# ispPAC®10

## In-System Programmable Analog Circuit

## Features

- **IN-SYSTEM PROGRAMMABLE (ISP™) ANALOG CIRCUIT**
  - **— Four Instrument Amplifier Gain/Attenuation Stages**
  - **— Signal Summation (Up to 4 Inputs)**
  - **— Precision Active Filtering (10kHz to 100kHz)**
  - **— No External Components Needed for Configuration**
  - **— Non-Volatile E²CMOS® Cells (10,000 Cycles)**
  - **— IEEE 1149.1 JTAG Serial Port Programming**
- **FOUR LINEAR ELEMENT BUILDING BLOCKS**
  - **— Programmable Gain Range (0dB to 80dB)**
  - **— Bandwidth of 550kHz (G=1), 330kHz (G=10)**
  - **— Low Distortion (THD < -74dB max @ 10kHz)**
  - **— Auto-Calibrated Input Offset Voltage**
- **TRUE DIFFERENTIAL I/O (±3V RANGE)**
  - **— High CMR (69dB) Instrument Amplifier Inputs**
  - **— 2.5V Common Mode Reference on Chip**
  - **— Four Rail-to-Rail Voltage Outputs**
- **28-PIN PLASTIC DIP OR SOIC PACKAGE**
  - **— Single Supply 5V Operation**
- **APPLICATIONS INCLUDE INTEGRATED:**
  - **— Single +5V Supply Signal Conditioning**
  - **— Active Filters, Gain Stages, Summing Blocks**
  - **— Analog Front Ends, 12-Bit Data Acq. Systems**
  - **— Sensor Signal Conditioning**

## Description

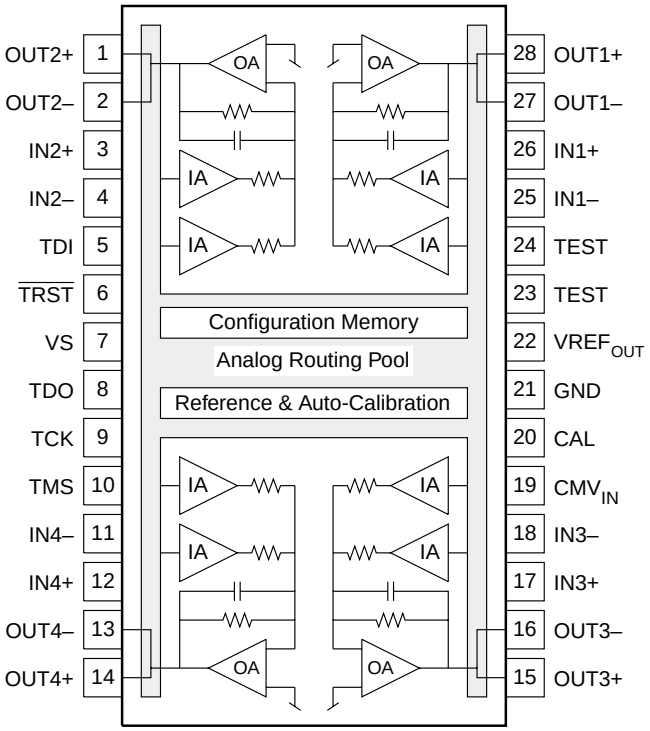
The ispPAC10 is a member of the Lattice family of In-System Programmable analog circuits, digitally configured via nonvolatile $E^2$CMOS technology.

Analog function modules, called PACblocks™, replace traditional analog components such as op amps and active filters, eliminating the need for most external resistors and capacitors. With no requirement for external configuration components, ispPAC10 expedites the design process, simplifying prototype circuit implementation and change, while providing high performance and integrated functionality.

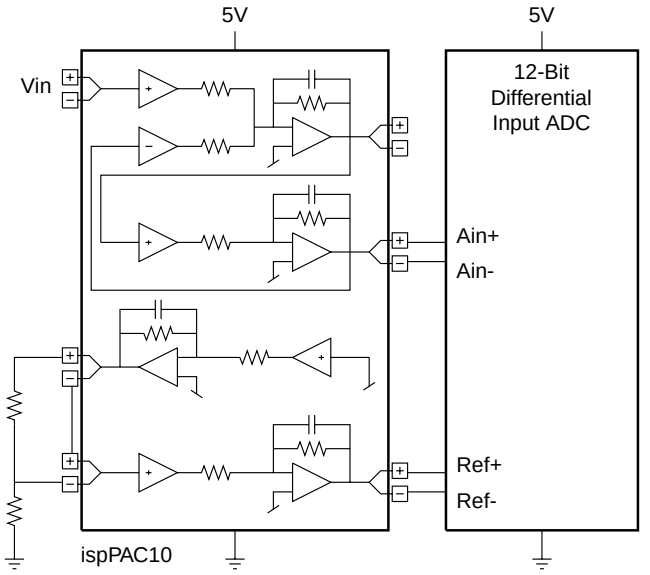
Designers configure the ispPAC10 and verify its performance using PAC-Designer®, an easy-to-use, Microsoft Windows® compatible development tool. Device programming is supported using PC parallel port I/O operations. A library of configurations is included with basic solutions and examples of advanced circuit techniques.

The ispPAC10 is configured through its IEEE Standard 1149.1 (JTAG) compliant serial port. The flexible In-System Programming capability enables programming, verification and reconfiguration if desired, directly on the printed circuit board.

## Functional Block Diagram




## Typical Application Diagram






LATTICE SEMICONDUCTOR CORP., 5555 Northeast Moore Ct., Hillsboro, Oregon 97124, U.S.A.
Tel. (503) 268-8000; 1-800-LATTICE; FAX (503) 268-8556; http://www.latticesemi.com

**September 2000**

**$T_A$ = 25°C; $V_S$ = 5.0V; Signal path = $V_{IN}$ to $V_{OUT}$ of one PACblock (second input unused); 1V ≤ $V_{OUT}$ ≤ 4V; Gain = 1; Output load = 200pf, 1MΩ. Feedback enabled; Feedback capacitor = minimum; Auto-Cal initiated immediately prior. (Unless otherwise specified).**

## DC Electrical Characteristics

| SYMBOL | PARAMETER | CONDITION | MIN. | TYP. | MAX. | UNITS |
|---|---|---|---|---|---|---|
| **Analog Input** | | | | | | |
| $V_{IN±}$ (1) | Input Voltage Range | Applied Either to $V_{IN+}$ or $V_{IN–}$ | 1 | | 4 | V |
| $V_{IN-DIFF}$ | Differential Input Voltage Swing (2) | 2\| $V_{IN+}$ – $V_{IN–}$ \| | 6 | | | $V_{p-p}$ |
| $V_{OS}$ (2) | Differential Offset Voltage (Input Referred) | G = 10 | | 20 | 100 | µV |
| | | G = 1 | | 0.2 | 1.0 | mV |
| $\Delta V_{OS}/\Delta T$ | Differential Offset Voltage Drift | -40 to +85°C | | 50 | | µV/°C |
| $R_{IN}$ | Input Resistance | | | $10^9$ | | Ω |
| $C_{IN}$ | Input Capacitance | | | 2 | | pF |
| $I_B$ | Input Bias Current | at DC | | 3 | | pA |
| $e_N$ | Input Noise Voltage Density | At 10kHz, Referred to Input, G = 10 | | 38 | | nV/√Hz |
| **Analog Output** | | | | | | |
| $V_{OUT±}$ | 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\| $V_{OUT+}$ – $V_{OUT–}$ \| | 9.6 | | | $V_{p-p}$ |
| $I_{OUT±}$ | Output Current | Source/Sink | 10 | | | mA |
| $V_{CM}$ | Common Mode Output Voltage | ($V_{OUT+}$ + $V_{OUT-}$)/2 ; $V_{IN+}$ = $V_{IN–}$ | 2.495 | 2.500 | 2.505 | V |
| **Static Performance** | | | | | | |
| G | Programmable Gain Range | Each Individual PACblock | 0 | | 20 | dB |
| | Gain Error | $R_L$ = 300Ω Differential | | | 4.0 | % |
| | Gain Matching | Between Two Inputs of Same PACblock | | | 3.0 | % |
| $\Delta G/\Delta T$ | Gain Drift | -40 to +85°C | | 20 | | ppm/°C |
| PSR | Power Supply Rejection | Differential at 1kHz | | 80 | | dB |
| | | Single-ended at 1kHz | | 77 | | dB |
| **Common Mode Reference Output ($VREF_{OUT}$)** | | | | | | |
| $VREF_{OUT}$ | Reference Output Voltage Range | Nominally 2.500V | -0.2 | | 0.2 | % |
| $CMV_{IN}$ (4) | Common Mode Voltage Input | Optional External Common-Mode Voltage | 1.25 | | 3.25 | V |
| | Reference Output Voltage Drift | -40 to +85°C | | 50 | | ppm/°C |
| $IREF_{OUT}$ | Reference Output Current | ($VREF_{OUT}$ = ±1%) Source | | 50 | | µA |
| | | ($VREF_{OUT}$ = ±1%) Sink | | 350 | | µA |
| | Reference Output Noise Voltage | 10MHz Bandwidth; 1µF Bypass Capacitor | | 40 | | $µV_{RMS}$ |
| | Reference Power Supply Rejection | 1kHz | | 80 | | dB |
| **Programming** | | | | | | |
| | Erase/Reprogram Cycles | | 10K | | | cycles |
| **Digital I/O** | | | | | | |
| $V_{IL}$ | Input Low Voltage | | 0 | | 0.8 | V |
| $V_{IH}$ | Input High Voltage | | 2.0 | | $V_S$ | V |
| $I_{IL}$, $I_{IH}$ | Input Leakage Current | 0V ≤ TCK Input ≤ $V_S$ | | | ±10 | µA |
| | | 0V ≤ CAL, TDI, TMS, TRST Inputs ≤ $V_S$ | | | +40/-70 | 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 |
| **Power Supplies** | | | | | | |
| $V_S$ | Operating Supply Voltage | | 4.75 | 5.0 | 5.25 | V |
| $I_S$ | Supply Current | $V_S$ = 5.0V | | | 23 | mA |
| $P_D$ | Power Dissipation | $V_S$ = 5.0V | | | 115 | mW |
| **Temperature Range** | | | | | | |
| | Operation | | -40 | | +85 | °C |
| | Storage | | -65 | | +150 | °C |

## AC Electrical Characteristics

| SYMBOL | PARAMETER | | CONDITION | MIN. | TYP. | MAX. | UNITS |
|---|---|---|---|---|---|---|---|
| **Dynamic Performance** | | | | | | | |
| THD | Total Harmonic Distortion | Differential | $F_{IN}$ = 10kHz | | -88 | -74 | dB |
| | | Single-Ended | | | -72 | | dB |
| | | Differential | $F_{IN}$ = 100kHz | | -67 | -62 | dB |
| | | Single-Ended | | | -63 | | dB |
| SNR | Signal to Noise | G = 1 to 10 | 0.1Hz to 100kHz | | 103 | | dB |
| CMR | Common Mode Rejection ($V_{IN}$ = 1V to 4V) | | 10kHz | | 69 | | dB |
| | Note: $V_{IN+}$ and $V_{IN-}$ connected together | | 100kHz | | 55 | | dB |
| BW | Small Signal Bandwidth | G = 1 | | | 550 | | kHz |
| | | G = 10 | | | 330 | | kHz |
| $BW_{FP}$ | Full Power Bandwidth | $V_{IN}$ = 6 | $V_{DIFF}$, $V_{OUT}$ = -3dB; G=1 | | 330 | | kHz |
| SR | Slew Rate | | | 5.0 | 7.5 | | V/µs |
| $t_S$ | Settling Time | 0.1% | $6V_{DIFF}$ Input Step | | 4.0 | | µs |
| | Crosstalk | | Between Any Two Channels | | -90 | | dB |
| **Filter Characteristics** | | | | | | | |
| | Filter Pole Programming Range | | Number of Poles in Range > 120 | 10 | | 100 | kHz |
| $F_0$ | Absolute Pole Frequency Accuracy | | Deviation From Calculated Value | | 1.0 | 5.0 | % |
| $\Delta F_0$ | Pole Step Size (Between Calculated Poles) | | 10kHz to 100kHz | | | 3.2 | % |
| $\Delta F_0/\Delta T$ | Pole Frequency Change vs. Temperature | | -40 to +85°C | | 0.02 | | %/°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 datasheet. (2) Refer to theory of operation section later in this datasheet 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) The user-provided voltage on this pin ($CMV_{IN}$) becomes an optional (selected via programming) alternative to the default 2.5V $VREF_{OUT}$.

## Absolute Maximum Ratings

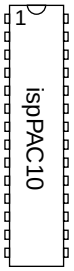
Supply Voltage $V_S$ ..... -0.5 to +7V
Logic and Analog Input Voltage Applied ..... 0 to $V_S$
Logic and Analog Output Short Circuit Duration ..... Indefinite
Lead Temperature (Soldering, 10 sec.) ..... 260°C
Ambient Temperature with Power Applied ... -55 to 125°C
Storage Temperature ..... -65 to 150°C

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.

## ispPAC10 Ordering Information

| Ordering Number | Package |
|---|---|
| ispPAC10-01PI | 28-Pin DIP |
| ispPAC10-01SI | 28-Pin SOIC |

## Package Options




**28-Pin PDIP**


**28-Pin SOIC**

## Part Number Description

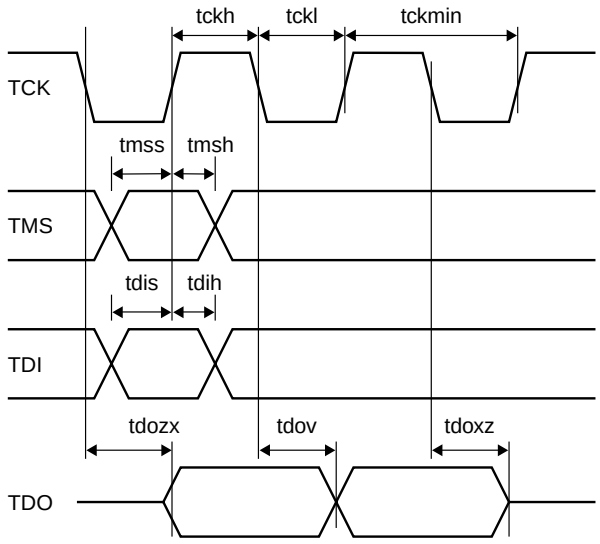
ispPAC 10 – XX X X

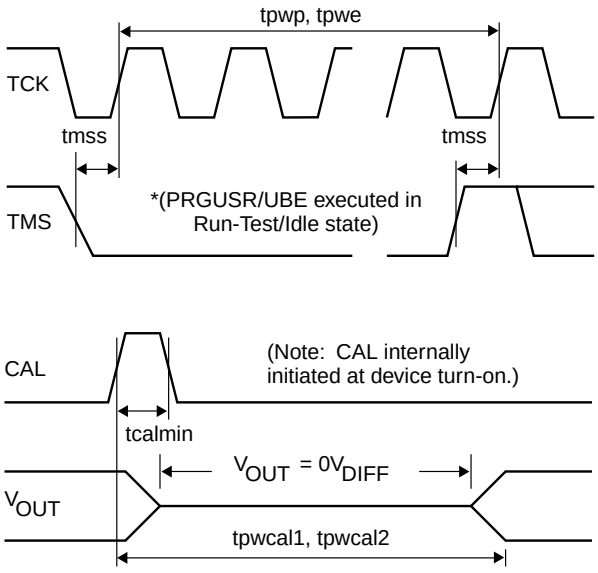
**Device Family**
**Device Number**
**Performance Grade**
01 = Standard
**Package**
P = PDIP, S = SOIC
**Grade**
Blank = Commercial
I = Industrial Temperature

## Timing Specifications

**$T_A$ = 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 | TDO Valid to Float Delay | | | | 60 | ns |
| trstmin | Minimum reset pulse width | | 40 | | | 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 outputs will stop responding to normal input stimulus. This is because all configuration information is erased and then re-written as part of a normal programming cycle, momentarily disrupting the input to output signal path. Behavior is not predictable during either of these steps since the analog outputs are not clamped during a programming cycle. Usually, however, the outputs will slew to either 0V (Ground) or 5V ($V_{supply}$) or 2.5V ($VREF_{OUT}$). This behavior is partially determined by conditions existing immediately prior to device reprogramming and intermediate configurations that occur during the process.

## Pin Descriptions

| Pin | Symbol | Name | Description |
|---|---|---|---|
| 1 | OUT2+ | Output 2(+) | Differential output pin, $V_{OUT+}$. (Plus complement of $V_{OUT}$ with respect to $VREF_{OUT}$, where differential $V_{OUT} = V_{OUT+} - V_{OUT-}$). |
| 2 | OUT2- | Output 2(-) | Differential output pin, $V_{OUT-}$. (Minus component, where differential $V_{OUT} = V_{OUT+} - V_{OUT-}$). |
| 3 | IN2+ | Input 2(+) | Differential input pin, $V_{IN+}$. (Plus $V_{IN}$, where differential $V_{IN} = V_{IN+} - V_{IN-}$). |
| 4 | IN2- | Input 2(-) | Differential input pin, $V_{IN-}$. (Minus component of differential $V_{IN}$, where $V_{IN} = V_{IN+} - V_{IN-}$). |
| 5 | TDI | Test Data In | Serial interface logic input pin. Input data valid on rising edge of TCK. |
| 6 | $\overline{TRST}$ | Test Reset | Serial interface logic reset pin (input). Asynchronously resets logic controller. Active low. Reset is equivalent of power-on default. |
| 7 | VS | Supply Voltage | Analog supply voltage pin (5V nominal). Should be bypassed to GND with 1µF and .01µF capacitors. |
| 8 | TDO | Test Data Out | Serial interface logic output pin. Input data valid on falling edge of TCK. |
| 9 | TCK | Test Clock | Serial interface logic clock pin (input). Best analog performance when TCK is idle. |
| 10 | TMS | Test Mode Select | Serial interface logic mode select pin (input). |
| 11 | IN4- | Input 4(-) | Differential input pin, $V_{IN-}$ |
| 12 | IN4+ | Input 4(+) | Differential input pin, $V_{IN+}$ |
| 13 | OUT4- | Output 4(-) | Differential output pin, $V_{OUT-}$ |
| 14 | OUT4+ | Output 4(+) | Differential output pin, $V_{OUT+}$ |
| 15 | OUT3+ | Output 3(+) | Differential output pin, $V_{OUT+}$ |
| 16 | OUT3- | Output 3(-) | Differential output pin, $V_{OUT-}$ |
| 17 | IN3+ | Input 3(+) | Differential input pin, $V_{IN+}$ |
| 18 | IN3- | Input 3(-) | Differential input pin, $V_{IN-}$ |
| 19 | $CMV_{IN}$ | Input for $V_{CM}$ Reference | Input pin for optional (external) analog Common-Mode Voltage ($V_{CM}$). Replaces $VREF_{OUT}$ (+2.5V) for any so programmed PACblock as its common-mode output voltage value. |
| 20 | CAL | Auto-Calibrate | Digital input pin. Commands an auto-calibration sequence on a rising edge. |
| 21 | GND | Ground | Ground pin. Should normally be connected to analog ground plane. |
| 22 | $VREF_{OUT}$ | Common-Mode Reference | Common-mode voltage reference output pin (+2.5V nominal). Must be bypassed to GND with a 0.1µF capacitor. |
| 23 | TEST | Test Pin | Manufacturing test pin. Connect to GND for proper circuit operation. |
| 24 | TEST | Test Pin | Manufacturing test pin. Connect to GND for proper circuit operation. |
| 25 | IN1- | Input 1(-) | Differential input pin, $V_{IN-}$ |
| 26 | IN1+ | Input 1(+) | Differential input pin, $V_{IN+}$ |
| 27 | OUT1- | Output 1(-) | Differential output pin, $V_{OUT-}$ |
| 28 | OUT1+ | Output 1(+) | Differential output pin, $V_{OUT+}$ |

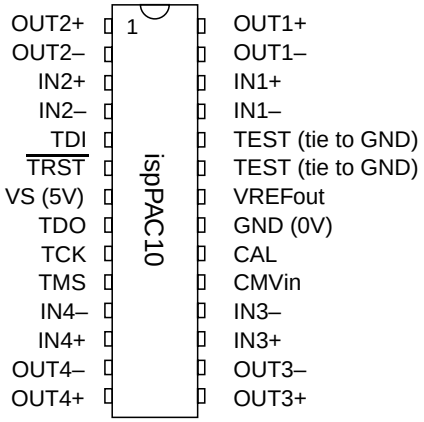
## Connection Notes

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 or internally under user programmable control.

2. All analog output pins are “hard-wired” to internal output devices and should be left open if not used. Outputs of uncommitted PACblocks are forced to $VREF_{OUT}$ (2.5V) and can be used as low impedance reference output buffers. $V_{OUT+}$ 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).

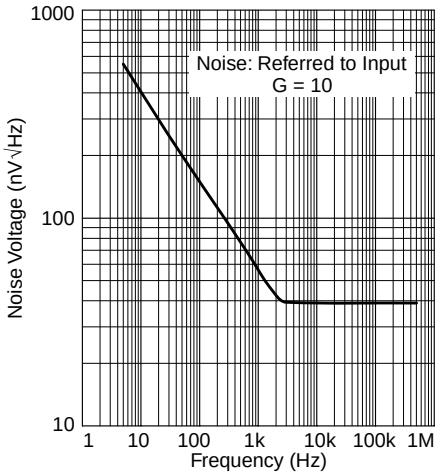
## Pin Configuration



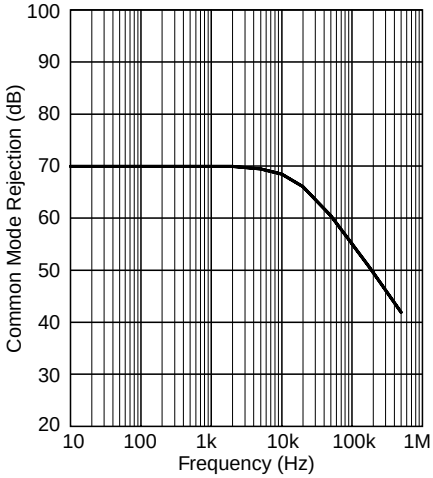

**28-Pin**
**Top View**

## Typical Performance Characteristics

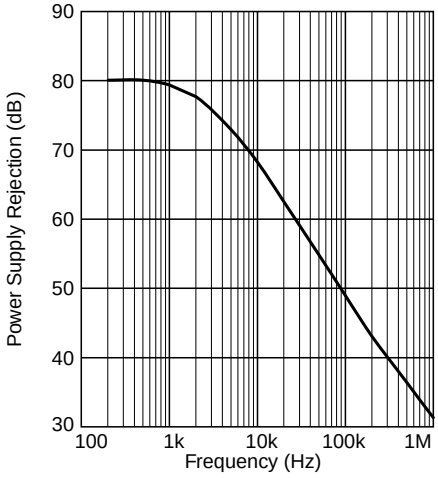
### Input Noise Spectrum



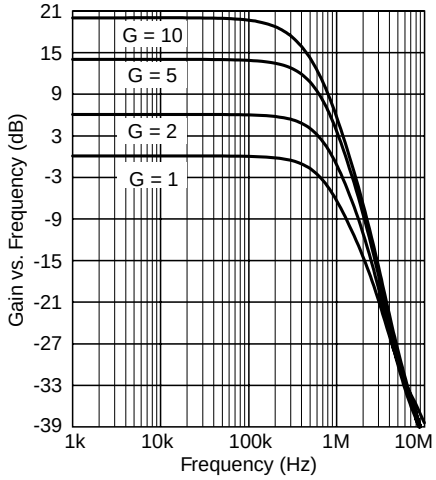

### CMR vs. Frequency



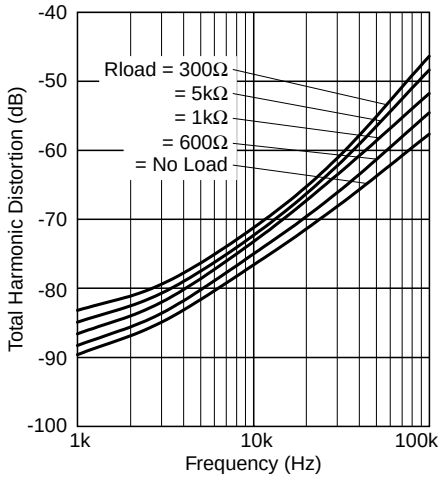

### PSR vs. Frequency



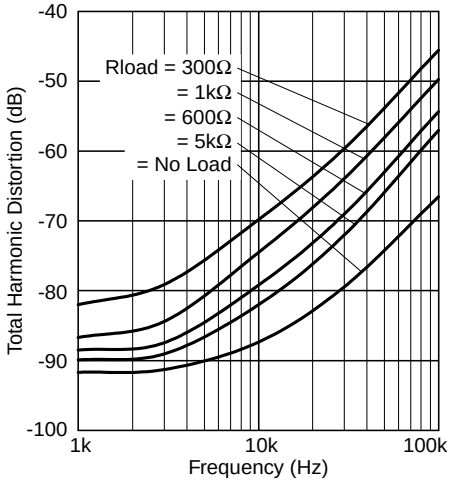

### Small Signal BW vs. Gain



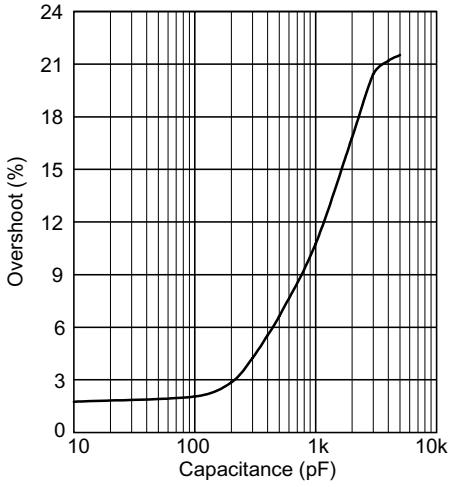

### THD vs. Frequency (Gain=1)



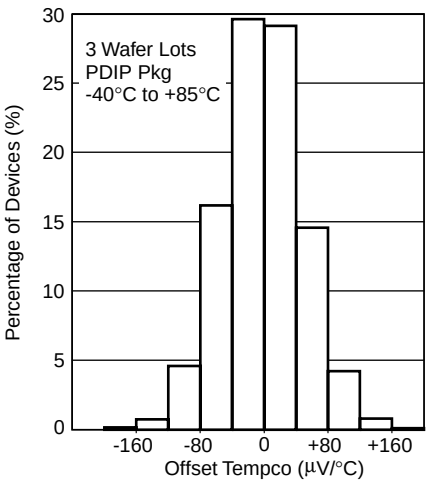

### THD vs. Frequency (Gain=10)



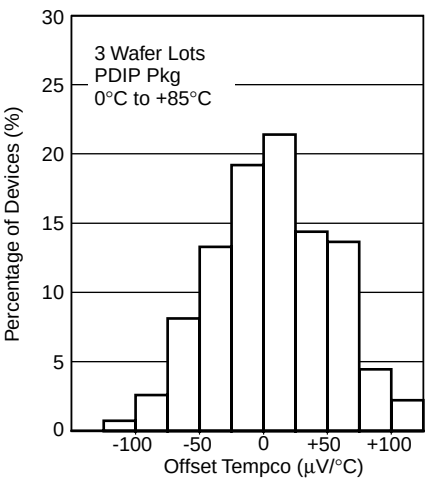

### Capacitive Load Handling




### $V_{OS}$ Tempco

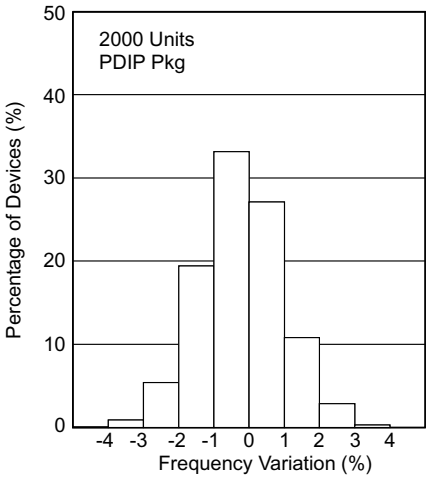



### $VREF_{OUT}$ Tempco

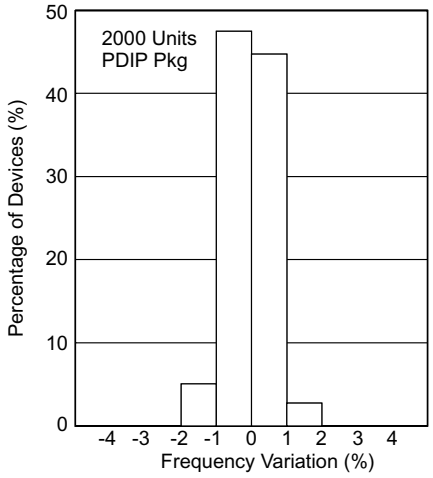


## Typical Performance Characteristics

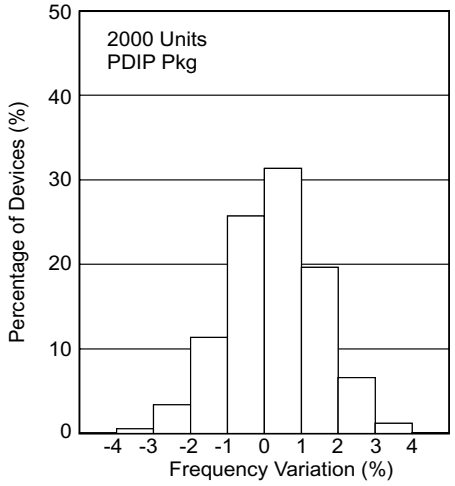
**10.34kHz Filter $F_c$ Accuracy**



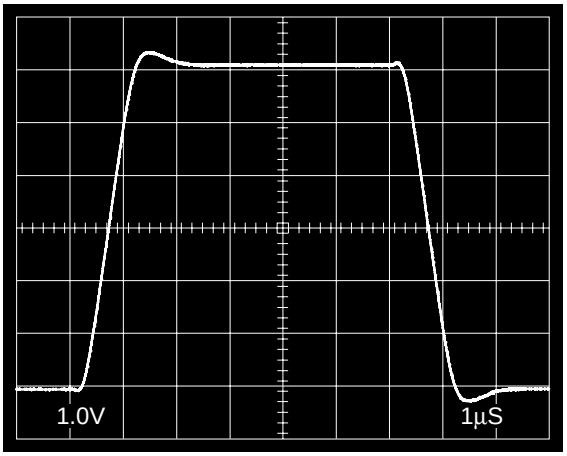

**46.46kHz Filter $F_c$ Accuracy**




**91.98kHz Filter $F_c$ Accuracy**

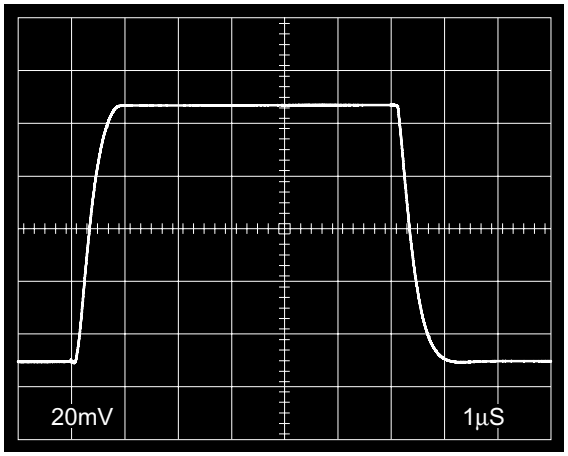



**Large-Signal Response**



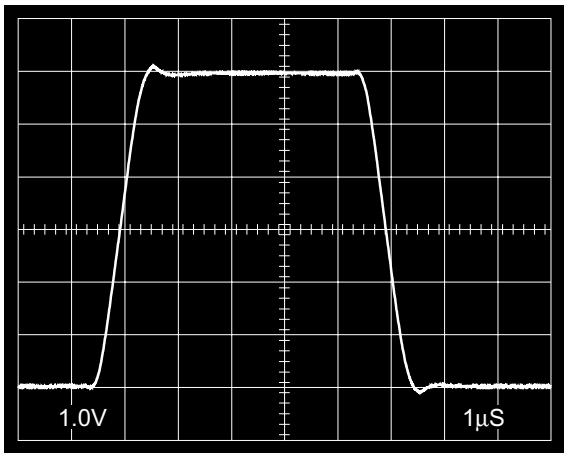

Gain = 1
Load = No Load

**Small-Signal Response**



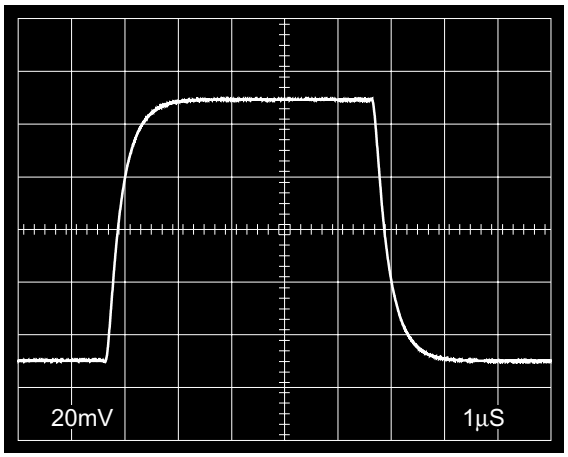

Gain = 1
Load = No Load

**Large-Signal Response with 600pF Load**




Gain = 1
Load = 600pF

**Small-Signal Response with 600pF Load**




Gain = 1
Load = 600pF

## Theory of Operation

### Introduction

The ispPAC10 consists of four programmable analog macrocells called PACblocks, each emulating a collection of operational amplifiers, resistors and capacitors. Requiring no external components, it flexibly implements basic analog functions such as precision filtering, summing/differencing, gain/attenuation and integration. Each PACblock contains a summing amplifier, two differential input instrument amplifiers, and an array of feedback capacitors. The capacitors, combined with a fixed value feedback element, provide more than 120 programmable poles between 10kHz to 100kHz with an absolute accuracy of 5.0 percent. Variable gain input instrument amplifiers make it possible to program any PACblock gain in integer steps between ±1 and ±10. More complex signal processing functions are performed by configuring additional PACblocks in combination with each other to achieve a variety of circuit functions.

The ispPAC10 architecture is fully differential from input to output. This effectively doubles dynamic range versus single-ended I/O. It also affords improved performance with regard to specifications such as input common mode rejection (CMR) and total harmonic distortion (THD).

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 AN6019.

Input polarity is programmable without affecting input impedance or dynamic performance, since no internal change is made other than routing to the input amplifier. Single-ended operation is achieved by using either one input and/or one output pin, as required, and adjusting gain settings to achieve desired output levels.

The ispPAC10 operates on a single 5V supply and includes an internal reference generating 2.5V. This reference is made available externally through the voltage common-mode reference or $VREF_{OUT}$ pin (Pin 22). The output common mode voltage is always referenced to 2.5V, regardless of the input common mode level. It is possible, when desired, to use an externally supplied voltage instead of $VREF_{OUT}$, however. This optional common-mode output voltage ($V_{CM}$) must be provided by the user via the $CMV_{IN}$ input pin (Pin 19). The only limitation is this reference voltage must be between 1.25V and 3.25V. When an external voltage is present, an ispPAC10 must be programmed, on a per-PACblock basis, to use the external reference instead of the internal 2.5V.

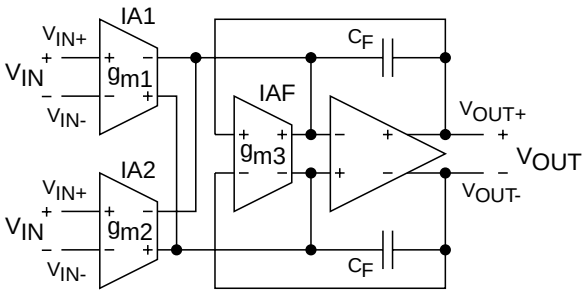
Configuring an ispPAC10 is accomplished using PAC-Designer, a Windows-based design environment. PAC-Designer includes an AC simulator for design verification prior to programming. The user can download the design to the ispPAC10 at any time via the device's IEEE Standard 1149.1 (JTAG) compliant serial port directly from the parallel port of a PC using an ispDOWNLOAD™ cable. Once downloaded, the circuit topology and component values are stored in non-volatile digital $E^2$CMOS cells on the ispPAC10 without any need for external programming voltages.

### Architecture

In all ispPAC products, individual programmable circuit functions called PACells™ are carefully combined to form larger analog macrocells or PACblocks. The ispPAC10 has four such PACblocks that incorporate specially configured PACells to perform amplification, summation, integration and filtering. Each of the four filtering/summation or "FilSum" PACblocks within ispPAC10 is comprised of three separate PACells, two input instrument amplifiers and an output summing amplifier (see Figure 1). The input amplifier PACells act as front-end gain stages for the FilSum PACblock and allow multiple signals to be summed together. The PACblock's output amplifier is similar to the familiar operational amplifier except that it has true differential outputs. Also included with each output amplifier is a filter capacitor array and switchable DC feedback path element. These components in combination enable the filtering and integrating functions of the FilSum PACblock.

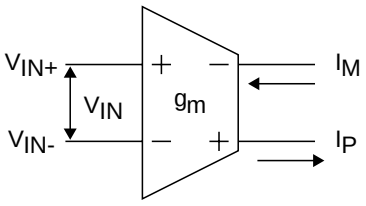
## Theory of Operation (Continued)

**Figure 1. FilSum (Filtering/Summation) PACblock Diagram**




Each FilSum PACblock actually employs three instrument amplifier (IA) PACells: two at the input (IA1 and IA2) and one as a feedback element around the op amp (IAF). The instrument amplifier PACells all have differential I/O and convert an input voltage to an output current (refer to Figure 2). This type of amplifier is sometimes referred to as an operational transconductance amplifier or OTA. When a differential input voltage is applied to these IAs, it is converted to a current proportional to the input signal. Because an AC signal common to both of the high impedance inputs of the IA does not create a net difference in the input signal, it is rejected by the amplifier. This characterizes the function of what is commonly known as an instrument amplifier and is a very desirable property because it acts to preserve the integrity of small signals in the presence of otherwise overwhelming noise.

**Figure 2. Instrument Amplifier PACell**



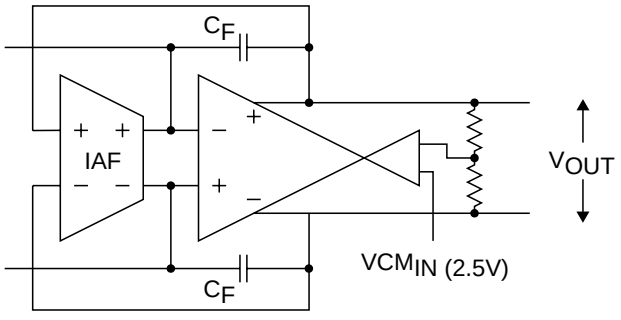

The two input instrument amplifiers have a programmable transconductance ($g_m$) value in 10 steps between 2µA/V and 20µA/V with programmable input polarity, whereas the feedback amplifier is fixed at 2µA/V. The IA PACells exhibit extremely high input impedance so they don't load circuitry driving them and their outputs can be enabled or disabled under $E^2$CMOS control, effectively switching them in and out of the FilSum PACblock circuitry. These simple characteristics permit a great deal of functionality: Signals can be summed, the resistive amplifier feedback can be removed to create an integrator, the sign of PACblock transfer function can be changed without changing the input or output loading characteristics. The FilSum PACblock can precisely filter, amplify or attenuate signals, always maintaining the high impedance input qualities of instrumentation amplifiers.

### FilSum PACblock Operation

All ispPAC10 inputs are differential, the input signal being the difference between input amplifier (IA) PACell pins $V_{IN+}$ (Positive Input) and $V_{IN-}$ (Minus Input). The common mode value of the input is ignored, and as long as the inputs are not within one volt of the supply rails, the part is in its linear operating region. As the input signal range exceeds these limits, distortion begins to increase until clipping occurs. This is discussed further in the advanced topics section.

The output is also differential, being the difference between output amplifier (OA) PACell pins $V_{OUT+}$ and $V_{OUT-}$. The output maintains high linearity to within 100mV of the supply rails under minimum load. The output has short circuit protection and is capable of driving resistive loads as low as 300Ω or capacitances as large as 1000pF. The output common mode voltage is maintained at $VREF_{OUT}$ independent of the input common mode level. That is, the output amplifier PACell "re-references" the common mode level of the input signal. This is accomplished by continuously sensing the output common mode voltage and comparing it to $VREF_{OUT}$ as shown in Figure 3, and makes it possible to use an individual FilSum PACblock as a $VREF_{OUT}$ reference as discussed in the section titled "Using $VREF_{OUT}$".

**Figure 3. Output $VREF_{OUT}$ Re-Referencing**




***Input Offset Auto-Calibration.*** A unique feature of the ispPAC10 is its ability to automatically calibrate itself to achieve very low offset error. This is done utilizing on-chip circuitry to perform an auto-calibration (auto-cal)

## Theory of Operation (Continued)

sequence every time the device is turned on, or anytime it is commanded externally via the CAL pin or by a JTAG programming command. With this feature, the degradation of device offset performance that could occur over time and temperature is dramatically reduced. Specifically, this means one PACblock of an ispPAC10 in a gain configuration of one is guaranteed to never have an input offset error greater than 1mV, after being auto-calibrated. For higher gain settings when offset is especially important, the error is not multiplied by gain, but is instead divided by it, due to the unique architecture of the ispPAC10. When an individual PACblock is configured in a gain of ten, that results in an input referred offset error that never exceeds 100µV.

Internally, auto-calibration is accomplished by simultaneous successive approximation routines (SAR) to determine the amount of offset error referred to each of the four PACblock output amplifiers of the ispPAC10. That error is then nulled by a calibration DAC for each output amplifier. The calibration constant is not stored in $E^2$CMOS memory, but is recomputed each time the device is powered up or auto-cal is otherwise initiated. Initiation of auto-cal occurs when an ispPAC10 is powered on as part of its normal power on routine, or by a positive going pulse to the CAL pin (Pin 20), or by issuing the appropriate JTAG command.

During auto-cal, all ispPAC10 outputs are driven to 0V and remain there until calibration is complete. The timing for the calibration process is generated internally. At power on, the sequence takes a maximum of 250ms, and when auto-cal is initiated via the CAL pin or by JTAG programming, it takes a maximum of 100ms to complete. The longer time required at power on insures the device power supply reaches its final value before calibration begins. Additional attempts to initiate auto-cal once calibration is in progress are ignored. Finally, the only direct indication of auto-cal completion will be the device's outputs returning to operational values from the 0V clamped state.

To insure maximum accuracy of the auto-cal procedure, all digital signals to the ispPAC10 should be suspended when calibration is in progress to avoid feed-through of noise to critical analog circuitry. This is especially true when auto-cal is initiated via JTAG command and the programming port is in use. There is sufficient time, however, to clock the JTAG controller back to its "reset" state without affecting the calibration process.

***Bandwidth Trim.*** The bandwidth of an OA PACell is trimmed during manufacturing by adjusting the amplifier's feedback capacitance to optimize the step response. The trimmed step response resembles that of a critically damped system with minimum overshoot.

The bandwidth trim ensures a nominal feedback capacitance is always present, limiting the small signal bandwidth of an OA PACell to about 600kHz when configured in a gain of 1 (G=1). This should not be confused with the gain-bandwidth product of the op amp within the output amplifier PACells which is approximately 5MHz. It is important to note that the individual output amplifiers are always in essentially the same fixed gain configuration and do not, therefore, contribute to a decrease in signal bandwidth at higher PACblock gain settings. Since the gain of an individual PACblock is determined by varying the $g_m$ of the input amplifier, bandwidth is not reduced in direct proportion to gain, as it would be in a traditional voltage feedback amplifier configuration. Specifically, small signal bandwidth is only reduced by a factor of 2, not the expected 10, with a PACblock gain setting change of G=1 to G=10. This is a significant advantage of the PACblock architecture.

***Pole Accuracy Trim***. Separate from the bandwidth trim capacitance, each FilSum PACblock contains a range of user selectable op amp feedback capacitance. This is made possible by a parallel arrangement of seven capacitors, each in series with an $E^2$CMOS switch. The user controls the position of the switches when selecting from the available capacitor values. The resulting capacitance is in parallel with the op amp feedback element, IAF, making 128 possible pole locations available. The capacitor values are not binarily weighted, instead they are chosen to optimize and concentrate pole spacing below 100kHz. There are 122 poles between 10kHz and 96kHz, which guarantees a step of no greater than 3.2% anywhere in that frequency range (to the nearest computed pole location). In fact, step size in over 50% of that range is less than 1.0%. Finally, capacitors are trimmed to achieve 5.0% accuracy (absolute) with regard to their nominal value.

## PACblock Transfer Function

The block diagram for a PACblock is shown in Figure 1. The transfer function for a transconductor is:

$$I_P = - g_m \cdot V_{IN} \quad (1)$$

$$I_M = g_m \cdot V_{IN} \quad (2)$$

Using KCL (Kirchoff's current law) at the op amp inputs and assuming the input is connected to IA1 only:

## Theory of Operation (Continued)

$$- V_{IN}\, g_{m1} + V_{OUT}\, g_{m3} + (V_{OUT+} - (V-))sC_F \quad (3a)$$

$$V_{IN}\, g_{m1} - V_{OUT}\, g_{m3} + (V_{OUT-} - (V+))sC_F \quad (3b)$$

where V- and V+ are the voltages at the op amp inverting and non-inverting inputs respectively. Because of feedback they are equal, so

$$\begin{aligned} -V_{IN}\, g_{m1} + V_{OUT}\, g_{m3} + (V_{OUT+} sC_F) \\ = V_{IN}\, g_{m1} - V_{OUT}\, g_{m3} + (V_{OUT-} sC_F) \end{aligned} \quad (4)$$

and the differential output voltage $V_{OUT}$ is the difference $V_{OUT+} - V_{OUT-}$ ,

$$\frac{V_{OUT}}{V_{IN}} = \frac{g_{m1}}{g_{m3} + \frac{sC_F}{2}} \quad (5a)$$

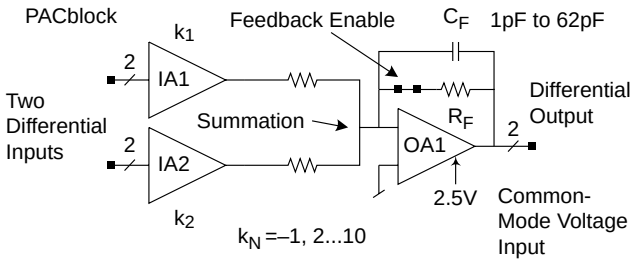
Since the PACblock has two separate inputs (IA1 and IA2) summed at the output amplifier input:

$$V_{OUT} = \frac{k_1 g_m V_{IN1} + k_2 g_m V_{IN2}}{g_{m3} + \frac{sC_F}{2}} \quad (5b)$$

The input amplifiers have a programmable gain of k·2µA/V ($g_{m1}$ and $g_{m2}$) where k is an integer from -10 to 10. The feedback amplifier transconductance $g_{m3}$ is fixed at 2µA/V, but may be disabled ($g_{m3}$ = 0) to open-circuit the output amplifier's resistive feedback. The programmable feedback capacitance lies in the range 1pF to 62pF.

The PACblock model from PAC-Designer is shown in Figure 4. The output amplifier is configured as an inverting mode op amp and illustrates the summing configuration. The input instrument amplifiers are shown to make it clear that unlike a typical inverting op amp, the PACblock input impedance is extremely high. The input amplifier (IA) transconductance (gain) is shown as the value (k) above or below each amplifier. The gain of IA1 and IA2 are independently programmable. Because the feedback transconductor IAF (designated here as $R_F$) can be disabled by the user, a user configurable switch is shown in series.

**Figure 4. PAC-Designer FilSum PACblock**

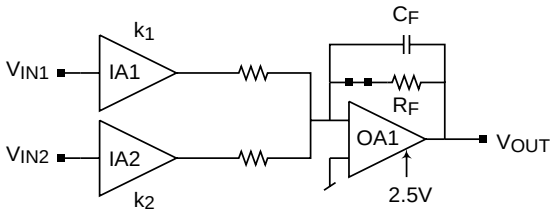



The FilSum PACblock implements two primary functions: the lossy integrator (low pass filter) and the integrator, both with gain.

***Lossy Integrator***. The lossy integrator's schematic within PAC-Designer is shown in Figure 5. Manipulating the PACblock transfer function of Equation 5 to better show the pole frequency yields:

$$V_{OUT} = \frac{k_1 V_{IN1} + k_2 V_{IN2}}{1 + \frac{sC_F}{2g_m}} \quad (6)$$

**Figure 5. PAC-Designer PACblock Lossy Integrator**



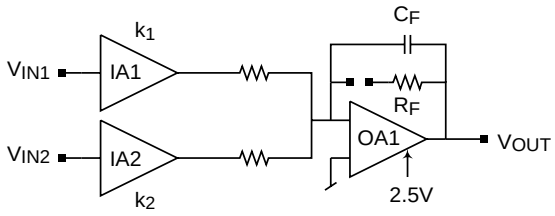

The DC gain of each input is set by $k_1$ or $k_2$ respectively, the gain constant for the input amplifiers. Below the pole frequency, this circuit can be viewed as a gain block. Because of the bandwidth trim capacitance, there is a minimum value of $C_F$ causing the bandwidth to be approximately 550kHz when the DC gain is one. For larger gains, the input amplifier bandwidth begins to dominate the overall PACblock response, limiting the bandwidth to about 330kHz when the gain is 10.

Examining this transfer function shows the pole frequency is $(1/2\pi)(2g_m/C)$. Since $g_m$ = 2µA/V and 1pF ≤ $C_F$ ≤ 62pF, then 600kHz ≥ $f_P$ ≥ 10kHz. Due to the selection options for feedback capacitance, there are at least 120 poles between 10kHz and 100kHz.

## Theory of Operation (Continued)

***Integrator***. Switching out $R_F$ (turning off IAF) removes the feedback element as shown in Figure 6. The integrator's transfer function can be derived from Equation 5b by setting $g_{m3} = 0$ (open circuit IAF ($R_F$)).

**Figure 6. PAC-Designer PACblock Integrator (IAF Disabled; $g_{m3} = 0$)**



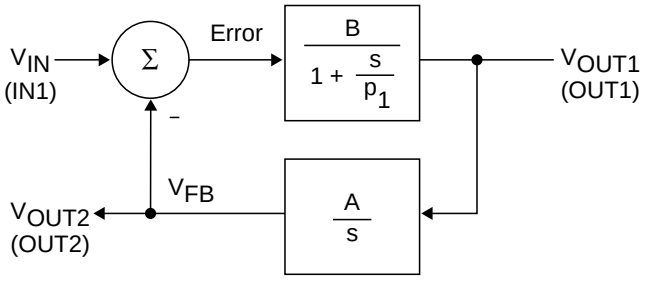

$$V_{OUT} = \frac{k_1 V_{IN1} + k_2 V_{IN2}}{\frac{sC_F}{2g_m}} \quad (7)$$

The integrator slope is proportional to 1/f and, for the case of a single input, the transfer function magnitude equals |k| when the frequency is $(1/2\pi)(2g_m/C)$. The integrator should not be used as a stand-alone circuit element. It needs to be used in configurations that provide DC feedback to ensure the output does not saturate, as illustrated by the biquad filter circuit below.

## Application Examples

***Biquad Filter***. By simply combining the two structures, the integrator providing feedback around the lossy integrator, creates a useful circuit. The block diagram is shown in Figure 7a

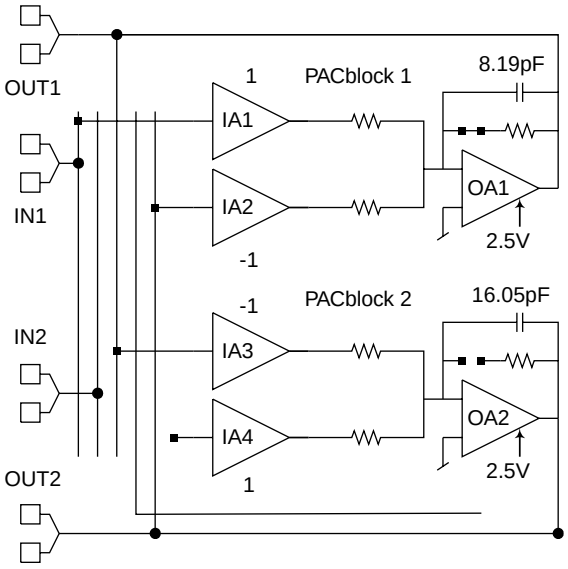
**Figure 7a. Biquad Bandpass Filter Block Diagram**




and the schematic from PAC-Designer is shown in Figure 7b. The transfer function OUT1(s)/IN1(s) is a band pass filter with programmable gain, Q and center frequency. Note the presence of DC feedback around the integrator.

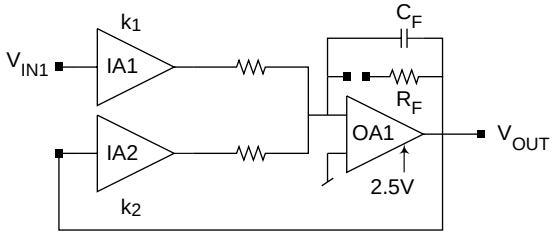
It can also be seen that the transfer function $V_{FB}(s)/V_{IN}(s)$ implements a lowpass filter. This application is discussed further in a separate application note.

**Figure 7b. Biquad Bandpass Filter Schematic**




***Attenuator***. The PACblock architecture makes variations possible on these two basic building blocks just described. An example uses summation to connect an input amplifier (IA2) in parallel with the feedback element ($R_F$), as shown in Figure 8.

**Figure 8. PACblock $A_v < 1$**




The result is a circuit whose transfer function is:

$$\frac{V_{OUT}}{V_{IN}} = -\frac{k_1}{k_2 - \frac{sC_F}{2g_m}} \quad (8)$$

The gains $k_1$ and $k_2$ are independently set by the user; this circuit can either amplify or attenuate an input signal. The one in the denominator is due to $R_F$; if $R_F$ is disabled,

## Theory of Operation (Continued)

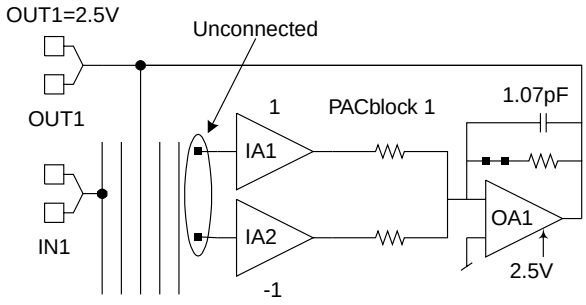
this term is eliminated. The level of attainable attenuation is as low as 1/11 (-20.8dB) with $R_F$ enabled or 1/10 (-20dB) with $R_F$ disabled.

When configuring a PACblock to attenuate, it is necessary to increase the value of feedback capacitance to maintain stability. Increasing feedback capacitance has the same beneficial effect as for a discrete op amp: It increases the network's phase margin which assists in maintaining stability.

### Using $VREF_{OUT}$

The $VREF_{OUT}$ output is high impedance and it should be buffered when used as a reference. A PACblock can be made into a $VREF_{OUT}$ buffer as shown in Figure 9. The PACblock inputs are left unconnected and the feedback closed. In this condition the input amplifiers are tied to $VREF_{OUT}$ and the output amplifier's outputs are thus forced to $VREF_{OUT}$ or 2.5V. Either output is now a $VREF_{OUT}$ voltage source. This reference has the same drive capabilities of any ispPAC10 output. However, do not short the two outputs together. There is a small potential difference between them which will cause a steady state current to flow, thus needlessly dissipating power.

**Figure 9. PACblock as $VREF_{OUT}$ Buffer**

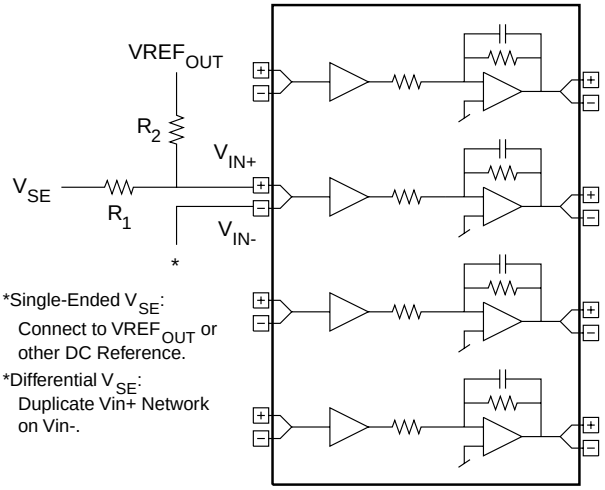



It is not always necessary to buffer the $VREF_{OUT}$ output. If it is used to reference a high impedance source, i.e., one that does not require more than 10µA, then it can be directly connected. An example is shifting the DC level of a signal connected to the input of a PACblock. In this case, the signal is AC coupled and "terminated" in $VREF_{OUT}$ through a minimum total resistance of 100kΩ. Referring to Figure 10b, if $R_{IN}$ is greater than 200kΩ then the $VREF_{OUT}$ pin may be used without buffering.

### Interfacing

When used in a single-supply system where the system common mode voltage is near $V_S/2$, signals may be directly connected to the ispPAC10 input. If the input signal does not have such a DC bias, then one needs to be added to the signal in order to accommodate the input requirements for the ispPAC10. A DC coupled bias can be added to a signal by using a voltage divider circuit as shown for one-half of the differential input in Figure 10a. Normally the choice for the reference DC voltage is the supply voltage, but other values may be used if necessary (and available).

**Figure 10a. DC Biasing an Input Signal**

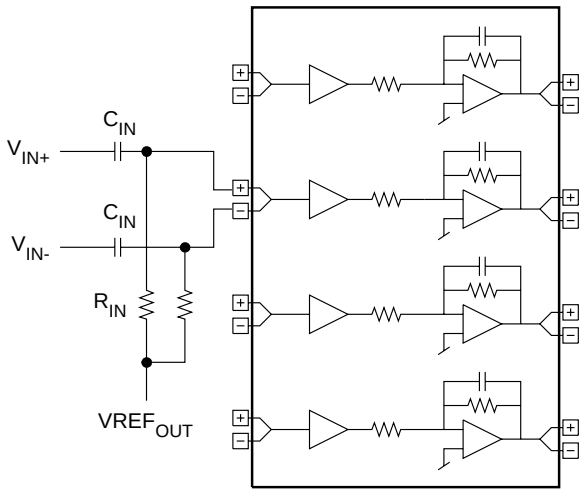



$$V_{IN+} = \frac{V_{SE}\ R_2}{R_1 + R_2} + \frac{VREF_{OUT}\ R_1}{R_1 + R_2}$$

Where DC coupling is not required, the input signal may be AC coupled as shown in Figure 10b. This circuit forms a high pass filter with a cutoff frequency of $1/(2\pi RC)$ and adds the necessary DC bias to the signal to accommodate the ispPAC10 input requirements. The DC reference should equal $V_S/2$, making $VREF_{OUT}$ the natural choice. The minimum resistance when using the $VREF_{OUT}$ buffer circuit of Figure 9 is 600Ω; when using the $VREF_{OUT}$ output pin it is 200kΩ (as discussed earlier).

## Theory of Operation (Continued)

**Figure 10b. AC-coupled Input with DC Bias**




### Single-ended Operation

Single-ended signals may be connected to the ispPAC10 input and one of the two differential ispPAC10 outputs can be used to drive single-ended circuitry. So, in addition to fully differential I/O, either the input, output or both may be used single-ended.

***Single-ended Input***. To connect the ispPAC10 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 (as in Figure 10b) 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 $VREF_{OUT}$ due to the re-referencing aspect of the FilSum PACblock. 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 ($3V_{PP}$ versus $6V_{PP}$) and since the output current capacity is the same whether driving differentially or single-ended, a single output can drive twice the load as the differential output (150Ω vs. 300Ω or 2000pF vs. 1000pF). If the load requires DC current, the amount available for voltage swing is reduced. The output is capable of 10mA, so any DC current raises the minimum allowable load impedance.

### Noise vs. Gain

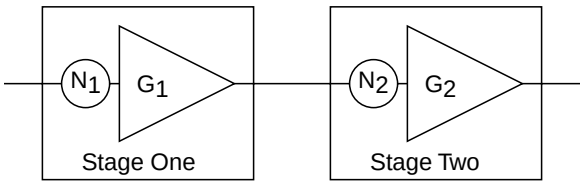
Noise gain is the gain of a circuit configuration to its combined input-referred circuit noise. The noise gain of an inverting op amp circuit is:

$$\text{Noise Gain} = 1 + \text{Closed Loop Voltage Gain} \qquad (9)$$

In this case, the noise gain of the circuit increases proportionally to the circuit gain.

A FilSum PACblock contains an input amplifier stage followed by an output amplifier. In this way it can be viewed as a system, with each of the components having its own contribution to the overall noise as shown in Figure 11. Both the output amplifier noise ($N_2$) and input amplifier noise ($N_1$) contribute to the overall noise performance, but the contribution due to the output amplifier dominates except at input gains near 10. The result is that the SNR of a FilSum PACblock is nearly constant versus gain. This is different than the behavior predicted by Equation 9.

**Figure 11. Multistage ispPAC Noise Diagram**




$G_2$ = Constant

$$\text{Output Noise Voltage} = G_1 G_2 \sqrt{N_1^2 + \left(\frac{N_2}{G_1}\right)^2} \qquad (10a)$$

If $N_2/G_1 > 3 \cdot N_1$, then

$$\text{Output Noise Voltage} \cong G_2 N_2 \qquad (10b)$$

There is a few dB decrease in SNR as the gain approaches 10. This characteristic implies the input amplifier noise contribution is approaching that of the op amp. As the gain of the input amplifier nears 10, its noise contribution in Equation 10a ($N_1$) approaches that of the op amp and becomes a factor in the overall output noise voltage, causing it to increase.

## Theory of Operation (Continued)

### Input Common-Mode Voltage Range

For the ispPAC10, 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 an extended 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 1, 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 1. Input Common-Mode Voltage Range Limitations**

| Input Voltage Magnitude (Volts-Peak) | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| $V_{CM-}$ | $V_{CM+}$ | G=1 | G=2 | G=3 | G=4 | G=5 | G=6 | G=7 | G=8 | G=9 | G=10 |
| **1.000** | **4.000** | 0.557 | 0.278 | 0.186 | 0.139 | 0.111 | 0.093 | 0.080 | 0.070 | 0.062 | 0.056 |
| **1.100** | **3.900** | 0.728 | 0.364 | 0.243 | 0.182 | 0.146 | 0.121 | 0.104 | 0.091 | 0.081 | 0.073 |
| **1.200** | **3.800** | 0.899 | 0.450 | 0.300 | 0.225 | 0.180 | 0.150 | 0.128 | 0.112 | 0.100 | 0.090 |
| **1.300** | **3.700** | 1.071 | 0.535 | 0.357 | 0.268 | 0.214 | 0.178 | 0.153 | 0.134 | 0.119 | 0.107 |
| **1.400** | **3.600** | 1.242 | 0.621 | 0.414 | 0.310 | 0.248 | 0.207 | 0.177 | 0.155 | 0.138 | 0.124 |
| **1.500** | **3.500** | 1.413 | 0.707 | 0.471 | 0.353 | 0.283 | 0.236 | 0.202 | 0.177 | 0.157 | 0.141 |
| **1.600** | **3.400** | 1.584 | 0.792 | 0.528 | 0.396 | 0.317 | 0.264 | 0.226 | 0.198 | 0.176 | 0.158 |
| **1.700** | **3.300** | 1.756 | 0.878 | 0.585 | 0.439 | 0.351 | 0.293 | 0.251 | 0.219 | 0.195 | 0.176 |
| **1.800** | **3.200** | 1.927 | 0.964 | 0.642 | 0.482 | 0.385 | 0.321 | 0.275 | 0.241 | 0.214 | 0.193 |
| **1.900** | **3.100** | 2.098 | 1.049 | 0.699 | 0.525 | 0.420 | 0.350 | 0.300 | 0.262 | 0.233 | 0.210 |
| **2.000** | **3.000** | 2.270 | 1.135 | 0.757 | 0.567 | 0.454 | 0.378 | 0.324 | 0.284 | 0.252 | 0.227 |
| **2.100** | **2.900** | 2.441 | 1.220 | 0.814 | 0.610 | 0.488 | 0.407 | 0.349 | 0.305 | 0.271 | 0.244 |
| **2.200** | **2.800** | 2.612 | 1.306 | 0.871 | 0.653 | 0.522 | 0.435 | 0.373 | 0.327 | 0.290 | 0.261 |
| **2.300** | **2.700** | 2.783 | 1.392 | 0.928 | 0.696 | 0.557 | 0.464 | 0.398 | 0.348 | 0.309 | 0.278 |
| **2.400** | **2.600** | 2.955 | 1.477 | 0.985 | 0.739 | 0.591 | 0.492 | 0.422 | 0.369 | 0.328 | 0.295 |
| **2.426** | **2.574** | 3.000* | 1.500* | 1.000* | 0.750* | 0.600* | 0.500* | 0.429* | 0.375* | 0.333* | 0.300* |
| **2.500** | **2.500** | 3.126 | 1.563 | 1.042 | 0.782 | 0.625 | 0.521 | 0.447 | 0.391 | 0.347 | 0.313 |

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

## Software-Based Design Environment

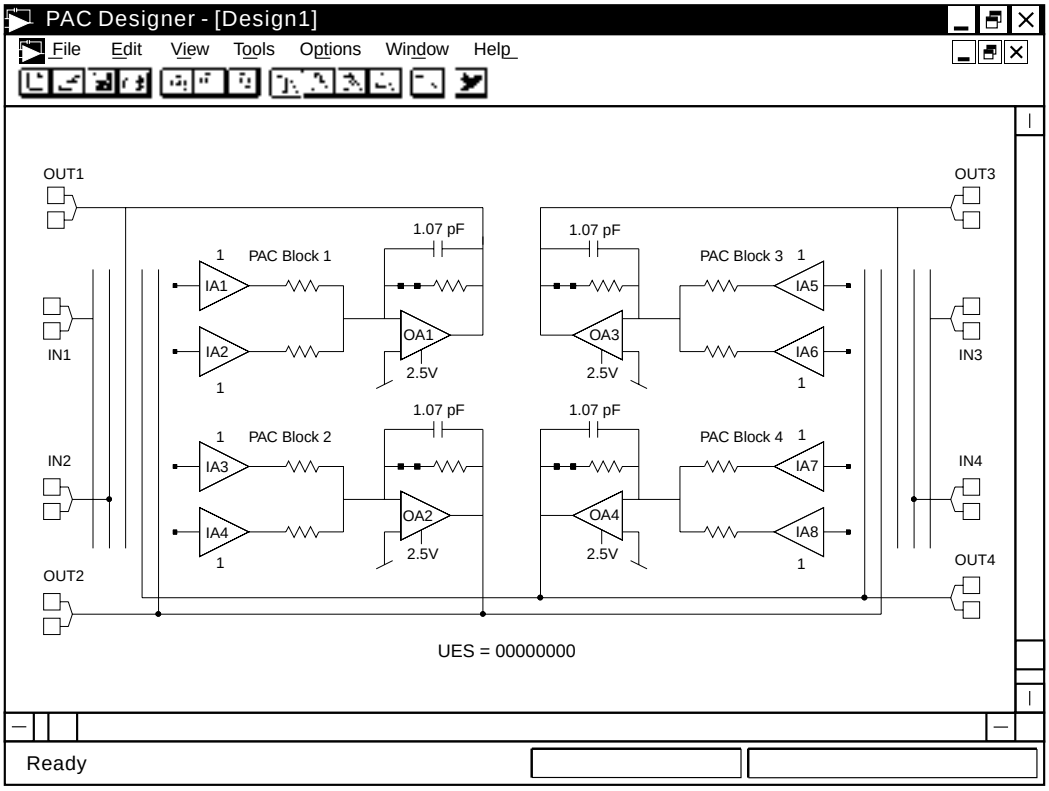
### Design Entry Software

Designers configure the ispPAC10 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 ispPAC10. A library of configurations is included with basic solutions and examples of advanced circuit techniques. 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 12, provides access to all configurable ispPAC10 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. These stand-alone programs are called circuit generator macros. An example of such a macro is the biquad filter generator supplied with PAC-Designer. With this macro, filter parameters such as gain, Q and corner frequency are input directly and then automatically converted to a schematic configuration. The application example shown in Figure 7b was generated using the biquad filter generator macro. More information on this and other topics is included in the on-line documentation as well as ispPAC application notes.

**Figure 12. Initial PAC-Designer Schematic Design Entry Screen**



## Software-Based Design Environment (Continued)

### 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 13 shows a typical screen plot of the gain/phase simulator. In it are the input to output response curves of a 2nd order biquad filter similar to the implementation illustrated in Figure 7b. In this example, the lowpass and bandpass characteristics of the filter are seen.

The simulator is capable of displaying up to four separate input to output responses. This allows multiple signal paths 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.

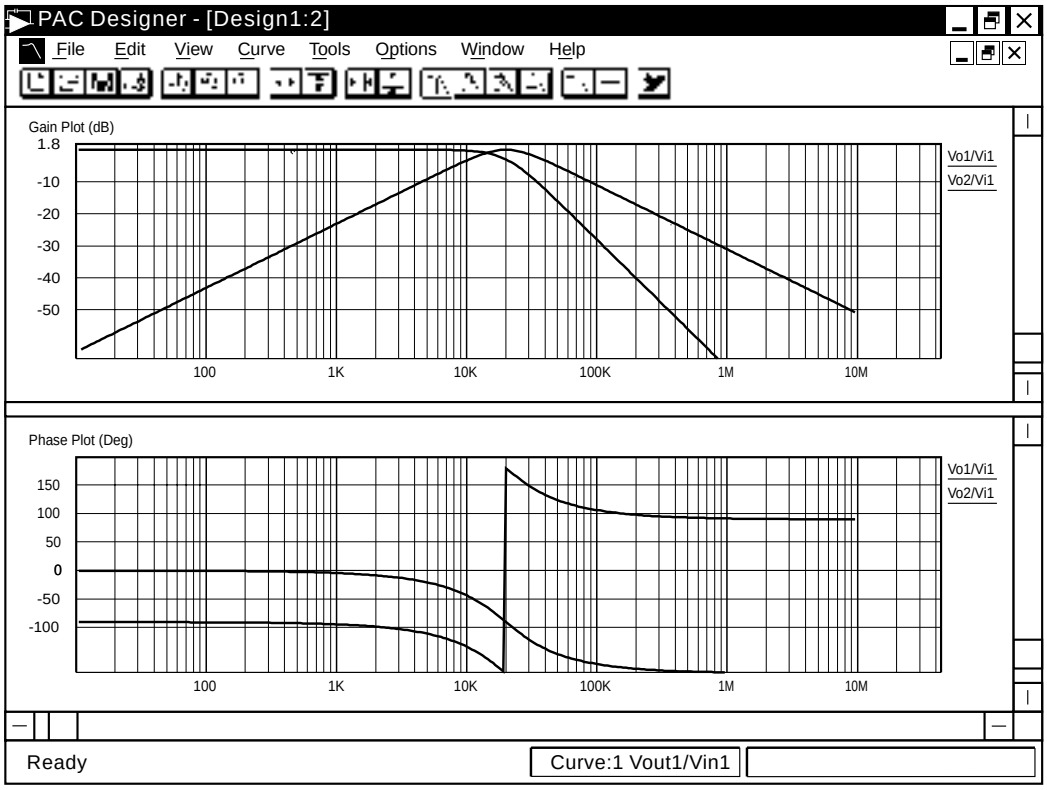
### In-System Programming

The ispPAC10 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 (JTAG) compliant serial port interface at normal logic levels. Once a device is programmed, all configuration information is stored in on-chip, non-volatile $E^2$CMOS 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 ispPAC10. It contains 8 bits that can be configured by the user to store unique data such as ID codes, revision numbers or inventory control data.

**Figure 13. PAC-Designer Simulation Plot Screen (Biquad Filter Configuration)**



## In-System Programmability

### Electronic Security

An electronic security "fuse" (ESF) bit is provided in every ispPAC10 device to prevent unauthorized readout of the $E^2$CMOS 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. Parts can then be ordered through the usual supply channels with the user's specific configuration already preloaded into the parts. 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

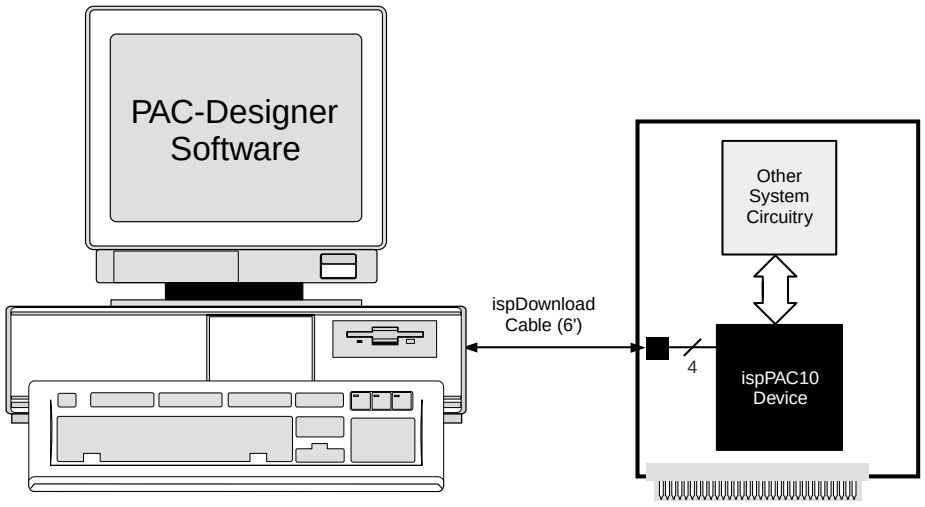
Included in the basic ispPAC10 Design Kit is an engineering prototype board that is connected to the parallel port of a PC. It demonstrates proper layout techniques for the ispPAC10 and can be used in real time to check circuit operation as part of the design process. Input and output connections as well as a "breadboard" circuit area are provided to speed debugging of the circuit.

### Serial Port Programming Interface

Communication with the ispPAC10 is facilitated via an IEEE 1149.1 test access port (TAP). It is used by the ispPAC10 as a serial programming interface, and not for boundary scan test purposes. There are no boundary scan logic cells in the ispPAC10 architecture. This does not prevent the ispPAC10 from functioning correctly, however, when placed in a valid serial chain with other IEEE 1149.1 compliant devices.

A brief description of the ispPAC10 serial interface follows. For complete details of the reference specification, refer to the publication, Standard Test Access Port and Boundary-Scan Architecture, IEEE Standard 1149.1-1990 (which now includes IEEE Standard 1149.1a-1993).

**Figure 14. Configuring the ispPAC10 "In-System" from a PC Parallel Port**

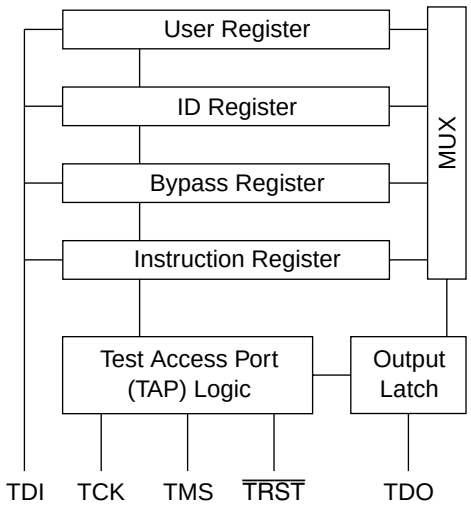


## IEEE Standard 1149.1 Interface

### Overview

An IEEE 1149.1 test access port (TAP) provides the control interface for serially accessing the digital I/O of the ispPAC10. 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 $E^2$CMOS cells. It is these non-volatile cells that determine the configuration of the ispPAC10. By cycling the TAP controller through the necessary states, data can also be shifted out of the user register to verify the current ispPAC10 configuration. Instructions exist to access all data registers and perform internal control operations.

**Figure 15. ispPAC10 TAP Registers**




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 ispPAC10, the bypass register is a 1-bit shift register that provides a short path through the device when boundary testing or other operations are not being performed. The ispPAC10, as mentioned, has no boundary-scan logic and therefore no boundary scan register. All instructions relating to boundary scan operations place the ispPAC10 in the BYPASS mode to maintain compliance with the specification. The optional identification register described in IEEE 1149.1 is also included in the ispPAC10. One additional data register included in the TAP of the ispPAC10 is the Lattice defined user register. Figure 15 shows how the instruction and various data registers are placed in an ispPAC10.

### 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 16. 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. Return to the Test-Logic-Reset state can also be immediately accomplished by placing a logic low on the Test-Reset (TRST#) pin. Test-Logic-Reset is also the power-on default state.

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. This condition will occur independently anytime a hardware reset (TRST#) is executed and is also the power-on default. It will always enable the ID Register for readout if no other instruction is loaded prior to a Shift-DR opera-

## IEEE Standard 1149.1 Interface (Continued)

tion. 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.

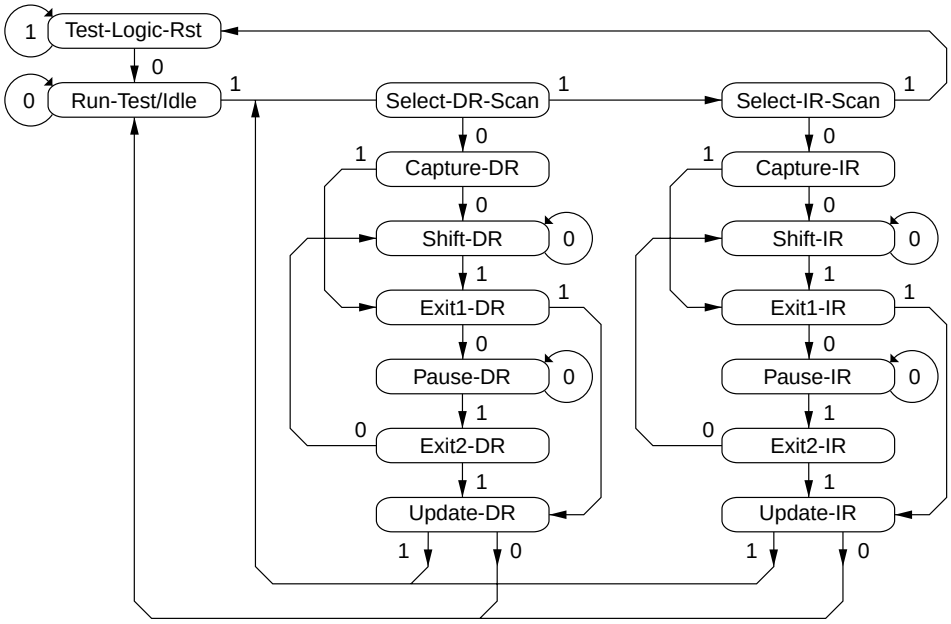
### Test Instructions

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 2 bits, with only the BYPASS and EXTEST instruction code patterns being specifically called out (all ones and all zeroes respectively). The ispPAC10 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 ispPAC10, the instruction word length is 5 bits. All ispPAC10 instructions available to users are shown in Table 2.

***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 ispPAC10. 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 ispPAC10 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 2.

**Figure 16. Test Access Port (TAP) Contoller State Diagram**




Note: The value shown adjacent to each state transition in this figure represents the signal present at TMS at the time of a rising edge at TCK.

## IEEE Standard 1149.1 Interface (Continued)

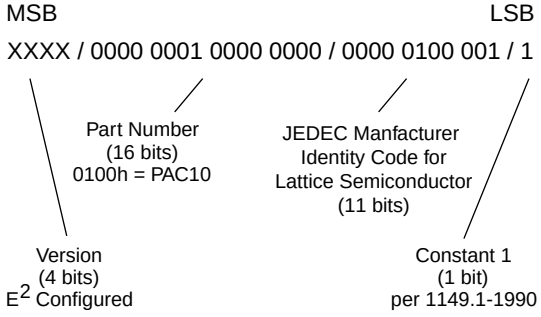
The ***EXTEST*** (external test) instruction is required and would normally place the device into an external boundary test mode while also enabling the Boundary-Scan Register to be connected between TDI and TDO. Again, since the ispPAC10 has no boundary-scan logic, the device is put in the BYPASS mode to ensure specification compatibility. The bit code of this instruction is defined by the 1149.1 standard to be all zeros.

**Table 2. ispPAC10 TAP Instructions**

| Instruction | Code | Description |
|---|---|---|
| EXTEST | 00000 | External test. Default to BYPASS. |
| ADDUSR | 00001 | Address User data register. |
| UBE | 00010 | User bulk erase. |
| VERUSR | 00011 | Verify User data register. |
| PRGUSR | 00100 | Program User data register. |
| IDCODE | 01101 | Read Identification data register. |
| ENCAL | 10000 | Enable Calibration sequence. |
| SAMPLE | 11110 | Sample/Preload. Default to BYPASS. |
| BYPASS | 11111 | Bypass (connect TDI to TDO). |

The optional ***IDCODE*** (identification code) instruction is incorporated in the ispPAC10 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 17). Access to the Identification Register is immediately available, via a TAP data scan operation, after power-up of the device, after a reset using the optional TRST pin, or by issuing a Test-Logic-Reset instruction. The bit code for this instruction is defined by Lattice as shown in Table 2.

**The Figure 17. Identification Code (IDCODE) 32-Bit Binary Word for Lattice ispPAC10**




***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 PROGUSR (program user) instruction to shift in a new configuration and follows a VERUSR (verify user) instruction to shift out the current configuration. The bit code for this instruction is shown in Table 2.

The ***PRGUSR*** (program user) is a Lattice instruction that enables the data shifted into the user register to be programmed into the non-volatile $E^2$CMOS memory of the ispPAC10 and thereby alter its configuration. The user register is a 109-bit shift register that contains all the user-controlled parametric and interconnect data pertaining to the configuration of the ispPAC10. 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 this instruction is shown in Table 2.

***VERUSR*** (verify user) is the next Lattice instruction and causes the current configuration of the ispPAC10 to be loaded into the user register. This operation doesn't interrupt operation of the device. The current configuration can then be shifted out of the user register immediately after an ADDUSR instruction is executed. The bit code for this instruction is shown in Table 2.

***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 0V 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 2.

## IEEE Standard 1149.1 Interface (Continued)

The last Lattice instruction is ***UBE*** (user bulk erase). Operation of the device is interrupted during UBE, after 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 UBE is used to return all user bits to a zero state at the same time. A UBE usually proceeds a PRGUSR 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 constraints apply to UBE as for PRGUSR. The bit code for this instruction is shown in Table 2.

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.

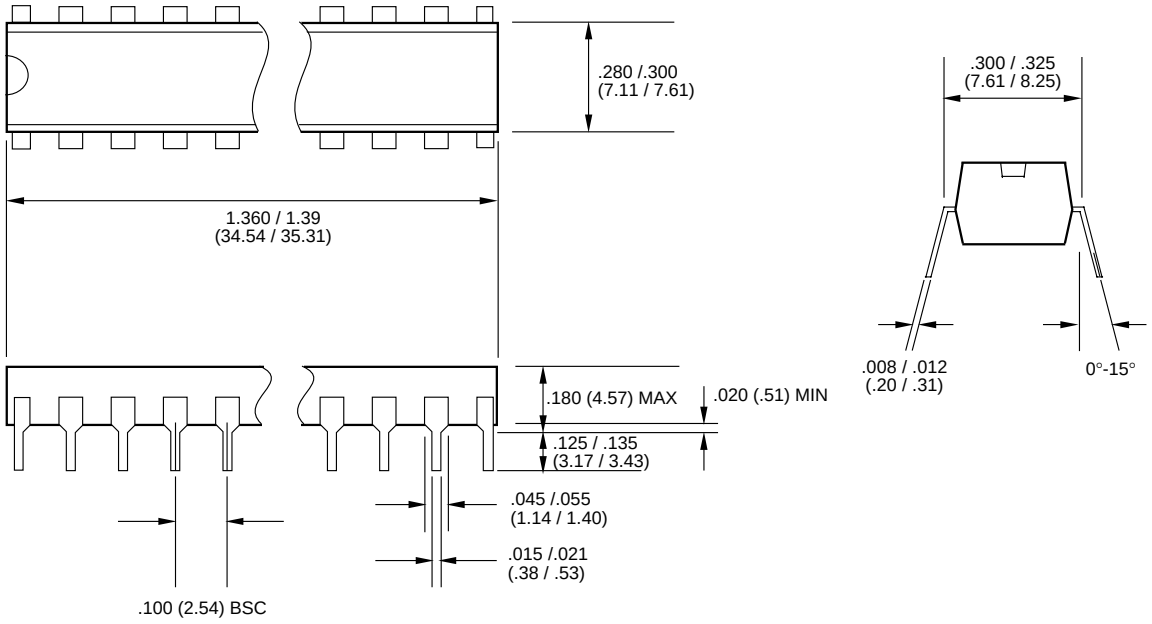
It is recommended that when all serial interface operations are completed, the TAP controller be reset and left in the Test-Logic-Reset state (the power-up default) and the TCK and TMS inputs idled. This will insure the best analog performance possible by minimizing the effects of digital logic “feed-through.”

## Package Diagrams

**28-Pin Plastic DIP**

**Dimensions in Inches MIN./MAX.**

(Dimensions in millimeters, shown in parenthesis, are for reference only)




**28-Pin Plastic SOIC**

**Dimensions in Inches MIN./MAX.**

(Dimensions in millimeters, shown in parenthesis, are for reference only)