ispPAC81

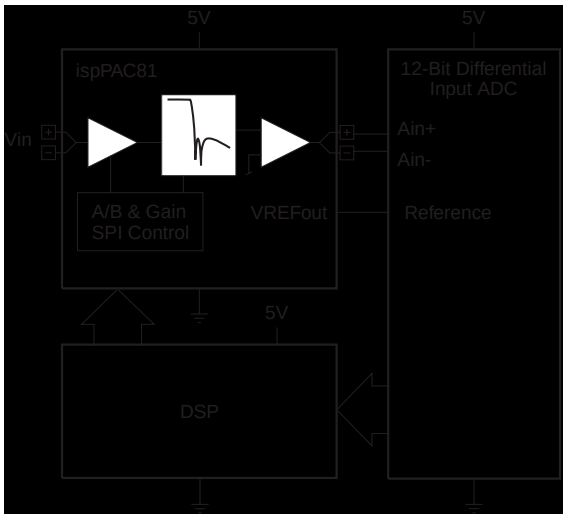
In-System Programmable Analog Circuit

October 2001 **Data Sheet**

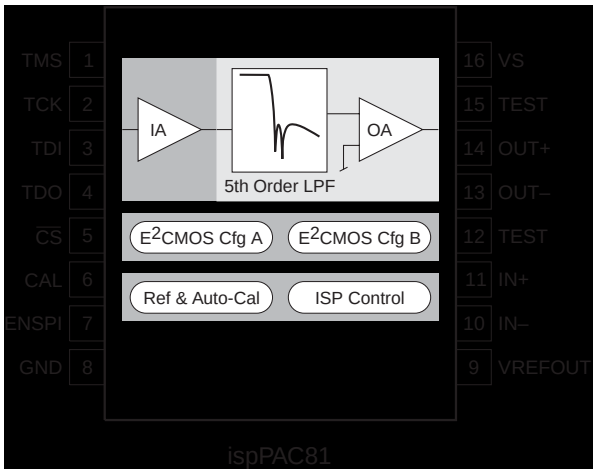
Features

- **In-System Programmable(ISP™) Analog**
  - Instrument Amplifier Gain Stage
  - Precision Active Filtering (10kHz to 75kHz)
  - Continuous-Time Fifth Order Low Pass Topology
  - Dual, A/B Configuration Memory
  - Non-Volatile $E^2CMOS^{®}$ Cells
  - IEEE 1149.1 JTAG Serial Port Programming
- **Unique Flexibility and Performance**
  - Programmable Gain Range (0dB to 20dB)
  - Implements Multiple Filter Types: Elliptical, Chebyshev, Butterworth
  - Low Distortion (THD < -80dB at 10kHz)
  - Auto-Calibrated Input Offset Voltage
- **True Differential I/O**
  - High CMR Instrument Amplifier Input
  - 2.5V Common Mode Reference on Chip
  - Rail-to-Rail Voltage Outputs
- **Single Supply 5V Operation**
  - Power Dissipation of 133mW
  - 16-Pin Plastic SOIC, PDIP Packages
- **Applications Include Integrated**
  - Single +5V Supply Signal Conditioning
  - Programmable Filters With Fully Differential I/O
  - Analog Front Ends, 12-Bit Data Acq. Systems
  - DSP System Front End Signal Conditioning
  - High-Performance Reconstruction Filters

Typical Application Diagram



Functional Block Diagram



Description

The ispPAC81 is a member of the Lattice family of In-System Programmable analog circuits, digitally configured via nonvolatile E^2CMOS technology.

Analog building blocks, called PACell™(s), replace traditional analog components such as opamps, eliminating the need for external resistors and capacitors. With no requirement for external configuration components, ispPAC81 expedites the design process, simplifying prototype circuit implementation and change, while providing high-performance integrated functionality. With all components on chip, there is no longer a concern of performance degradation due to component mismatch or other external factors. The ispPAC81 provides reliable and repeatable performance, every time.

Designers configure the ispPAC81 and verify its performance using PAC-Designer®, an easy-to-use, Microsoft Windows® compatible program. A filter configuration database is provided whereby thousands of different configurations can be realized. No special understanding of filter synthesis is required beyond that of general specifications such as corner frequency and stopband attenuation, etc. The software lists the possible choices that meet the designer's specifications which can then be loaded directly into either of two device (A/B) configurations from the lookup table. Device programming is supported using PC parallel port I/O operations.

The ispPAC81 is configured through its IEEE Standard 1149.1 compliant serial port. The flexible In-System Programming capability enables programming, verification and reconfiguration, if desired, directly on the printed circuit board.

DC Electrical Characteristics

T_A = 25°C; V_S = 5.0V; 1V < V_{OUT} < 4V; Gain = 1; Output load = 200pf, 1MΩ. Filter configuration = CC055042, F_P = 17.62kHz; Auto-Cal initiated immediately prior. (Unless otherwise specified).

Symbol	Parameter	Condition	Min.	Typ.	Max.	Units
Analog Input						
$V_{IN\pm}$ (1)	Input Voltage Range	Applied to either V_{IN+} or V_{IN-}	1		4	V
$V_{IN-DIFF}$	Differential Input Voltage Swing (2)	2 $\lvert V_{IN+} - V_{IN-} \rvert$	6			V_{P-P}
V_{OS} (2)	Differential Offset Voltage (Input Referred)	G=10 G = 1		250 2.5	1,000 10	µV mV
$\Delta V_{OS} / \Delta T$	Differential Offset Voltage Drift	-40 to +85°C		100		µV/°C
R_{IN}	Input Resistance			10^9		Ω
C_{IN}	Input Capacitance			2		pF
I_B	Input Bias Current	at DC		1		pA
e_N	Input Noise Voltage Density	At 10kHz, referred to input, G=10		71		$nV/\sqrt{Hz}$
Analog Output						
$V_{OUT\pm}$	Output Voltage Range	Present at either V_{OUT+} or V_{OUT-}	0.1		4.9	V
$V_{OUT-DIFF}$	Differential Output Voltage Swing (2)	2 $\lvert V_{OUT+} - V_{OUT-} \rvert$	9.6			V_{P-P}
$I_{OUT\pm}$	Output Current	Source/Sink	10			mA
V_{CM}	Common Mode Output Voltage	$(V_{OUT+} - V_{OUT-})/2$	2.495	2.5	2.505	V
Static Performance						
G	Programmable Gain Range	Individual gain amplifier (1, 2, 5, 10)	0		20	dB
	Gain Error	R_L = 300Ω differential		0.5	2.5	%
ΔG/ΔT	Gain Drift	-40 to +85°C		20		ppm/°C
PSR	Power Supply Rejection	Differential at 1kHz Single-ended at 1kHz		80 67		dB dB
Common Mode Reference Output ($VREF_{OUT}$)						
$VREF_{OUT}$	Reference Output Voltage Range	Nominally 2.500V	-0.2		0.2	%
	Reference Output Voltage Drift	-40 to +85°C		50		ppm/°C
$IREF_{OUT}$	Reference Output Current	($VREF_{OUT}$ = ±1%) source ($VREF_{OUT}$ = ±1%) sink		50 -350		µA µA
	Reference Output Noise Voltage	10MHz bandwidth		40		μV_{RMS}
	Reference Power Supply Rejection	1kHz		80		dB
Programming						
	Erase/Reprogram Cycles		10K	1M		cycles
Digital I/O						
V_{IL}	Input Low Voltage		0		0.8	V
V_{IH}	Input High Voltage		2		V_S	V
I_{IL}, I_{IH}	Input Leakage Current	0V ≤ TCK, ENSPI, CAL input ≤ VS 0V ≤ TDI, MTS, $\overline{CS}$ inputs ≤ VS			-10/+40 -70/+10	µA µA
V_{OL}	Output Low Voltage (TDO)	I_{OL} = 4.0mA			0.5	V
V_{OH}	Output High Voltage (TDO)	I_{OH} = -1.0mA	2.4			V

AC Electrical Characteristics

Symbol	Parameter	Condition	Min.	Typ.	Max.	Units
Dynamic Performance (4)						
SNR	Signal to Noise (G=1 to 10)	0.1Hz to 75kHz, F_C = 50kHz		86		dB
THD	Total Harmonic Distortion (Differential) Single-Ended (F_P = 50kHz)	F_{IN} = 10kHz, V_{IN} = $6V_{P-P}$ F_{IN} = 10kHz, V_{IN} = $6V_{P-P}$		-80 -80	-72 -72	dB dB
CMR	Common Mode Rejection (V_{IN} = 1V to 4V) Note: V_{IN+} and V_{IN-} connected together	10kHz, F_C = 50kHz	48	60		dB
Filter Characteristics (4)						
$\|F_C\|$	Absolute Corner Frequency Accuracy	Deviation from calculated -3db point F_C = 10kHz or 50kHz		0.8		%
ΔF_C	Max Delta Between Filters (Computed)	Butterworth, 10kHz to 60kHz		0.6	3.6	%
$\Delta F_C/\Delta T$	Corner Frequency Delta vs. Temperature	F_C = 10kHz		0.03		%/°C
$\Delta F_C/\Delta V$	Corner Frequency Delta vs. Supply Voltage	F_C = 10kHz		0.09		%/°V
Elliptic Filter Response (5)						
	Passband Ripple Passband Ripple	F_C = 10kHz F_C = 50kHz		1.4 1.5		dB dB
Power Supplies						
V_S	Operating Supply Voltage		4.75	5	5.25	V
I_S	Supply Current	V_S = 5.0V		26.5	33	mA
P_D	Power Dissipation	V_S = 5.0V		133	165	mW
Temperature Range						
	Operation		-40		85	°C
	Storage		-65		150	°C

Notes: (1) A wider input range of 0.7V to 4.3V is typical, but not guaranteed. Inputs larger than this will be clipped. Input signals are also subject to common-mode voltage limitations. Refer to the table of conditions in this data sheet. (2) Refer to theory of operation section later in this data sheet for explanation of differential voltage swing computation. (3) To insure full spec performance an additional auto-calibration should be performed after initial turn-on and the device reaches thermal stability. (4) Although many hundreds of thousands of filter configurations are available using ispPAC81, not every type will have corner frequencies available from exactly 10kHz to 75kHz, depending on the tables available from within PAC-Designer filter design tools. The general specifications given under this heading are realized using the Elliptic filter types. For more information on other types and/or frequencies not contained in the filter database, please contact Lattice Marketing. (5) A Cauer elliptic filter of type CC051042 (see data sheet text) is used to guarantee these specific filter accuracy specifications. It is assumed that all other configurations available in PAC-Designer will exhibit equivalent performance according to the applicability of the individual filter type. Necessary limitations will apply, however, when specifications do not directly apply. See the data sheet text, application notes and guides in PAC-Designer for specific filter type considerations.

Pin Descriptions

Pin(s)	Symbol	Name	Description
1	TMS	Test Mode Select	Serial interface logic mode select pin (input). JTAG interface mode only.
2	TCK	Test Clock	Serial interface logic clock pin (input). JTAG and SPI interface modes.
3	TDI	Test Data In	Serial interface logic pin (input) for both JTAG and SPI operation modes.
4	TDO	Test Data Out	Serial interface logic pin (output) for both JTAG and SPI operation modes. Input data valid on falling edge of TCK (JTAG), or on rising edge of CS (SPI).
5	$\overline{CS}$	Chip Select	Chip select logic input pin. SPI data latch.
6	CAL	Auto-Calibrate	Digital pin (input). Commands an auto-calibration sequence on a rising edge.
7	ENSPI	Enable SPI Mode	Enable SPI logic input pin. When high, causes serial port to run in SPI mode.
8	GND	Ground	Ground pin. Should normally be connected to the analog ground plane.
9	VREFOUT	Common-Mode Reference	Common-mode voltage reference output pin (+2.5V nominal). Must be bypassed to GND with a 1µF capacitor.
10, 11	IN	Inputs (+ or -)	Differential input pins, using two pins (e.g., IN+ and IN-). Plus or minus components of V_{IN}, where differential $V_{IN} = V_{IN+} - V_{IN-}$.
12, 15	TEST	Test Pin	Test pin. Connect to GND for proper circuit operation.
13, 14	OUT	Outputs (+ or -)	Differential output pins, using two pins (e.g., OUT+ and OUT-). Complementary with respect to VREFOUT. Differential $V_{OUT} = V_{OUT+} - V_{OUT-}$.
16	VS	Supply Voltage	Analog supply voltage pin (5V nominal). Should be bypassed to GND with 1µF and .01µF capacitors.

Connection Notes

1. All inputs and outputs are labeled with plus (+) and minus (-) signs. Polarity is labeled for reference and can be selected externally by reversing pin connections.
2. All analog output pins are “hard-wired” to internal output devices and should be left open if not used. V_{OUT+} and V_{OUT-} should not be tied together as unnecessary power will be dissipated.
3. When the signal input is single-ended, the other half of the unused differential input must be connected to a DC common-mode reference (usually $VREF_{OUT}$, 2.5V).

Absolute Maximum Ratings

Supply Voltage VS . . . . . .-0.5 to +7V

Logic and Analog Input Voltage Applied. . . . . .0 to VS

Logic and Analog Output Short Circuit Duration . . . . . Indefinite

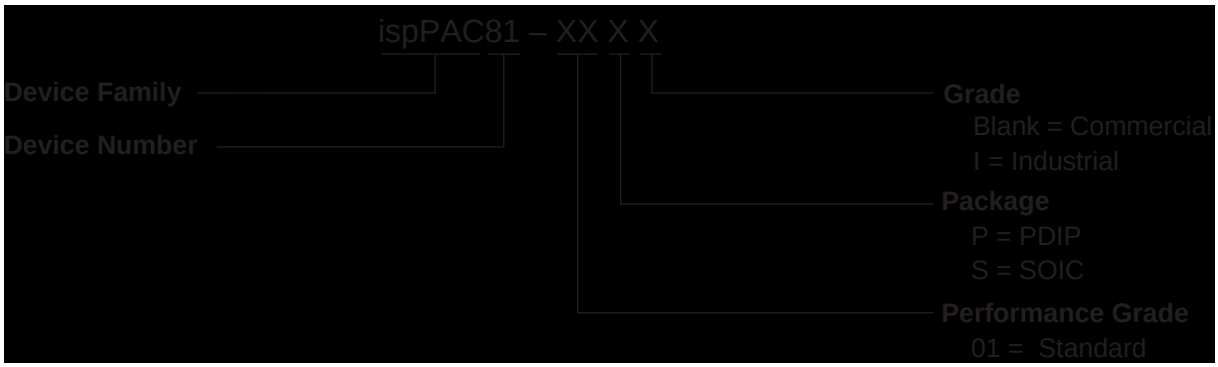
Lead Temperature (Soldering, 10 sec.) . . . . . .260°C

Ambient Temperature with Power Applied . . . . . -55 to 125°C

Storage Temperature . . . . . -65 to 150°C

Note: Stresses above those listed may cause permanent damage to the device. These are stress only ratings and functional operation of the device at these or at any other conditions above those indicated in the operational sections of this specification is not implied.

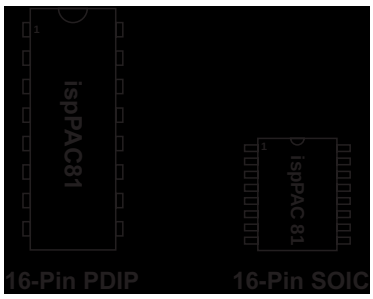
Part Number Description



ispPAC81 Ordering Information

Part Number	Package
ispPAC81-01PI	16-pin PDIP
ispPAC81-01SI	16-pin SOIC

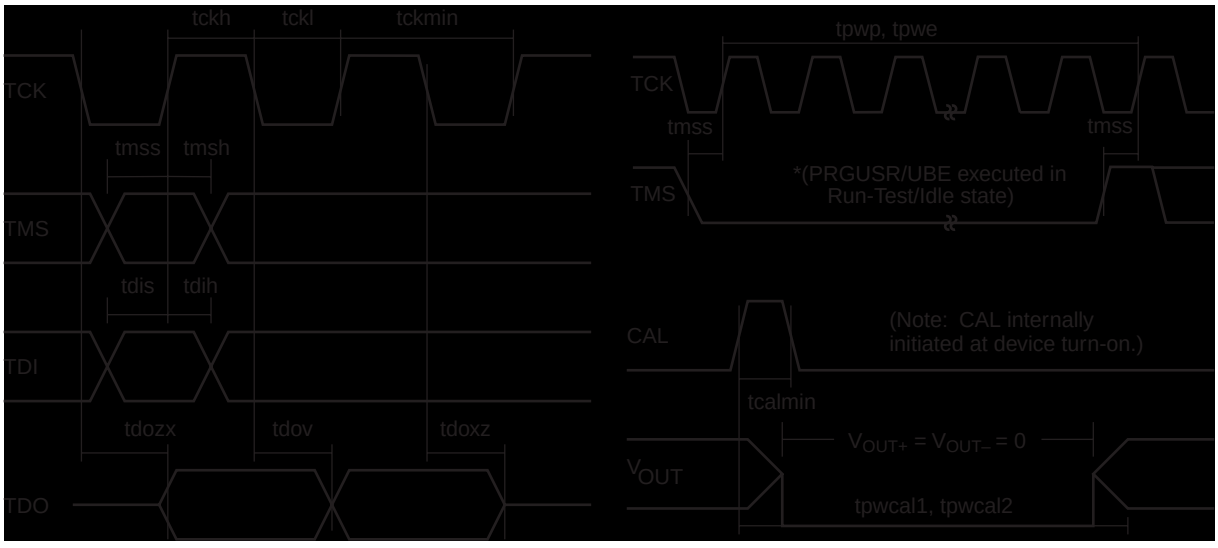
Package Options



Timing Specifications (JTAG Interface Mode)

T_A = 25°C; V_S = +5.0V (Unless otherwise specified)

Symbol	Parameter	Condition	Min.	Typ.	Max.	Units
Dynamic Performance						
tckmin	Minimum clock period		200			ns
tckh	TCK high time		50			ns
tckl	TCK low time		50			ns
tmss	TMS setup time		15			ns
tmsh	TMS hold time		10			ns
tdis	TDI setup time		15			ns
tdih	TDI hold time		10			ns
tdozx	TDO float to valid delay				60	ns
tdov	TDO valid delay				60	ns
tdoxz	TDI valid to float delay				60	ns
tpwp	Time for a programming operation	Executed in Run-Test/Idle	80		100	ms
tpwe	Time for an erase operation	Executed in Run-Test/Idle	80		100	ms
tpwcal1	Time for auto-cal operation on power-up	Automatically executed at power-up			250	ms
tcalmin	Minimum auto-cal pulse width		40			ns
tpwcal2	Time for user-initiated auto-cal operation	Executed on rising edge of CAL			100	ms

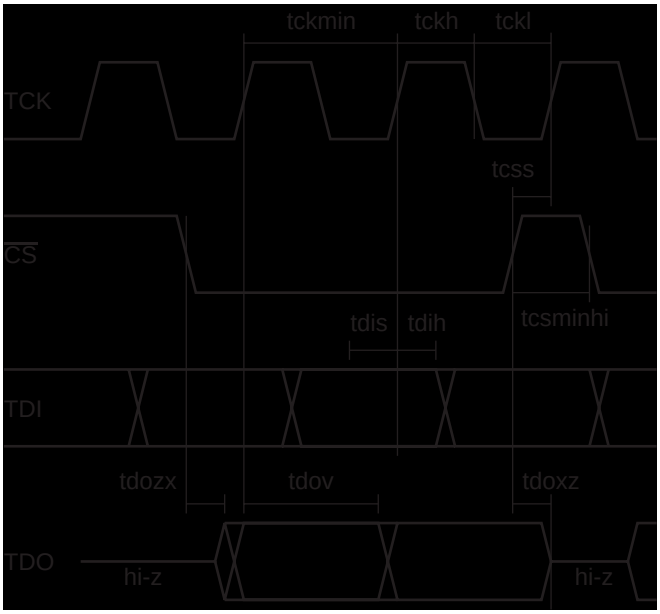


*Note: During device JTAG programming, analog output response will deviate from expected behavior. This is because all configuration information is erased and then re-written as part of a normal programming cycle, momentarily changing device filter and gain parameters. Behavior will deviate from that expected during both of these steps since the analog outputs are not clamped during a programming cycle. During erase, a drop in the filter corner frequency and an automatic change to the 10X gain setting can be expected (80ms minimum by specification) and will continue until bits go to their final state after a JTAG write command is issued (less than 2ms later, though the write cycle must still be maintained for a full 80ms to achieve specified data retention).

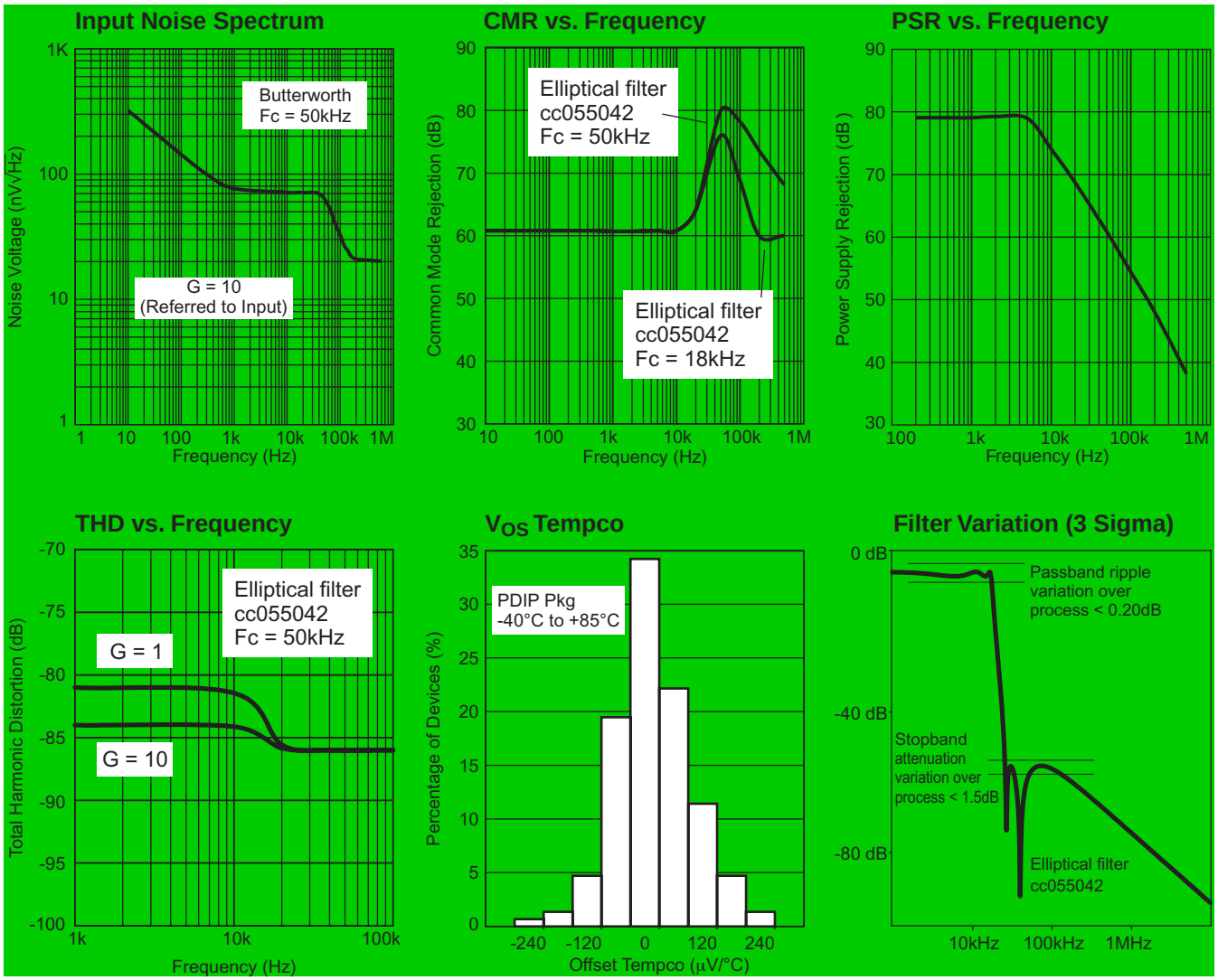
Timing Specifications (SPI Interface Mode)

T_A = 25°C; V_S = +5.0V (Unless otherwise specified)

Symbol	Parameter	Condition	Min.	Typ.	Max.	Units
Dynamic Performance						
tckmin	Minimum clock period		200			ns
tckh	TCK high time		100			ns
tckl	TCK low time		100			ns
tcss	CS setup time		20			ns
tcsminhi	Minimum CS pulse widths		40			ns
tdis	TDI setup time		15			ns
tdih	TDI hold time		10			ns
tdozx	TDO float to valid delay				60	ns
tdov	TDO valid delay				60	ns
tdoxz	TDO valid to float delay				60	ns



Typical Performance Characteristics

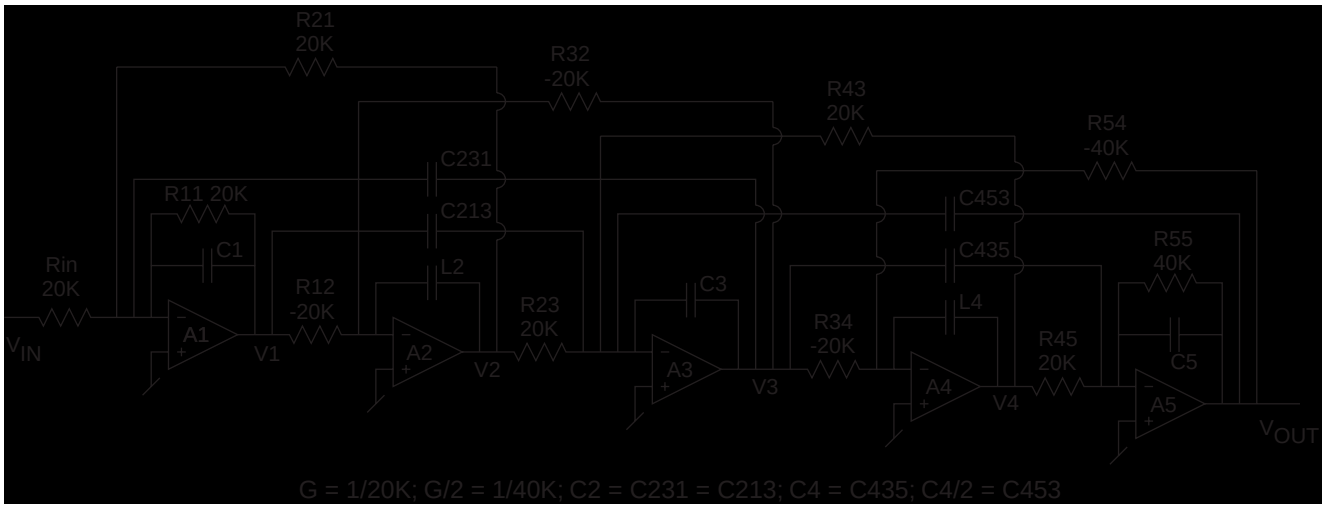


Theory of Operation

Reference Configuration

In the specifications table, specific filter configurations are referred to, such as the CC055042 at 17.62kHz. This is simply a shorthand notation for the classical filter type that includes all the characteristic parameters of the particular configuration. In this case, the CC refers to complete Chebyshev, or a Chebyshev ripple response in both the pass and stopbands of the filter. Another common name for this type of filter is the Elliptic family of filters. The next set of numbers, “05” refers to the order of the filter. In the case of ispPAC81 this will always be fifth order. The next two digits signify the reflection coefficient (rf), in this case 10%, and has a direct mathematical relationship to the passband ripple magnitude of the filter, where the passband ripple expressed in dB = 10*Log(1-rf^2). The final two digits are the passband to stopband attenuation ratio expressed as an angle, in this case 42 degrees or 1.49 (1/sin((π/180)*42)). This configuration corresponds to the elliptical filter with the ID# 1088 (Fp=17.62kHz; Fc=17.96kHz) in the filter configuration database utility of PAC-Designer. Because of the almost limitless number of configurations realizable with ispPAC81, standard test configurations had to be chosen. The Elliptical family was chosen since it has many parameters that can be easily and directly measured to insure that all the internal circuits of ispPAC81 are operating correctly.

Figure 1. Simplified ispPAC81 Filter Core Schematic



SPI Power-On Condition

The SPI shift register is always reset to all zeroes when an ispPAC81 powers on. That means that if the ENSPI pin is high at power on, the initial configuration will be set to a gain of 1X (0dB) and configuration “A” is selected as the “wake-up” configuration. The only way to prevent this behavior would be to hold the ENSPI pin low while applying power to the device. Because this is usually impractical, it is advised that if the ispPAC81 is used in SPI mode that it be reloaded to the desired first configuration every time power is cycled to the device and/or that the “A” configuration memory hold the desired “wake up” filter response.

A/B Configuration

Two complete configurations can be stored in the E^2 memory of the ispPAC81. Selection of either the “A” or “B” configuration in real time is accomplished with the device in the SPI interface mode (ENSPI pin = logic high). An eight-bit string is read into the ispPAC81 in the following order: four “don’t care” bits followed by a CAL command bit, the A/B configuration setting and gain bits PG2 and PG1.

Table 1. SPI Control Bit Sequence

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PG2	PG1	A/B	CAL	X	X	X	X

Table 2. Gain Bit Settings

Gain Setting	PG2	PG1
1X (0dB)	0	0
2X (6dB)	0	1
5X (14dB)	1	0
10X (20dB)	1	1

Table 3. JTAG User Configuration Bits

Symbol	Name	Description
FreqRange Bit	Hi/Lo Frequency Range Bit	Depending on the corner frequency, the frequency range bit is automatically set from within PAC-Designer to optimize the transfer function response of the ispPAC81. Exists for both the A and B user strings. Can be overridden from within PAC-Designer from the edit symbol dialog.
UES Bits	User Electronic Signature	These are uncommitted E^2 bits that can be used to store device information for future reference. The ispPAC81 contains 21 UES bits. These bits are accessible from within PAC-Designer by using the Edit Symbol, UES Bits command. Part of user configuration string A only.
Cap Bits	Capacitor Selection Bits	Varying length data words for each of the seven configuration capacitors of the ispPAC81. There is a complete set of 70 bits total for each user configuration string, A and B.
A/B Bit	Initial Configuration Select	With the A/B bit set to “A” (a logic 0), the device will power up in the configuration stored in user string A. The designations of A or B would have been determined initially in the design environment using PAC-Designer. It is also possible to designate the B user string as the initial or “wake up” configuration, although this is not recommended as it blocks the algorithm required to do a “blind” verification of the A configuration of a previously programmed device. This is determined from within PAC-Designer in the edit symbol dialog.
PG1 & PG2 Bits	Programmable Gain Bits	Contained only in the A configuration string. Can also be modified under SPI control. Refer to Table 2 for bit setting specifics.
ESF	Electronic Security Fuse	Setting this bit causes all subsequent readouts of the device configuration to be disabled (JTAG Verify commands). Can be reset by performing a JTAG user (USRA) bulk erase commands and reprogramming the device. This feature is used to prevent unauthorized readout of the device’s configuration.

JTAG User Bits

There are a number of user-configured E^2 bits that control various aspects of and can all be accessed in either the pull-down menus or directly in the schematic design entry screen of the PAC-Designer software interface to the ispPAC81. See the online help associated with the ispPAC81 in PAC-Designer for more details of how to set/program various operation modes. The list of control E^2 bits available is listed in Table 3.

Differential I/O

Differential peak-peak voltage is determined by knowing the signal extremes on both differential input or output pins. For example, if V(+) equals 4V and V(-) equals 1V, the differential voltage is defined as V(+) - V(-) = Vdiff, or 4V - 1V = +3V. Since either polarity can exist on differential I/O pins, it is also possible for the opposite extreme to exist and would mean when V(+) equals 1V and V(-) equals 4V, the differential voltage is now 1V - 4V = -3V. To calculate the differential peak-peak voltage or full signal swing, the absolute difference between the two extreme Vdiff's is calculated. Using the previous examples would result in |(+3V) - (-3V)| = 6V. It can be immediately seen that true differential signals result in a doubling of usable dynamic range. For more explanation of this and other differential circuit benefits, please refer to application note number AN6019, *Differential Signaling.*

Single-ended Input

To connect the ispPAC81 differential input to a single-ended signal, one of the differential inputs needs to be connected to a DC bias, preferably $VREF_{OUT}$. The input signal must either be AC coupled or have a DC bias equal to the DC level of the other input. Since the input voltage is defined as $V_{IN+} - V_{IN-}$, the common mode level is ignored. The signal information is only present on one input, the other being connected to a voltage reference.

Single-ended Output

Connecting the output to a single-ended circuit is simpler still. Simply connect one-half of the differential output, but not the other. Either output conveys the signal information, just at half the magnitude of the differential output. The DC level of the single-ended output will be $VREF_{OUT}$. If the load is not AC coupled and is at a DC potential other than $VREF_{OUT}$, the load draws a constant current. Using one of the differential outputs halves the available output voltage swing (3Vp-p versus 6Vp-p). If the load requires DC current, the available voltage swing is reduced. The output is capable of 10mA, so any DC current raises the minimum allowable load impedance.

Input Common-Mode Voltage Range

For the ispPAC81, both maximum input signal range and corresponding common-mode voltage range are a function of the input gain setting. The maximum input voltage times the gain of an individual PACblock cannot exceed the output range of that block or clipping will occur. The maximum guaranteed input range is 1V to 4V, with a typical range of 0.7V to 4.3V for a 5V supply voltage.

The input common-mode voltage is $V_{CM} = (V_{CM+} + V_{CM-})/2$. When the value of V_{CM} is 2.5V, there are no further input restrictions other than the previously mentioned clipping consideration. This is easily achieved when the input signal is true differential and referenced to 2.5V.

When V_{CM} is not 2.5V and the gain setting is greater than one, distortion will occur when the maximum input limit is reached for a particular gain. The lowest V_{CM} for a given gain setting is expressed by the formula, $V_{CM-} = 0.675V + 0.584G \cdot V_{IN}$ where G is the gain setting and V_{IN} is the peak input voltage, expressed as $|V_{IN+} - V_{IN-}|$ and the highest V_{CM} is $V_{CM+} = 5.0V - V_{CM-}$ where 5V is the nominal supply voltage.

In Table 4, the maximum V_{IN} for a given V_{CM-} to V_{CM+} range is given. If the maximum V_{IN} is known, find the equivalent or greater value under the appropriate gain column and the widest range for V_{CM} will be found horizontally across in the left-most two columns. Only a V_{CM} range equal to or less than this will give distortion-free performance. Conversely, if the maximum V_{CM} range is known, the largest acceptable peak value of V_{IN} can be found in the corresponding gain column. All values of V_{IN} less than this will give full rated performance.

Table 4. Input Common-Mode Voltage Range Limitations

Input Voltage Magnitude (Volts-Peak)					
V_{CM-}	V_{CM+}	G=1	G=2	G=5	G=10
1.000	**4.000**	0.557	0.278	0.111	0.056
1.100	**3.900**	0.728	0.364	0.146	0.073
1.200	**3.800**	0.899	0.450	0.180	0.090
1.300	**3.700**	1.071	0.535	0.214	0.107
1.400	**3.600**	1.242	0.621	0.248	0.124
1.500	**3.500**	1.413	0.707	0.283	0.141
1.600	**3.400**	1.584	0.792	0.317	0.158
1.700	**3.300**	1.756	0.878	0.351	0.176
1.800	**3.200**	1.927	0.964	0.385	0.193
1.900	**3.100**	2.098	1.049	0.420	0.210
2.000	**3.000**	2.270	1.135	0.454	0.227
2.100	**2.900**	2.441	1.220	0.488	0.244
2.200	**2.800**	2.612	1.306	0.522	0.261
2.300	**2.700**	2.783	1.392	0.557	0.278
2.400	**2.600**	2.955	1.477	0.591	0.295
2.426	**2.574**	3.000*	1.500*	0.600*	0.300*
2.500	**2.500**	3.126	1.563	0.625	0.313

*Peak input voltage for guaranteed performance at a given gain setting.

Software-Based Design Environment

Design Entry Software

Designers configure the ispPAC81 and verify its performance using PAC-Designer, an easy-to-use, Microsoft Windows compatible program. Circuit designs are entered graphically and then verified, all within the PAC-Designer environment. Full device programming is supported using PC parallel port I/O operations and a download cable connected to the serial programming interface of the ispPAC81. A database of filter configurations is included with thousands of possible implementations to choose from. In addition, comprehensive on-line and printed documentation is provided that covers all aspects of PAC-Designer operation.

The PAC-Designer schematic window, shown in Figure 2, provides access to all configurable ispPAC81 elements via its graphical user interface. All analog input and output pins are represented. Static or non-configurable pins such as power, ground, $VREF_{OUT}$, and the serial digital interface are omitted for clarity. Any element in the schematic window can be accessed via mouse operations as well as menu commands. When completed, configurations can be saved, simulated, and downloaded to devices.

PAC-Designer operation can be automated and extended by using custom-designed Visual Basic™ programs that set the interconnections and the parameters of ispPAC products. More information on this and other topics is included in the on-line documentation as well as the *PAC-Designer Getting Started Manual.*

Design Simulation Capability

A powerful feature of PAC-Designer is its simulation capability, enabling quick and accurate verification of circuit operation and performance. Once a circuit is configured via the interactive design process, gain and phase response between any input and output can then be determined. This function is part of the simulator capability which derives a transfer equation between the two points and then sweeps it over the user-specified frequency range. Figure 3 shows a typical screen plot of the gain/phase simulator. In it are the input to output response curves of an Elliptical and a Butterworth response stored in configuration A and B respectively. These are the two options specified in the design screen window shown in Figure 2.

The simulator is capable of displaying up to four separate input to output responses. This allows multiple signals to be viewed as well as intermediate results of component changes so performance comparisons can be made. There is also a user-positioned crosshair cursor that intersects the curves on the plot, and reads out the gain and frequency in the lower right hand corner of the plot window when activated.

Figure 2. Initial PAC-Designer Schematic Design Entry Screen

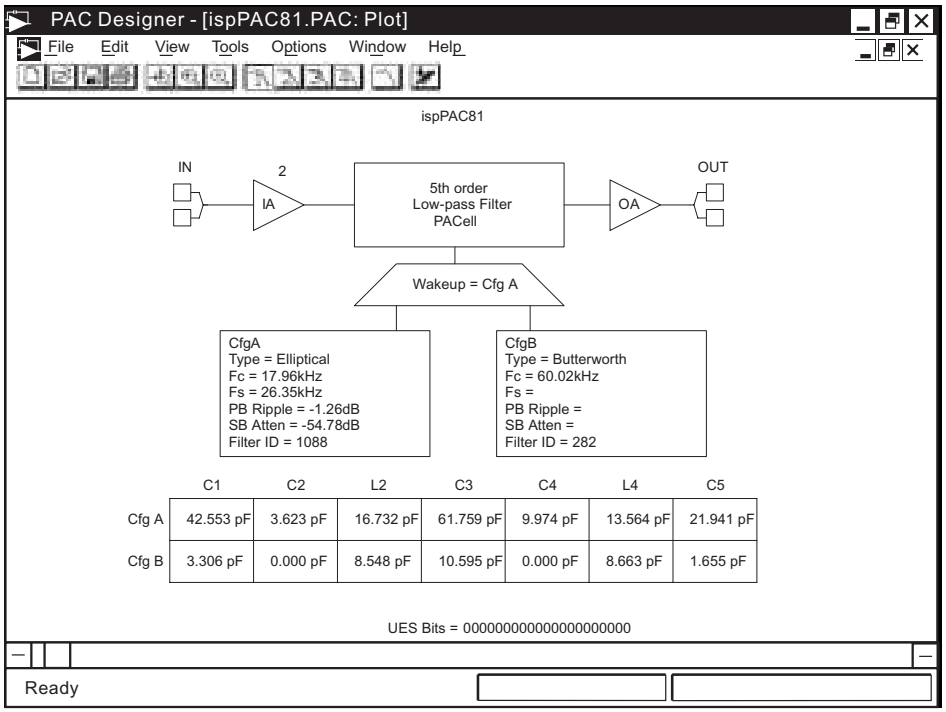
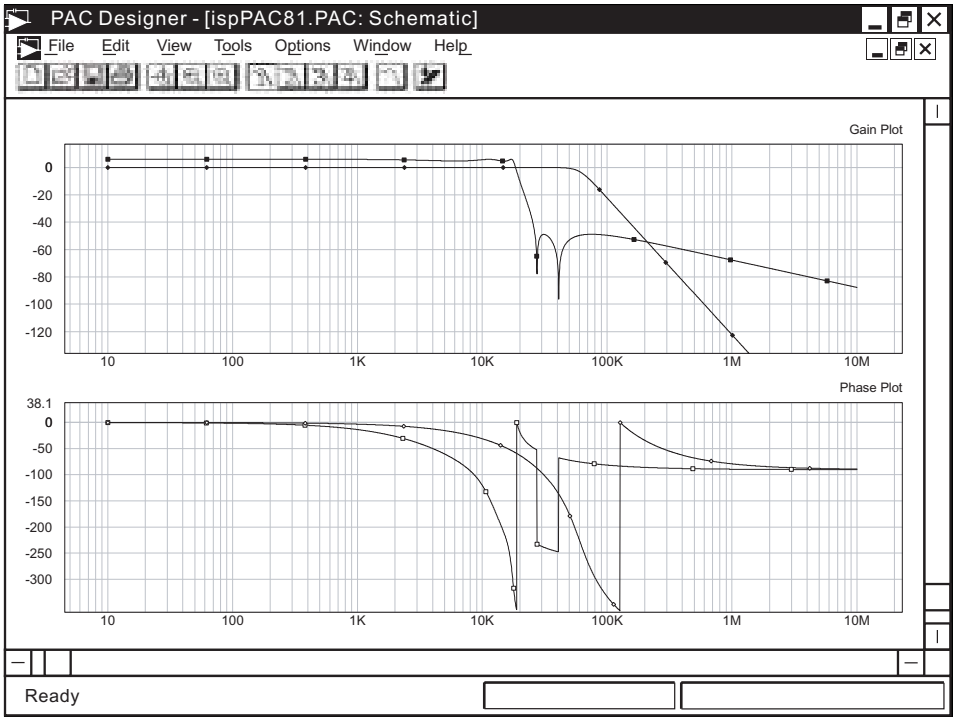


Figure 3. PAC-Designer Simulation Plot Screen



In-System Programmability

In-System Programming

The ispPAC81 is an in-system programmable device. This is accomplished by integrating all high voltage programming circuitry on-chip. Programming is performed through a 5-wire, IEEE 1149.1 compliant serial port interface at normal logic levels. Once a device is programmed, all configuration information is stored on-chip, in non-volatile E^2CMOS memory cells. The specifics of the IEEE 1149.1 serial interface are described in the interface section of this data sheet.

User Electronic Signature

A user electronic signature (UES) feature is included in the E^2 memory of the ispPAC81. It contains 21 bits that can be configured by the user to store unique data such as ID codes, revision numbers or inventory control data.

Electronic Security

An electronic security “fuse” (ESF) bit is provided in every ispPAC81 device to prevent unauthorized readout of the E^2CMOS user bit patterns. Once programmed, this cell prevents further access to the functional user bits in the device. This cell can only be erased by reprogramming the device, so the original configuration can not be examined once programmed. Usage of this feature is optional.

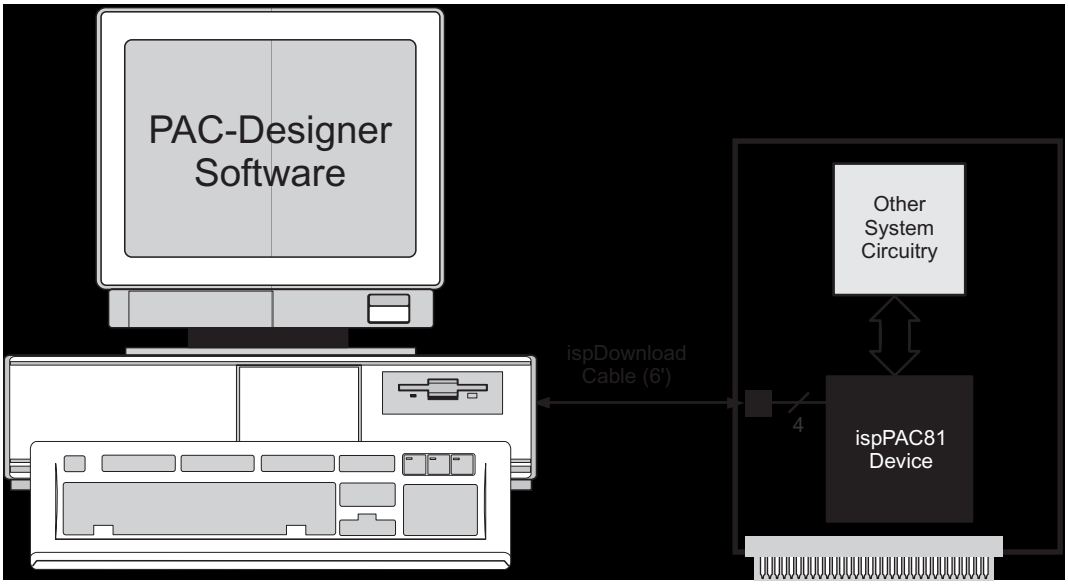
Production Programming Support

Once a final configuration is determined, an ASCII format JEDEC file is created using the PAC-Designer software. Devices can then be ordered through the usual supply channels with the user’s specific configuration already preloaded into the devices. By virtue of its standard interface, compatibility is maintained with existing production programming equipment, giving customers a wide degree of freedom and flexibility in production planning.

Evaluation Fixture

Included in the basic ispPAC81 Design Kit is an engineering prototype board that can be connected to the parallel port of a PC. It demonstrates proper layout techniques for the ispPAC81 and can be used in real time to check circuit operation as part of the design process. Input and output connections as well as a “breadboard” circuit area are provided to speed debugging of the circuit.

Figure 4. Configuring the ispPAC81 “In-System” from a PC Parallel Port



IEEE Standard 1149.1 Interface

Serial Port Programming Interface

Communication with the ispPAC81 is facilitated via an IEEE 1149.1 test access port (TAP). It is used by the ispPAC81 as a serial programming interface, and not for boundary scan test purposes. There are no boundary scan logic cells in the ispPAC81 architecture. This does not prevent the ispPAC81 from functioning correctly, however, when placed in a valid serial chain with other IEEE 1149.1 compliant devices.

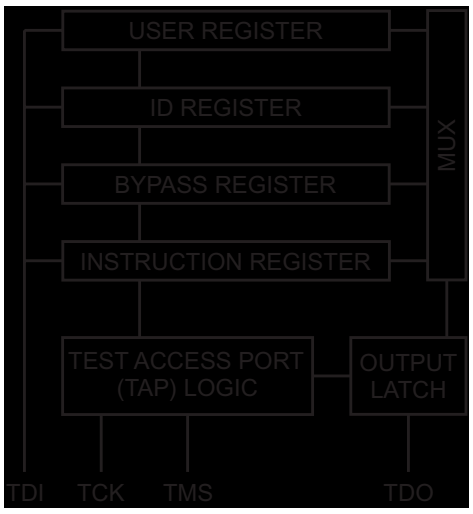
A brief description of the ispPAC81 serial interface follows. For complete details of the reference specification, refer to the publication, Standard Test Access Port and Boundary-Scan Architecture, IEEE Std 1149.1-1990 (which now includes IEEE Std 1149.1a-1993).

Overview

An IEEE 1149.1 test access port (TAP) provides the control interface for serially accessing the digital I/O of the ispPAC81. The TAP controller is a state machine driven with mode and clock inputs. Under the correct protocol, instructions are shifted into an instruction register which then determines subsequent data input, data output, and related operations. Device programming is performed by addressing the user register, shifting data in, and then executing a program user instruction, after which the data is transferred to internal E^2CMOS cells. It is these non-volatile cells that determine the configuration of the ispPAC81. By cycling the TAP controller through the necessary states, data can also be shifted out of the user register to verify the current ispPAC81 configuration. Instructions exist to access all data registers and perform internal control operations.

For compatibility between compliant devices, two data registers are mandated by the IEEE 1149.1 specification. Others are functionally specified, but inclusion is strictly optional. Finally, there are provisions for optional data registers defined by the manufacturer. The two required registers are the bypass and boundary-scan registers. For ispPAC81, the bypass register is a one-bit shift register that provides a short path through the device when boundary testing or other operations are not being performed. The ispPAC81, as mentioned, has no boundary scan logic and therefore no boundary scan register. All instructions relating to boundary scan operations place the ispPAC81 in the BYPASS mode to maintain compliance with the specification. The optional identification register described in IEEE 1149.1 is also included in the ispPAC81. One additional data register included in the TAP of the ispPAC81 is the Lattice defined user register. Figure 5 shows how the instruction and various data registers are placed in an ispPAC81.

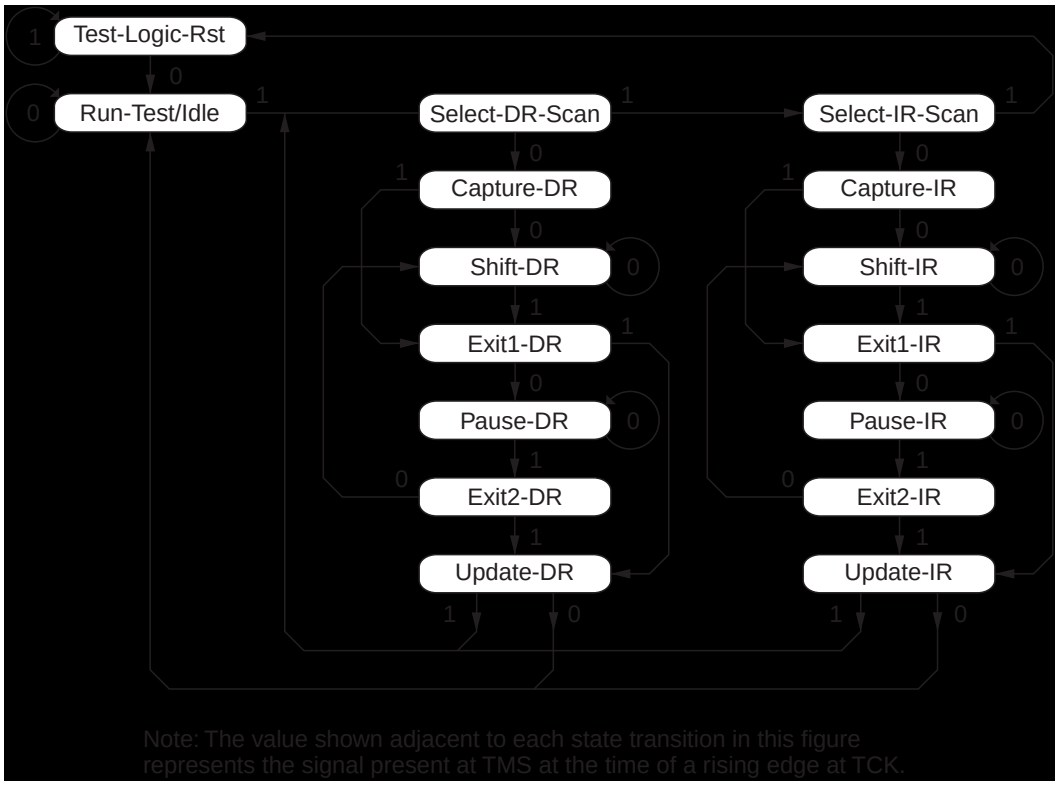
Figure 5. ispPAC81 TAP Registers



TAP Controller Specifics

The TAP is controlled by the Test Clock (TCK) and Test Mode Select (TMS) inputs. These inputs determine whether an Instruction Register or Data Register operation is performed. Driven by the TCK input, the TAP consists of a small 16-state controller design. In a given state, the controller responds according to the level on the TMS input as shown in Figure 6. Test Data In (TDI) and TMS are latched on the rising edge of TCK, with Test Data Out (TDO) becoming valid on the falling edge of TCK. There are six steady states within the controller: Test-Logic-Reset, Run-Test/Idle, Shift-Data-Register, Pause-Data-Register, Shift-Instruction-Register and Pause-Instruction-Register. But there is only one steady state for the condition when TMS is set high: the Test-Logic-Reset state. This allows a reset of the test logic within five TCKs or less by keeping the TMS input high. Test-Logic-Reset is the power-on default state.

Figure 6. Test Access Port (TAP) Controller State Diagram



When the correct logic sequence is applied to the TMS and TCK inputs, the TAP will exit the Test-Logic-Reset state and move to the desired state. The next state after Test-Logic-Reset is Run-Test/Idle. Until a data or instruction scan is performed, no action will occur in Run-Test/Idle (steady state = idle). After Run-Test/Idle, either a data or instruction scan is performed. The states of the Data and Instruction Register blocks are identical to each other differing only in their entry points. When either block is entered, the first action is a capture operation. For the Data Registers, the Capture-DR state is very simple: it captures (parallel loads) data onto the selected serial data path (previously chosen with the appropriate instruction). For the Instruction Register, the Capture-IR state will always load the IDCODE instruction. It will always enable the ID Register for readout if no other instruction is loaded prior to a Shift-DR operation. This, in conjunction with mandated bit codes, allows a “blind” interrogation of any device in a compliant IEEE 1149.1 serial chain.

From the Capture state, the TAP transitions to either the Shift or Exit1 state. Normally the Shift state follows the Capture state so that test data or status information can be shifted out or new data shifted in. Following the Shift state, the TAP either returns to the Run-Test/Idle state via the Exit1 and Update states or enters the Pause state via Exit1. The Pause state is used to temporarily suspend the shifting of data through either the Data or Instruction Register while an external operation is performed. From the Pause state, shifting can resume by reentering the

Shift state via the Exit2 state or be terminated by entering the Run-Test/Idle state via the Exit2 and Update states. If the proper instruction is shifted in during a Shift-IR operation, the next entry into Run-Test/Idle initiates the test mode (steady state = test). This is when the device is actually programmed, erased or verified. All other instructions are executed in the Update state.

Test Instructions

Like data registers, the IEEE 1149.1 standard also mandates the inclusion of certain instructions. It outlines the function of three required and six optional instructions. Any additional instructions are left exclusively for the manufacturer to determine. The instruction word length is not mandated other than to be a minimum of two bits, with only the BYPASS and EXTEST instruction code patterns being specifically called out (all ones and all zeroes respectively). The ispPAC81 contains the required minimum instruction set as well as one from the optional instruction set. In addition, there are several proprietary instructions that allow the device to be configured and verified. For ispPAC81, the instruction word length is five bits. All ispPAC81 instructions available to users are shown in Table 5.

Table 5. ispPAC81 TAP Instructions

Instruction	Code	Description
EXTEST	**00000**	External Test. Default to BYPASS.
ADDUSR	**00001**	Address User Data Register (A or B).
ABE	**00010**	User A Bulk Erase.
BBE	**00011**	User B Bulk Erase.
VERA	**00100**	Verify User A Data Register.
VERB	**00101**	Verify User B Data Register.
PRGA	**00110**	Program User A Data Register.
PRGB	**00111**	Program User B Data Register.
ENCAL	**01100**	Enable Calibration Sequence.
IDCODE	**01101**	Read Identification Data Register.
SAMPLE	**11110**	Sample/Preload. Default to BYPASS.
BYPASS	**11111**	Bypass (Connect TDI to TDO).

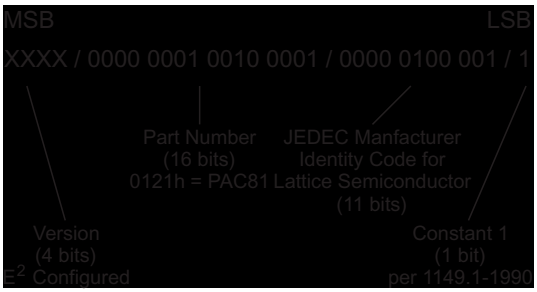
BYPASS is one of the three required instructions. It selects the Bypass Register to be connected between TDI and TDO and allows serial data to be transferred through the device without affecting the operation of the ispPAC81. The bit code of this instruction is defined to be all ones by the IEEE 1149.1 standard.

The required **SAMPLE/PRELOAD** instruction dictates the Boundary-Scan Register be connected between TDI and TDO. The ispPAC81 has no boundary-scan register, so for compatibility it defaults to the BYPASS mode whenever this instruction is received. The bit code for this instruction is defined by Lattice as shown in Table 5.

The **EXTEST** (External Test) instruction is required and would normally place the device into an external boundary test mode while also enabling the Boundary-Scan Register to be connected between TDI and TDO. Again, since the ispPAC81 has no boundary-scan logic, the device is put in the BYPASS mode to ensure specification compatibility. The bit code of this instruction is defined by the 1149.1 standard to be all zeros.

The optional **IDCODE** (Identification Code) instruction is incorporated in the ispPAC81 and leaves it in its functional mode when executed. It selects the Device Identification Register to be connected between TDI and TDO. The Identification Register is a 32-bit shift register containing information regarding the IC manufacturer, device type and version code (see Figure 7). Access to the Identification Register is immediately available, via a TAP data scan operation, after power-up of the device, or by issuing a Test-Logic-Reset instruction. The bit code for this instruction is defined by Lattice as shown in Table 5.

Figure 7. Identification Code (IDCODE) 32-Bit Binary Word for Lattice ispPAC81



ADDUSR (Address User Register) instruction is a Lattice-defined instruction that selects the user register to be shifted during a Shift-DR operation. Normal operation of a device is not interrupted by this instruction. It precedes a PRGA or PRGB (Program User A or B) instruction to shift in a new configuration from the user register into either the A or B configuration memory, and follows a VERA or VERB (Verify User A or B) instruction to shift out the current configuration of either A or B configuration memory into the user register. The bit code for this instruction is shown in Table 5.

The **PRGA** and **PRGB** (Program User A or B) are Lattice instructions that enable the data shifted into the user register to be programmed into the non-volatile E^2CMOS memory of the ispPAC81 and thereby alter either or both of its two user configurations. The user register is a 96-bit shift register that contains all the user-controlled parametric data pertaining to the configuration of the ispPAC81. NOTE: Although the user register length is 96 bits, only the “A” configuration is that long. The device gain setting bits, UES bits, and security fuse bit are all part of the “A” configuration memory and are not stored at all in “B” memory, which only contains the unique capacitor settings of that configuration. When initially programming or reprogramming the ispPAC81 with software other than PAC-Designer, or an authorized third-party programmer (e.g., via microcontroller, refer to the Lattice application note covering the required algorithms necessary for complete JTAG device programming control of the ispPAC81, specific bit assignments, word lengths, etc.). Normal operation of the device is interrupted during the actual programming time. A programming operation does not begin until entry of the Run-Test/Idle state. The time required to insure data retention is given in the TAP signal specifications table. The user must ensure that the recommended programming times are observed. The bit code for these instructions is shown in Table 5.

VERA and **VERB** (Verify User A or B) are the next Lattice instructions and cause the current A or B configurations of the ispPAC81 to be loaded into the user register. This operation doesn’t interrupt operation of the device. The current configuration of either the A or B configuration memory can then be shifted out of the user register immediately after an ADDUSR instruction is executed. NOTE: The verification of memory configuration “A” is possible only when the A/B bit is set to a logic 0. This must be taken into account if verify will be performed at a later time on parts with unknown configurations (refer to the Lattice application note covering the required algorithms necessary for complete JTAG device programming control of the ispPAC81, specific bit assignments, word lengths, etc.). If the A/B bit has been set to a logic 1, it will not be possible to do a VERA command properly. The bit code for this instruction is shown in Table 5.

ENCAL (Enable Calibration) is a Lattice instruction that enables the start of an auto-calibration sequence. This operation causes all outputs of the device to go to 2.5V until the calibration sequence is completed (see Timing Specifications). As with the programming instructions above, calibration does not begin until entry of the Run-Test/Idle state. The completion of the calibration is not dependent, however, on any further TAP control. This means the state of the TAP can be returned immediately to the Test-Logic-Reset state. The only consideration would be to not clock the TAP during critical analog operations. The first several milliseconds of the calibration routine are consumed waiting for configurations to settle, though, leaving more than enough time to clock the TAP back to the Test-Logic-Reset state. The bit code for this instruction is shown in Table 5.

The last Lattice instructions are **ABE** and **BBE** (User A or B Bulk Erase). Operation of the device is interrupted during an ABE or BBE, during which all inputs are disconnected and all outputs driven to $VREF_{OUT}$ (2.5V). To economize internal circuitry, programming can only be selectively done in one direction (from zeroes to ones). The ABE

and BBE are used to return all user bits to a zero state at the same time. An ABE or BBE usually proceeds a PRGA or PRGB operation, otherwise one to zero changes would not be implemented. It can also be used to erase all configuration information from a device and is the default condition of parts shipped from the factory. The same programming time constraints apply to ABE and BBE as for PRGA and PRGB. The bit code for this instruction is shown in Table 5.

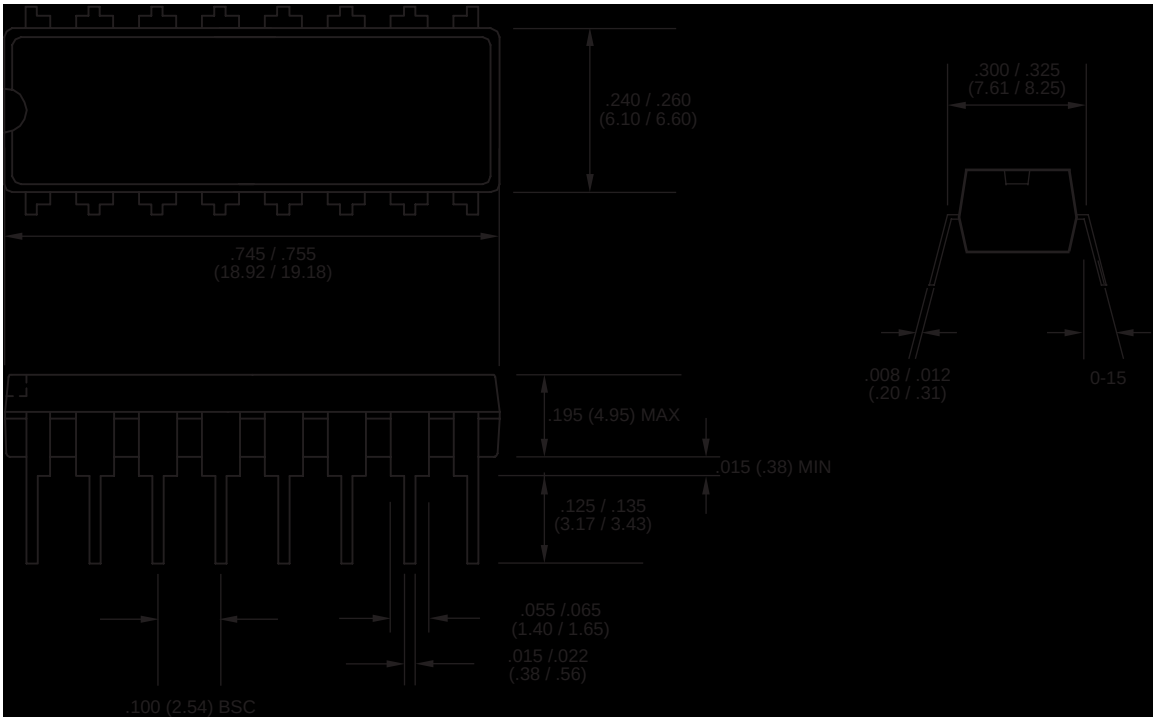
The ADDUSR, BYPASS, EXTEST, IDCODE and SAMPLE/PRELOAD instructions are all executed in the Update-IR state. Other instructions: PRGUSR, VERUSR and UBE are executed upon entry of the Run-Test/Idle state.

It is recommended that when all serial interface operations are completed, the TAP controller be reset and left in the Test-Logic-Reset state (the power-up default) and the TCK and TMS inputs idled. This will insure the best analog performance possible by minimizing the effects of digital logic “feed-through.”

Package Diagrams

16-Pin Plastic PDIP

Dimensions in Inches MIN./MAX. Dimensions in millimeters, shown in parenthesis, are for reference only.



16-Pin Plastic SOIC

Dimensions in Inches MIN./MAX. Dimensions in millimeters, shown in parenthesis, are for reference only.