or if the current executed instruction has its Loop bit (Bit 1 in any Instruction RAM ''00'') set high (''1''). It increments at the end of the currently executed instruction and points to the next instruction. It will continue to increment up to 111 unless an instruction's Loop bit is set. If this bit is set, the counter resets to ''000'' and execution begins again with the first instruction. If all instructions have their Loop bit reset to ''0'', the Sequencer will execute all eight instructions continuously. Therefore, it is important to realize that if less than eight instructions are programmed, the Loop bit on the last instruction must be set. Leaving this bit reset to ''0'' allows the Sequencer to execute ''unprogrammed'' instructions, the results of which may be unpredictable.

The Sequencer's Instruction Pointer value is readable at any time and is found in the Status register at Bits 8–10. *Figure 10* illustrates the instruction execution flow as performed by the sequencer. The Sequencer can go through eight states during instruction execution:

State 0: The current instruction's first 16 bits are read from the Instruction RAM ''00''. This state is one clock cycle long.

State 1: Checks the state of the Calibration and Start bits. This is the ''rest'' state whenever the Sequencer is stopped using the reset, a Pause command, or the Start bit is reset low (''0''). When the Start bit is set to a ''1'', this state is one clock cycle long.

State 2: Perform calibration. If bit 2 or bit 6 of the Configuration register is set to a ''1'', state 2 is 76 clock cycles long. If the Configuration register's bit 3 is set to a ''1'', state 2 is 4944 clock cycles long.

State 3: Run the internal 16-bit Timer. The number of clock cycles for this state varies according to the value stored in the Timer register. The number of clock cycles is found by using the expression below

$$32T + 2$$

where $0 \leq T \leq 2^{16} - 1$.

State 7: Sample the input signal and read Limit #1's value if needed. The number of clock cycles for acquiring the input signal in the 12-bit + sign mode varies according to

$$9 + 2D$$

where D is the user-programmable 4-bit value stored in bits 12–15 of Instruction RAM ''00'' and is limited to $0 \leq D \leq 15$.

The number of clock cycles for acquiring the input signal in the 8-bit + sign or ''watchdog'' mode varies according to

$$2 + 2D$$

State 6: Perform first watchdog comparison. This state is 5 clock cycles long.

State 4: Read Limit #2. This state is 1 clock cycle long.

State 5: Perform a conversion or second watchdog comparison. This state takes 44 clock cycles for a 12-bit + sign conversions or 21 clock cycles for a 8-bit + sign conversions. The ''watchdog'' comparison mode takes 5 clock cycles.

6.0 Operational Information (Continued)

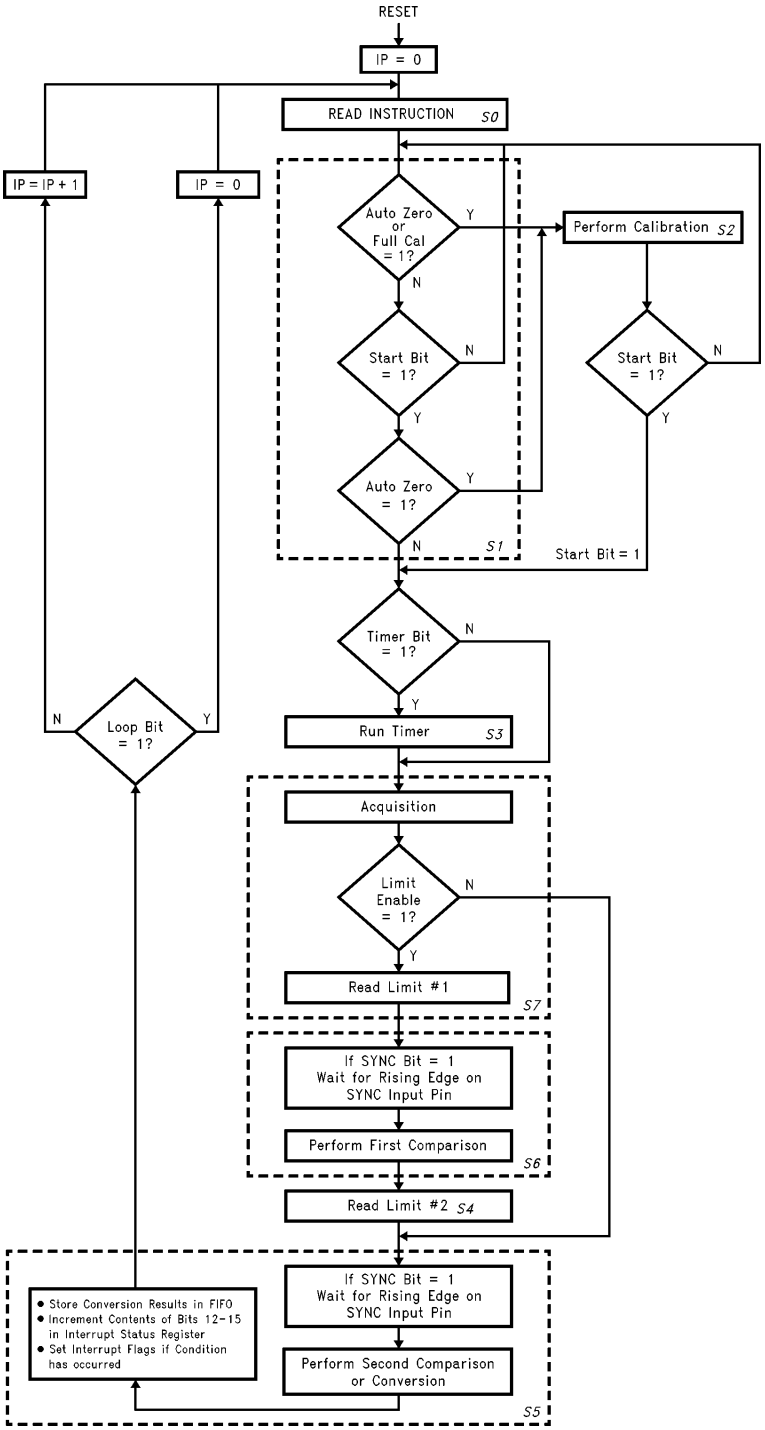


FIGURE 10. Sequencer Logic Flow Chart (IP = Instruction Pointer)

7.0 Digital Interface

In order to read from or write to the registers of the LM12434 and LM12{L}438 a very flexible serial synchronous interface is provided. Communication between the LM12434 and LM12{L}438 and microcontrollers, microprocessors and other circuitry is accomplished through this serial interface. The serial interface is designed to directly communicate with synchronous serial interface of the most popular microprocessors and I2C serial protocol with no additional hardware required. The interface has been also designed to accommodate easy and straightforward software programming.

The LM12434 and LM12{L}438 supports four selectable protocols as shown in Table VI. The MODESEL1 and MODESEL2 inputs select the desired protocol. These pins are normally hardwired for a selected protocol, but they can also be controlled by the system in case a protocol change within the system is required. P1–P5 are multi-function serial interface input or output pins that have different assignments depending on the selected interface mode.

The ''Standard'' interface mode uses a simple shift register type of serial data transfer. It supports several microcontrollers' serial synchronous protocols, including: National Semiconductor's MICROWIRE/PLUS, Motorola's SPI, QSPI, and Hitachi's synchronous SCI. Section 7.1.1 shows general block diagrams of how the serial DAS, configured in the Standard Interface Mode, can be connected to the HPC and 68HC11. Also, detailed assembly routines are included for single writes, single reads and burst read operations.

The ''8051'' mode supports the synchronous serial interface of the 8051 family of microcontrollers (8051 serial interface Mode 0). It is also compatible with the serial interface in the MCS-96 family of 16-bit microcontrollers. Section 7.2.1 shows a general block diagram of how the serial DAS, configured in the 8051 Interface Modes can be connected to the 8051 family of μCs. Also, detailed assembly routines for a single write, single read and burst read operations are included.

The ''TMS320'' mode is designed to directly interact with the serial interface of the TMS320C3x and TMS320C5x families of digital signal processors. This interface is also compatible with the similar serial interfaces on the DSP56000 and the ADSP2100 families of DSP processors. Section 7.3.1 shows a general block diagram of how the serial DAS, configured in the TMS320 interface mode, can be connected to the TMS320C3x family of DSP processors. Also, detailed assembly routines for a single write, single read and burst read operations are included.

The ''I2C'' mode supports the Philips' I2C bus specification for both the standard (100 kHz maximum data rate) and the fast (400 kHz maximum data rate) modes of operation. The DAS behaves as a slave device on the I2C bus and receives and transmits the information under the control of a bus master. Section 7.4.1 shows a general block diagram of how the serial DAS, configured in the I2C Interface mode, can be connected to an I2C bus using an I2C controller (PCD8584).

All the serial interface modes allow for three basic types of data transfer; these are single write, single read and burst read. In a single write or read, 16 bits (2 bytes) of data is written to or read from one of the registers inside the DAS. In a burst read, multiple reads are performed from one register without having to repeatedly send the control and register address information for each read. The burst read can be performed on any LM12434 and LM12{L}438's register, however it is primarily provided for multiple reads from the FIFO register (one address, 32 locations), where a sequence of conversion results is stored.

7.1 STANDARD INTERFACE MODE

The standard interface mode is a simple shift register type of serial data transfer. The serial clock synchronizes the transfer of data to and from the LM12434 and LM12{L}438. The interface uses 4 lines: 2 data lines (DI and DO), a serial clock line (SCLK) and a chip-select (CS) line. More than one device can share the data and serial clock lines provided that each device has its own chip-select line.

The LM12434 and LM12{L}438 standard mode is selected when the MODESEL1 and MODESEL2 pins have the logic state of ''01''. *Figure 12* shows a typical connection diagram for the LM12434 and LM12{L}438 standard mode serial interface. The CS, DI, DO, and SCLK lines are respectively assigned to interface pins P2 through P5. The P1 pin is assigned to a signal called R/F (Rise/Fall). The logic level on this pin specifies the polarity of the serial clock:

— If R/F = 1, data is shifted after falling edge and is stable and captured at the rising edge of the SCLK.

— If R/F = 0, data is shifted after rising edge and is stable and captured at the falling edge of the SCLK.

TABLE VI. LM12434 and LM12{L}438 Interface Modes and Pin Assignments

Interface Mode	M0DESEL1	MODESEL2	P1	P2	P3	P4	P5
Standard	0	1	$\overline{R}$/F	$\overline{CS}$	DI	DO	SCLK
8051	0	0	1*	1*	$\overline{CS}$	RXD	TXD
TMS320	1	1	FSR	FSX	DX	DR	CLK
I2C	1	0	Slave AD0	Slave AD1	Slave AD2	SDA	SCL

*Internally pulled-up

7.0 Digital Interface (Continued)

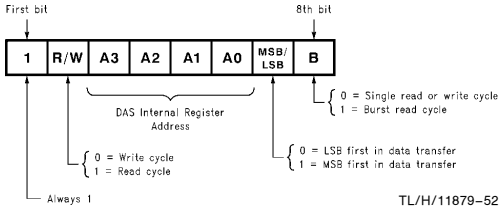
In both cases the data transfer is insensitive to idle state of the SCLK. SCLK can stay at either logic level high or low when not clocking (see *Figure 11*)

Data transfer in this mode is basically byte-oriented. This is compatible with the serial interface of the target microcontrollers and microprocessors. As mentioned, the LM12434 and LM12{L}438 have three different communication cycles: write cycle, read cycle and burst read cycle. At the start of each data transfer cycle, ''command byte'' is written to the serial DAS, followed by write or read data. The command byte informs the LM12434 and LM12{L}438 about the communication cycle. The command byte carries the following information:

— what type of data transfer (communication cycle) is started

— which device register to be accessed

The command byte has the following format:



TL/H/11879–52

Note that the first bit may be either the MSB or the LSB of the byte depending on the processor type, but it must be the first bit transmitted to the LM12434 and LM12{L}438.

Figure 11 shows the timing diagrams for different communication cycles. *Figures 11a, b, c, d* show write cycles for various combinations of R/F pin logic level and SCLK idle state. *Figures 11e, f, g, h* show read cycles for similar sets of conditions. *Figure 11i* shows a burst read cycle for the case of R/F = 0 and low SCLK idle state. Note that these timing diagrams depict general relationships between the SCLK edges, the data bits and $\overline{CS}$. These diagrams are not meant to show guaranteed timing. (See specification tables for parametric switching characteristics.)

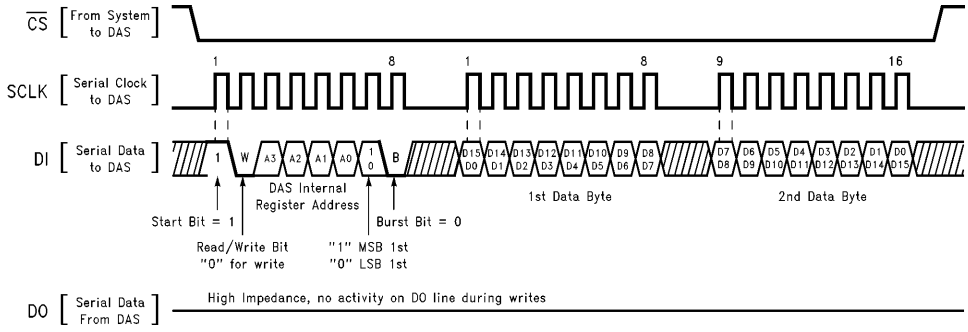
Write cycle: A write cycle begins with the falling edge of $\overline{CS}$. Then a command byte is written to the DAS on the DI line synchronized by SCLK. The command byte has the R/W and B bits equal to zero. Following the command byte, 16 bits of data (2 bytes) is shifted in on the same DI line. This data is written to the register addressed in the command byte (A3, A2, A1, A0). The data is interpreted as MSB or LSB first based on the logic level of the 7th bit (MSB/LSB) in the command byte. There is no activity on the DO line during write cycles and the DAS leaves the DO line in the high impedance state. $\overline{CS}$ will go high after the transfer of the last bit, thus completing the write cycle.

Read cycle: A read cycle starts the same way as a write cycle, except that the command byte's R/W bits equal to one. Following the command byte, the DAS outputs the data on the DO line synchronized with the microcontroller's SCLK. The data is read from the register addressed in the command byte. Data is shifted out MSB or LSB first, depending on the logic level of the MSB/LSB bit. The logic state of the DI line is ''don't care'' after the command byte. $\overline{CS}$ will go high after the transfer of the last data bit, then completing the read cycle.

Burst read cycle: A burst read cycle starts the same way as a single read cycle, but the B bit in the command byte is set to one, indicating a burst read cycle. Following the command byte the data is output on the DO line as long as the DAS receives SCLK from the system. To tell the DAS when a burst read cycle is completed pull $\overline{CS}$ high after the 8th and before the 15th SCLK cycle during the last data byte transfer (see *Figure 11i*). After $\overline{CS}$ high is detected and the last data bit is transferred, the DAS is ready for a new communication cycle to begin.

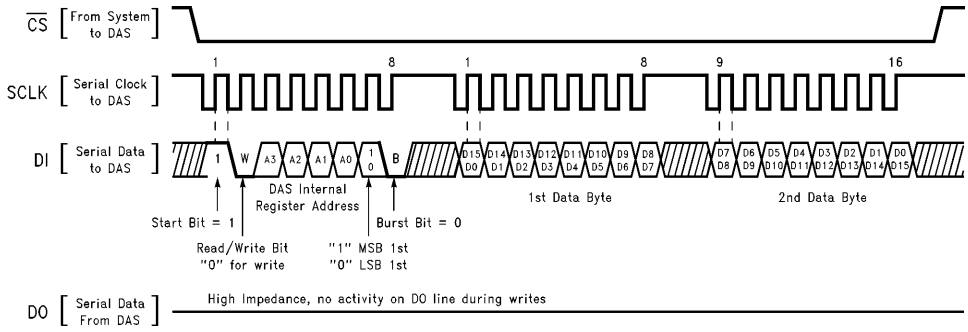
The timing diagrams in *Figure 11* show the transfer of data in packets of 8 bits (bytes). This represents the way the serial ports of most microcontrollers and microprocessors produce serial clock and data. The DAS does not require a gap between the first and second byte of the data; 16 continuous clock cycles will transfer the data word. However, there should be a gap equal to 3 CLK (the DAS main clock input, not the SCLK) cycles between the end of the command byte and the start of the data during a read cycle. This is not a concern in most systems for two reasons. First, the processor generally has some inherent gap between byte transfers. Second, the SCLK frequency is usually significantly slower than the CLK frequency. For example, a 68HC11 processor with an 8 MHz crystal generates a maximum SCLK frequency of 1 MHz. If the DAS is running with a 6 MHz CLK, there are 6 cycles of CLK within each cycle of SCLK and the requirement is satisfied even if SCLK operates continuously during and after the command byte.

7.0 Digital Interface (Continued)



TL/H/11879–30

(a) Write Cycle, R/F Input (P1) = 1
Idle State of SCLK = 0, Data Stable at Rising Edge and Shifted at Falling Edge of the SCLK

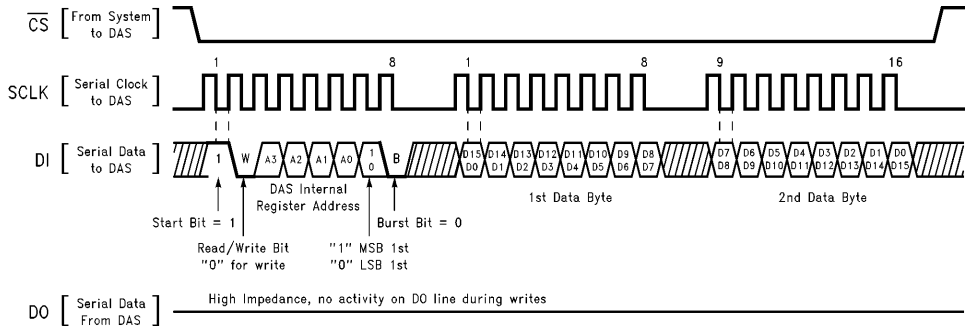


TL/H/11879–31

(b) Write Cycle, R/F Input (P1) = 1
Idle State of SCLK = 1, Data Stable at Rising Edge and Shifted at Falling Edge of the SCLK

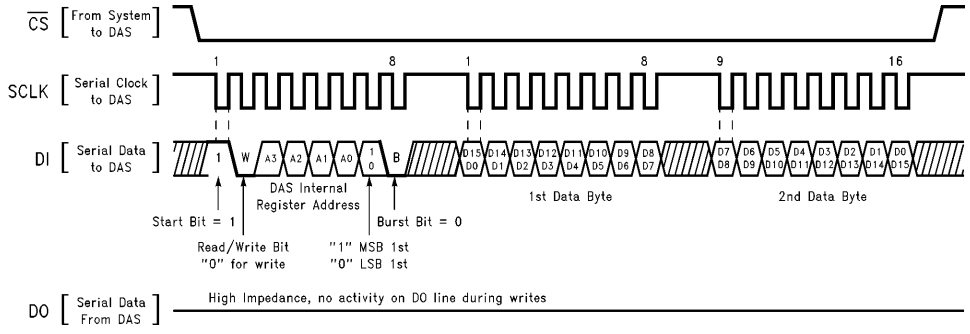
FIGURE 11. Timing Diagrams for LM12434 and LM12{L}438 Standard Serial Interface

7.0 Digital Interface (Continued)



TL/H/11879–32

(c) Write Cycle, R/F Input (P1) = 0
Idle State of SCLK = 0, Data Stable at Falling Edge and Shifted at Rising Edge of the SCLK

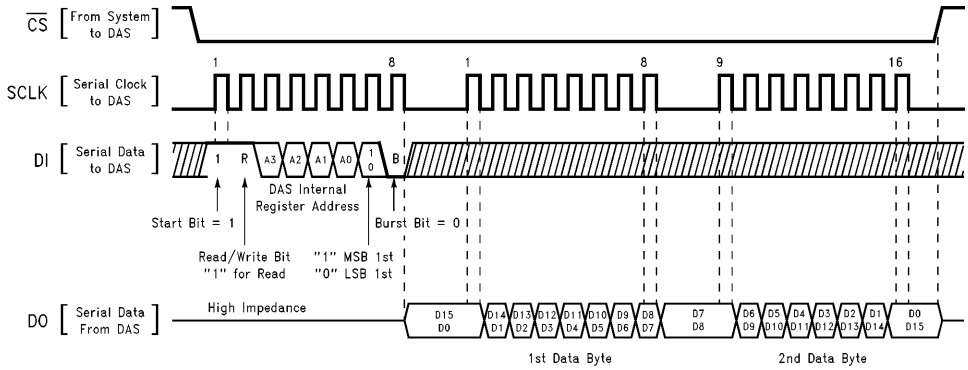


TL/H/11879–33

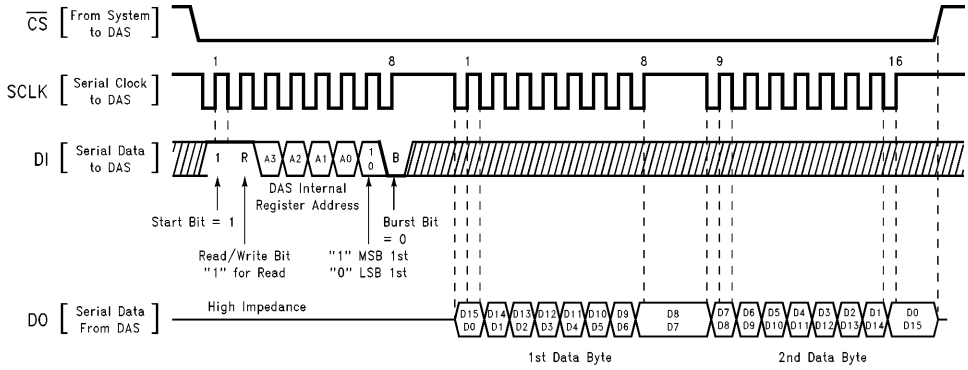
(d) Write Cycle, R/F Input (P1) = 0
Idle State of SCLK = 1, Data Stable at Falling Edge and Shifted at Rising Edge of the SCLK

FIGURE 11. Timing Diagrams for LM12434 and LM12{L}438 Standard Serial Interface (Continued)

7.0 Digital Interface (Continued)



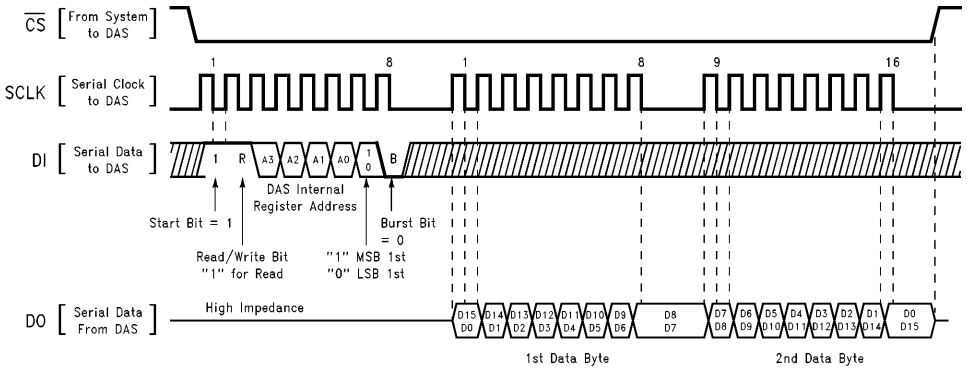
(e) Read Cycle, R/F Input (P1) = 1
Idle State of SCLK = 0, Data Stable at Rising Edge and Shifted at Falling Edge of the SCLK



(f) Read Cycle, R/F Input (P1) = 1
Idle State of SCLK = 1, Data Stable at Rising Edge and Shifted at Falling Edge of the SCLK

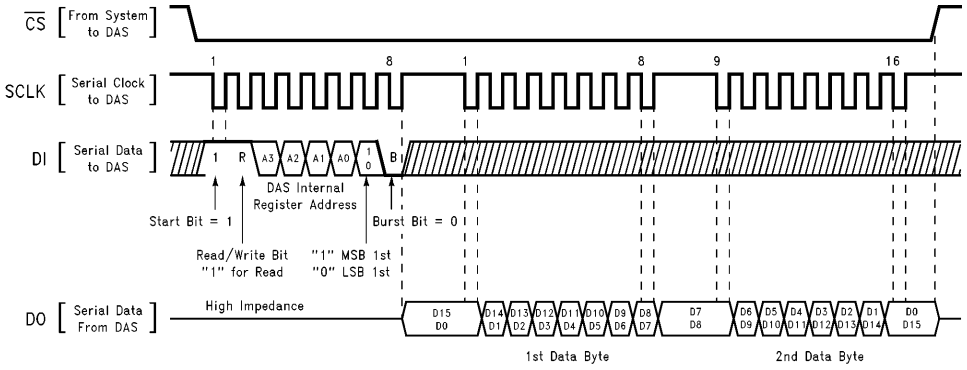
FIGURE 11. Timing Diagrams for LM12434 and LM12{L}438 Standard Serial Interface (Continued)

7.0 Digital Interface (Continued)



TL/H/11879–36

(g) Read Cycle, R/F Input (P1) = 0
Idle State of SCLK = 0, Data Stable at Falling Edge and Shifted at Rising Edge of the SCLK

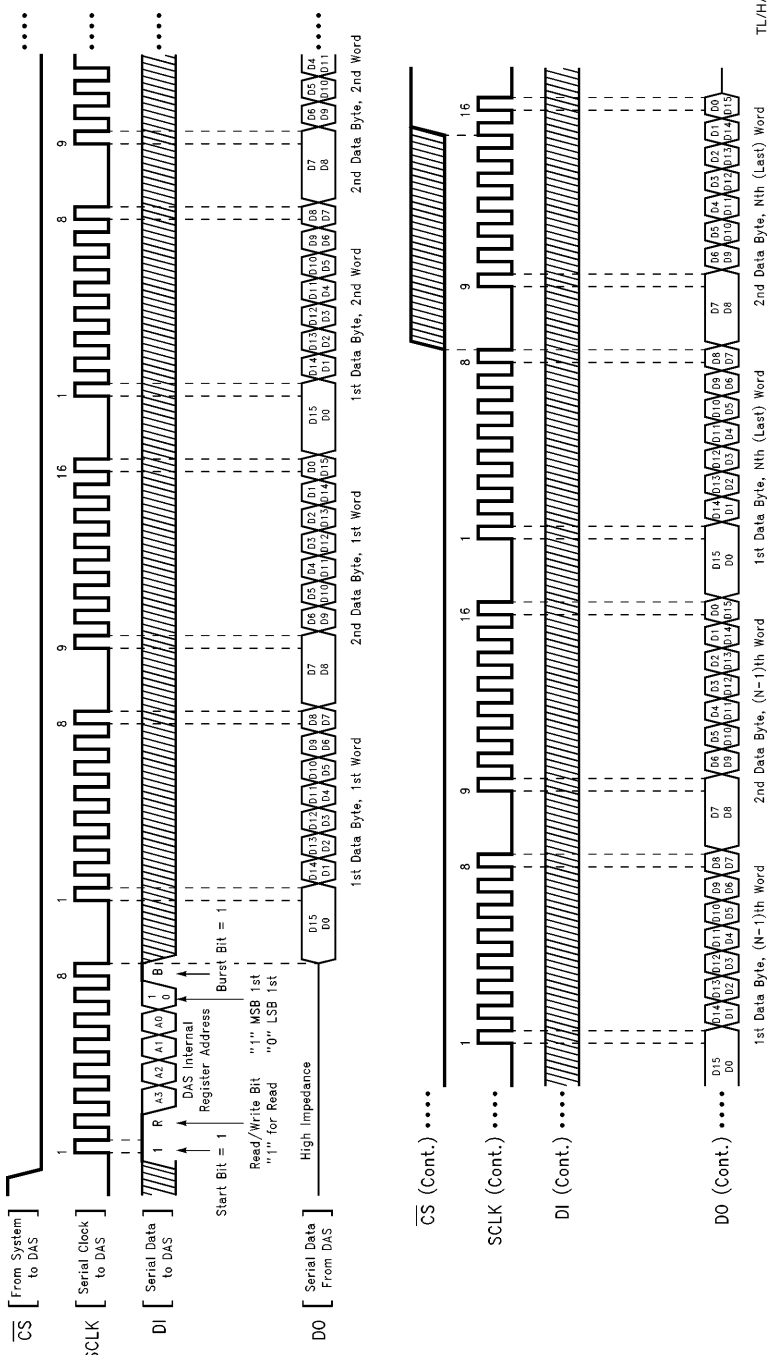


TL/H/11879–37

(h) Read Cycle, R/F Input (P1) = 0
Idle State of SCLK = 1, Data Stable at Falling Edge and Shifted at Rising Edge of the SCLK

FIGURE 11. Timing Diagrams for LM12434 and LM12{L}438 Standard Serial Interface (Continued)

7.0 Digital Interface (Continued)

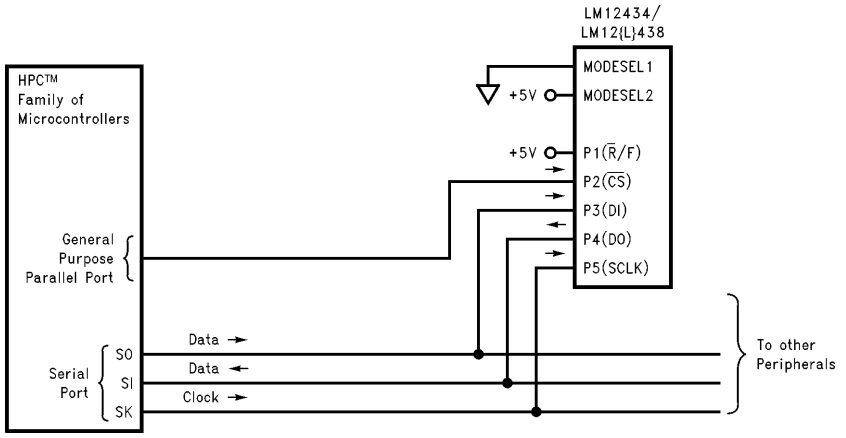


(i) Burst Read Cycle, R/F Input (P1) = 1
Idle State of SCLK = 0, Data Stable at Rising Edge and Shifted at Falling Edge of the SCLK

FIGURE 11. Timing Diagrams for LM12434 and LM12{L}438 Standard Serial Interface (Continued)

7.0 Digital Interface (Continued)

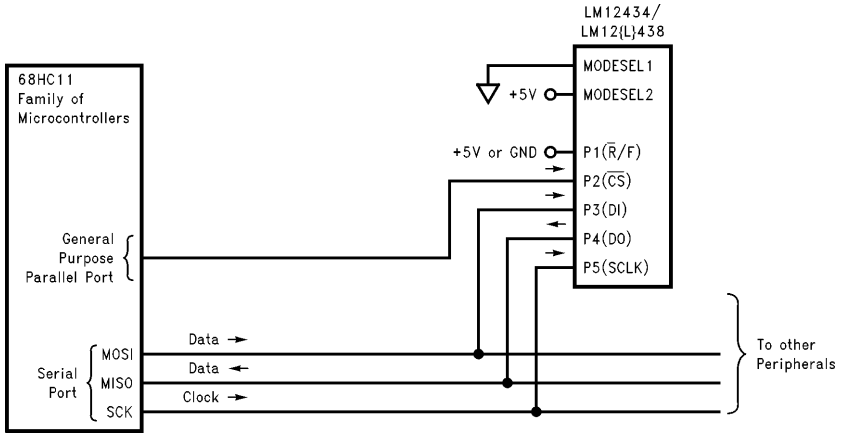
7.1.1 Examples of Interfacing to the HPC's MICROWIRE/PLUS and 68HC11's SPI



TL/H/11879–65

Note: Other device pins are not shown.

FIGURE 12a. LM12434 and LM12{L}438 Standard Mode Interface to the HPC's MICROWIRE/PLUS™



TL/H/11879–66

Note: Other device pins are not shown.

FIGURE 12b. LM12434 and LM12{L}438 Standard Mode Interface to the 68HC11's SPI

7.0 Digital Interface (Continued)

HPC Assembly Code Example

```
;*****************************************************************************
; THE HPC MICROCONTROLLER ASSEMBLY SUBROUTINES FOR INTERFACE TO THE LM12434 AND LM12L{438
; SERIAL DATA ACQUISITION SYSTEM (SDAS) CHIP.
;*****************************************************************************

;*****************************************************************************
; HPC's CONTROL REGISTER ADDRESSES SYMBOLIC DEFINITIONS, USED IN
; INTERFACE ROUTINES
;*****************************************************************************

;       A       = 0x00C8        ;ACCUMULATOR
;       B       = 0x00CC        ;B REGISTER
;       K       = 0x00CA        ;K REGISTER
;       X       = 0x00CE        ;X REGISTER
        PORTB   = 0x00E2        ;PORT B DATA REGISTER
        IRPD    = 0x00D2        ;INTERRUPT PENDING REGISTER
        SIO     = 0x00D6        ;MICROWIRE INPUT/OUTPUT SHIFT REGISTER
        AL      = 0x00C8        ;ACCUMULATOR LOW ORDER BYTE
        AH      = 0x00C9        ;ACCUMULATOR HIGH ORDER BYTE
        uWDONE  = 0x00          ;SYMBOL FOR BIT-0 IN IRPD REGISTER TO TEST
                                ;THE END OF MICROWIRE TRANSFER
;*****************************************************************************
; SERIAL DAS RELATED REGISTERS, CONSTANTS AND MEMORY BLOCK BASE ADDRESSES
; SYMBOLIC DEFINITIONS
;*****************************************************************************

        RINSTR0 = 0xC2          ;SERIAL DAS INSTRUCTION RAM AND LIMITS 1 & 2
        WINSTR0 = 0x82          ;READ AND WRITE CONTROL BYTES. THESE BYTES
        RINSTR1 = 0xC6          ;CONTAIN THE ADDRESS OF THE SDAS REGISTER, THE
        WINSTR1 = 0x86          ;READ/WRITE BIT AND THE MSB/LSB BIT
        RINSTR2 = 0xCA          ;PREDEFINED.
        WINSTR2 = 0x8A          ; "
        RINSTR3 = 0xCE          ; "
        WINSTR3 = 0x8E          ; "
        RINSTR4 = 0xD2          ; "
        WINSTR4 = 0x92          ; "
        RINSTR5 = 0xD6          ; "
        WINSTR5 = 0x96          ; "
        RINSTR6 = 0xDA          ; "
        WINSTR6 = 0x9A          ; "
        RINSTR7 = 0xDE          ; "
        WINSTR7 = 0x9E          ; "

        RCONFIG = 0xE2          ;SDAS CONFIGURATION REG. READ CONTROL BYTE.
        WCONFIG = 0xA2          ;SDAS CONFIGURATION REG. WRITE CONTROL BYTE.
        RINTEN  = 0xE6          ;SDAS INTERRUPT ENABLE REG. READ CONTROL BYTE.
        WINTEN  = 0xA6          ;SDAS INTERRUPT ENABLE REG. WRITE CONTROL BYTE.
        RINTSTAT= 0xEA          ;SDAS INTERRUPT STATUS REG. READ CONTROL BYTE.
        RTIMER  = 0xEE          ;SDAS TIMER REG. READ CONTROL BYTE.
        WTIMER  = 0xAE          ;SDAS TIMER REG. WRITE CONTROL BYTE.
        RSFIFO  = 0xF2          ;SDAS FIFO , SINGLE READ CONTROL BYTE.
        RBFIFO  = 0xF3          ;SDAS FIFO , BURST READ CONTROL BYTE.
        RLMTSTAT= 0xF6          ;SDAS LIMIT STATUS REG. READ CONTROL BYTE.

        DAS_CS  = 0x0X          ;BIT-X OF HPC PORT B USED FOR SDAS CHIP
                                ;SELECT.

        DATA_BLK= 0xXXXX        ;SYMBOLIC STARING ADDRESS OF THE DATA BLOCK
                                ;IN SYSTEM MEMORY, USED TO STORE THE
                                ;CONVERSION RESULTS READ FROM FIFO IN BURST
                                ;READ ROUTINE.
        DATA_BUF= 0xXXXX        ;SYMBOLIC ADDRESS FOR A 16 BIT DATA BUFFER
        CNTRL_BUF=0xXXXX        ;SYMBOLIC ADDRESS FOR AN 8 BIT BUFFER USED
                                ;IN ROUTINES FOR CONTROL BYTE.
        RSLT_NUM= 0xXXXX        ;SYMBOLIC DEFINITION FOR THE NUMBER OF
                                ;RESULTS TO BE READ FROM FIFO IN BURST READ

;*****************************************************************************
; SERIAL DAS READS AND WRITES ARE PERFORMED BY SUBROUTINES SER_WR & SER_RD,
; THESE ROUTINES USE THE CNTRL_BUF REGISTER AS CONTROL INPUT AND THE DATA_BUF
; REGISTER AS DATA BUFFER, FOR WRITES DATA IS LOADED IN THE DATA_BUF REG. AND
; FOR READS DATA RETURNS IN THE DATA_BUF REGISTER.
;*****************************************************************************
```

TL/H/11879–56

7.0 Digital Interface (Continued)

HPC Assembly Code Example (Continued)

```
;--- AN EXAMPLE OF A WRITE TO CONFIGURATION REGISTER:

        LD      CNTRL_BUF.B,#WCONFIG    ;CONFIGURATION REG. WRITE COMMAND
                                        ;LOADED IN THE CNTRL_BUF.
        LD      DATA_BUF.W,#0x0002      ;DATA LOADED ON THE DATA_BUF REG. RESET SDAS,
                                        ;PAUSE=1, RAM POINTER=00.
        JSR     SER_WR                  ;CALLING SER_WR FOR DATA TRANSFER.

;--- AN EXAMPLE OF A READ FROM CONFIGURATION REGISTER:

        LD      B.B,#RCONFIG            ;CONFIGURATION REG. READ COMMAND
                                        ;LOADED IN THE CNTRL_BUF.
        JSR     SER_RD                  ;CALLING SER_RD FOR DATA TRANSFER.

;*************************************************************************
; DATA WRITE SUBROUTINE "SER_WR", FOR SERIAL I/O TRANSFER OF DATA BETWEEN THE
; HPC AND THE SERIAL DAS WITH uW SERIAL INTERFACE. BEFORE CALLING THE ROUTINE,
; THE DATA TRANSFER CONTROL BYTE SHOULD BE LOADED IN THE CNTRL_BUF AND THE
; DATA TO BE WRITTEN TO THE SDAS SHOULD BE LOADED IN THE DATA_BUF.
;*************************************************************************

SER_WR:
        RBIT    DAS_CS,PORTB.B          ;RESET THE PORT B BIT-X TO SELECT
                                        ;THE SDAS1.
        LD      SIO.B,CNTRL_BUF.B       ;LOAD THE CONTROL BYTE TO HPC's SIO
                                        ;REGISTER, BYTE TRANSFER IS STARTED.
WAIT1:  IFBIT   uWDONE,IRPD.B           ;WAIT AND CHECK THE uWDONE BIT FOR
        JP      WBYTE1                  ;COMPLETION OF DATA TRANSFER. WHEN DONE,
        JP      WAIT1                   ;GO AHEAD FOR FIRST DATA BYTE TRANSFER.

WBYTE1: LD      SIO.B,(DATA_BUF+1).B    ;LOAD HIGH ORDER BYTE OF DATA TO SIO
                                        ;REGISTER, TRANSFER IS STARTED.
WAIT2:  IFBIT   uWDONE,IRPD.B           ;WAIT AND CHECK THE uWDONE BIT FOR
        JP      WBYTE2                  ;COMPLETION OF DATA TRANSFER. WHEN DONE,
        JP      WAIT2                   ;GO AHEAD FOR SECOND DATA BYTE TRANSFER.

WBYTE2: LD      SIO.B,DATA_BUF.B        ;LOAD LOW ORDER BYTE OF DATA TO SIO
                                        ;REGISTER, TRANSFER IS STARTED.
WAIT3:  IFBIT   uWDONE,IRPD.B           ;WAIT AND CHECK THE uWDONE BIT FOR
        JP      WDONE                   ;COMPLETION OF DATA TRANSFER. WHEN DONE,
        JP      WAIT3                   ;DESELECT THE SDAS AND RETURN.

WDONE:  SBIT    DAS_CS,PORTB.B          ;SET THE BIT TO DESELECT THE SDAS.
        RET                             ;RETURN FROM SUBROUTINE.

;*************************************************************************
; DATA READ SUBROUTINE "SER_RD", FOR SERIAL I/O TRANSFER OF DATA BETWEEN THE
; HPC AND THE SERIAL DAS WITH uW SERIAL INTERFACE. BEFORE CALLING THE ROUTINE,
; THE DATA TRANSFER CONTROL BYTE SHOULD BE LOADED IN THE CNTRL_BUF AND THE
; DATA IS LOADED IN THE DATA_BUF UPON RETURN FROM SUBROUTINE.
;*************************************************************************

SER_RD:
        RBIT    DAS_CS,PORTB.B          ;RESET THE PORT B BIT-X TO SELECT
                                        ;THE SDAS1.
        LD      SIO.B,CNTRL_BUF.B       ;LOAD THE CONTROL BYTE TO HPC's SIO
                                        ;REGISTER, BYTE TRANSFER IS STARTED.
WAIT4:  IFBIT   uWDONE,IRPD.B           ;WAIT AND CHECK THE uWDONE BIT FOR
        JP      RBYTE1                  ;COMPLETION OF DATA TRANSFER. WHEN DONE,
        JP      WAIT4                   ;GO AHEAD FOR FIRST DATA BYTE TRANSFER.

RBYTE1: LD      SIO.B,#0x00             ;LOAD THE SIO REGISTER WITH ZERO,
                                        ;THIS IS JUST A DUMMY LOAD TO START
                                        ;THE DATA TRANSFER
WAIT5:  IFBIT   uWDONE,IRPD.B           ;WAIT AND CHECK THE uWDONE BIT FOR
        JP      RBYTE2                  ;COMPLETION OF DATA TRANSFER. WHEN DONE,
        JP      WAIT5                   ;GO AHEAD FOR SECOND DATA BYTE TRANSFER.

RBYTE2: LD      (DATA_BUF+1).B,SIO.B    ;LOAD HIGH ORDER BYTE OF THE DATA_BUF REGISTER
                                        ;WITH DATA JUST READ FROM SDAS.
        LD      SIO.B,#0x00             ;LOAD THE SIO REGISTER WITH ZERO,
                                        ;THIS IS JUST A DUMMY LOAD TO START
```

TL/H/11879–57

7.0 Digital Interface (Continued)

HPC Assembly Code Example (Continued)

```
                                        ;THE DATA TRANSFER
WAIT6:  IFBIT   uWDONE,IRPD.B           ;WAIT AND CHECK THE uWDONE BIT FOR
        JP      RDONE                   ;COMPLETION OF DATA TRANSFER. WHEN DONE,
        JP      WAIT6                   ;LOAD THE READ DATA TO AL, DESELECT THE
                                        ;SDAS AND RETURN.

RDONE:  LD      DATA_BUF.B,SIO.B        ;LOAD LOW ORDER BYTE OF THE DATA_BUF REGISTER
                                        ;WITH THE DATA JUST READ FROM SDAS.
        SBIT    DAS_CS,PORTB.B          ;SET THE BX TO DESELECT THE SDAS.
        RET                             ;RETURN FROM SUBROUTINE.

;*************************************************************************
; FIFO BURST READ SUBROUTINE "RD_FIFO",USED FOR READING THE CONVERSION RESULTS
; FROM FIFO IN BURST READ MODE. DATA IS READ FROM FIFO AND STORED IN THE
; SYSTEM MEMORY STARTING FROM THE DATA_BLK ADDRESS. NUMBER OF CONVERSION
; RESULTS BEING READ IS RSLT_NUM WHICH IS LOADED IN THE X REGISTER. IT IS ASSUMED
; THAT THE HPC IS USING 16 BIT DATA BUS.
;*************************************************************************

RD_FIFO:

        LD      BK.W,#DATA_BLK,#(DATA_BLK+2*RSLT_NUM-1)
                                        ;SET B FOR STARTING ADDRESS OF MEMORY
                                        ;AND K FOR ENDING ADDRESS MINUS ONE
LPFIFO:
        LD      X.W,#RSLY_NUM           ;A COUNTER TO KEEP TRACK OF # OF FIFO
                                        ;READS FOR TERMINATION OF BURST MODE
                                        ;BY PULLING THE CHIP SELECT HIGH DURING
                                        ;THE LAST READ CYCLE AND BEFORE THE
                                        ;14TH CLOCK EDGE.
        RBIT    DAS_CS,PORTB.B          ;RESET THE PORT B BIT-X TO SELECT
                                        ;THE SDAS.
        LD      SIO.B,#RBFIFO           ;LOAD THE BURST FIFO READ CONTROL BYTE
                                        ;TO SIO REG. BYTE TRANSFER IS STARTED.
WAIT7:  IFBIT   uWDONE,IRPD.B           ;WAIT AND CHECK THE uWDONE BIT FOR
        JP      MSBYTE                  ;COMPLETION OF DATA TRANSFER. WHEN DONE,
        JP      WAIT7                   ;GO AHEAD FOR FIRST DATA BYTE READ.

MSBYTE: LD      SIO.B,#0x00             ;LOAD THE SIO WITH 0, THIS IS JUST A
                                        ;DUMMY LOAD TO START THE DATA TRANSFER
WAIT8:  IFBIT   uWDONE,IRPD.B           ;WAIT AND CHECK THE uWDONE BIT FOR
        JP      LSBYTE                  ;COMPLETION OF DATA TRANSFER. WHEN DONE,
        JP      WAIT8                   ;GO AHEAD FOR SECOND DATA BYTE READ.

LSBYTE: LD      AH.B,SIO.B              ;LOAD HIGH ORDER BYTE OF THE A REGISTER
                                        ;WITH DATA JUST READ FROM THE SDAS.
        LD      SIO.B,#0x00             ;LOAD THE SIO WITH 0, THIS IS JUST A
                                        ;DUMMY LOAD TO START THE DATA TRANSFER.

        DECSZ   X                       ;DECREMENT X AND SET THE SDAS CHIP-
        JP      WAIT9                   ;SELECT BIT IF LAST READ CYCLE (X=0),
        SBIT    DAS_CS,PORTB.B          ;OTHERWISE CONTINUE.

WAIT9:  IFBIT   uWDONE,IRPD.B           ;WAIT AND CHECK THE uWDONE BIT FOR
        JP      CMPLT                   ;COMPLETION OF DATA TRANSFER. WHEN DONE,
        JP      WAIT9                   ;LOAD THE READ DATA TO AL.

CMPLT:  LD      AL.B,SIO.B              ;LOAD LOW ORDER BYTE OF THE A REGISTER
                                        ;WITH THE DATA JUST READ FROM THE SDAS.
        XS      A,[B+].W                ;STORE A TO THE DATA_BLK WITH B AUTO-
                                        ;INCREMENT AND SKIP IF GREATER THAN K.
        JP      MSBYTE                  ;GO FOR THE NEXT FIFO DATA

        RET

;*************************************************************************
; THIS ROUTINE INITIALIZES THE SDAS SERIAL INTERFACE IN CASE A
; COMMUNICATION CYCLE IS INTERRUPTED IN THE MIDDLE OF A CYCLE FOR ANY REASON.
;*************************************************************************

SDAS_SER_PORT_RST:

        SBIT    DAS_CS,PORTB.B          ;DESELECT THE SDAS
```

TL/H/11879–58

7.0 Digital Interface (Continued)

HPC Assembly Code Example (Continued)

```
        LD      SIO.B,#0x00             ;RESET SEQUENCE FOR THE SDAS INTER-
RWAIT1: IFBIT   uWDONE,IRPD.B           ;FACE TO BRING IT OUT OF A HANGUP BY
        JP      R_NXT2                  ;APPLYING 24 SERIAL CLOCK PULSE WHILE
        JP      RWAIT1                  ;CHIP SELECT IS HIGH, THIS IS EQUAL TO
R_NXT2: LD      SIO.B,#0x00             ;POWER UP RESET FOR THE INTERFACE
RWAIT2: IFBIT   uWDONE,IRPD.B           ;FACE TO BRING IT OUT OF A HANGUP
        JP      R_NXT3                  ;NOTE THAT THIS ROUTINE DOES NOT RESET
        JP      RWAIT2                  ;THE SERIAL DAS BUT ONLY THE SERIAL INTERFACE
R_NXT3: LD      SIO.B,#0x00             ;THIS ROUTINE IS USEFUL DURING
RWAIT3: IFBIT   uWDONE,IRPD.B           ;SOFTWARE DEVELOPMENT OR IN CASE THAT
        RET                             ;A COMMUNICATION CYCLE NEEDS TO BE
        JP      RWAIT3                  ;INTERRUPTED BY SYSTEM REQUIREMENTS.
```

TL/H/11879–59

68HC11 Assembly Code Example

```
*****************************************************************************
* THE 68HC11 MICROCONTROLLERS FAMILY ASSEMBLY SUBROUTINES FOR INTERFACE TO
* THE LM12434 AND LM12{L}438 SERIAL DATA ACQUISITION SYSTEM (SDAS) CHIP.
*****************************************************************************

*****************************************************************************
* 68HC11 CONTROLLER REGISTER'S ADDRESSES SYMBOLIC DEFINITIONS, USED IN
* INTERFACE ROUTINES
*****************************************************************************
PORTD   EQU     $1008   ; Port D data register
*                       ; " -    , -  ,SS* ,SCK ;MOSI,MISO,TxD ,RxD "
*                       ; PORT D "SS" BIT IS USED FOR SDAS CHIP SELECT
DDRD    EQU     $1009   ; Port D data direction
SPCR    EQU     $1028   ; SPI control register
*                       ; "SPIE,SPE ,DWOM,MSTR;CPOL,CPHA,SPR1,SPR0"
SPSR    EQU     $1029   ; SPI status register
*                       ; "SPIF,WCOL, -  ,MODF; -    , -  , -  , -  "
SPDR    EQU     $102A   ; SPI data register; Read-Buffer; Write-Shifter
*
*****************************************************************************
* SERIAL DAS RELATED REGISTERS, CONSTANTS AND MEMORY BLOCKS BASE ADDRESSES
* SYMBOLIC DEFINITIONS
*****************************************************************************
*
RINSTR0   EQU   $C2          ;SERIAL DAS INSTRUCTION RAM AND LIMITS 1 & 2
WINSTR0   EQU   $82          ;READ AND WRITE CONTROL BYTES. THESE BYTES
RINSTR1   EQU   $C6          ;CONTAIN ADDRESSES OF THE SDAS REGISTER, THE
WINSTR1   EQU   $86          ;READ/WRITE BIT AND THE MSB/LSB BIT
RINSTR2   EQU   $CA          ;PREDEFINED.
WINSTR2   EQU   $8A          ; "
RINSTR3   EQU   $CE          ; "
WINSTR3   EQU   $8E          ; "
RINSTR4   EQU   $D2          ; "
WINSTR4   EQU   $92          ; "
RINSTR5   EQU   $D6          ; "
WINSTR5   EQU   $96          ; "
RINSTR6   EQU   $DA          ; "
WINSTR6   EQU   $9A          ; "
RINSTR7   EQU   $DE          ; "
WINSTR7   EQU   $9E          ; "

RCONFIG   EQU   $E2          ;SDAS CONFIGURATION REG. READ CONTROL BYTE.
WCONFIG   EQU   $A2          ;SDAS CONFIGURATION REG. WRITE CONTROL BYTE.
RINTEN    EQU   $E6          ;SDAS INTERRUPT ENABLE REG. READ CONTROL BYTE.
WINTEN    EQU   $A6          ;SDAS INTERRUPT ENABLE REG. WRITE CONTROL BYTE.
RINTSTAT  EQU   $EA          ;SDAS INTERRUPT STATUS REG. READ CONTROL BYTE.
RTIMER    EQU   $EE          ;SDAS TIMER REG. READ CONTROL BYTE.
WTIMER    EQU   $AE          ;SDAS TIMER REG. WRITE CONTROL BYTE.
RSFIFO    EQU   $F2          ;SDAS FIFO , SINGLE READ CONTROL BYTE.
RBFIFO    EQU   $F3          ;SDAS FIFO , BURST READ CONTROL BYTE.
RLMTSTAT  EQU   $F6          ;SDAS LIMIT STATUS REG. READ CONTROL BYTE.
*
*
```

TL/H/11879–85

7.0 Digital Interface (Continued)

68HC11 Assembly Code Example (Continued)

```
DATA_BLK  EQU   $10          ;SYMBOLIC STARTING ADDRESS OF THE DATA BLOCK
*                            ;IN SYSTEM MEMORY, USED TO STORE THE
*                            ;CONVERSION RESULTS READ FROM FIFO IN BURST
*                            ;READ ROUTINE.
DATA_BUF  EQU   $42          ;SYMBOLIC ADDRESS FOR A 16 BIT DATA BUFFER
CNTRL_BUF EQU   $40          ;SYMBOLIC ADDRESS FOR AN 8 BIT BUFFER USED
*                            ;IN ROUTINES FOR CONTROL BYTE.
RSLT_NUM  EQU   $10          ;SYMBOLIC DEFINITION FOR THE NUMBER OF
*                            ;RESULTS TO BE READ FROM FIFO IN BURST READ
*
*******************************************************************************
* SERIAL DAS READS AND WRITES ARE PERFORMED BY SUBROUTINE SER_IO.
* THIS ROUTINE USES THE CNTRL_BUF REGISTER AS CONTROL INPUT AND THE DATA_BUF
* REGISTER AS DATA BUFFER, FOR WRITES DATA IS LOADED ON THE DATA_BUF REG. AND
* FOR READS DATA RETURNS IN THE DATA_BUF REGISTER.
*******************************************************************************

*--- AN EXAMPLE OF A WRITE TO CONFIGURATION REGISTER:

        LDAA    #WCONFIG             ;CONFIGURATION REG. WRITE COMMAND
        STAA    CNTRL_BUF            ;LOADED IN THE CNTRL_BUF.
        LDD     #$0010               ;DATA LOADED ON THE DATA_BUF REG.
        STD     DATA_BUF             ;RESET= 1, RAM POINTER=00.
        JSR     SER_IO               ;CALLING SER_WR FOR DATA TRANSFER.

*--- AN EXAMPLE OF A READ FROM CONFIGURATION REGISTER:

        LDAA    #RCONFIG             ;CONFIGURATION REG. READ COMMAND
        STAA    CNTRL_BUF            ;LOADED IN THE CNTRL_BUF.
        JSR     SER_IO               ;CALLING SER_RD FOR DATA TRANSFER.

*******************************************************************************
* DATA WRITE/READ SUBROUTINE "SER_IO", FOR SERIAL I/O TRANSFER OF DATA BETWEEN
* THE 68HC11 AND THE SERIAL DAS WITH SPI SERIAL INTERFACE. BEFORE CALLING THE
* ROUTINE, THE DATA TRANSFER CONTROL BYTE SHOULD BE LOADED IN THE CNTRL_BUF.
* FOR WRITES THE DATA TO BE WRITTEN TO THE SDAS SHOULD BE LOADED IN DATA_BUF.
* FOR READS, DATA IS LOADED INTO THE DATA_BUF UPON RETURN FROM THIS SUBROUTINE.
*******************************************************************************

SER_IO:
        BCLR    PORTD,Y $20          ; DROP CHIP SELECT
        LDAA    CNTRL_BUF            ; LOAD A WITH CONTROL BYTE
        STAA    SPDR                 ; START SPI SEND
SEND1   LDAA    SPSR                 ; GET SPI STATUS TO WAIT FOR SPIF
        ANDA    #$80                 ; MASKING THE EIGHTH BIT WITH THE SPIF BIT
        BEQ     SEND1                ; IF SPIF=0 THEN BRANCH, ELSE SKIP
        LDAA    DATA_BUF             ; GET MSB DATA BYTE AND SEND
        STAA    SPDR                 ; START SPI SEND. THIS WILL ALSO CLEAR THE SPIF BIT
SEND2   LDAA    SPSR                 ; GET SPI STATUS TO WAIT FOR SPIF
        ANDA    #$80                 ; MASKING THE EIGHTH BIT WITH THE SPIF BIT
        BEQ     SEND2                ; IF SPIF=0 THEN BRANCH, ELSE SKIP
        LDAA    SPDR                 ; LOADS 1 DATA BYTE (MSB/LSB) SENT FROM DAS INTO ACC A
        STAA    DATA_BUF             ; STORE MSB DATA BYTE IN RAM BUFFER
        LDAA    DATA_BUF+1           ; GET LSB DATA BYTE TO SEND
        STAA    SPDR                 ; START SPI SEND
SEND3   LDAA    SPSR                 ; GET SPI STATUS TO WAIT FOR SPIF
        ANDA    #$80                 ; MASKING THE EIGHTH BIT WITH THE SPIF BIT
        BEQ     SEND3                ; IF SPIF=0 THEN BRANCH, ELSE SKIP
        LDAA    SPDR                 ; LOADS 1 DATA BYTE (MSB/LSB) SENT FROM DAS INTO ACC A
        STAA    DATA_BUF             ; STORE MSB DATA BYTE IN RAM BUFFER
        BSET    PORTD,Y $20          ; DONE -- RAISE CS
        RTS
```

TL/H/11879–86

7.0 Digital Interface (Continued)

68HC11 Assembly Code Example (Continued)

```
*****************************************************************************
* FIFO BURST READ SUBROUTINE "RD_FIFO", FOR READING THE CONVERSION RESULTS
* FROM FIFO IN BURST READ MODE. DATA IS READ FROM FIFO AND STORED IN THE
* SYSTEM MEMORY STARTING FROM THE DATA_BLK ADDRESS. NUMBER OF CONVERSION
* RESULTS BEING READ IS RSLT_NUM WHICH IS LOADED IN THE X REGISTER. IT IS ASSUMED
* THAT THE HPC IS USING 16 BIT DATA BUS.
*****************************************************************************
RD_FIFO:
        LDX     #DATA_BLK               ; LOAD X WITH DATA BLOCK BASE ADDRESS
        LDAB    #RSLT_NUM               ; LOAD B WITH NUMBER OF RESULTS
        LSLB                            ; MAKE INTO BYTE COUNT
        DECB                            ; ONE LESS FOR LAST BYTE
        BCLR    PORTD,Y $20             ; DROP CHIP SELECT
        LDAA    #RBFIFO                 ; LOAD A WITH BURST READ COMMAND
        STAA    SPDR                    ; SEND COMMAND
BURST1  LDAA    SPSR                    ; GET SPI STATUS TO WAIT FOR SPIF
        ANDA    #$80                    ; MASKING THE EIGHTH BIT WITH THE SPIF BIT
        BEQ     BURST1                  ; IF SPIF=0 THEN BRANCH, ELSE SKIP

BLOOP
        CLRA                            ; CLEAR DATA BYTE TO SEND
        STAA    SPDR                    ; START SPI, RECEIVE A DATA BYTE
BURST2  LDAA    SPSR                    ; GET SPI STATUS TO WAIT FOR SPIF
        ANDA    #$80                    ; MASKING THE EIGHTH BIT WITH THE SPIF BIT
        BEQ     BURST2                  ; IF SPIF=0 THEN BRANCH, ELSE SKIP
        LDAA    SPDR                    ; GET THE RECEIVED DATA BYTE
        STAA    0,X                     ; STORE DATA BYTE
        INX                             ; POINT TO NEXT DATA BYTE
        DECB                            ; COUNTING DOWN # OF BYTES
        BNE     BLOOP                   ; STILL MORE DATA BYTES TO GET
        BSET    PORTD,Y $20             ; RAISE CS TO END BURST READ
        STAA    SPDR                    ; START SPI, RECEIVE LAST BYTE
BURST3  LDAA    SPSR                    ; GET SPI STATUS TO WAIT FOR SPIF
        ANDA    #$80                    ; MASKING THE EIGHTH BIT WITH THE SPIF BIT
        BEQ     BURST3                  ; IF SPIF=0 THEN BRANCH, ELSE SKIP
        LDAA    SPDR                    ; GET RECEIVED DATA BYTE
        STAA    0,X                     ; STORE DATA BYTE IN RAM BUFFER
        RTS
```

TL/H/11879–87

7.0 Digital Interface (Continued)

7.2 8051 INTERFACE MODE

The 8051 interface mode is designed to work directly with the 8051 family of microcontrollers' mode 0 serial interface. This interface mode is a simple shift register type of serial data transfer. The serial clock synchronizes the transfer of data to and from the LM12434 and LM12{L}438. The interface uses 3 lines: a bidirectional data line (RXD), a serial clock line (TXD) and a chip-select ($\overline{CS}$) line. More than one device can share the data and serial clock lines provided that each device has its own chip-select line.

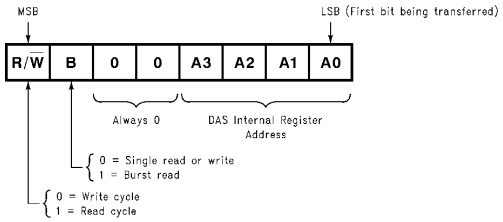
The 8051 mode is selected when the MODESEL1 and MODESEL2 pins have the logic state of ''00''. *Figure 14* shows a typical connection diagram for the 8051 mode serial interface. The $\overline{CS}$, RXD and TXD lines are respectively assigned to interface pins P3 through P5. The P1 and P2 pins are not used in this mode and should be left open or connected to logic ''1''. In this interface the idle state of the serial clock TXD is logic ''1''. The data is stable at both edges of the TXD clock and is shifted after its rising edge. The interface has a bidirectional RXD data line. The LM12434 and LM12{L}438 leaves the RXD line in a high impedance state whenever it is not outputting any data.

Data transfer in this mode is byte oriented. As mentioned, the LM12434 and LM12{L}438 has three different communication cycles: write cycle, read cycle and burst read cycle. At the start of each data transfer cycle, ''command byte'' is written to the LM12434 and LM12{L}438, followed by write or read data. The command byte informs the LM12434 and LM12{L}438 about the communication cycle and carries the following information:

— what type of data transfer (communication cycle) is started

— which device register is to be accessed

The command byte has the following format:



TL/H/11879–53

The first bit is the LSB of the byte based on the 8051 mode 0 serial interface protocol.

Figure 13 shows the timing diagrams for different communication cycles. *Figure 13a* shows a write cycle. *Figure 13b* shows a read cycle. *Figure 13c* shows a burst read cycle. Note that these timing diagrams depict general relationships between the SCLK edges, the data bits and $\overline{CS}$. These diagrams are not meant to show guaranteed timing performance. (See specification tables for parametric switching characteristics.)

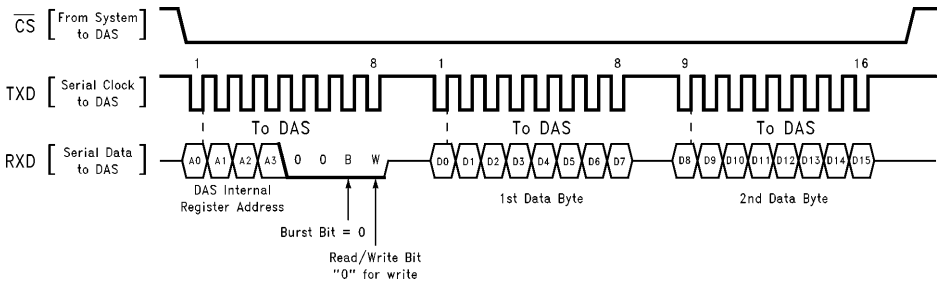
Write cycle: A write cycle begins with the falling edge of the $\overline{CS}$. Then a command byte is written to the DAS on the RXD line synchronized by TXD clock. The command byte has the R/W and B bits equal to zero. Following the command byte, 16 bits of data (2 bytes) is shifted in on the RXD line. The data is written to the register addressed in the command byte (A3, A2, A1, A0). The data is always LSB first in this interface. $\overline{CS}$ will go high after the transfer of the last bit, thus completing the write cycle.

Read cycle: A read cycle starts the same way as a write cycle, except that the command bytes R/$\overline{W}$ bit is equal to one. Following the command byte, the DAS outputs the data on the RXD line synchronized with the microcontroller's TXD clock. The data is read from the register addressed in the command byte. Data is shifted in LSB first. Again, $\overline{CS}$ will go high after the transfer of the last data bit, thus completing the read cycle.

7.0 Digital Interface (Continued)

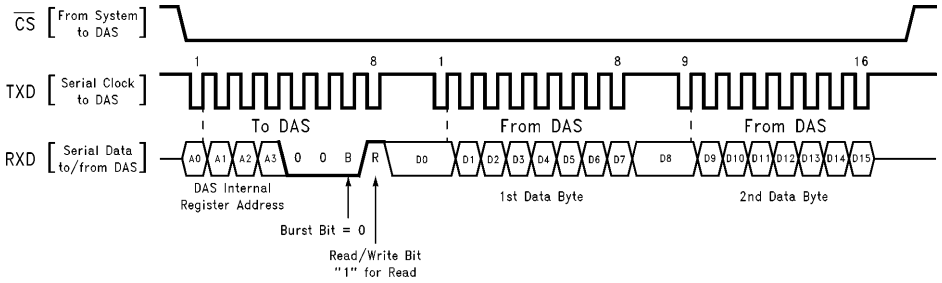
Burst read cycle: A burst read cycle starts the same way as a single read cycle, but the B bit in the command byte is set to one, indicating a burst read cycle. Following the command byte the data is output on the RXD line as long as the DAS receives TXD clock from the system. To tell the DAS when a burst read cycle is completed, $\overline{CS}$ should be set high after the 8th and before the 15th SCLK cycle during the last data byte transfer (see *Figure 13c*). After $\overline{CS}$ high is detected and the last data bit is transferred, the DAS is ready for a new communication cycle to begin.

The timing diagrams in *Figure 13* show the transfer of data in packets of 8 bits (bytes). This represents the way the serial ports of the 8051 family of microcontrollers produce the serial clock and data. The DAS does not require a gap between the first and second bytes of the data; 16 continuous clock cycles will transfer the data word. However, there should be a gap equal to 3 CLK (the DAS main clock input, not the TXD clock) cycles between the end of the command byte and the start of the data during a read cycle. This is not concerned in most systems for two reasons. First, the processor generally has some inherent gap between byte transfers. Second, the TXD frequency is usually significantly slower than the CLK frequency. For example, an 8051 processor with 12 MHz crystal generates a TXD of 1 MHz. If the DAS is running with 6 MHz CLK, there are 6 cycles of CLK within each cycle of TXD and the requirement is satisfied even if TXD comes continuously after command byte. The user should pay attention to this requirement if running the TXD with a speed near or higher than CLK.



TL/H/11879–40

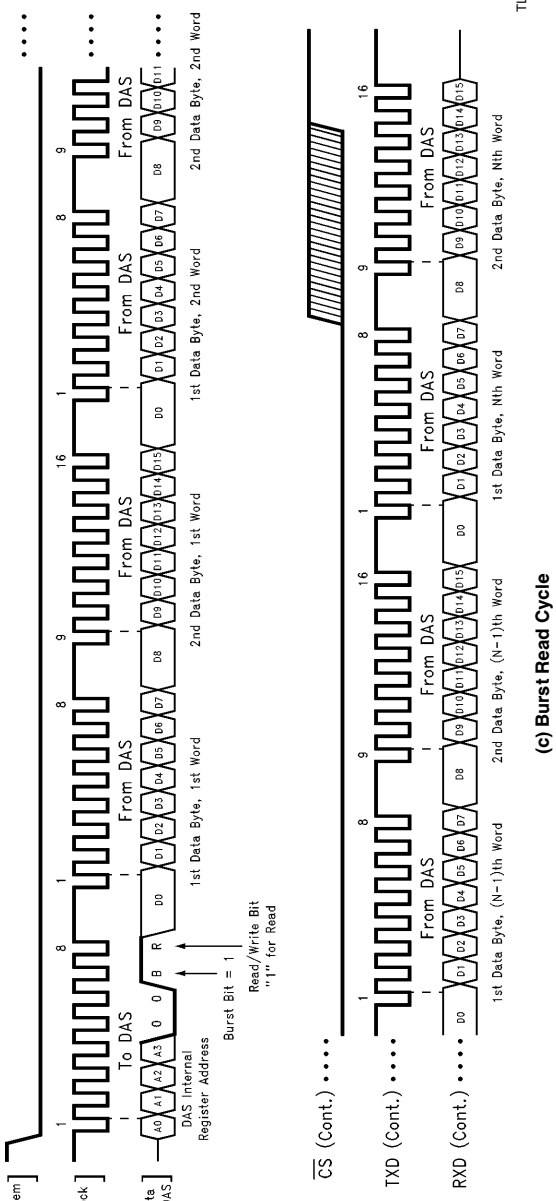
(a) Write Cycle
Idle State of SCLK = 1, Data Shifted at the Rising Edge of the SCLK



TL/H/11879–41

(b) Read Cycle
Idle State of SCLK = 1, Data Shifted at the Rising Edge of the SCLK

FIGURE 13. Timing Diagrams for LM12434 and LM12{L}438 8051 Serial Interface Mode



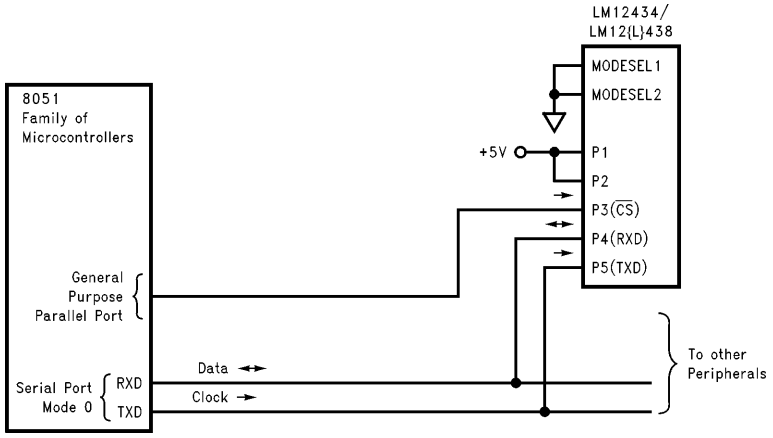
(c) Burst Read Cycle

Idle State of SCLK = 1, Data Shifted after the Rising Edge of the SCLK

FIGURE 13. Timing Diagrams for LM12434 and LM12(L)438 8051 Serial Interface Mode (Continued)

7.0 Digital Interface (Continued)

7.2.1 Example of Interfacing to the 8051



TL/H/11879–67

FIGURE 14. LM12434 and LM12{L}438 in the 8051 Interface Mode

8051 Assembly Code Example

```
;*****************************************************************************
; THE 8051 MICROCONTROLLERS FAMILY ASSEMBLY SUBROUTINES FOR INTERFACE TO
; THE LM12434 and LM12{L}438, SERIAL INTERFACE DATA ACQUISITION SYSTEM (SDAS) CHIP.
;*****************************************************************************

;*****************************************************************************
; 8051 CONTROLLER REGISTER, BITS SYMBOLIC DEFINITIONS, USED IN INTERFACE
; ROUTINES
;*****************************************************************************

;SCON                    ;SERIAL PORT CONTROL REGISTER
R_DONE     BIT    SCON.0  ;RECEIVE CYCLE COMPLETE FLAG, BIT #0 OF SCON
S_DONE     BIT    SCON.1  ;SEND CYCLE COMPLETE FLAG, BIT #1 OF SCON
R_EN       BIT    SCON.4  ;RECEIVE CYCLE ENABLE BIT, BIT #4 OF SCON
;SBUF                    ;SERIAL PORT DATA BUFFER FOR SEND AND RECEIVE
SDAS_SLCT BIT     P3.4    ;PIN #4 OF PORT 3 USED FOR THE SDAS CHIP SELECT

;*****************************************************************************
; SERIAL DAS RELATED REGISTERS, CONSTANTS AND MEMORY BLOCK BASE ADDRESSES
; SYMBOLIC DEFINITIONS
;*****************************************************************************

RINSTR0    EQU    80H     ;SERIAL DAS INSTRUCTION RAM AND LIMITS 1 & 2
WINSTR0    EQU    00H     ;READ AND WRITE CONTROL BYTES. THESE BYTES
RINSTR1    EQU    81H     ;CONTAIN ADDRESSES OF THE SDAS REGISTERS, THE
WINSTR1    EQU    01H     ;READ/WRITE BIT AND THE BURST READ BIT
RINSTR2    EQU    82H     ;PREDEFINED.
WINSTR2    EQU    02H     ; "
RINSTR3    EQU    83H     ; "
WINSTR3    EQU    03H     ; "
RINSTR4    EQU    84H     ; "
WINSTR4    EQU    04H     ; "
RINSTR5    EQU    85H     ; "
WINSTR5    EQU    05H     ; "
RINSTR6    EQU    86H     ; "
WINSTR6    EQU    06H     ; "
RINSTR7    EQU    87H     ; "
WINSTR7    EQU    07H     ; "
```

TL/H/11879–89

7.0 Digital Interface (Continued)

8051 Assembly Code Example (Continued)

```
RCONFIG   EQU   88H     ;SDAS CONFIGURATION REG. READ CONTROL BYTE.
WCONFIG   EQU   08H     ;SDAS CONFIGURATION REG. WRITE CONTROL BYTE.
RINTEN    EQU   89H     ;SDAS INTERRUPT ENABLE REG. READ CONTROL BYTE.
WINTEN    EQU   09H     ;SDAS INTERRUPT ENABLE REG. WRITE CONTROL BYTE.
RINTSTAT  EQU   8AH     ;SDAS INTERRUPT STATUS REG. READ CONTROL BYTE.
RTIMER    EQU   8BH     ;SDAS TIMER REG. READ CONTROL BYTE.
WTIMER    EQU   0BH     ;SDAS TIMER REG. WRITE CONTROL BYTE.
RSFIFO    EQU   8CH     ;SDAS FIFO , SINGLE READ CONTROL BYTE.
RBFIFO    EQU   0CCH    ;SDAS FIFO , BURST READ CONTROL BYTE.
RLMTSTAT  EQU   8DH     ;SDAS LIMIT STATUS REG. READ CONTROL BYTE.


DATA_BLK  EQU   0XXH    ;SYMBOLIC STARTING ADDRESS OF THE DATA BLOCK
                        ;IN SYSTEM MEMORY, USED TO STORE THE
                        ;CONVERSION RESULTS READ FROM FIFO IN BURST
                        ;READ ROUTINE.
DATA_BUF  EQU   0XXH    ;SYMBOLIC ADDRESS FOR A 16 BIT DATA BUFFER
CNTRL_BUF EQU   0XXH    ;SYMBOLIC ADDRESS FOR AN 8 BIT BUFFER USED
                        ;IN ROUTINES FOR CONTROL BYTE.
RSLT_NUM  EQU   0XXH    ;SYMBOLIC DEFINITION FOR THE NUMBER OF
                        ;RESULTS TO BE READ FROM FIFO IN BURST READ

;*****************************************************************************
; SERIAL DAS READS AND WRITES ARE PERFORMED BY SUBROUTINES SER_WR & SER_RD,
; THESE ROUTINES USE THE "CNTRL_BUF" REGISTER AS CONTROL INPUT AND THE
; "DATA_BUF" REGISTER AS DATA BUFFER, FOR WRITES DATA IS LOADED ON THE
; "DATA_BUF" REGISTER, AND FOR READS DATA RETURNS IN THE "DATA_BUF" REGISTER.
;*****************************************************************************

;--- AN EXAMPLE OF A WRITE TO CONFIGURATION REGISTER:

        MOV     CNTRL_BUF,#WCONFIG      ;LOAD CNTRL_BUF WITH WRITE CONTROL
                                        ;BYTE
        MOV     DATA_BUF,#02H           ;LOAD LOW ORDER BYTE OF DATA TO
                                        ;DATA_BUF
        MOV     DATA_BUF+1,#00H         ;LOAD HIGH ORDER BYTE OF DATA TO
                                        ;DATA_BUF
        LCALL   SER_WR                  ;SER_WR ROUTINE TRANSFERS THE DATA

;--- AN EXAMPLE OF A READ FROM CONFIGURATION REGISTER:

        MOV     CNTRL_BUF,#RTIMER       ;LOAD CNTRL_BUF WITH READ CONTROL
                                        ;BYTE
        LCALL   SER_RD                  ;SER_RD ROUTINE READS THE DATA
```

TL/H/11879–90

7.0 Digital Interface (Continued)

8051 Assembly Code Example (Continued)

```
;*****************************************************************************
; DATA WRITE SUBROUTINE "SER_WR", FOR A SERIAL WRITE TO THE DAS. BEFORE CALLING THE
; ROUTINE, THE WRITE CONTROL BYTE SHOULD BE LOADED IN THE "CNTRL_BUF"
; AND THE DATA TO BE WRITTEN TO THE SDAS SHOULD BE LOADED IN THE "DATA_BUF".
;*****************************************************************************

SER_WR:
        CLR     SDAS_SLCT               ;SELECT THE SDAS, CHIP SELECT=0
        CLR     S_DONE                  ;CLEAR SEND CYCLE DONE FLAG
        MOV     SBUF,CNTRL_BUF          ;START SENDING THE WRITE CONTROL BYTE
SENDW:  JNB     S_DONE,SENDW            ;WAIT HERE UNTIL SEND CYCLE COMPLETED

        CLR     S_DONE                  ;CLEAR SEND CYCLE DONE FLAG
        MOV     SBUF,DATA_BUF           ;START SENDING LOW ORDER BYTE OF DATA
SEND1:  JNB     S_DONE,SEND1            ;WAIT HERE UNTIL SEND CYCLE COMPLETED

        CLR     S_DONE                  ;CLEAR SEND CYCLE DONE FLAG
        MOV     SBUF,DATA_BUF+1         ;START SENDING HIGH ORDER BYTE OF DATA
SEND2:  JNB     S_DONE,SEND2            ;WAIT HERE UNTIL SEND CYCLE COMPLETED

        SETB    SDAS_SLCT               ;DESELECT THE SDAS, CHIP SELECT=1
        RET
;*****************************************************************************
; DATA READ SUBROUTINE "SER_RD", FOR A SERIAL READ  FROM THE DAS. BEFORE CALLING THE
; ROUTINE, THE READ CONTROL BYTE SHOULD BE LOADED ON THE "CNTRL_BUF"
; AND THE DATA IS LOADED IN THE "DATA_BUF" UPON RETURN FROM SUBROUTINE.
;*****************************************************************************
SER_RD:
        CLR     SDAS_SLCT               ;SELECT THE SDAS, CHIP SELECT=0
        CLR     S_DONE                  ;CLEAR SEND CYCLE DONE FLAG
        MOV     SBUF,CNTRL_BUF          ;START SENDING THE READ CONTROL BYTE
SENDR:  JNB     S_DONE,SENDR            ;WAIT HERE UNTIL SEND CYCLE COMPLETED

        SETB    R_EN                    ;ENABLE DATA RECEIVE CYCLES

        CLR     R_DONE                  ;START A DATA BYTE RECEIVE CYCLE
RCV1:   JNB     R_DONE,RCV1             ;WAIT HERE UNTIL RECEIVE COMPLETED
        MOV     DATA_BUF,SBUF           ;STORE LOW ORDER BYTE IN DATA_BUF

        CLR     R_DONE                  ;START A DATA BYTE RECEIVE CYCLE
RCV2:   JNB     R_DONE,RCV2             ;WAIT HERE UNTIL RECEIVE COMPLETED
        MOV     DATA_BUF+1,SBUF         ;STORE HIGH ORDER BYTE IN DATA_BUF

        SETB    SDAS_SLCT               ;DESELECT THE SDAS, CHIP SELECT=1
        CLR     R_EN                    ;DISABLE DATA RECEIVE CYCLES
        RET
```

TL/H/11879–91

7.0 Digital Interface (Continued)

8051 Assembly Code Example (Continued)

```
;*****************************************************************************
; FIFO BURST READ SUBROUTINE "RD_FIFO", FOR READING THE CONVERSION RESULTS
; FROM FIFO IN BURST READ MODE. DATA IS READ FROM FIFO AND STORED IN THE
; SYSTEM MEMORY STARTING FROM THE "DATA_BLK" ADDRESS. NUMBER OF CONVERSION
; RESULTS BEING READ IS "RSLT_NUM". THIS ROUTINE USES THE R0 AND R1 REGISTERS.
; IT IS ASSUMED THAT THEY ARE IN THE PRESENT REGISTER BANK.
; R0 IS THE POINTER TO "DATA_BLK" WHERE THE CONVERSION RESULTS ARE STORED.
; R1 IS USED AS A COUNTER TO KEEP TRACK OF THE NUMBER OF RESULTS TO BE READ
; FROM FIFO.
;*****************************************************************************
RD_FIFO:
        MOV     R0,DATA_BLK             ;SETTING DATA BLOCK POINTER
        MOV     A,#RSLT_NUM             ;NUMBER OF RESULTS TO BE READ IN ACC
        RL      A                       ;CALCULATING # OF DATA BYTES TO BE
                                        ;READ FROM FIFO, EACH CONVERSION
                                        ;RESULTS IS 2 BYTES
        MOV     R1,A                    ;NUMBER OF DATA BYTES TO R1 COUNTER
        DEC     R1                      ;TOTAL DATA BYTES MINUS 1 IN COUNTER

        CLR     SDAS_SLCT               ;SELECT THE SDAS, CHIP SELECT=0
        CLR     S_DONE                  ;CLEAR SEND CYCLE DONE FLAG
        MOV     SBUF,#RBFIFO            ;START SENDING THE FIFO BURST READ
                                        ;CONTROL BYTE
SENDB:  JNB     S_DONE,SENDB            ;WAIT HERE UNTIL SEND CYCLE COMPLETED

        SETB    R_EN                    ;ENABLE DATA RECEIVE CYCLES

RD_LP:  CLR     R_DONE                  ;START A DATA BYTE RECEIVE CYCLE
RCVB:   JNB     R_DONE,RCVB             ;WAIT HERE UNTIL RECEIVE COMPLETED
        MOV     @R0,SBUF                ;STORE DATA BYTES IN DATA_BLK
        INC     R0                      ;POINTING TO NEXT DATA LOCATION
        DJNZ    R1,RD_LP                ;READ NEXT BYTE IF NOT THE LAST ONE

        SETB    SDAS_SLCT               ;DESELECT THE SDAS, BEFOR READING
                                        ;THE LAST BYTE, BURST READ TERMINATION
        CLR     R_DONE                  ;START A DATA BYTE RECEIVE CYCLE
RCVL:   JNB     R_DONE,RCVL             ;WAIT HERE UNTIL RECEIVE COMPLETED
        MOV     @R0,SBUF                ;STORE THE LAST DATA BYTE

        CLR     R_EN                    ;DISABLE DATA RECEIVE CYCLES
        RET
;*****************************************************************************
; THIS ROUTINE INITIALIZES THE SDAS SERIAL INTERFACE IN CASE THAT A
; COMMUNICATION CYCLE HAS BEEN INTERRUPTED. THIS ROUTINE APPLYS 24
; SERIAL CLOCK PULSES TO THE DAS WHILE ITS CHIP SELECT
; IS HIGH. THIS ROUTINE CAN BE USED AT THE START OF THE PROGRAM DURING CODE
; DEVELOPMENT OR ANYWHERE THAT A READ OR WRITE CYCLE MUST BE INTERRUPTED
; BECAUSE OF THE SYSTEM REQUIREMENT.
;*****************************************************************************
SDAS_SER_PORT_RST:
        SETB    SDAS_SLCT               ;DESELECT THE SDAS, CHIP SELECT=1
        SETB    R_EN                    ;ENABLE DATA RECEIVE CYCLES

        CLR     R_DONE                  ;START A CYCLE, 8 PULSES APPLIED
TRY1:   JNB     R_DONE,TRY1             ;WAIT HERE UNTIL CYCLE COMPLETED
        CLR     R_DONE                  ;START A CYCLE, 8 PULSES APPLIED
TRY2:   JNB     R_DONE,TRY2             ;WAIT HERE UNTIL CYCLE COMPLETED
        CLR     R_DONE                  ;START A CYCLE, 8 PULSES APPLIED
TRY3:   JNB     R_DONE,TRY3             ;WAIT HERE UNTIL CYCLE COMPLETED

        CLR     R_EN                    ;DISABLE DATA RECEIVE CYCLES
        RET
```

TL/H/11879–96

7.0 Digital Interface (Continued)

7.3 TMS320 INTERFACE MODE

The TMS320 interface mode is designed to work directly with the serial interface port of the TMS320C3x and TMS320C5x families of digital signal processors. This interface uses five lines: two data lines (DX, DR), two frame synchronization signal lines (FSX, FSR), and a serial clock line (SCLK). Note that the TMS320C3x/5x serial interface has two separate serial clock lines for transmit and receive called CLKX and CLKR, but the LM12434 and LM12{L}438 only uses one clock input for both receive and transmit. Typically, CLKX is specified as an output and drives SCLK as well as CLKR (defined as an input). The serial clock for this interface mode is a free running clock, with the data stream synchronized by SCLK. The start of each data transfer (the beginning of a data packet) is synchronized by FSX (Transmit Frame Sync) or FSR (Receive Frame Sync). This interface can communicate with one device; no device select signal is used. The following discussion assumes that the reader has a basic knowledge of the architecture and operation of the TMS320C3x/5x serial interface port.

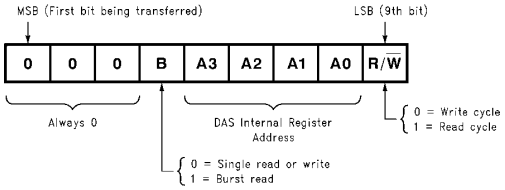
The TMS320 interface mode is selected when the MODESEL1 and MODESEL2 pins have the logic state of ''11''. *Figure 16* shows a typical connection diagram for the LM12434 and LM12{L}438 in the TMS320 serial interface mode. The FSR, FSX, DX, DR, and SCLK lines are assigned to interface pins P1 through P5.

Data transfer in this mode is programmable by the processor for 8-, 16-, 24-, or 32-bit data packets for the TMS320C3x and 8-, or 16-bit data packets for TMS320C5x. The LM12434 and LM12{L}438 uses 16-bit and 32-bit data packets. For the TMS320C5x the 32-bit packet is composed of two successive 16-bit packets with no gaps between them. The data bits in each packet are transferred MSB first, and are shifted in on the rising edge of SCLK and are stable and captured at the falling edge of the SCLK. As with the ''Standard'' and ''8051'' interface modes, the LM12434 and LM12{L}438 has three different communication cycles: write cycle, read cycle and burst read cycle. At the start of each data transfer cycle, a stream of 9 data bits (the ''command packet'') is written to the LM12434 and LM12{L}438 and informs it about the communication cycle. The placement of these 9 bits in the data packet is different in the read and write cycles and is discussed for each case separately. The command packet carries the following information:

— what type of data transfer (communication cycle) is started

— which device register is to be accessed

The command packet has the following format:



TL/H/11879–54

The first bit of the command packet is always the MSB of the data packet to to be transferred.

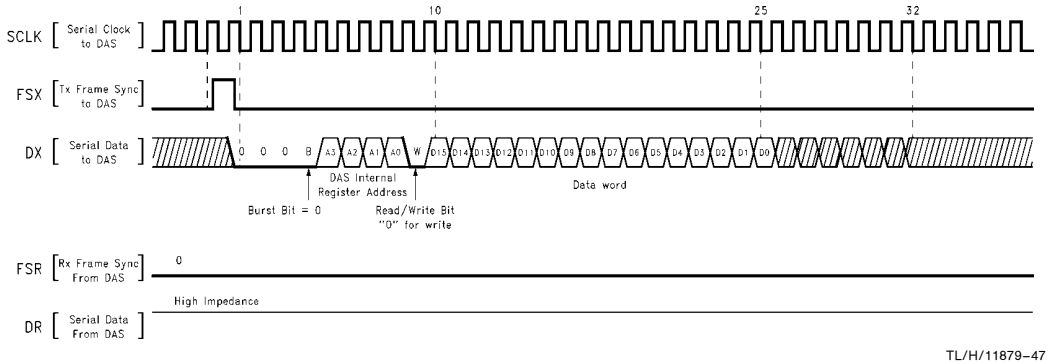
Figure 15 shows the timing diagrams for the three communication cycles. *Figure 15a* shows a write cycle. *Figure 15b* shows a read cycle, and *Figure 15c* shows a burst read cycle. Note that these timing diagrams depict general relationships between the SCLK edges, the data bits and the frame synchronization signals (FSX, FSR). These diagrams are not meant to show guaranteed timing performance. (See specification tables for parametric switching characteristics.)

Write cycle: A write cycle begins with an FSX pulse from the processor. The first data bit is received by the DAS on the DX line during the next SCLK falling edge after the falling edge of FSX. A 32-bit data packet is written to the DAS. The TMS320C3x does this with a 32-bit transfer, using its serial port 32-bit register. With the TMS320C5x family two successive 16-bit transfers are initiated without any gap in between. The first 9 bits (MSBs) of the data are the command packet with the R/$\overline{W}$ bit and B bit equal to zero. Following the command packet, a 16-bit data stream starts on the falling edge of the 10th SCLK cycle and continues through the 25th cycle. The last 7 bits in the 32-bit data packet are ''don't care'' and are ignored by the DAS. The data is written to the register addressed in the command packet (A3, A2, A1, A0). There is no activity on the FSR and DR lines during a write cycle. The write cycle is completed after the last data bit is transferred.

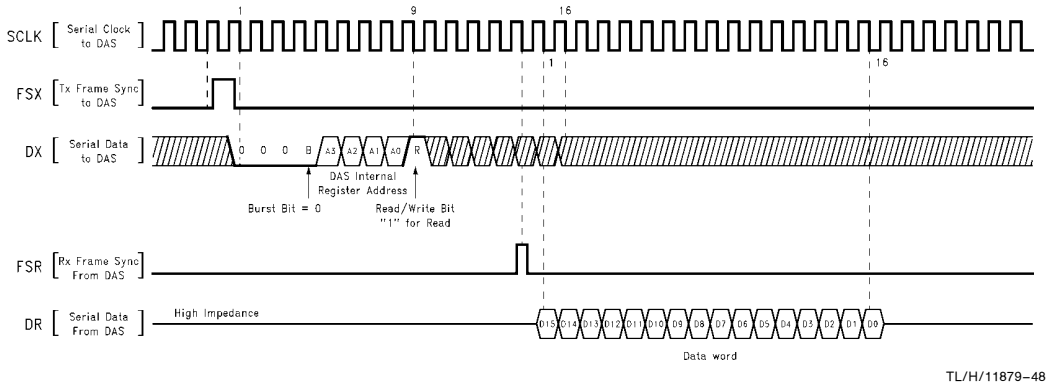
Read cycle: A read cycle also begins with an FSX pulse from the processor. The read cycle uses 16-bit data transfer. Following the FSX pulse, 16 bits of data are written to the DAS on the DX line. The first 9 bits (MSBs) of data are the command packet with the R/$\overline{W}$ bit equal to one and the B bit equal to zero. The last 7 bits (LSBs) are ''don't care'' and are ignored by the DAS. About 3 to 4 CLK (the DAS main clock input, not the SCLK) cycles after the R/$\overline{W}$ bit is received, the DAS generates an FSR pulse to initiate the data transfer. Following the FSR pulse, the DAS will send 16 bits of data to the processor on the DR line. The first bit (MSB) of the data appears on the DR line on the next SCLK cycle following the FSR pulse. The data is read from the register addressed in the command packet. The read cycle is completed after the last data bit is transferred.

7.0 Digital Interface (Continued)

Burst read cycle: A burst read cycle starts the same way as a single read cycle, but the B bit in the command packet is set to one, indicating a burst read cycle. After the first 16 bits of data carrying the command packet is written to the DAS, the DAS begins to send out the data words from the addressed register on the DR line repeatedly. Each data word is preceded by an FSR pulse for synchronization. To terminate a burst read cycle, the processor does a dummy read from the configuration register during the last data word. This dummy read should be started so that its FSR pulse occurs during the 15th to 17th SCLK cycle of the last data word as shown in *Figure 15c*. The dummy read terminates the burst read cycle and shifts out the contents of the configuration register on the DR line. This data can be discarded. After transfer of the last data bit from the configuration register, the DAS is ready for a new communication cycle to begin.



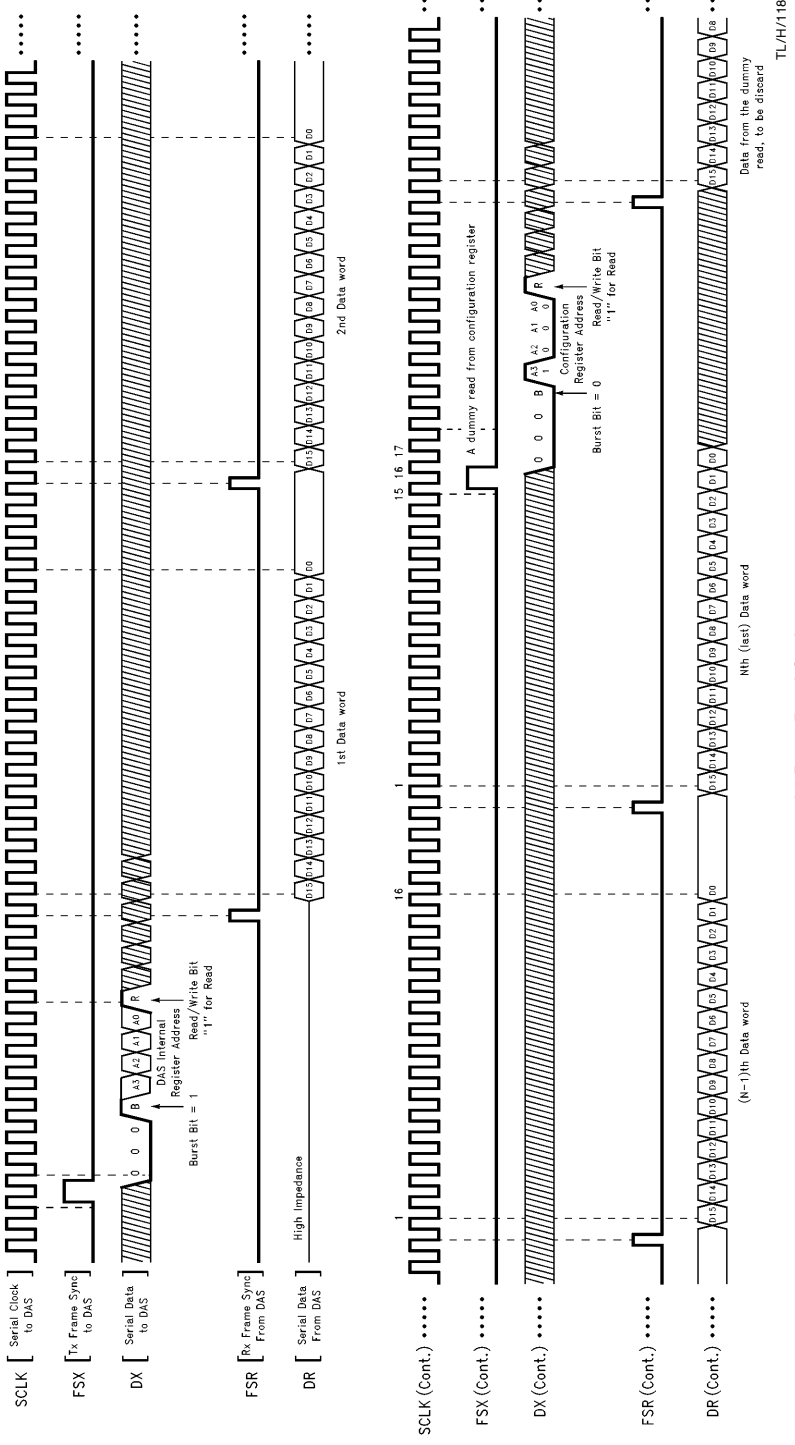
(a) Write Cycle



(b) Read Cycle

FIGURE 15. Timing Diagram for LM12434 and LM12{L}438 TMS320 Serial Interface Mode

7.0 Digital Interface (Continued)

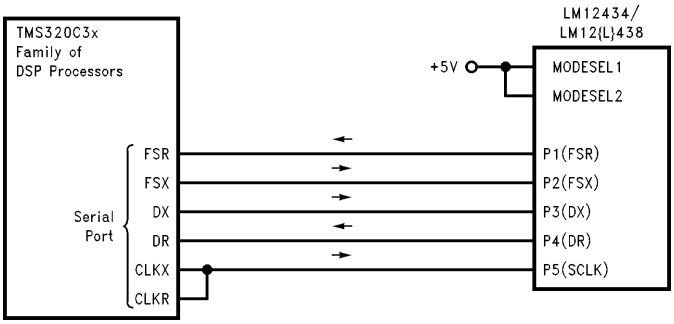


(c) Burst Read Cycle

FIGURE 15. Timing Diagram for LM12434 and LM12{L}438 TMS320 Serial Interface Mode (Continued)

7.0 Digital Interface (Continued)

7.3.1 Example of Interfacing to the TMS320C3x



TL/H/11879–75

Note: Other device pins are not shown.

FIGURE 16. LM12434 and LM12{L}438 in the TMS320 Interface Mode

TMS320C3x Assembly Code Example

```
; TMS320C3x ASSEMBLY ROUTINES FOR INTERFACE TO THE LM12434 AND LM12{L}438 SERIAL DAS
; LM12438 REGISTER ADDRESSES
CONFIG          .word 00000008H         ; CONFIGUREATION REGISTER
INTEN           .word 00000009H         ; INTERRUPT ENABLE REGISTER
INTSTAT         .word 0000000AH         ; INTERRUPT STATUS REGISTER
TIMER           .word 0000000BH         ; TIMER REGISTER
FIFO            .word 0000000CH         ; FIFO
FIFODATA        .word 00000000H
                .space 32               ; 32 reserved spaces for FIFO data
LIMIT           .word 0000000DH         ; LIMIT REGISTER

; INSTRUCTION RAM 0-8 (NOTE: CONFIG. REG. RAM POINTER SELECTS BANKS 0, 1 OR 2)

RAM0            .word 00000000H         ; INSTRUCTION RAM 0
RAM1            .word 00000001H         ; INSTRUCTION RAM 1
RAM2            .word 00000002H         ; INSTRUCTION RAM 2
RAM3            .word 00000003H         ; INSTRUCTION RAM 3
RAM4            .word 00000004H         ; INSTRUCTION RAM 4
RAM5            .word 00000005H         ; INSTRUCTION RAM 5
RAM6            .word 00000006H         ; INSTRUCTION RAM 6
RAM7            .word 00000007H         ; INSTRUCTION RAM 7
CLNDATA         .word 0000FFFFH         ; USED FOR ZEROING DON'T CARE DATA BITS

        .text
* THE PROCESSOR IS INITIALIZED. THE REMAINING APPLICATION-
* DEPENDENT PART OF THE SYSTEM (BOTH ON- AND OFF-CHIP SHOULD
* NOW BE INITIALIZED.
*
* FIRST, INITIALIZE THE CONTROL REGISTER. IN THIS EXAMPLE,
* EVERYTHING IS INITIALIZED TO ZERO SINCE THE ACTUAL INITIALIZATION
* IS APPLICATION DEPENDENT.
*
        LDI     @CTRL,AR0       ; LOAD in AR0 the pointer to control
*                               ; registers
        LDI     @DMACTL,R0
        STI     R0,*+AR0(0)     ; Init DMA control
        LDI     @TIM0CTL,R0
        STI     R0,*+AR0(32)    ; Init timer 0 control
        LDI     @TIM1CTL,R0
        STI     R0,*+AR0 (48)   ; Init timer 1 control
        LDI     @SERGLOB0,R0
        STI     R0,*+AR0(64)    ; Init serial 0 global control
        LDI     @SERPRTX0,R0
        STI     R0,*+AR0(66)    ; Init serial 0 xmt control
        LDI     @SERPRTR0,R0
        STI     R0,*+AR0(67)    ; Init serial 0 rcv control
```

TL/H/11879–92

7.0 Digital Interface (Continued)

TMS320C3x Assembly Code Example (Continued)

```
        LDI     @SERTIM0,R0
        STI     R0,*+AR0(68)    ; Init serial 0 timer control
        LDI     @SERGLOB1,R0
        STI     R0,*+AR0(80)    ; Init serial 1 global control
        LDI     @SERPRTX1,R0
        STI     R0,*+AR0(82)    ; Init serial 1 xmt control
        LDI     @SERPRTR1,R0
        STI     R0,*+AR0(83)    ; Init serial 1 rcv control
        LDI     @SERTIM1,R0
        STI     R0,*+AR0(84)    ; Init serial 1 timer control
;       LDI     @STIMCNT1,R0
;       STI     R0,*+AR0(85)    ; Init serial 1 timer counter
;       LDI     @STIMPRD1,R0
;       STI     R0,*+AR0(86)    ; Init serial 1 timer period
        LDI     @PARINT,R0
        STI     R0,*+AR0(100)   ; Init parallel interface control (c30 only)
        LDI     @IOINT,R0
        STI     R0,*+AR0(96)    ; Init I/O interface control
*
        LDI     @STCK,SP        ; Initialize the stack pointer
        OR      2000H,ST        ; Global interrupt enable
*
        BR      BEGIN           ; Branch to the beginning of application.
BEGIN   NOP
        LDI     0,IOF           ; PROGRAM XF1 PORT AS AN INPUT PORT
        LDI     @CTRL,AR0       ; LOAD in AR0 the pointer to control
        LDI     @CONFIG,R0
        LDI     0082H,R1        ; SYNC. PIN OUTPUT
        CALL    SWRITE          ; SOFT RESET LM12438

        LDI     @INTEN,R0
        LDI     0714H,R1        ; 32 CONVERSIONS
        LDI     0C714H,R1       ; 24 CONVERSIONS
        CALL    SWRITE          ; INIT. INTERRUPT ENABLE REG.

        LDI     @TIMER,R0
        LDI     0AAAAH,R1
        CALL    SWRITE          ; LOAD SOME VALUE IN TIMER

        LDI     @RAM0,R0        ; INSTRUCTIONS FOR 8 CONVERSION
        LDI     0000H,R1        ; ON EACH CHANNEL (0-7) ALL SINGLE ENDED
        CALL    SWRITE          ; SET RAM0
        LDI     @RAM1,R0
        LDI     0004H,R1
        CALL    SWRITE          ; SET RAM1
        LDI     @RAM2,R0
        LDI     0008H,R1
        CALL    SWRITE          ; SET RAM2
        LDI     @RAM3,R0
        LDI     000CH,R1
        CALL    SWRITE          ; SET RAM3
        LDI     @RAM4,R0
        LDI     0010H,R1
        CALL    SWRITE          ; SET RAM4
        LDI     @RAM5,R0
        LDI     0014H,R1
        CALL    SWRITE          ; SET RAM5
        LDI     @RAM6,R0
        LDI     0018H,R1
        CALL    SWRITE          ; SET RAM6
        LDI     @RAM7,R0
        LDI     001CH,R1
        CALL    SWRITE          ; SET RAM7

        LDI     @CONFIG,R0      ; START FULL CALIBRATION
        LDI     0088H,R1        ; SYNC. PIN OUTPUT
        CALL    SWRITE
```

TL/H/11879–93

7.0 Digital Interface (Continued)

TMS320C3x Assembly Code Example (Continued)

```
CHKINT1 TSTB    80H,IOF         ; TEST XF1 INPUT CONNECTED TO LM12438'S INTERRUPT
        BNZ     CHKINT1         ; FOR COMPLETION OF FULL CALIBRATION

        LDI     @INTSTAT,R0
        CALL    SREAD           ; READ INTERRUPT STATUS REG.

DOAGAIN LDI     @CONFIG,R0      ; IF FULL CALIBRATION IS DONE SET THE START BIT
        LDI     0081H,R1        ; OF LM12438 CONFIG. REG. (SYNC. PIN OUTPUT)
        CALL    SWRITE          ; START LM12438 SEQUENCER

CHKINT2 TSTB    80H,IOF         ; TEST XF1 INPUT CONNECTED TO LM12438'S INTERRUPT
        BNZ     CHKINT2         ; (COMPLETION OF 24 FIFO CONVERSIONS)

        LDI     @CONFIG,R0
        LDI     0080H,R1        ; STOP THE CONVERSION (IS NOT NECESSARY)
        CALL    SWRITE
        LDI     @INTSTAT,R0
        CALL    SREAD           ; READ INTERRUPT STATUS REG.
        LDI     32,R4
FLOOP   LDI     R4,R1
        LDI     @FIFO,R0        ; READ FIFO
        CALL    BREAD
        SUBB    1,R4
        BNZ     FLOOP
        LDI     @CONFIG,R0
        LDI     0002H,R1        ; RESET LM12438 (SYNC. PIN OUTPUT)
        CALL    SWRITE
        BR      DOAGAIN
        IDLE

; LM12438 BURST READ ROUTINE THROUGH SERIAL PORT1
BREAD   PUSH    ST              ; SAVE STATUS REG.
        PUSH    AR0             ; SAVE AR0
        PUSH    AR1             ; SAVE AR1
        PUSH    AR2             ; SAVE AR2
        PUSH    R0              ; SAVE R0
        PUSH    R1              ; SAVE R1
        PUSH    R2              ; SAVE R2
        PUSH    R3              ; SAVE R3
        PUSH    R4              ; SAVE R4

        LDI     @CTRL,AR0       ; LOAD in AR0 the pointer to control
        LDI     @FIFODATA,AR2   ; USE AR2 AS POINTER TO FIFO DATA
        LDI     @SERGLOB1R,R2   ; PREPARE FOR 16 BIT TRANSMIT

        CMPI    0,R1            ; IF COUNTER IS 0 (USER'S ERROR)
        BZ      BDONE2          ; TERMINATE NOW ELSE CONTINUE

        CMPI    1,R1            ; IF A SINGLE READ REQUIRED THEN
        BZ      SINGLE          ; CALL THE SINGLE READ SUBROUTINE
        BR      MLTIPLE         ; ELSE GO ON

SINGLE  LDI     @FIFO,R0        ; FOR SINGLE READ FIFO ADDRESS IS
        CALL    SREAD           ; CALLING SINGLE READ ROUTINE
        STI     R1,*AR2++(1)    ; STORE READ DATA INTO FIFODATA
        BR      BDONE2          ; TERMINATE

MLTIPLE LDI     13,R4           ; SET UP R4 AS THE DELAY COUNTER
        LDI     @SERGLOB1R,R2   ; PREPARE FOR 16 BIT TRANSMIT
        STI     R2,*+AR0(80)    ; Init serial 1 global control

        RPTS    7               ; POSITION THE ADDRESS
        ROL     R0              ; TO START AT BIT #10
        OR      1080H,R0        ; SET THE READ BIT
        LDI     R0,R3           ; R0 IS FREED FOR LAST READ

        LDI     @CONFIG,R0      ; PREPARE FOR A LAST CONFIG REG. READ
        RPTS    7               ; WHICH WILL STOP LM12438 FROM GENERATING
        ROL     R0              ; FURTHER BURST READS
        OR      80H,R0          ; SET THE READ BIT
```

TL/H/11879–94

7.0 Digital Interface (Continued)

TMS320C3x Assembly Code Example (Continued)

```
        STI     R3,*+AR0(88)    ; Init serial 1 data xmt register
        RPTS    2H              ; PROVIDE DELAY FOR UPDATE OF
        NOP                     ; XSREMPTY BIT IN GLOB CONT REG.
        LDI     1000B,R3
CHKR2   TSTB    *+AR0(80),R3    ; CHECK SER. 1 CONTROL XSREMPTY
        BNZ     CHKR2           ; IF XSREMPTY=1 THEN KEEP CHECKING

COUNT   NOP
        SUBB    1,R1            ; DECREMENT COUNTER (R1) AND CHECK FOR ZERO
        LDI     1,R3            ; PREPARE R3 FOR CHECKING RRDY BIT

RCONT3  TSTB    *+AR0(80),R3
        BZ      RCONT3          ; IF RRDY=0 THEN CHECK AGAIN

RDONE2  LDI     *+AR0(92),R3    ; LOAD DRR (RECEIVED DATA), RRDY IS CLEARED
        AND     0FFFFH,R3       ; CLEAN UP THE UPPER BITS
        STI     R3,*AR2++(1)    ; PLACE READ DATA IN FIFODATA
        CMPI    1,R1            ; IF COUNTER=1 THEN TERMINATE
        BNZ     COUNT           ; ELSE CONTINUE

BSTDONE RPTS    R4              ; WAIT FOR THE 16TH CLOCK RISE
        NOP                     ; BEFORE SENDING THE TERMINATING
                                ; READ FROM CONFIG. REG
        STI     R0,*+AR0(88)    ; XMT FOR LAST READ FROM CONFIG REG
RCONT4  LDI     *+AR0(80),R3    ; READ SER. 1 CONTROL REGISTER TO
        TSTB    0001B,R3        ; CHECK FOR RRDY BIT
        BZ      RCONT4          ; IF RRDY IS 1, EXIT THE BURST ROUTINE

RCONT5  LDI     *+AR0(92),R3    ; READ THE LAST BURST DATA IN DRR
        AND     0FFFFH,R3       ; CLEAN UP THE UPPER BITS
        STI     R3,*AR2++(1)    ; PLACE IT IN FIFODATA

RCONT6  LDI     *+AR0(80),R3    ; READ SER. 1 CONTROL REGISTER TO
        TSTB    0001B,R3          CHECK FOR RRDY BIT
        BZ      RCONT6

BDONE2  LDI     *+AR0(92),R3    ; READ THE DRR (CLEAR RRDY BIT)
        POP     R4              ; RESTORE R4
        POP     R3              ; RESTORE R3
        POP     R2              ; RESTORE R2
        POP     R1              ; RESTORE R1
        POP     R0              ; RESTORE R0
        POP     AR2             ; RESTORE AR2
        POP     AR1             ; RESTORE AR1
        POP     AR0             ; RESTORE AR0
        POP     ST              ; RESTORE ST
        RETS
```

TL/H/11879–95

7.0 Digital Interface (Continued)

TMS320C3x Assembly Code Example (Continued)

```
; LM12438 SINGLE READ ROUTINE THROUGH SERIAL PORT1
SREAD   PUSH    ST              ; SAVE STATUS REG.
        PUSH    AR0             ; SAVE AR0
        PUSH    R0              ; SAVE R0  THE READ ADDRESS
        PUSH    R2              ; SAVE R2
        LDI     @CTRL,AR0       ; LOAD in AR0 the pointer to control
        LDI     @SERGLOB1R,R2   ; PREPARE FOR 16 BIT TRANSMIT
                                ; AND 16 BIT RECIEVE
        STI     R2,*+AR0(80)    ; Init serial 1 global control
        RPTS    7               ; POSITION THE ADDRESS
        ROL     R0              ; TO START AT BIT #10
        OR      80H,R0          ; SET THE READ BIT
        STI     R0,*+AR0(88)    ; Init serial 1 data xmt register
        RPTS    2H              ; PROVIDE DELAY FOR UPDATE OF
        NOP                     ; XSREMPTY BIT IN GLOB CONT REG.
CHKR1   LDI     *+AR0(80),R0    ; READ SER. 1 CONTROL XSREMPTY
        TSTB    1000B,R0        ; CHECK
        BNZ     CHKR1           ; IF IT IS CLEAR (TRANSMIT COMPLETE) CONTINUE

RCONT1  LDI     *+AR0(80),R0    ; READ SER. 1 CONTROL
        TSTB    0001B,R0        ; CHEK RRDY BIT
        BZ      RCONT1          ; IF RRDY IS 1 (RECEIVE COMPLETE) CONTINUE

RDONE1  LDI     *+AR0(92),R1    ; LOAD DRR (RECEIVED DATA) INTO R1
        AND     0FFFFH,R1       ; CLEAN UP UPPER BITS
        POP     R2              ; RESTORE R2
        POP     R0              ; RESTORE R0    THE READ ADDRESS
        POP     AR0             ; RESTORE AR0
        POP     ST              ; RESTORE ST
        RETS
; LM12438 WRITE ROUTINE THROUGH SERIAL PORT1
SWRITE  PUSH    ST              ; SAVE STATUS REG.
        PUSH    AR0             ; SAVE AR0
        PUSH    R2              ; SAVE R2
        LDI     @CTRL,AR0       ; LOAD in AR0 the pointer to control
        LDI     @SERGLOB1W,R2   ; PREPARE FOR 32 BIT TRANSMIT
        STI     R2,*+AR0(80)    ; Init serial 1 global control
        AND     @CLNDATA,R1     ; CLEAN UP UNUSED ADD. BITS
        RPTS    23              ; POSITION THE ADDRESS TO START AT BIT #27
        ROL     R0              ;
        RPTS    6               ; POSITION DATA TO START AT BIT #22
        ROL     R1              ;
        OR      R1,R0           ;
        STI     R0,*+AR0(88)    ; Init serial 1 data xmt register
        RPTS    2H              ; PROVIDE DELAY FOR UPDATE OF
        NOP                     ; XSREMPTY BIT IN GLOB CONT REG.
CHK1    LDI     *+AR0(80),R0    ; READ SER. 1 CONTROL XSREMPTY
        TSTB    1000B,R0        ; CHECK
        BNZ     CHK1            ; IF IT IS CLEAR (TRANSMIT COMPLETE) CONTINUE

WDONE1  POP     R2              ; RESTORE R2
        POP     AR0             ; RESTORE AR0
        POP     ST              ; RESTORE ST
        RETS
```

TL/H/11879–97

7.0 Digital Interface (Continued)

7.4 I²C BUS INTERFACE

The I²C bus is a serial synchronous bus structure. It is a multi-master bus, which means that more than one device capable of controlling the bus can be connected to it. The bus uses 2 wires, serial data (SDA) and serial clock (SCL), to carry information between the devices connected to the bus. Both data and clock lines are bidirectional and are connected to the positive power supply via a pull-up resistor. Each device is identified by a unique address, whether it is a microprocessor/controller or a peripheral such as memory, keyboard, data-converter or display. Each device can operate as either transmitter or receiver, depending on the function of the device. In addition to transmitters and receivers, devices can also be considered as masters and slaves when performing data transfer. A master is the device that initiates a data transfer on the bus and generates the clock signals to permit that transfer. At that time, any device addressed is considered slave. It should be apparent that the I²C bus is not merely an interconnecting wire, it embodies comprehensive formats and procedures for addressing, transfer cycles start and stop, clock generation/synchronization and bus arbitration. The following discussion assumes that the reader is familiar with the specification and architecture of the I²C bus.

The LM12434 and LM12{L}438's I²C bus interface is selected when the MODESEL1 and MODESEL2 pins have the logic state of ''10''. *Figure 18* shows a typical connection diagram for the LM12434 and LM12{L}438 to the I²C bus. As was mentioned, communication on the I²C bus is performed on 2 lines, SCL (serial clock) and SDA (serial data); pins P5 and P4 are assigned to these lines. The DAS operates as a slave on the I²C bus. As a result, the SCL line is an input (no clock is generated by the LM12434 and LM12{L}438) and the SDA line is a bi-directional serial data path. According to I²C bus specifications, the DAS has a 7-bit slave address. The four most significant bits of the slave address are hard wired inside the LM12434 and LM12{L}438 and are ''0101''. The three least significant bits of the address are assigned to pins P3–P1. Therefore, the LM12434 and LM12{L}438 I²C slave address is:

0	1	0	1	P3	P2	P1
MSB						LSB

Tying the P3–P1 pins to different logic levels allows up to eight LM12434 and LM12{L}438's to be addressed on a single I²C bus.

Figure 17 shows the timing diagram for the read and write cycles for the LM12434 and LM12{L}438's I²C interface. This timing diagram depicts the general relationship between the serial clock edges and the data bits. It is not meant to show guaranteed timing performance. (See specification tables for parametric switching characteristics.) The DAS's I²C interface timing parameters fully meet or exceed the I²C bus specification. Data transfer on the I²C bus is byte oriented and the 16-bit data to be written to or read from each register is transferred in two bytes.

Write cycle: A write cycle is illustrated in *Figure 17a*. Communication is initiated with a start condition generated by a master (I²C bus specification), followed by a byte of the DAS's slave address with the read/write bit (8th bit) being ''0'', indicating a write cycle will follow. At the 9th SCL clock pulse of the first data packet, the DAS pulls the SDA line low (''0'') to acknowledge that it has been addressed. The next byte is the address of the DAS register to be accessed. The format of this byte is three ''0's'' (MSBs) followed by four bits of register address (MSB first as shown) and a ''0'' as the last bit (LSB). After the DAS acknowledges the address byte, the 16-bit data proceeds in two bytes, beginning with the high order byte (MSB first). The direction of the data in a write cycle is from master to DAS with acknowledgement given by the DAS at the end of each byte. The cycle is completed by a stop condition generated by the master.

Read/burst read cycle: The read and burst read cycles for the I²C interface are combined in a single format. A read cycle is shown in *Figure 17b*. A read cycle starts the same as a write with a slave address byte for write followed by a register address byte. After the register address byte is written to the DAS, the bus should be released without any stop condition. The master then applies a repeat start condition followed by the DAS's slave address, but with the read/write bit being ''1'', indicating a read request from the master. The DAS (slave) acknowledges its address and beginning with the next byte, the direction of the data will be from DAS to master. The DAS starts to transmit the contents of its register (addressed previously at second byte of the cycle) synchronized with the clocks applied by the master. An even number of data bytes should be read from the DAS (two bytes per register). At the end of each byte received from the DAS the bus master generates an acknowledge. The DAS continues to repeat transmitting its register contents as long as the master is transmitting clocks and acknowledges at the end of each byte. The DAS recognizes the end of the transfer whenever the master does not acknowledge at the end of an even numbered byte. At this point, the master should generate a stop condition as required by the I²C bus specification. Notice that the master may read only one word (single read) or as many words (two bytes each) as it needs using the read procedure.

7.0 Digital Interface (Continued)

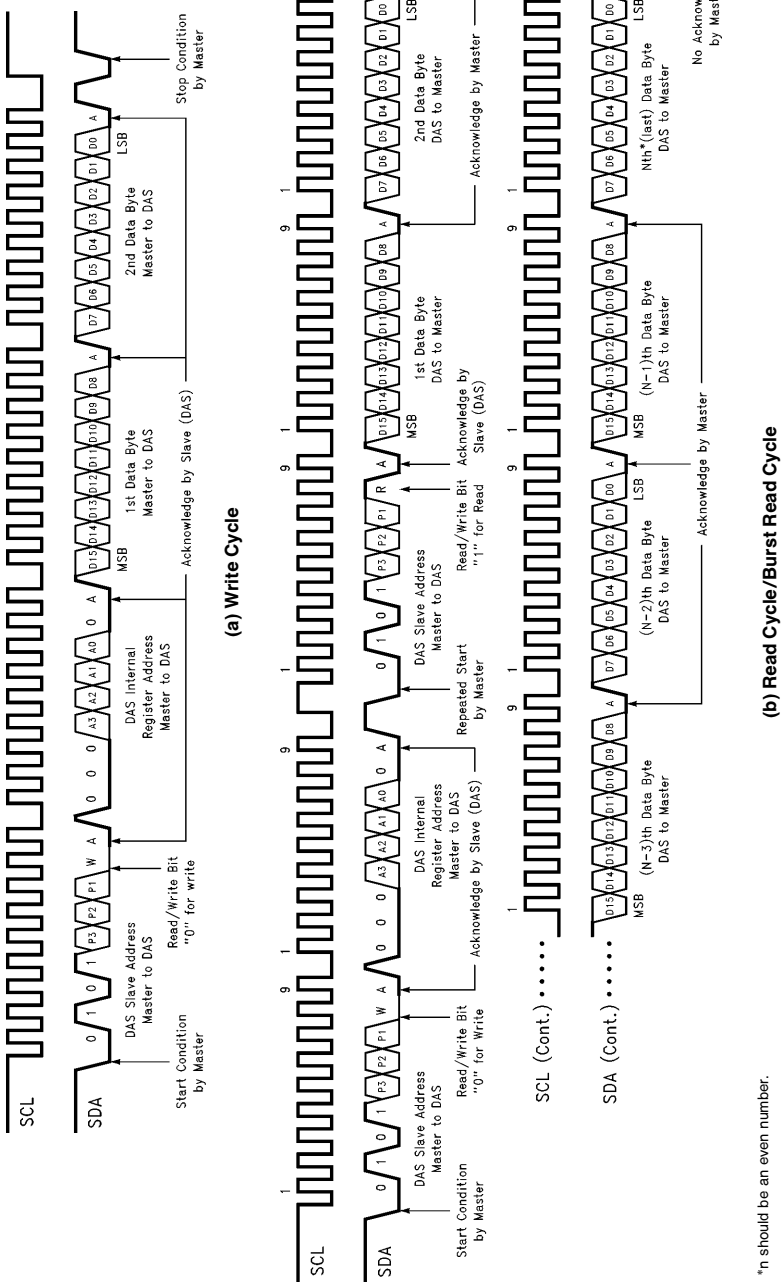
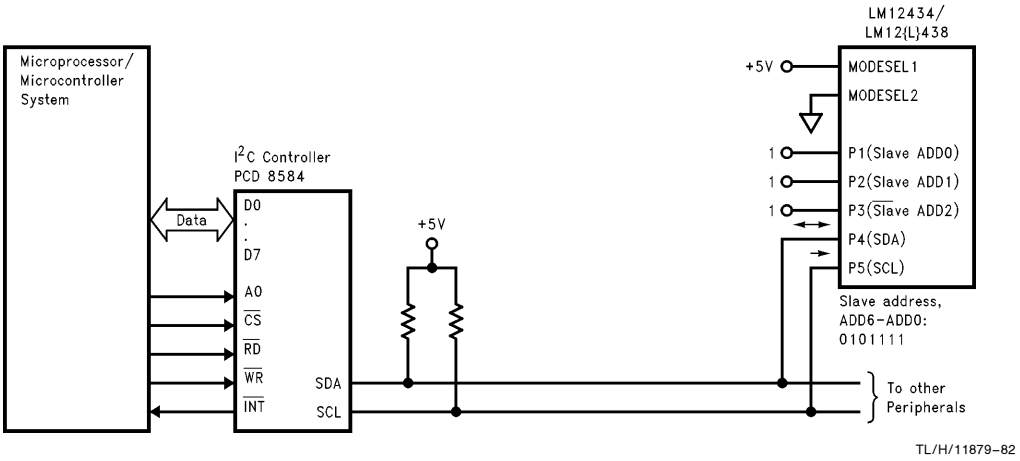


FIGURE 17. Timing Diagrams for LM12434 and LM12(L)438 I²C Interface

7.0 Digital Interface (Continued)

7.4.1 Example of Interfacing to an I²C Bus Controller (No Assembly Code)



Note: Other device pins are not shown.

FIGURE 18. Interfacing the DAS to an I²C Bus Controller

8.0 Analog Considerations

8.1 REFERENCE VOLTAGE

The difference between the voltages applied to the V_{REF+} and V_{REF-} is the analog input voltage span (the difference between the voltages applied across two multiplexer inputs or the voltage applied to one of the multiplexer inputs and analog ground, over which 4095 positive and 4096 negative codes exist). The voltage sources driving V_{REF+} or V_{REF-} must have very low output impedance and noise. The circuit in *Figure 19* is an example of a very stable reference appropriate for use with the LM12434 and LM12{L}438.

The ADC can be used in either ratiometric or absolute reference applications. In ratiometric systems, the analog input voltage is proportional to the voltage used for the ADC's reference voltage. When this voltage is the system power supply, the V_{REF+} pin is connected to V_A^+ and V_{REF-} is connected to GND. This technique relaxes the system reference stability requirements because the analog input voltage and the ADC reference voltage move together. This maintains the same output code for given input conditions.

For absolute accuracy, where the analog input voltage varies between very specific voltage limits, a time and temperature stable voltage source can be connected to the reference inputs. Typically, the reference voltage's magnitude will require an initial adjustment to null reference voltage induced full-scale errors.

8.2 INPUT RANGE

The LM12434 and LM12{L}438's fully differential ADC and reference voltage inputs generate a two's-complement output that is found by using the equation below.

$$\text{output code} = \frac{V_{IN+} - V_{IN-}}{V_{REF+} - V_{REF-}} (4096) - ½ \qquad \text{(12-bit)}$$

$$\text{output code} = \frac{V_{IN+} - V_{IN-}}{V_{REF+} - V_{REF-}} (256) - ½ \qquad \text{(8-bit)}$$

Round up to the next integer value between −4096 to 4095 for 12-bit resolution and between −256 to 255 for 8-bit resolution if the result of the above equation is not a whole number. As an example, V_{REF+} = 2.5V, V_{REF-} = 1V, V_{IN+} = 1.5V and V_{IN-} = GND. The 12-bit + sign output code is positive full-scale, or 0,1111,1111,1111. If V_{REF+} = 5V, V_{REF-} = 1V, V_{IN+} = 3V, and V_{IN-} = GND, the 12-bit + sign output code is 0,1100,0000,0000.

8.3 INPUT CURRENT

A charging current flows into or out of (depending on the input voltage polarity) the analog input pins, IN0–IN7 at the start of the analog input acquisition time (t_{ACQ}). This current's peak value will depend on the actual input voltage applied.

8.4 INPUT SOURCE RESISTANCE

For low impedance voltage sources (<60Ω for 8 MHz operation), the input charging current will decay, before the end of the S/H's acquisition time, to a value that will not introduce any conversion errors. For higher source impedances, the S/H's acquisition time can be increased. As an example, operating with a 8 MHz clock frequency and maximum acquisition time, the LM12434 and LM12438's analog inputs can handle source impedances as high as 4.17 kΩ. Refer to Section 6.2.1, Instruction RAM "00", Bits 12–15 for further information.

8.5 INPUT BYPASS CAPACITANCE

External capacitors (0.01 μF–0.1 μF) can be connected between the analog input pins, IN0–IN7, and analog ground to filter any noise caused by inductive pickup associated with long input leads. These capacitors will not degrade the conversion accuracy.

8.6 INPUT NOISE

The leads to each of the analog multiplexer input pins should be kept as short as possible. This will minimize input noise and clock frequency coupling that can cause conversion errors. Input filtering can be used to reduce the effects of the noise sources.

8.7 POWER SUPPLY CONSIDERATIONS

Decoupling and bypassing the power supply on a high resolution ADC is an important design task. Noise spikes on the V_A+ (analog supply) or V_D+ (digital supply) can cause conversion errors. The analog comparator used in the ADC will respond to power supply noise and will make erroneous conversion decisions. The DAS is especially sensitive to power supply spikes that occur during the auto-zero or linearity calibration cycles.

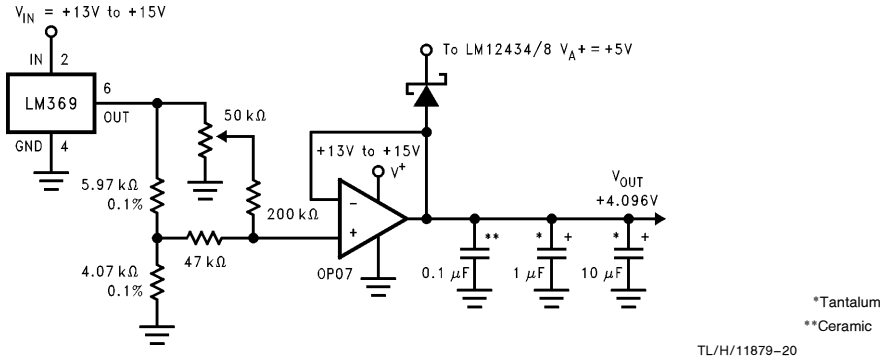


FIGURE 19. Low Drift Extremely Stable Reference Circuit

8.0 Analog Considerations (Continued)

The LM12434/8 is designed to operate from a single +5V power supply. The LM12{L}438 is designed to operate from a single +3.3V supply. The separate supply and ground pins for the analog and digital portions of the circuit allow separate external bypassing. To minimize power supply noise and ripple adequate bypass capacitors should be placed directly between power supply pins and their associated grounds. Both supply pins are generally connected to the same supply source. In systems with separate analog and digital supplies, the DAS should be powered from the analog supply. At least a 10 μF tantalum electrolytic capacitor in parallel with a 0.1 μF monolithic ceramic capacitor is recommended for bypassing each power supply. The key consideration for these capacitors is to have the low series resistance and inductance. The capacitors should be placed as close as physically possible to the supply and ground pins with the smaller capacitor closer to the device. The capacitors also should have the shortest possible leads in order to minimize series lead inductance. Surface mount chip capacitors are optimal in this respect and should be used when possible.

When the power supply regulator is not local on the board, adequate bypassing (a high value electrolytic capacitor) should be placed at the power entry point. The value of the capacitor depends on the total supply current of the circuits on the PC board. All supply currents should be supplied by the capacitor instead of being drawn from the external supply lines, while the external supply charges the capacitor at a steady rate.

The DAS has two V_D+ and DGND pins on two sides of its package. It is recommended to use a 0.1 μF plus a 10 μF capacitor between pins 15 and 16 (V_D+) and 14 (DGND) and a 0.1 μF capacitor between pins 28 (V_D+) and 1 (DGND) for the PLCC package. The respective pins for the SO package are 21 and 22 (V_D+) and 20 (DGND), 6 (V_D+) and 7 (DGND). The layout diagrams in Section 8.8 show the recommended placement for the supply bypass capacitors.

8.8 PC BOARD LAYOUT AND GROUNDING CONSIDERATIONS

To get the best possible performance from the LM12434 and LM12{L}438, the printed circuit boards should have separate analog and digital ground planes. The reason for using two ground planes is to prevent digital and analog ground currents from sharing the same path until they reach a very low impedance power supply point. This will prevent noisy digital switching currents from being injected into the analog ground.

Figure 20 illustrates a favorable layout for ground planes, power supply and reference input bypass capacitors. *Figure 20a* shows a layout using a 28-pin PLCC socket and through-hole assembly. *Figure 20b* shows a surface mount layout for the same 28-pin PLCC package. A similar approach should be used for the SO package.

The analog ground plane should encompass the area under the analog pins and any other analog components such as the reference circuit, input amplifiers, signal conditioning circuits, and analog signal traces.

The digital ground plane should encompass the area under the digital circuits and the digital input/output pins of the DAS. Having a continuous digital ground plane under the data and clock traces is very important. This reduces the overshoot/undershoot and high frequency ringing on these lines that can be capacitively coupled to analog circuitry sections through stray capacitances.

The AGND and DGND in the LM12434 and LM12{L}438 are not internally connected together. They should be connected together on the PC board right at the chip. This will provide the shortest return path for the signals being exchanged between the internal analog and digital sections of the DAS.

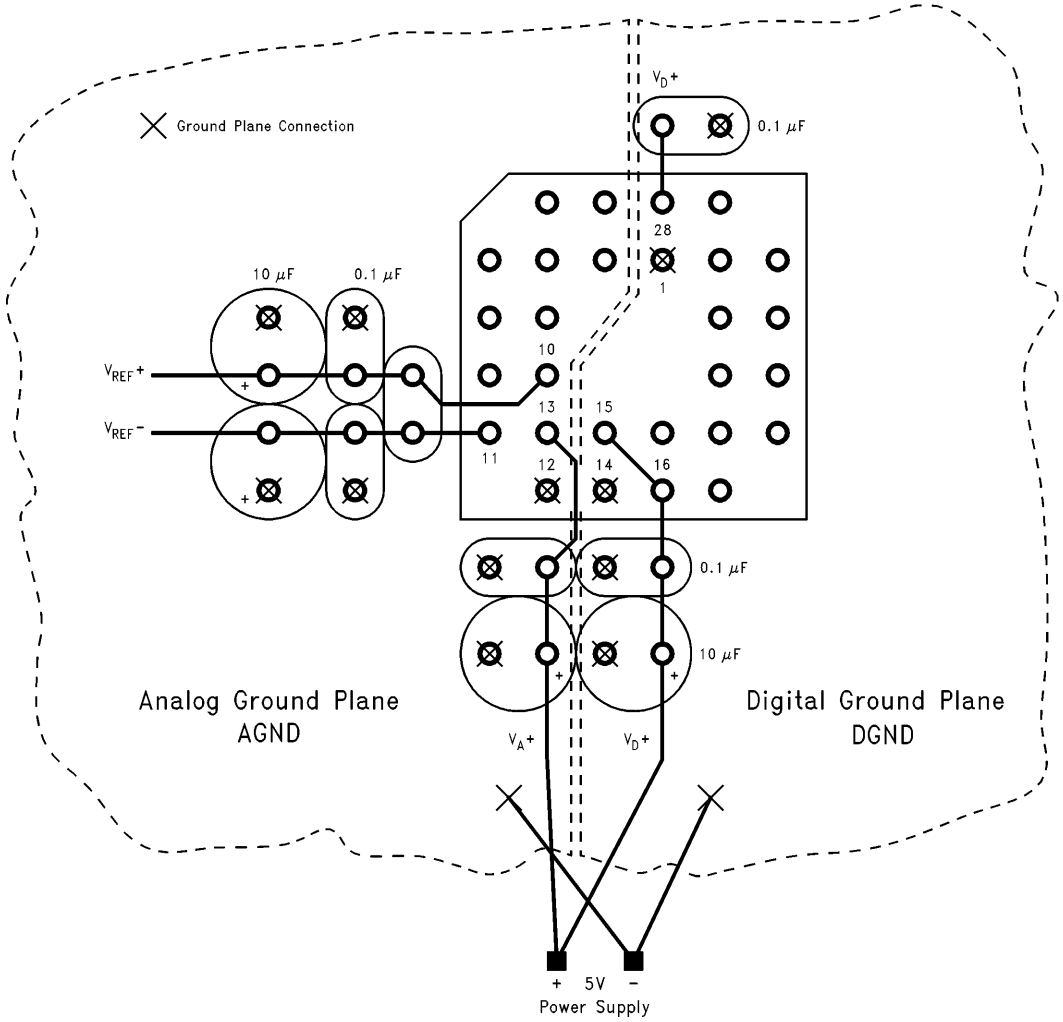
It is also a good design practice to have power plane layers in the PC board. This will improve the supply bypassing (an effective distributed capacitance between power and ground plane layers) and voltage drops on the supply lines. However, power planes are not essential as ground planes are for the performance of the DAS. If power planes are used, they should be separated into two planes and the area and connections should follow the same guidelines as mentioned for the ground planes. Each power plane should be laid out over its associated ground planes, avoiding any overlap between power and ground planes of different types. When the power planes are not used, it is recommended to use separate supply traces for the V_A+ and V_D+ pins from a low impedance supply point (the regulator output or the power entry point to the PC board). This will help ensure that the noisy digital supply does not corrupt the analog supply.

When measuring AC input signals with the DAS, any cross-talk between analog input/output lines and the reference lines (IN0–IN7, MUXOUT±, S/H IN±, V_{REF}±) should be minimized. Cross talk is minimized by reducing any stray capacitance between the lines. This can be done by increasing the clearance between traces, keeping the traces as short as possible, shielding traces from each other by placing them on different sides of the AGND plane, or running AGND traces between them.

Figure 20 also shows the reference input bypass capacitors. Here the reference inputs are considered to be differential. The performance of the DAS improves by having a 0.1 μF capacitor between the $V_{REF}+$ and $V_{REF}-$, and by bypassing in a manner similar to that described in Section 8.7 for the supply pins. When a single ended reference is used, $V_{REF}-$ is connected to AGND and only two capacitors are used between $V_{REF}+$ and $V_{REF}-$ (0.1 μF + 10 μF). It is recommended to directly connect the AGND side of these capacitors to the $V_{REF}-$ instead of connecting $V_{REF}-$ and the ground sides of the capacitors separately to the ground planes. This provides a significantly lower-impedance connection when using surface mount technology.

Figure 21 is intended to give a general idea of how the DAS should be wired and interfaced to a μC that operates in the Standard Interface mode. All necessary analog and digital power supply and voltage reference bypass capacitors are shown. A voltage reference of 4.096V generated by the LM4040-4.1 is connected to the $V_{REF}+$ of the DAS and the $V_{REF}-$ is connected to analog ground. The serial interface pins P1 through P5 of the DAS are connected to the μC's serial control lines and the interrupt pin of the DAS is wired directly to the interrupt of the μC. In this diagram the DAS runs on a separate clock than the μC, however, in some applications the DAS analog clock (CLK) may be a derivative of the μC's clock.

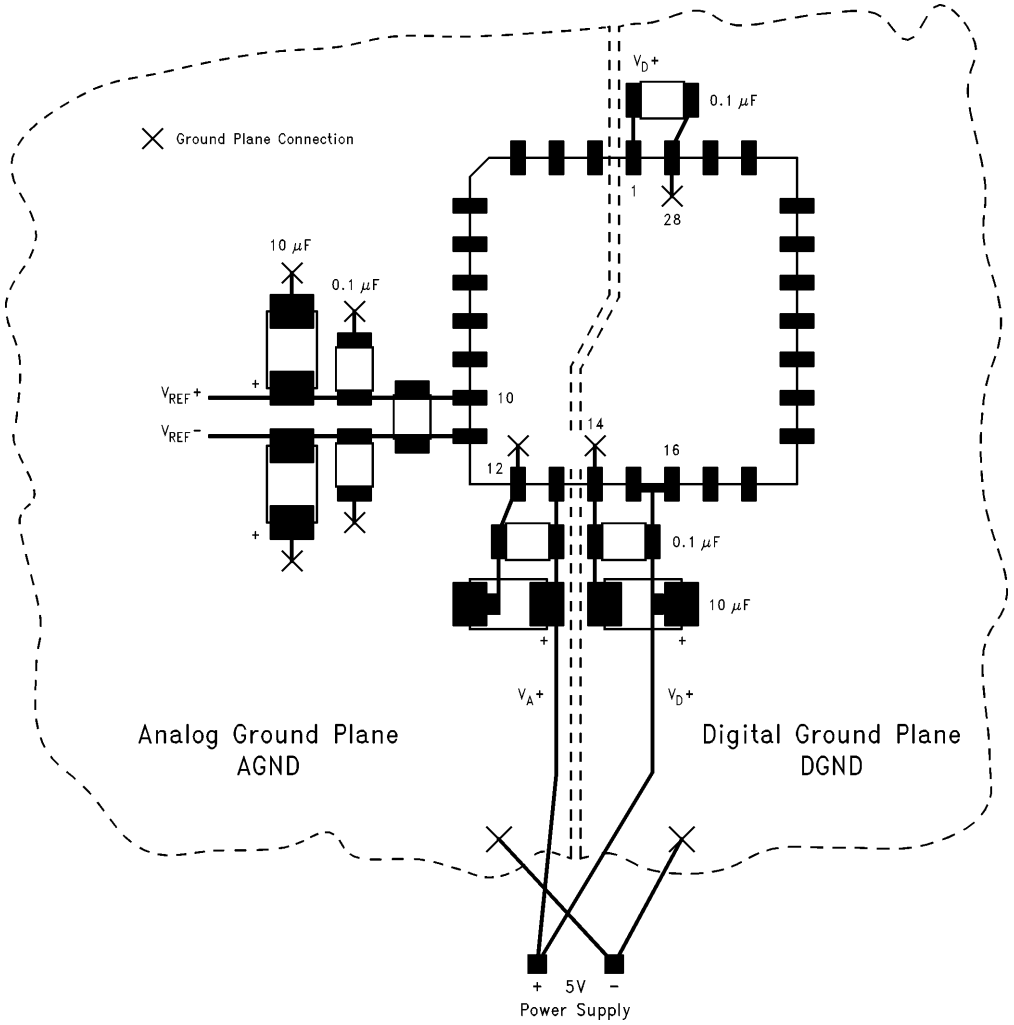
8.0 Analog Considerations (Continued)



(a) Through Hole Technology with 28-Pin PLCC Socket

FIGURE 20. Printed Circuit Board Layout for LM12434 and LM12{L}438

8.0 Analog Considerations (Continued)



(b) Surface Mount Technology for 28-Pin PLCC Package

FIGURE 20. Printed Circuit Board Layout for LM12434 and LM12{L}438 (Continued)

8.0 Analog Considerations (Continued)

Microcontroller (Standard Interface Mode)

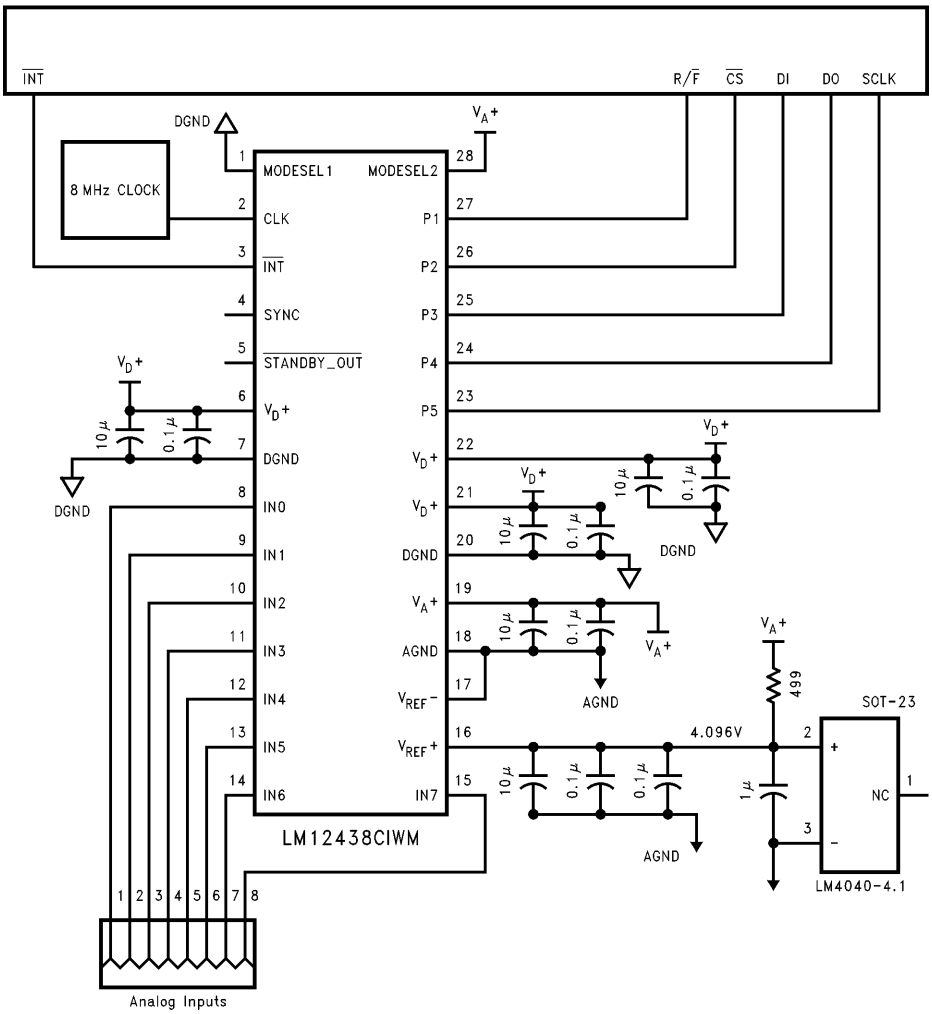
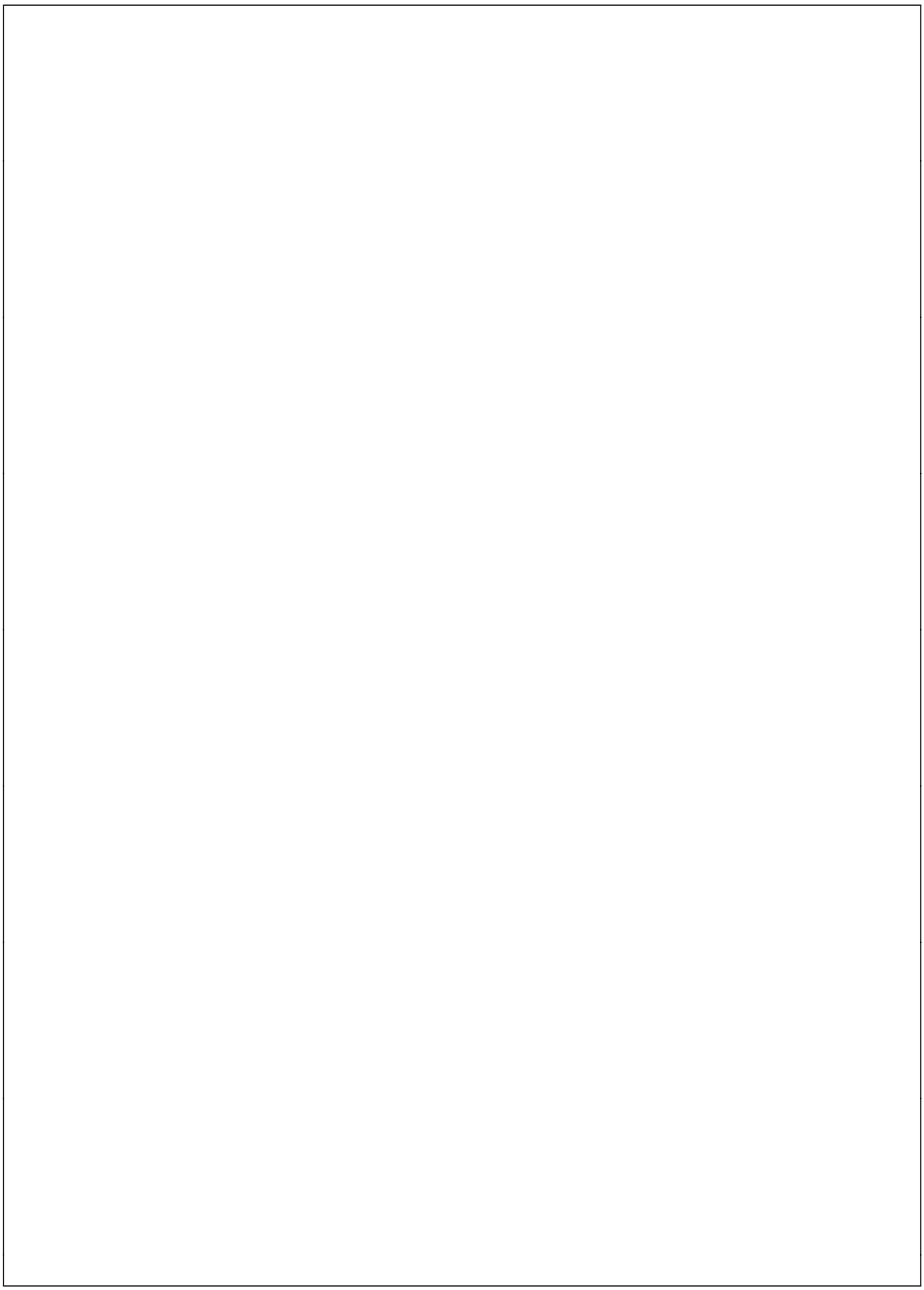
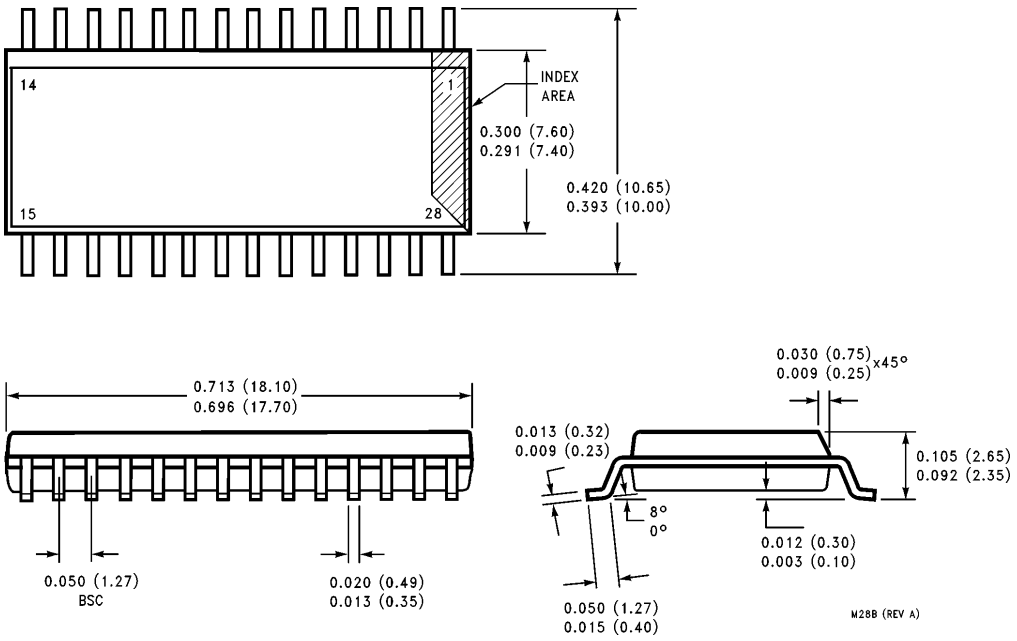


FIGURE 21. General Schematic of the DAS Operating in Standard Interface Mode

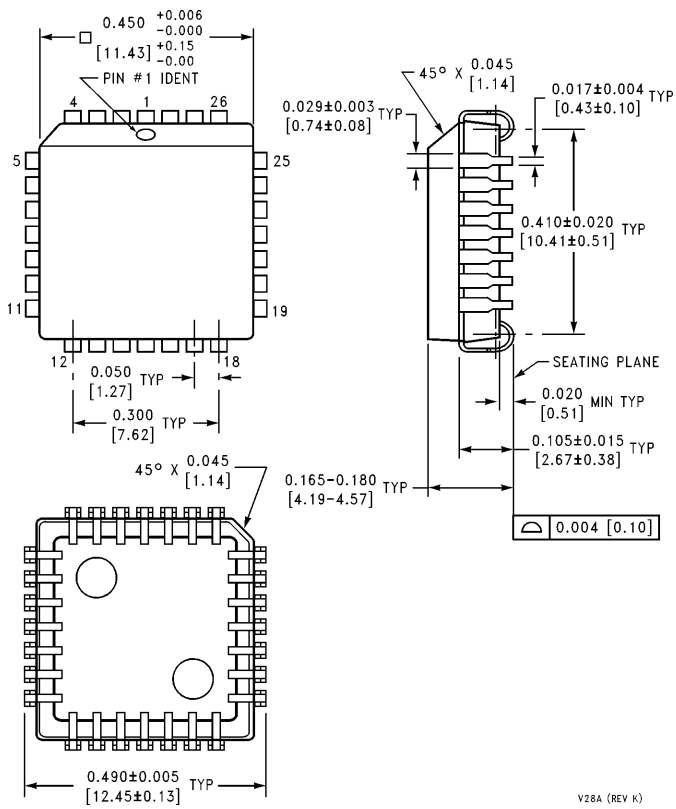


Physical Dimensions inches (millimeters)



Order Number LM12434CIWM, LM12438CIWM or LM12L438CIWM
NS Package Number M28B

Physical Dimensions inches (millimeters) (Continued)



Order Number LM12434CIV, LM12438CIV or LM12L438CIV
NS Package Number V28A

LIFE SUPPORT POLICY

NATIONAL'S PRODUCTS ARE NOT AUTHORIZED FOR USE AS CRITICAL COMPONENTS IN LIFE SUPPORT DEVICES OR SYSTEMS WITHOUT THE EXPRESS WRITTEN APPROVAL OF THE PRESIDENT OF NATIONAL SEMICONDUCTOR CORPORATION. As used herein:

1. Life support devices or systems are devices or systems which, (a) are intended for surgical implant into the body, or (b) support or sustain life, and whose failure to perform, when properly used in accordance with instructions for use provided in the labeling, can be reasonably expected to result in a significant injury to the user.
2. A critical component is any component of a life support device or system whose failure to perform can be reasonably expected to cause the failure of the life support device or system, or to affect its safety or effectiveness.

National Semiconductor Corporation
2900 Semiconductor Drive
P.O. Box 58090
Santa Clara, CA 95052-8090
Tel: 1(800) 272-9959
TWX: (910) 339-9240

National Semiconductor GmbH
Livry-Gargan-Str. 10
D-82256 Fürstenfeldbruck
Germany
Tel: (81-41) 35-0
Telex: 527649
Fax: (81-41) 35-1

National Semiconductor Japan Ltd.
Sumitomo Chemical
Engineering Center
Bldg. 7F
1-7-1, Nakase, Mihama-Ku
Chiba-City,
Ciba Prefecture 261
Tel: (043) 299-2300
Fax: (043) 299-2500

National Semiconductor Hong Kong Ltd.
13th Floor, Straight Block,
Ocean Centre, 5 Canton Rd.
Tsimshatsui, Kowloon
Hong Kong
Tel: (852) 2737-1600
Fax: (852) 2736-9960

National Semiconductores Do Brazil Ltda.
Rue Deputado Lacorda Franco
120-3A
Sao Paulo-SP
Brazil 05418-000
Tel: (55-11) 212-5066
Telex: 391-1131931 NSBR BR
Fax: (55-11) 212-1181

National Semiconductor (Australia) Pty, Ltd.
Building 16
Business Park Drive
Monash Business Park
Nottinghill, Melbourne
Victoria 3168 Australia
Tel: (3) 558-9999
Fax: (3) 558-9998