Agilent 34950A
64-Bit Digital I/O and Counter Module

User's Guide

Agilent Technologies, Inc.
Printed in Malaysia
Edition 1
June 2008 E0608

34980-90050
34980-90050

Agilent Technologies

Notices

Manual Part Number

34980-90050

Edition

First Edition, June 2008

Printed in Malaysia

Agilent Technologies, Inc.
3501 Stevens Creek Blvd
Santa Clara, CA 95052 USA

Software Revision

This guide is valid for the firmware that was installed in the instrument at the time of manufacture. However, upgrading the firmware may add or change product features. For the latest firmware and documentation, go to the product page at:

www.agilent.com/find/34980A

Warranty

The material contained in this document is provided "as is," and is subject to being changed, without notice, in future editions. Further, to the maximum extent permitted by applicable law, Agilent disclaims all warranties, either express or implied, with regard to this manual and any information contained herein, including but not limited to the implied warranties of merchantability and fitness for a particular purpose. Agilent shall not be liable for errors or for incidental or consequential damages in connection with the furnishing, use, or performance of this document or of any information contained herein. Should Agilent and the user have a separate written agreement with warranty terms covering the material in this document that conflict with these terms, the warranty terms in the separate agreement shall control.

Technology Licenses

The hardware and/or software described in this document are furnished under a license and may be used or copied only in accordance with the terms of such license.

Restricted Rights Legend

U.S. Government Restricted Rights. Software and technical data rights granted to the federal government include only those rights customarily provided to end user customers. Agilent provides this customary commercial license in Software and technical data pursuant to FAR 12.211 (Technical Data) and 12.212 (Computer Software) and, for the Department of Defense, DFARS 252.227-7015 (Technical Data - Commercial Items) and DFARS 227.7202-3 (Rights in Commercial Computer Software or Computer Software Documentation).

Safety Notices

CAUTION

A **CAUTION** notice denotes a hazard. It calls attention to an operating procedure, practice, or the like that, if not correctly performed or adhered to, could result in damage to the product or loss of important data. Do not proceed beyond a **CAUTION** notice until the indicated conditions are fully understood and met.

WARNING

A WARNING notice denotes a hazard. It calls attention to an operating procedure, practice, or the like that, if not correctly performed or adhered to, could result in personal injury or death. Do not proceed beyond a WARNING notice until the indicated conditions are fully understood and met.

Additional Safety Notices

The following general safety precautions must be observed during all phases of operation of this instrument. Failure to comply with these precautions or with specific warnings or instructions elsewhere in this manual violates safety standards of design, manufacture, and intended use of the instrument. Agilent Technologies assumes no liability of the customer's failure to comply with the requirements.

General

Do not use this products in any manner not specified by the manufacturer. The protective features of this product may be impaired if it is used in a manner not specified in the operation instructions.

Before Applying Power

Verify that all safety precautions are taken. Make all connections to the unit before applying power.

Ground the Instrument

This product is provided with protective earth terminals. To minimize shock hazard, the instrument must be connected to the ac power mains through a grounded power cable, with the ground wire firmly connected to an electrical ground (safety ground) at the power outlet. Any interruption of the protective (grounding) conductor or disconnection of the protective earth terminal will cause a potential shock hazard that could result in personal injury.

Do Not Operate in an Explosive Atmosphere

Do not operate the instrument in the presence of flammable gases or fumes.

Do Not Remove the Instrument Cover

Only qualified, service-trained personal who are aware of the hazards involved should remove instrument covers. Always disconnect the power cable and any external circuits before removing the instrument cover.

Do Not Modify the Instrument

Do not install substitute parts or perform any unauthorized modification to the product. Return the product to an Agilent Sales and Service Office for service and repair to ensure that safety features are maintained.

In Case of Damage

Instruments that appear damaged or defective should be made inoperative and secured against unintended operation until they can be repaired by qualified service personnel.

Safety Symbols

∼	Alternating current
⏚	Frame or chassis terminal
⏻	Standby supply. Unit is not completely disconnected from ac mains when switch is off
⚡	Caution, risk of electric shock
⚠	Caution, refer to accompanying description

Waste Electrical and Electronic Equipment (WEEE) Directive 2002/96/EC

This product complies with the WEEE Directive (2002/96/EC) marking requirement. The affixed product label (see above) indicates that you **must not discard this electrical/electronic product in domestic household waste**.

Product Category: With reference to the equipment types in the WEEE directive Annex 1, this product is classified as a "Monitoring and Control instrumentation" product.

To return unwanted products, contact your local Agilent office, or go to www.agilent.com/environment/product for more information.

Technical Support

If you have questions about your shipment, or if you need information about warranty, service, or technical support, contact Agilent Technologies:

In the United States: (800) 829-4444
In Europe: 31 20 547 2111
In Japan: 0120-421-345

Or go to www.agilent.com/find/assist for information on contacting Agilent in your country of specific location. You can also contact your Agilent Technologies Representative.

The **Declaration of Conformity (DoC)** for the 34980A mainframe instrument can be found on page iii in the *34980A Mainframe User's Guide*. That DoC applies to the 34980A mainframe and all available plug-in modules.

Contents

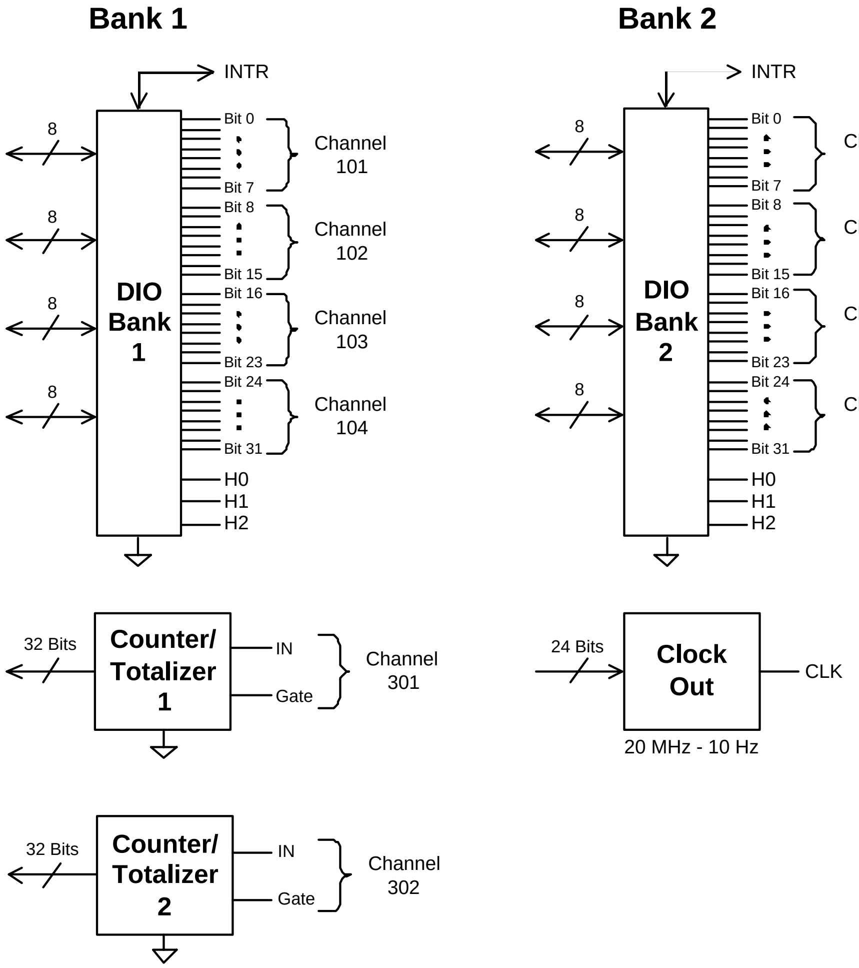
34950A 64-Bit Digital I/O Module with Memory and Counter

The 34950A has 64-bits of general-purpose digital I/O grouped in 8-bit channels with programmable polarity, input thresholds, and output levels. The module is segmented into two banks of four 8-bit channels. Each bank has 64 Kb of volatile memory for pattern capture and pattern generation with hardware interrupt capability. Up to three pins of handshaking are available for each bank of 32 bits.

The module also has two 10 MHz frequency counter/totalizer measurement input channels and a programmable clock output for frequency synchronization or general clocking needs.

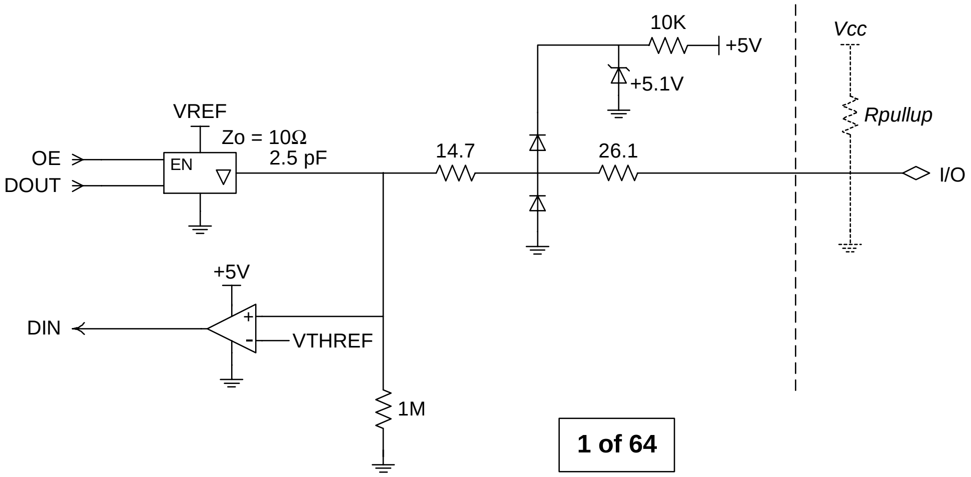
Bank and Channel Assignments

The digital channels are numbered by bank; 101 through 104 and 201 through 204 for banks 1 and 2 respectively. The counter/totalizer channels are assigned channel numbers 301 and 302. The programmable clock is not assigned a channel number.



Electrical Characteristics for Digital I/O Lines

The following simplified schematic shows the interface circuitry and specifications *for each* of the 64 digital I/O lines:



Active Drive:	
V_{in}	0V – 5V
V_{out} (L)	$0.24V < V_{out} < 0.55V$ $4mA < I_{out} < 24mA$
V_{out} (H)	$1.6V < V_{out} < 5V$ $-4mA < I_{out} < -24mA$

Open Collector:	
V_{out}	0V – 5V $-4\ mA < I_{out} < -24mA$
V_{cc} (< 2V) V_{cc} (> 2V)	$215\Omega < R_{pullup} < 1\ k\Omega$ $215\Omega < R_{pullup} < 10\ k\Omega$

Operating Considerations

See the *Introduction to the Plug In Modules* chapter of the 34980A Mainframe User’s Guide for detailed environmental operating conditions for the 34980A mainframe and its installed modules. That guidance sets maximum per channel current and power ratings at rated voltage for pollution degree 1 (dry) and pollution degree 2 (possible condensation) conditions, for the Digital I/O module.

Basic Digital I/O Operations

Channel Numbering and Width

The digital channels are numbered by bank; 101 through 104 and 201 through 204 for banks 1 and 2 respectively.

Using SCPI commands you can group digital I/O channels together to allow 16- or 32-bit operations. The first and third channels on a bank can be control channels. Width and direction of the memory operations are controlled by the width and direction of the first channel on the bank (i.e., 101 or 201). In the SCPI language for the 34950A, BYTE refers to 8-bit operations, WORD refers to 16-bit operations, and LWORd refers to 32-bit operations.

This table illustrates how the channels are numbered for each operating configuration.

	Bank 1				**Bank 2**			
	Channel							
BYTE (default)	**101**	**102**	**103**	**104**	**201**	**202**	**203**	**204**
	8-bits	8-bits	8-bits	8-bits	8-bits	8-bits	8-bits	8-bits
WORD	**101**		**103**		**201**		**203**	
	16-bits		16-bits		16-bits		16-bits	
LWORd	**101**				**201**			
	32-bits				32-bits			

Reading Digital Data

The simplest way to read a digital channel is using the MEASure:DIGital? query. This query sets the channel to be an input channel and sets all other channel parameters to the default settings.

For example, sending the following SCPI command to a Digital I/O module installed in slot 1 of the mainframe will read the value of the 8-bit channel 102. An unsigned integer value is returned that represents the state of the 8 bits on channel 102.

```
MEAS:DIG? BYTE, (@1102)
```

By adding parameters to the command, you can set the channel width, polarity, and threshold for read. For example, sending the following SCPI command you can read the 32-bit channel 201.

```
MEAS:DIG? LWOR, (@1201)
```

To read digital data with more control over the channel parameters, use the SCPI CONFigure and SENSe commands. The CONFigure commands set up the digital I/O channel parameters. For example, sending the following SCPI command to a Digital I/O module installed in slot 1 of the mainframe, sets a 16-bit input channel (103) to use a 2.5 V input threshold, and normal polarity.

```
CONF:DIG WORD, 2.5, NORM, (@1103)
```

Once configured, the data is read using the following command.

```
SENS:DIG:DATA:WORD? (@1103)
```

You may also read an individual bit using the SENSe commands. This allows you to check the state of an individual bit in a channel without having to create an input mask. For example, the following command returns the state of bit 3 in the channel 101 byte.

```
SENS:DIG:DATA:BIT? 3, (@1101)
```

The acceptable range for the bit parameter is based on the channel width as shown below:

- BYTE (8-bit): <bit> can range from ‘0’ to ‘7’
- WORD (16-bit): <bit> can range from ‘0’ to ‘15’
- LWORd (32-bit): <bit> can range from ‘0’ to ‘31’

The SENSe command differs from the MEASure command in that it will not change the direction (input or output) of the channel. If the channel is configured as an output, the SENSe command will return the value being driven.

Writing Digital Data

To write digital data, set the channel output parameters using the SOURce commands. For example, sending the following SCPI commands to a Digital I/O module in slot 1 sets a 32-bit channel to use normal polarity, with active drive and a 'set' output voltage of 4 volts.

```
CONF:DIG:WIDT LWOR,(@1201)
CONF:DIG:POL NORM,(@1201)
SOUR:DIG:DRIV ACT,(@1201)
SOUR:DIG:LEV 4,(@1201)
```

The width and polarity parameters apply to both input and output operations.

You can set a channel to output in either active drive or open collector configurations. When set to ACTive, the module drives the digital lines for both high and low. The voltage level that represents a logic '1' can be set using the SOURce:DIGital:LEVel command. Output voltages can range from 1.65 V to 5 V.

When the channel is set to OCOLlector, lines are driven low, but set to high impedance (Hi-Z) when asserted. In the open collector mode, multiple lines can be connected together by providing external pull-ups.

NOTE When using external pull-ups in the open collector mode, the outputs will not exceed 5 V.

Once a channel has been configured, write digital data to the channel using the SOURce:DIGital:DATA command.

```
SOUR:DIG:DATA:LWOR 26503,(@1201)
```

You may also use a hexadecimal format to represent values in the commands. For example, to send the decimal value of 26503 in hex use the command form:

```
SOUR:DIG:DATA:LWOR #h6787,(@1201)
```

NOTE Writing to a channel automatically configures the channel as an output.

Note that the data should match the channel width configured using CONFigure:DIGital:DATA:WIDTh command. The data written is masked by the configured width so that any extra bytes will be discarded.
For example: sending the value 65531 to a byte wide channel will result in the channel discarding the upper byte and outputting 251.

Channel Width and Polarity, Threshold, Level, and Drive

When the width of a channel is set to `WORD` or `LWORd`, the channel direction (input or output) of the channels spanned by the width is controlled by the channel in operation. That is, all grouped channels are automatically set to the same input or output operation.

Channel settings of polarity, threshold, level, and drive mode are unchanged when channels are combined. For example, consider the following command sequence.

```
CONF:DIG:POL NORM,(@1101)
CONF:DIG:POL INV,(@1102)
CONF:DIG:WIDT WORD,(@1101)
```

This command sequence set the first 8 bits (channel 101) to normal polarity for input and output operations, set the next 8 bits (channel 102) to inverted polarity, and then combines the bits into a 16-bit channel. When this WORD channel is used, the first eight bits will input or output using normal polarity but the next 8 bits will read or written using inverted polarity.

Threshold, level, and drive settings all behave in the same manner as the polarity setting described above.

Handshaking

Handshaking provides a means to synchronize the input or output of digital data. By default, no handshaking is used; data is input or output as the command is executed. The handshake is configured per bank.

The 34950A provides a synchronous handshake mode (strobe handshake). You can use this mode with basic input and output operations. You *must* use this handshake mode to use buffered I/O (see "Buffered I/O Operations" on page 14).

The handshake is performed using three lines on each bank. The lines are labeled H0, H1, and H2. The function of each line is set by the input or output mode in use. Since there are only three handshake lines per bank, the SCPI handshake commands are only valid for the first channel in a bank. Once handshaking is enabled, it applies to the width of the first channel in the bank.

The three handshaking lines on each bank also differ slightly if you are using buffered (memory) I/O (see page 14) or unbuffered I/O operations. You can also perform unbuffered operations without any handshake. The function of each line for each mode of operation is defined in the table below.

	H0	H1	H2
Unbuffered Synchronous Input	I/O Direction (output)	Strobe (output)	Not Used (Hi-Z)
Unbuffered Synchronous Output	I/O Direction (output)	Strobe (output)	Not Used (Hi-Z)
Buffered Synchronous Input	Start/Stop (output)	Not Used (Hi-Z)	Input Strobe (input)
Buffered Synchronous Output (internal clock)	Start/Stop (output)	Strobe (output)	Not Used (Hi-Z)
Buffered Synchronous Output (external clock)	Start/Stop (output)	Not Used (Hi-Z)	Output Strobe (input)

The following handshake command sets the synchronous handshaking mode for the channels in bank 1.

```
CONF:DIG:HAND SYNC, (@1101)
```

This form of the handshaking command also allows you to optionally set the input threshold, output drive level, and polarity of all the handshake lines. For example, the following command sets bank 2 to use synchronous handshaking, with an input threshold of 2.5 V, an output drive level of 2.5 V, and normal polarity. Other parameters such as the handshake timing are set to default values (refer to the *Programmer's Reference Help file* for details).

```
CONF:DIG:HAND SYNC, 2.5, 2.5, NORM, (@1201)
```

You can set parameters by using a sequence of commands instead of the `CONFigure` macro command. For example, the following command sequence sets the handshaking mode to synchronous, the output drive to open collector, and the handshake rate to 1 MHz.

```
CONF:DIG:HAND:MODE SYNC, (@1101)
CONF:DIG:HAND:DRIV OCOL, (@1101)
CONF:DIG:HAND:RATE 1000000, (@1101)
```

Setting the Handshake Line Parameters

You can set the handshake lines' input threshold, output drive mode, and output drive voltage. These settings affect all the handshake lines in the bank. Handshake line polarity can be set for each individual handshake line.

For example, you can invert the polarity of the handshake line H1 on bank 2 with the following command.

```
CONF:DIG:HAND:POL INV, H1, (@1201)
```

You can set the output drive mode, output voltage, and input threshold for all handshake lines in each bank. For example, the following commands set the drive mode to active, the drive voltage to 4.5 V, and the input threshold to 1.0 V on bank 2.

```
CONF:DIG:HAND:DRIV ACT, (@1201)
SOUR:DIG:HAND:LEV 4.5, (@1201)
SENS:DIG:HAND:THR 1, (@1202)
```

NOTE The settings for drive mode, output drive level, and input threshold also apply to the bank's interrupt line.

NOTE When using external pull-ups in the open collector mode, the outputs will not exceed 5 V.

Synchronous Handshake Mode

In the synchronous handshake mode, a strobe or clock signal is used to transfer data to or from an external device. The strobe line (H1) is an output and is pulsed once for each transfer.

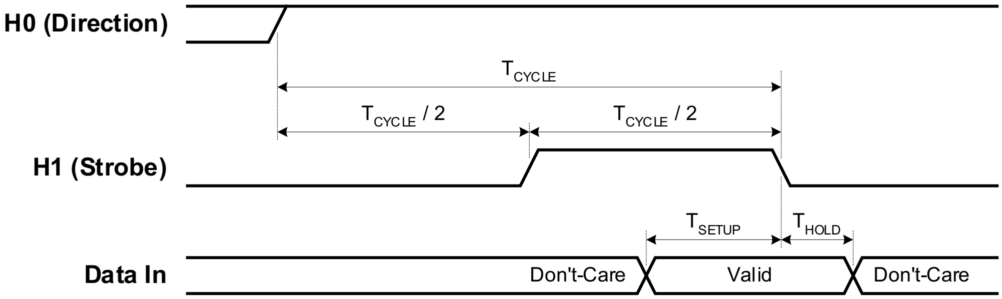
Synchronous Unbuffered Inputs

For synchronous handshake unbuffered inputs the H0 line indicates the direction of the transfer. This line is set high to indicate an input operation. The H0 line will remain in the high state until the 34950A direction is changed. The H1 line is the strobe output line. The H2 line is not used and is set to high impedance.

The timing of the input operation is controlled by the T_{CYCLE} parameter set using the `CONFigure:DIGital:HANDshake:RATE` command. This setting affects strobe width, memory clock rate, as well as the setup and hold times. Alternatively, the reciprocal form of the command `CONFigure:DIGital:HANDshake:CTIMe` can be used to specify the speed in terms of time instead of a rate. T_{CYCLE} begins when the 34950A executes one of the input commands.

The timing should be set such that the device sending the data ensures the data lines are valid prior to T_{SETUP} time. The trailing edge of the strobe line indicates the 34950A will latch the data within the T_{HOLD} time. T_{SETUP} is 90 ns and T_{HOLD} is 0 ns. Since T_{HOLD} = 0 µs, the sending device can use the trailing edge of the strobe to initiate a change in the data lines.

A synchronous unbuffered input is shown in the diagram below (default handshake line polarity).



For example, the following SCPI commands set a 34950A in slot 5 to have a 16-bit input using synchronous handshake. Two data inputs are then performed and the strobe line is pulsed for each query. The I/O direction line is set high following the first `SENSe:DIGital:DATA:WORD?` query and remains high until the digital channel is reset or reconfigured.

```
CONF:DIG:WIDT WORD, (@5101)
CONF:DIG:DIR INP, (@5101)
CONF:DIG:HAND SYNC, (@5101)
SENS:DIG:DATA:WORD? (@5101)
SENS:DIG:DATA:WORD? (@5101)
```

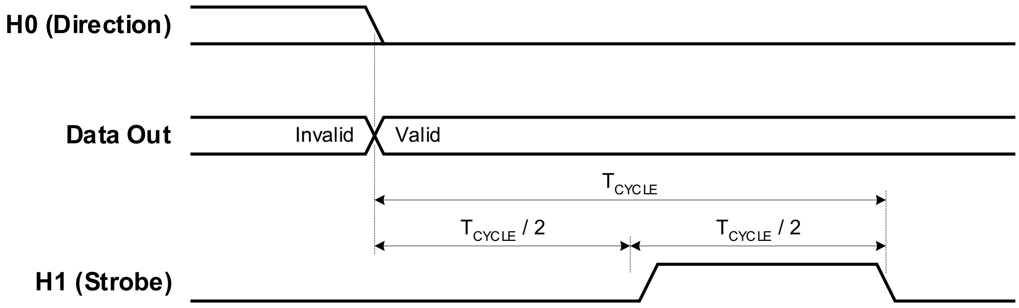
Synchronous Unbuffered Outputs

For synchronous handshake unbuffered outputs, the H0 line indicates the direction of the transfer. This line is set low to indicate an output operation. The H0 line will remain in the low state until the 34950A direction is changed. The H1 line is the strobe output line.

When the 34950A executes an output command, it sets the data lines and waits for $T_{CYCLE}/2$ before asserting the strobe line. The leading edge of the strobe indicates the data is valid. The strobe line is asserted for T_{CYCLE} /2 and then de-asserted. The H2 line is not used and is set to high impedance.

The timing of the output operation is controlled by the T_{CYCLE} parameter set using the `CONFigure:DIGital:HANDshake:RATE` command. This setting affects strobe width, memory clock rate, as well as the setup and hold times. Alternatively, the reciprocal form of the command `CONFigure:DIGital:HANDshake:CTIMe` can be used to specify the speed in terms of time instead of a rate. The timing should be set such that the device receiving the data can read the data lines during the $T_{CYCLE}/2$ time.

A synchronous unbuffered output is shown in the diagram below (default handshake line polarity).



For example, the following SCPI commands set a 34950A in slot 5 to have a 16-bit output using synchronous handshake. Two data outputs are then performed and the strobe line is pulsed for each. The I/O direction line is set low following the first `SOURce:DIGital:DATA:WORD` command and remains low until the digital channel is reset of reconfigured.

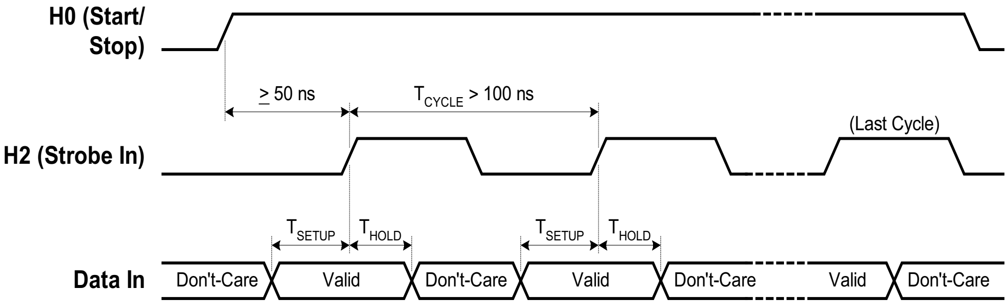
```
CONF:DIG:WIDT WORD, (@5101)
CONF:DIG:DIR OUTP, (@5101)
CONF:DIG:HAND SYNC, (@5101)
SOUR:DIG:DATA:WORD #hFFFF, (@5101)
SOUR:DIG:DATA:WORD #h4DB5, (@5101)
```

Synchronous Buffered Inputs

You can use synchronous mode handshake with buffered (memory) input operations. (Buffered operations are described in more detail beginning on page 14.) For buffered input operations, the H0 line acts as a start/stop line. This line will be set high when the memory input command is executed and will return low when the memory input operation has completed. The H1 line is not used and is set to high impedance.

An external strobe input on the H2 line controls the pace of memory transfers. The sending device must ensure the data is valid before the T_{SETUP} and stays valid until after T_{HOLD}. T_{SETUP} is 46 ns and T_{HOLD} is 10 ns.

A synchronous buffered input using an external clock is shown in the diagram below (default handshake line polarity).



For example, the following SCPI commands set a 34950A in slot 5 to have an 8-bit input using synchronous handshake with an external strobe input. The number of bytes to read into memory is set to infinite (continuous reading into memory until the memory is stopped). The memory is enabled and then triggered. The start/stop line is set high following the first byte handshake and remains high until the last byte is captured.

```
CONF:DIG:WIDT BYTE, (@5101)
CONF:DIG:DIR INP, (@5101)
CONF:DIG:HAND SYNC, (@5101)
SENS:DIG:MEM:SAMP:COUN 0, (@5101)
SENS:DIG:MEM:ENAB ON, (@5101)
SENS:DIG:MEM:STAR (@5101)
```

Synchronous Buffered Outputs

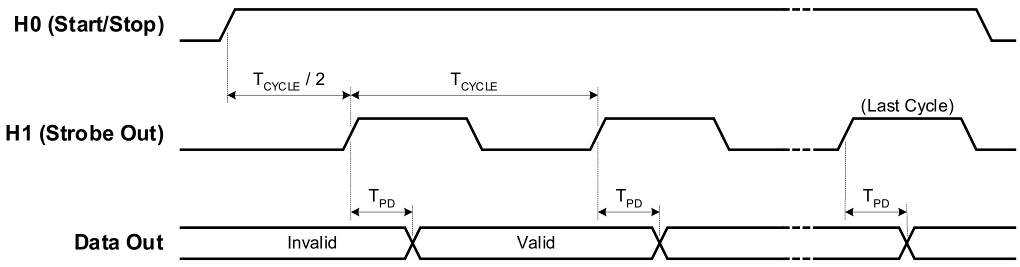
You can use synchronous mode handshake with buffered (memory) output operations. (Buffered operations are described in more detail beginning on page 14.) For buffered output operations, the H0 line acts as a start/stop line. This line will be set high when the memory output command is executed by the 34950A and will return low when the memory output operation has completed.

Synchronous memory output operations can be paced using either the *internal* strobe or an *external* strobe.

When using the *internal* strobe, the H1 line is the strobe output line. The timing of the output operation when using the default `INTernal` clock is controlled by the `CONFigure:DIGital:HANDshake:RATE` command. This setting affects strobe width, memory clock rate, as well as the setup and hold times. Alternatively, the reciprocal form of the command `CONFigure:DIGital:HANDshake:CTIMe` can be used to specify the speed in terms of time instead of a rate. The timing should be set such that the device receiving the data can latch the data lines during the T_{CYCLE} time.

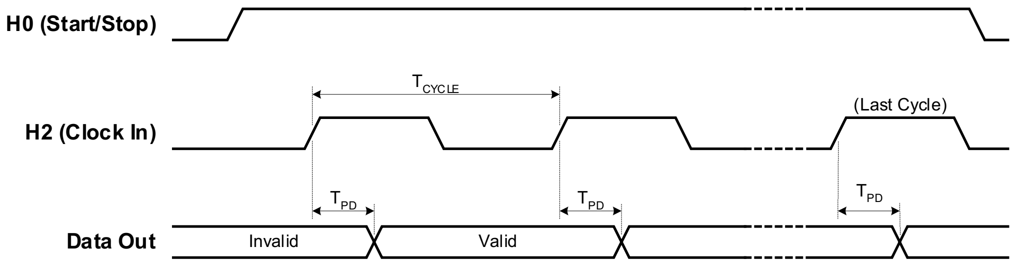
The receiving device should detect the leading edge of the strobe line, wait for the 34950A to set the data (T_{PD}) and then latch the data. Latching the data on the trailing edge of the strobe is recommended, however, you can the data following T_{PD}. T_{PD} ranges from -23 to 23 ns.

A synchronous buffered output using the internal clock is shown in the diagram below (default handshake line polarity).



Optionally, you can provide an *external* strobe input on the H2 line to control the memory transfers. If you pace the memory inputs from an external clock, the 34950A will sense the leading edge of the strobe and set the data. The data will be valid after T_{PD} and the receiving device may latch the data. T_{PD} ranges from 82 ns to 47 ns.

A synchronous buffered output using an external clock is shown in the diagram below (default handshake line polarity).



For example, using the *internal* strobe, the following SCPI commands set a 34950A in slot 5 to have a 32-bit output using synchronous handshake. The number of times to output the traces is set to 4. A trace is then loaded into memory and assigned to the channel. The memory is enabled and then triggered. The start/stop line is set high following the first byte handshake and remains high until the last byte is output.

```
CONF:DIG:WIDT LWOR, (@5101)
CONF:DIG:DIR OUTP, (@5101)
CONF:DIG:HAND SYNC, (@5101)
SOUR:DIG:MEM:NCYC 4, (@5101)
TRAC:DATA:DIG:LWOR (@5101), mytrace, #hFFEEFFEE, #hBCBC9999
SOUR:DIG:MEM:TRAC mytrace,(@5101)
SOUR:DIG:MEM:ENAB ON, (@5101)
SOUR:DIG:MEM:STAR (@5101)
```

Using an *external* strobe, the following SCPI commands set a 34950A in slot 5 to have an 8-bit output using synchronous handshake with an external strobe input. The number of times to output the traces is set to infinite (continuous output until the memory is stopped). The memory is enabled and then triggered. The start/stop line is set high following the first byte handshake and remains high until the last byte is output.

```
CONF:DIG:WIDT BYTE, (@5101)
CONF:DIG:DIR OUTP, (@5101)
CONF:DIG:HAND SYNC, (@5101)
CONF:DIG:HAND:SYNC:STR:SOUR EXT, (@5101)
SOUR:DIG:MEM:NCYC 0, (@5101)
TRAC:DATA:DIG:BYTE (@5101), mytrace, 260, 139
SOUR:DIG:MEM:TRAC mytrace,(@5101)
SOUR:DIG:MEM:ENAB ON, (@5101)
SOUR:DIG:MEM:STAR (@5101)
```

Buffered I/O Operations

Each of the two banks on the 34950A has its own memory that can be used to store patterns to output (traces) or to store input patterns. The width of the first channel in each bank controls the width of the memory operations. Memory may be used as:

- 64K x 8 bits
- 64K x 16 bits
- 32K x 32 bits

Buffered (Memory) Output

Each bank on the 34950A has its own memory for use in buffered transfers. Changing a bank from an input to an output will clear all memory for that bank. For buffered outputs, you download “traces” of digital data to the memory. Multiple traces (up to 32) can be downloaded into each bank. A specified trace is then output using the handshaking parameters set.

The general steps to output from memory are:

1. Set the channel width and direction.
2. Set the handshake mode.
3. Set the trigger source.
4. Set the number of times to output the trace.
5. Load the trace(s) into memory.
6. Set which trace to use.
7. Enable the memory.
8. Trigger the output.

Set the channel width and direction. Use the `SOURce:DIGital:DATA` command to set the channel width and set the channel as an output. Additionally, the data specified in the command will be the initial state of the data lines before the memory operation begins.

Set the handshake mode. You must use synchronous handshaking mode. You can use either an internal or external strobe (clock) to pace the outputs. Handshaking is described in more detail on page 7.

Set the trigger source. By default, the trigger source is set to use a software trigger. You can also use one of the interrupt lines (see page page 18) as a trigger source.

Set the number of times to output the trace. Each trace can be output once, multiple times, or infinitely. The `SOURce:DIGital:MEMory:NCYCles` command sets the number of times to output the trace. If not set to infinite, you can output the trace from 1 to 255 times (the output is controlled by the handshake).

Load the trace(s) into memory. Named traces are downloaded using the `TRACe:DATA:DIGital` command. The channel width used should match the width of the channel set in step 1. If you change the width of a bank, all traces in memory are cleared. The trace names must start with a character and may be up to 12 characters in length. The trace name used must be unique to the bank. Up to 32 traces may be downloaded (up to the maximum memory).

Traces can be added or deleted only when memory is disabled. Memory output cannot be enabled unless the bank has a trace assigned to it.

For example, the following commands load two traces into memory for bank 1 of a module in slot 1. In this example, each byte of the LWORd to output is sent as a separate byte.

```
TRAC:DATA:DIG:LWOR (@1101), MyTrace1, 255, 200, 128, 0
TRAC:DATA:DIG:LWOR (@1101), MyTrace2, 254, 192, 64, 32
```

You can also send trace data in IEEE-488 block format using this command.

The 34950A also has two special built-in traces for your use. You can generate and download a count-up trace and a walking 1's pattern using the `TRACe:DATA:DIGital:FUNCtion` command. See the *Programmer's Reference Help file* for more details.

NOTE

You can generate a count-down or walking zero pattern by inverting the data line polarity.

Set which trace to use. The `SOURce:DIGital:MEMory:TRACe` command assigns the desired trace to the bank. This command allows you to switch between the traces pre-loaded into the bank's memory.

Enable the memory. Enable the memory on the bank using the `SOURce:DIGital:MEMory:ENABle` command. This command sets the selected trace to be the output and puts the bank in the wait-for-trigger state.

Trigger the output. When the default trigger source is used, the `SOURce:DIGital:MEMory:STARt` command triggers the output. The selected trace will be output when the handshake occurs.

If the trigger source has been set to one of the interrupt lines (see page page 18), the output will wait for the interrupt to occur and then the handshake to occur before the trace is output.

You can also output the trace one sample at a time on the data lines using the `SOURce:DIGital:MEMory:STEP` command. This command outputs one sample and then puts the memory in the stopped state. The STEP command also overrides the interrupt line so it can be used to trigger a transfer even if the interrupt line is set to be the trigger source.

Deleting Trace

You can delete traces in memory to recover the memory space. Use the `TRACe:DELete:NAME` command to delete a specific trace. Note that deleting a specific trace does not de-fragment the memory. You can delete all traces using the `TRACe:DELete:ALL` command.

Buffered (Memory) Input

Each bank on the 34950A has its own memory for use in buffered transfers. Changing a bank from an output to an input will clear all memory for that bank. The general steps to use input memory are:

1. Set the channel width and parameters.
2. Set the handshake mode.
3. Set the number of samples to collect.
4. Start the capture.
5. Check the status of the transfer.
6. Retrieve the captured data.

Set the channel width and direction. Use the `CONFigure:DIGital` command to set the channel width, direction, thresholds, and polarity. See page 4 for basic input operations.

Set the handshake mode. You must use synchronous handshaking mode. Handshaking is described in more detail on page 7.

Set the number of samples to collect. The `SENSe:DIGital:MEMory:SAMPle:COUNt` command sets the number of samples to capture. If you set the number of counts to infinite (0 = default), the bank will capture data until a STOP is received. Older samples are overwritten if memory gets full. Allowed sample counts depend upon the channel width as follows:

- BYTE (8-bit) 1 to 65535
- WORD (16-bit) 1 to 65535
- LWORd (32-bit) 1 to 32767

Start the capture. The `SENSe:DIGital:MEMory:STARt` command sets the channel to begin the data capture. The capture begins when the handshake occurs.

Check the status of the transfer. You can use the `SENSe:DIGital:MEMory:DATA:POINts?` query to return the number of samples currently in memory.

Retrieve the captured data. Set the desired memory retrieval format using the `SENSe:DIGital:DATA:FORMat` command. You can set the memory to be read as either `LIST` or `BLOCk`. The `LIST` parameter (default) returns the data as comma separated ASCII values. `BLOCk` returns the data in IEEE-488 block format.

Before you can read the data in memory, you must stop the memory operations using the `SENSe:DIGital:MEMory:ENABle OFF` command.

Read all the captured data using the `SENSe:DIGital:MEMory:DATA:ALL?` query. This performs a non-destructive read of all data in the bank's memory.

To read specific captures, use the `SENSe:DIGital:MEMory:DATA?` form of the command. This command takes *index* and *count* parameters to specify which data to retrieve. The oldest data in memory has an *index* of '0'. The *count* specifies the number of samples to read. *count* + *index* must be less than the number of captured points.

Both these data reads are non-destructive to the bank memory.
To clear the memory for new data, send the `SENSe:DIGital:MEMory:CLEar` command.

Interrupt Lines

Each bank has an interrupt line that can be used with memory input or output operations. When a bank is set to input data, the interrupt line is an output. When a bank is set to output data, the interrupt line is set to be an input. You can set the polarity of the interrupt line for input and output operations using the `CONFigure:DIGital:INTerrupt:POLarity` command.

You can configure the interrupt line drive mode, output drive level, and input threshold. These parameters are set for both the handshake lines and interrupt line on a bank. See page 8 for details about setting these parameters.

Memory Output Operations

For memory output operations, the interrupt line is sensed and can be used to start or stop memory output operations. This provides a hardware means to control the data output.

The `SOURce:DIGital:INTerrupt:MODE` command sets how the bank will behave when using memory output. The mode can be set to one of three values:

- `STARt`: The memory output will begin on the rising edge of the interrupt line.
- `STOP`: The memory output is halted on the rising edge of the interrupt line.
- `GATE`: The interrupt line acts a a gate for the memory output. The bank can output when the interrupt line is asserted, and will stop when the interrupt line is de-asserted.

When you have set the polarity and mode, enable the interrupt using the `SOURce:DIGital:INTerrupt:ENABle` command.

NOTE

To prevent an unexpected trigger, make sure that the pattern interrupt signal is not floating.

Memory Input Operations

For memory input operations, the interrupt line is an output and is set on a pattern match or when the memory has been filled. You can set the interrupt line to be driven or open collector using the `SENSe:DIGital:HANDshake:DRIVe` command.

NOTE The settings for drive mode, output drive level, and input threshold also apply to the bank’s handshake lines.

When set to `ACTive` the interrupt line will be driven by the module. The high output voltage is set for both the handshaking and interrupt line on a bank with the `SOURce:DIGital:HANDshake:LEVel` command.

When set to `OCOLlector` the interrupt line will be driven low, but will go to high impedance mode when in the ‘High’ state. The open collector mode requires external pull-ups.

The `SENSe:DIGital:INTerrupt:MODE` command sets the condition that will cause the interrupt to be asserted. When set to `MFULl` the interrupt is given when the memory is full. When set to `COMPare` the interrupt is asserted when the pattern is detected (see page page 21). When either condition is removed, the interrupt is de-asserted.

The interrupt line is enabled by the `SENSe:DIGital:INTerrupt:ENABle` command and the status can be checked using the SCPI Status System (refer to the *Programmer's Reference Help file*).

Byte Ordering

When using buffered memory operations, the width of the data sets how the memory data is interpreted. Changing the width of the first channel in a bank invalidates any traces stored or captured.

Output Operations For output operations (see page 14), traces are put into memory using the `TRACe:DATA:DIGital` command.

For output operations, the data stored in memory is output as follows:

- `BYTE` output - first byte in memory on the first handshake, next byte in memory on the second handshake, and so on.
- `WORD` output - first and second byte in memory on the first handshake, next two bytes in memory on the second handshake, and so on.
- `LWORd` output - first four bytes in memory on the first handshake, next four bytes in memory on the second handshake, and so on.

Note that for `WORD` outputs the first byte in memory is considered the most significant byte and is output on the upper bits (8 through 15). For `LWORd` outputs the first byte is output on bits 24 through 31.

You can change the byte order reported using the `FORMat:BORDer` command. This command allows you to swap the most-significant and least-significant byte ordering for all data transfer operations. The command is applied globally and cannot be assigned to an individual slot or channel.

Input Operations For input operations (see page page 16), bytes are read into memory as follows:

- `BYTE` input - the first byte in memory was read on the first handshake, the next byte in memory was read on the second handshake, and so on.
- `WORD` input - first and second byte in memory were read on the first handshake, next two bytes in memory were read on the second handshake, and so on.
- `LWORd` input - first four bytes in memory were read on the first handshake, next four bytes in memory were read on the second handshake, and so on.

Note that for `WORD` inputs the first byte in memory is considered the most significant byte and was read on the upper bits (8 through 15). For `LWORd` inputs the first byte was read on bits 24 through 31.

Pattern Matching

Pattern matching can be used on input channels only. Pattern matching can be done with or without handshaking. When a pattern match occurs, the 34950A can set an interrupt line or system alarm. A pattern match can also be used to start or stop a buffered (memory) transfer.

Pattern matching is done on a per bank basis and always starts at the first channel of a bank and works up to encompass the configured width of the channel.

Patterns are set up and enabled using the `CALCulate` subsystem of SCPI commands. For example, the following commands set up a pattern match (#HF00F) and assert the interrupt line when the input pattern is equal to the match pattern.

```
CONF:DIG:WIDT WORD, (@1101)
CALC:COMP:DATA #HF00F, (@1101)
CALC:COMP:TYPE EQUAL, (@1101)
SENS:DIG:INT:MODE COMP, (@1101)
SENS:DIG:INT:ENAB ON, (@1101)
CALC:COMP:STAT ON, (@1101)
```

Once the pattern matching state is turned on, the 34950A polls for the pattern #HF00F to appear on the data lines of channel 101. The interrupt line will be asserted when the pattern is matched. In the example above the last command, `CALCulate:COMPare:STATe`, also sets the mainframe alarm on a pattern match.

You can use pattern matching to start or stop a buffered (memory) input transfer. When the desired pattern is found, the 34950A can be set to start or stop a capture.

For example, the following commands establish a byte pattern match on channels 101 and 201. When the pattern is found, 200 samples are captured.

```
CONF:DIG:WIDTH BYTE,(@3101,3201)
CALC:COMP:DATA:BYTE 140,(@3101,3201)
CALC:COMP:STAT ON,(@3101,3201)
DIG:MEM:SAMP:COUN 200,(@3101,3201)
DIG:MEM:COMP:ACT STAR,(@3101,3201)
DIG:MEM:ENAB ON,(@3101,3201)
```

Counter

The 34950A has two 10 MHz frequency counter/totalizer measurement input channels. The counters can operate in two general modes: *Totalizer* mode, and *Initiated Measurement* mode. In the totalizer mode, the counter acts as a basic totalizer. In the initiated measurement mode, the counter can make frequency, period, duty cycle, and pulse width measurements.

Totalizer Mode

Totalizer mode is the default operating mode for the counters. When the counter is configured for `TOTalizer` mode, it automatically starts running. The totalized count can be read, reset, scanned, and monitored.

The simplest way to take a totalizer measurement is to use the `MEASure` form of the command. For example, the following command configures the totalizer on the first bank, initiates the measurement, and returns the result. The data is returned in a floating point format.

```
MEAS:COUN:TOT? READ, (@1301)
```

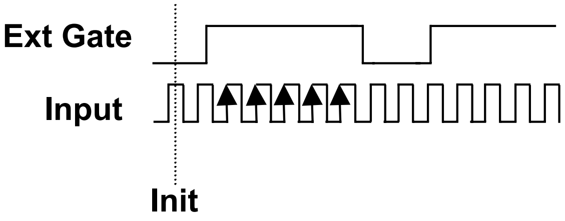
You can also reset the totalizer count by setting the parameter to `RRESet`. For example, the following command configures the totalizer on the first bank, initiates the measurement, and returns the result. The totalize count is reset when the data is read.

```
MEAS:COUN:TOT? RRES, (@1301)
```

Totalizer counts begin as soon as the channel is configured for the totalize measurement. You can stop a count by sending `SENSe:COUNter:ABORt` command and restart the count using the `SENSe:COUNter:INITiate` command.

The slope of the edges being counted can be configured using the `SENSe:COUNter:SLOPe` command. By default, when started, the totalizer counts rising edges.

Additionally, you can control when the edges are counted by setting the gate source to external and providing a gate signal on the gate input. In external gate mode the counter totalizes when the gate is asserted. The gate time setting controls how long the counter totalizes. Once the external gate has been de-asserted a new measurement must be armed via the `SENSe:COUNter:INITiate` command. The figure below shows an externally gated totalizer measurement. The number of totalized counts is '5' in this particular example.



Initiated Measurement Mode

Measurements such as frequency, period, duty cycle, and pulse width require an initiate command and a gate. The SENSe:COUNter:INITiate command is used to initiate (arm) the measurement. The measurement
is gated by either an internal (default) or the external gate source.
For measurements the external gate acts like an external trigger which triggers the internal gate timer.

The gate source is set using the SENSe:COUNter:GATE:SOURce command. The default gate source is INTernal. The gate is the aperture over which the signal data is gathered. When the gate is internal, the measurement begins as soon as the INITiate command is received.

Since the measurements are all derived from the same basic measurement, you can retrieve the measured frequency, period, duty-cycle, and pulse width from the same initiated and gated measurement. For example, the following commands set the counter to measure the input signal for 1 ms using the internal gate. The frequency, period, duty cycle, and pulse width are returned as floating point numbers.

```
CONF:COUN:FREQ 1e-3, (@1301)
SENS:COUN:INIT (@1301)
SENS:COUN:FREQ? (@1301)
SENS:COUN:PER? (@1301)
SENS:COUN:PWID? (@1301)
SENS:COUN:DCYC? (@1301)
```

The CONFigure:COUNter:FREQuency command parameter sets the internal gate time (to 1e-3 or 1 ms in the above example). You can also set the gate time using the SENSe:COUNter:GATE:TIME command.

Threshold Voltages

The counter channels of the 34950A allow you to specify input and external gate threshold voltages. These threshold voltages are set with the commands:

```
SENS:COUN:THR:VOLT <voltage>|MIN|MAX|DEF, (@ch_list)
SENS:MOD:COUN:GATE:THR:VOLT <voltage>|MIN|MAX|DEF, <slot>
```

The *input* threshold voltage may be set independently for each counter channel. The input threshold level is stored in volatile memory and is set to 0.8V when power is cycled or if the instrument is reset.

The external *gate* threshold voltage applies to both channels. It is also stored in volatile memory and is set to 0.8V when power is cycled or following a reset.

Note that the gate source (INTernal or EXTernal) is set with the command:
```
SENS:COUN:GATE:SOUR <source>, @(ch_list)
```

The polarity (NORMal or INVerted) of the gate signal is set with the command:
```
SENS:COUN:GATE:POL <polarity>, (@ch_list)
```

Clock

The general-purpose clock output is derived from the internal time base. The output clock is divided down from the time base clock such that:

Clock Output (Hz) = (time base frequency)/(divisor)

The time base frequency is 40 MHz. The divisor can be an integer from 2 to 4^6 providing a range of 20 MHz to 10 Hz for the clock output. The valid values for the clock output rate are: 20 MHz, 13.33 MHz, 10 MHz, 8 MHz, 6.667 MHz, ... 10Hz. The clock output frequency will round to the nearest achievable frequency.

The commands used to control the clock output are:

```
SOUR:MOD:CLOC:FREQ {<freq>|MIN|MAX|DEF},<slot>
SOUR:MOD:CLOC {OFF|ON|0|1},<slot>
```

You can obtain the rounded value of the currently set clock frequency using the following query.

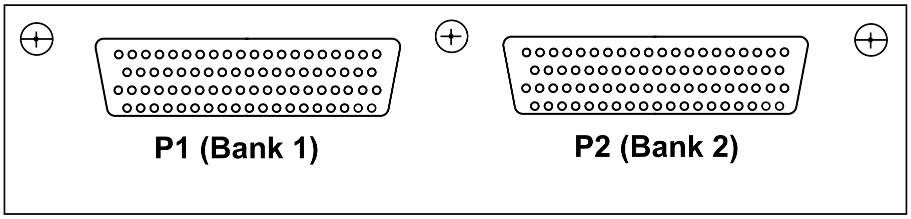
```
SOUR:MOD:CLOC:FREQ?
```

You can also set the logic “1” voltage level for external clock output. For example, the following command sets the output clock level to 4.5 V for the module in slot 5.

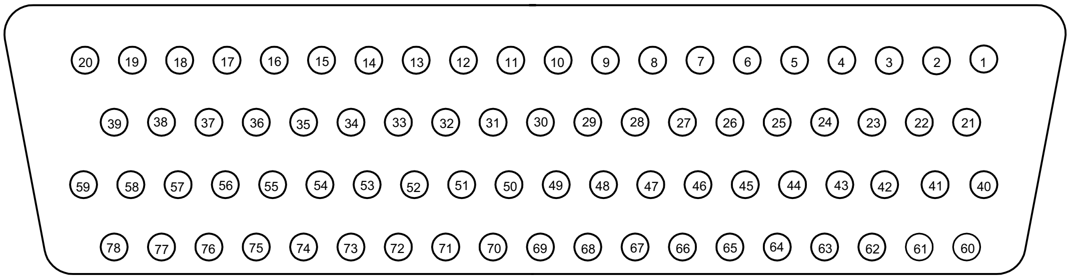
```
SOUR:MOD:CLOC:LEV 4.5, 5
```

34950A D-Sub Connectors

The 34950A uses two D-sub 78-pin female connectors. Each connector provides contains one bank of the module. As viewed from the rear panel, the connectors and their banks are shown below.



As viewed from the rear panel, the pins in each connector are numbered as shown below.



P1 (Bank 1) Connector Pin Assignments

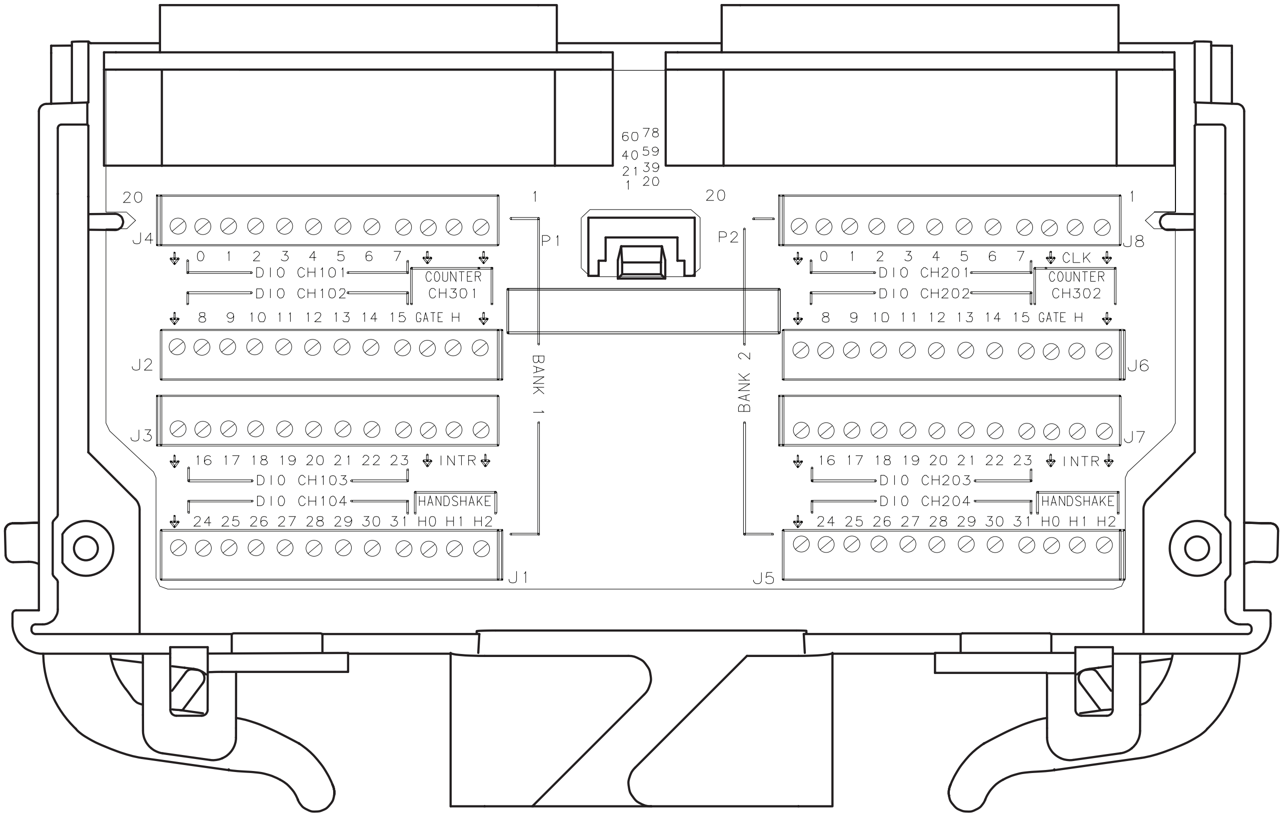
Pin	Signal		Pin	Signal		Pin	Signal		Pin	Signal	
1	GND	CH301	21	GND	CH104	40	18	CH103	60	8	CH102
2	CNTR		22	27		41	GND		61	GND	
3	GND		23	GND		42	17		62	NC	
4	GATE		24	26		43	GND		63	GND	CH101
5	GND		25	GND		44	16		64	7	
6	INTR		26	25		45	GND	CH102	65	GND	
7	GND		27	GND		46	15		66	6	
8	H2		28	24		47	GND		67	GND	
9	GND		29	GND	CH103	48	14		68	5	
10	H1		30	23		49	GND		69	GND	
11	GND		31	GND		50	13		70	4	
12	H0		32	22		51	GND		71	GND	
13	GND	CH104	33	GND		52	12		72	3	
14	31		34	21		53	GND		73	GND	
15	GND		35	GND		54	11		74	2	
16	30		36	20		55	GND		75	GND	
17	GND		37	GND		56	10		76	1	
18	29		38	19		57	GND		77	GND	
19	GND		39	GND		58	9		78	0	
20	28					59	GND				

P2 (Bank 2) Connector Pin Assignments

Pin	Signal		Pin	Signal		Pin	Signal		Pin	Signal	
1	GND	CH302	21	GND	CH204	40	18	CH203	60	8	CH202
2	CNTR		22	27		41	GND		61	GND	
3	GND		23	GND		42	17		62	CLK	
4	GATE		24	26		43	GND		63	GND	CH201
5	GND		25	GND		44	16		64	7	
6	INTR		26	25		45	GND	CH202	65	GND	
7	GND		27	GND		46	15		66	6	
8	H2		28	24		47	GND		67	GND	
9	GND		29	GND	CH203	48	14		68	5	
10	H1		30	23		49	GND		69	GND	
11	GND		31	GND		50	13		70	4	
12	H0		32	22		51	GND		71	GND	
13	GND	CH204	33	GND		52	12		72	3	
14	31		34	21		53	GND		73	GND	
15	GND		35	GND		54	11		74	2	
16	30		36	20		55	GND		75	GND	
17	GND		37	GND		56	10		76	1	
18	29		38	19		57	GND		77	GND	
19	GND		39	GND		58	9		78	0	
20	28					59	GND				

34950T Terminal Block

The optional 34950T terminal block has screw type connections and the terminal are labeled with the channel and bit information. The 34980A Product Reference CD (shipped with the instrument) contains a 34950T Wiring Log for you to document your wiring configuration for this module. You can open the wiring log file in Microsoft® Excel® or Adobe® Acrobat® format.



Wire Size:

20 AWG Typical
18 AWG Max

Index