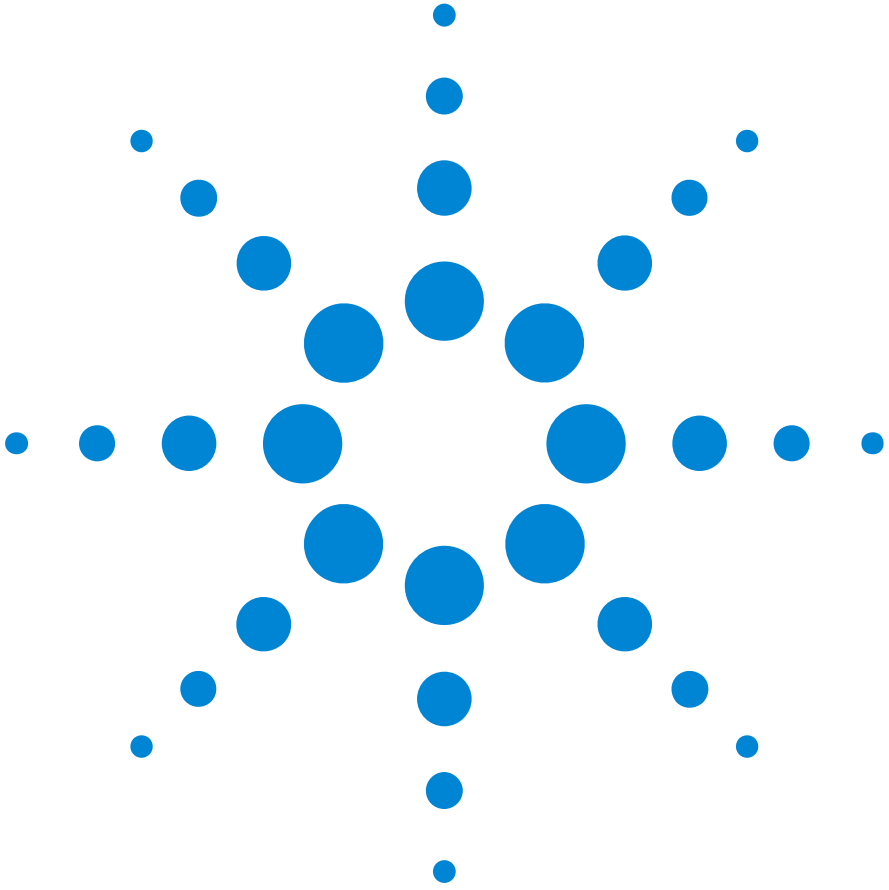


Agilent 81250 Parallel Bit Error Ratio Tester

User Guide

Agilent Technologies

Important Notice

Authors: t3 medien GmbH

Contents

What's New in This Release? 9

Major Changes from Rev. 1.0 to 1.1 9

New Module and Frontend 9

Introduction to the System 11

System Capabilities 12

Verify and Characterize Digital Devices 12

Key Features 12

System Components 13

Mainframes and Controllers 14

Modules 16

Frontends 19

Trigger Pod 22

Identification of Hardware Resources 22

Summary of Hardware-Related Terms 23

Operating Principles 24

Virtual Instruments/Systems 24

Hardware and Setup Models 25

Software Structure 28

Summary of Setup-Related Terms 30

Timing Principles 30

Choice of Clock Sources 31

FMR and Segment Resolution 31

Timely Alignment of Generated and Captured Signals 33

Trigger-Controlled Start and Stop 34

Summary of Timing-Related Terms 35

Data Generation Principles 35

Emulate Real Pattern and Waveform Conditions 36

Data Sequences 36

Data Blocks 37

Data Segments 38

Properties of Real Segments 39

Loops 41

Hardware Dependencies 42
Summary of Data-Related Terms 43
Data Capturing and Analysis Principles 43
Functional Tests 44
Error Analysis and Marginal Tests 45
Analyzer Sampling Point Adjustment 46
Display of Test Results 51
Summary of Analysis-Related Terms 51
Event Handling Principles 52
Usage of Events 52
What is an Event? 53
Actions Upon an Event 53
Summary of Event-Related Terms 54

Test Development Overview 55

Procedure for Setting Up the Test 56
Procedure for Running the Test 58
Procedure for Viewing Test Results 59
Procedure for Saving the Test Setting 60

System Start and User Interface 61

How to Start the System 62
How to Start the Agilent 81250 Software 62
Overview of the Windows 66
Overview of Test Setup Windows 66
Overview of Test Result Windows 67
Operating the User Interface 68
How to Use the Mouse or Touchpad 68
How to Navigate With the Keyboard 68
How to Change Units and/or Vernier Steps 68
How to Use the Window Selection Box 69
Items of the Main Menu 71
File Menu 73
Edit Menu 78
Tools Menu 82
View Menu 83
Go Menu 85

Control Menu 86
System Menu 88
Window Menu 89
Help Menu 90

Setting Global System Parameters 91

How to Start the Parameter Editor for Global Parameters 92
How to Set the Clock Frequency 93
How to Set the General System Frequency 94
How to Use Multiple Frequencies 96
How to Choose the Clock Source 100
How to Set the Characteristics of the External Input 101
How to Set the Characteristics of the Trigger Output 103

Connecting the DUT 105

How to Start the Connection Editor 106
Contents of the Connection Editor Window 106
How to Create a Port 107
How to Change the Characteristics of a Port 109
How to Delete a Port 109
How to Rename a Port 109
How to Add a Terminal to a Port 110
How to Change the Characteristics of a Terminal 110
How to Rename a Terminal 111
How to Delete a Terminal 111
How to Move a Terminal 111
How to Connect a Terminal 112
How to Disconnect a Terminal 114

Setting Up Ports and Channels 115

How to Start the Parameter Editor for Ports/Channels 116
How to Set Up a DUT Input Port or Generator Channel 117
How to Set Generator Timing Parameters 118
How to Set Additional Generator Parameters 119
How to Add Channels in Analog Mode 122

How to Set Up a DUT Output Port or Analyzer Channel 125
How to Set Analyzer Timing Parameters 126
How to Set Additional Analyzer Parameters 126
How to Combine Generator Channels 129
How to Start the Channel Configuration Editor 130
How to Use the Channel Configuration Editor 131

Choosing the Kind of Measurement 135

How to Access the Measurement Configuration Window 136
How to Set the Measurement Configuration 136
Capture Data 137
Error Rate Measurement 137
Compare and Acquire Around Error 137
Compare and Capture 138

Creating the Stream of Generated and Expected Data 139

The Standard Mode Sequence Editor 140
How to Use the Standard Mode Sequence Editor 141
How to Synchronize an Analyzer With Incoming Data 145
Special Characteristics of the Standard Mode Sequence Editor 149
The Detail Mode Sequence Editor 152
Contents of the Detail Mode Sequence Editor Window 153
How to Add, Move or Delete Blocks 154
How to Change Block Properties 155
How to Use a Block for Analyzer Sampling Point Adjustment 157
How to Replace the Current Segment 158
How to Create and Change Loops 162
How to Specify Events and Reactions Upon Events 164
Before You Start Using Events 165
How to Define Events 168
How to Specify the Reactions on Events 170

Creating and Editing Segments 175

How to Create a New Segment 176
How to Start Creating a New Segment 176
How to Create a Memory Segment 178
How to Create a PRBS/PRWS Segment 185
How to Save a New or Changed Segment 186
How to Edit a Stored Segment 187
How to Select a Segment 187
How to Edit a Memory Segment 188
How to Edit a PRBS/PRWS Segment 188

Using the Data/Sequence Editor 189

How to Start the Data/Sequence Editor 190
Contents of the Data/Sequence Editor Window 190
How to Customize the Data/Sequence Display 191
How to Change the Width of the Columns 192
How to Change the Height of a Block 193
How to Change the Format of Displayed Addresses 194
How to Change the Labels of Displayed Traces 194
How to Change the Sequence or Edit Segments 195
How to Change the Sequence Characteristics 195
How to Replace a Segment 196
How to Edit the Contents of a Segment 197

Running the Test 199

How to Download the Test Sequence 200
How to View BER Test Results 200
How to Start/Stop the Test 201

Viewing Generated and Captured Data 203

How to View Captured Test Results 204
How to Start the Error State Display 204
How to Operate the Error State Display 204
How to Transfer Captured Data Into a Segment 206

How to View Waveforms 209
How to Start the Waveform Viewer 209
Description of the Waveform Viewer Display 210
How to Operate the Waveform Viewer 211

Using Auxiliary Functions 215

How to Compensate for Internal and External Delays 216
How to Start the Deskew Editor 216
How to Adjust the Instrument Connectors 217
How to Compensate for Cable Delays 218
How to Compensate for Cable and DUT Board Delays 220
How to Export/Import Settings or Segments 222
Export/Import of a Setting 222
Export/Import of Segments 224
How to Execute Firmware Commands 225
How to Start the Command Line Editor 225
How to Use the Command Line Editor 226

Appendix A: How Do I ... ? 229

How Can I Generate a Clock Signal With a Data Module? 230
How Do I Use Events? 232
How Do I Select Between Two Different Tests? 232
How Do I Set a Trigger on Error? 233
How Do I Allow the DUT to Stabilize? 233
How Can I Return Pass/Fail Information to another Test System? 234
How Can I Execute Different Tests Embedded in One Sequence? 236
How Can I Change all Traces of a Port to Don't Care? 236
How Do I Set Up a Multiplexer BER Test? 240
How Do I Use Automatic Sampling Point Adjustment? 243
How Can I Synchronize a MUX Test With Two Systems? 243
How Can I Synchronize a DEMUX Test With Two Systems? 246
How Do I Use the AUX OUT of E4863A/E4865A Frontends? 249

Appendix B: PRBS/PRWS Data Segments 251

Pure and Distorted PRBS 252
Variable Mark Density 253
Extended Zeros/Ones 254
Error Insertion 254
Pure and Distorted PRWS 255
Pure PRWS 255
Distorted PRWS 256

Contents

What's New in This Release?

This chapter gives an overview of the most important changes and enhancements of the Agilent 81250.

The information is organized as follows:

"Major Changes from Rev. 1.0 to 1.1" on page 9

Major Changes from Rev. 1.0 to 1.1

In addition to the E4861A 2.67 GHz data generator/analyzer module, the Agilent 81250 system software now supports also the E4832A module.

New Module and Frontend

Module E4832A

The new 667 MHz data generator/analyzer module E4832A has four slots for four frontends. It can accomodate the frontends:

- E4838A, 667 Mbit/s, differential output, low voltage amplitude/offset and variable slopes generator frontend
- E4843A, 667 Mbit/s, NRZ/RZ, differential generator frontend
- New E4835A frontend, a pair of 667 Msa/s, differential or single-ended input high sensitivity analyzer frontends

Compared with the E4841A data generator/analyzer module, the E4832A has twice the memory capacity and supports Compare and Capture or Compare and Acquire Around Error tests at double speed (clock rates up to 667 MHz).

Introduction to the System

This chapter makes you familiar with the Agilent 81250 Parallel Bit Error Ratio Tester, its components, operating principles, and terms.

The information is organized as follows:

"System Capabilities" on page 12

"System Components" on page 13

"Operating Principles" on page 24

"Timing Principles" on page 30

"Data Generation Principles" on page 35

"Data Capturing and Analysis Principles" on page 43

"Event Handling Principles" on page 52

System Capabilities

The Agilent 81250 Parallel Bit Error Ratio Tester is first of all meant for testing high-speed data communication equipment (DCE), but can also be used as a multi-purpose digital stimulus/response system.

The system can be operated from the graphical user interface or controlled via LAN or GPIB, for example in an automated test rack. It can also control other GPIB instruments. It has a programming interface to VEE and C/C++.

Verify and Characterize Digital Devices

The device under test (DUT) and its application setup are modeled in the software.

The graphical user interface shows a raw DUT scheme in the Connection Editor window. In this window it is possible to define groups of signals for the DUT—these groups of signals are called “ports”. The DUT template offers two types of ports:

- Data ports are used for data signals such as stimulus data and response data.
- Pulse ports are used for pure parametric signals such as clock signals.

All signal parameters can be set up conveniently for a group of pins (a port) as well as separately for single DUT pins (terminals).

The system has data generating and analyzing frontends.

Cable delays and signal skew in the test setup can be compensated by using the deskew feature.

Key Features

Just a summary of the most important features:

- Stimulus as required, real-time error analysis and margin test.
- Data rates up to 2.67 Gbit/s.
- Up to 8 Mbit memory per channel.
- Up to 64 generator/analyzer channels at 1.33 or 2.67 Gbit/s. Up to 128 channels at 667 Mbit/s.
- 2 ps timing resolution.

- Logical XOR addition of two or four 667 MHz generator channels.
- Pattern formats NRZ, DNRZ, RZ, R1.
- PRBS (up to 2^{15}–1 plus 2^{23}–1 and 2^{31}–1).
- Automatic analyzer sampling point adjustment.
- Sequencing with up to five loop levels (nested loops).
- Variable delay, width, transition times, voltage levels—individually adjustable for each channel.
- Semi-automatic signal delay compensation.
- Event recognition and reactions upon events.
- Tabular and graphical result presentation.
- Modular hardware structure.
- Frontends for differential and low voltage signals.

For details please refer to the *Agilent 81250 Technical Specifications*.

System Components

The Agilent 81250 Parallel Bit Error Ratio Tester is available in several configurations.

Standard configurations comprise:

"Mainframes and Controllers" on page 14

"Modules" on page 16

"Frontends" on page 19

"Trigger Pod" on page 22

Mainframes and Controllers

An Agilent 81250 Parallel Bit Error Ratio Tester consists of a VXI mainframe, a controller, and modules plugged into the mainframe.

Figure 1 E4860A Mainframe Configuration

Mainframes

13-slot Mainframe

The standard mainframe is the E4803A VXI mainframe with 13 VXI slots.

Figure 2 E4803A VXI Mainframe with Modules

Up to two expander frames E4848B can be added. An E4848B expander frame consists of an E8403A mainframe, two E1482B VXI bus extender modules, and connection cables.

The VXI bus extender modules require one slot in both the basic and the added frame.

Controller Options

Two options are available for controlling the system:

Embedded Controller

- The built-in 2-slot VXI controller E9850A (opt. #012).

  This controller is a PC that includes harddisk, diskette drive, a serial and a parallel interface, SCSI controller, GPIB and LAN interface. Operating system and ParBERT software are readily installed.

  Opt. #012 leaves 11 mainframe slots for modules.

  A system with built-in controller requires a monitor, keyboard and mouse (Opt. #010).

External Controller

- The IEEE 1394 PC link to VXI (opt. #013).

  This option allows to use an external PC running under Windows NT as system controller. The option includes a PCI board to be installed in the computer, a 1-slot VXI module to be installed in the mainframe, and all required software.

  Opt. #013 leaves 12 mainframe slots for modules.

Open VXI Configurations

The Agilent 81200 Data Generator/Analyzer Platform supports also Open VXI configurations. Open VXI enables you to set up your test equipment as compact as required.

As long as free slots are available, VXI modules of other systems can be plugged into the Agilent 81250 mainframe. You only need to take care that the Agilent 81250 modules start from the leftmost slot and remain in contiguous slots. Additional software can be installed on the built-in harddisk or the external PC to operate these modules.

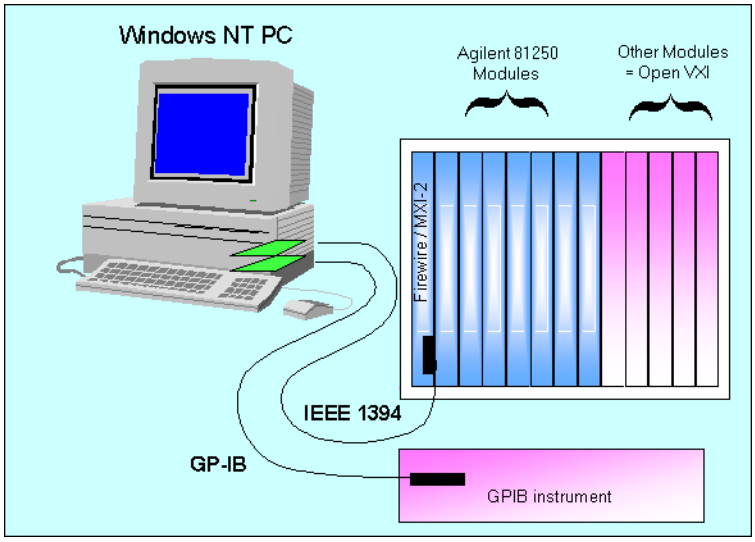


Figure 3 IEEE 1394 PC Link and Open VXI Configuration

For details please refer to the *Installation Guide* and the *Configuration Guide*.

Modules

An Agilent 81250 Parallel Bit Error Ratio Tester comprises at least one clock module and one data generator/analyzer module with frontends.

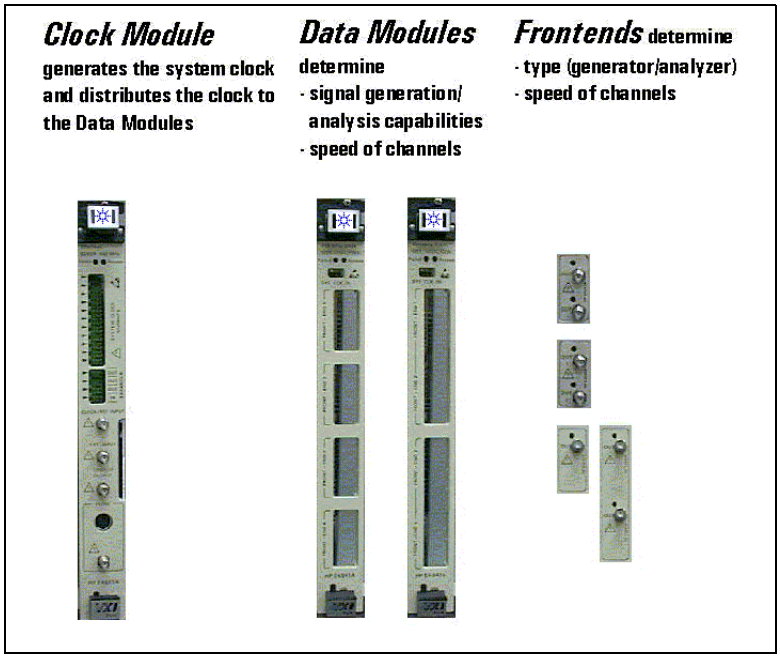


Figure 4 Modules and Frontends

Clock Modules

The clock module generates the system clock and synchronizes all data generator and analyzer channels in a mainframe. The clock module provides the sequencing capability of a system and can use either its internal synthesized clock source or an external clock source. The internal clock synthesis can be locked to a common frequency standard using the PLL reference input.

The following clock module is used:

E4805B Central Clock Module

- E4805B Central Clock Module

  This module synchronizes up to 11 data analyzer/generator modules. It can also control up to two slave clock modules. Slaves can be plugged into the same mainframe as the master clock module or into an expander frame. A deskew probe can be connected and the Agilent 81200 Trigger Pod can be attached to the master clock module.

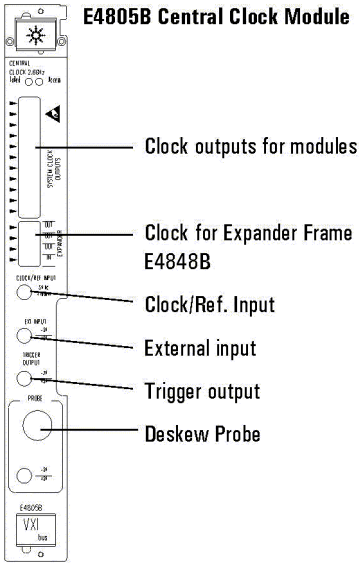


Figure 5 E4805B Clock Module

The E4805B central clock module permits to use up to 5 loop levels within generated data sequences.

NOTE The Agilent 81200 Data Generator/Analyzer Platform comprises also the

- E4805A central clock module and the
- E4831A central clock and data generator module.

These modules do not provide the automatic analyzer sampling point adjustment which is an important feature of the Agilent 81250 Parallel Bit Error Ratio Tester.

Data Generator/Analyzer Modules

A data generator/analyzer module houses 2 or 4 frontends. There are frontends for generating and sourcing signals to the DUT and others for capturing and analyzing signals from the DUT.

Any combination of input and output frontends within a module is possible, but not recommended, as this may conflict with the concept of grouping output and input signals into “ports” and generating or expecting pseudo random word stream signals (PRWS).

The following data generator/analyzer modules are used:

E4832A Module

- E4832A 667 MHz Data Generator/Analyzer Module:
  This module provides four slots for four frontends, with one or two channels each.

  Compared with the E4841A data generator/analyzer module, the E4832A has twice the memory capacity and supports Compare and Capture or Compare and Acquire Around Error tests at double speed (clock rates up to 667 MHz compared to 330 MHz). It supports also the phase vernier that can be used for moving the sampling point of an analyzer.

E4861A Module

- E4861A 2.67 GHz Data Generator/Analyzer Module:
  This module provides two slots for two high-speed frontends with maximum data rates of 1.33 Gbit/s or 2.67 Gbit/s. It supports the phase vernier that can be used for moving the sampling point of an analyzer.

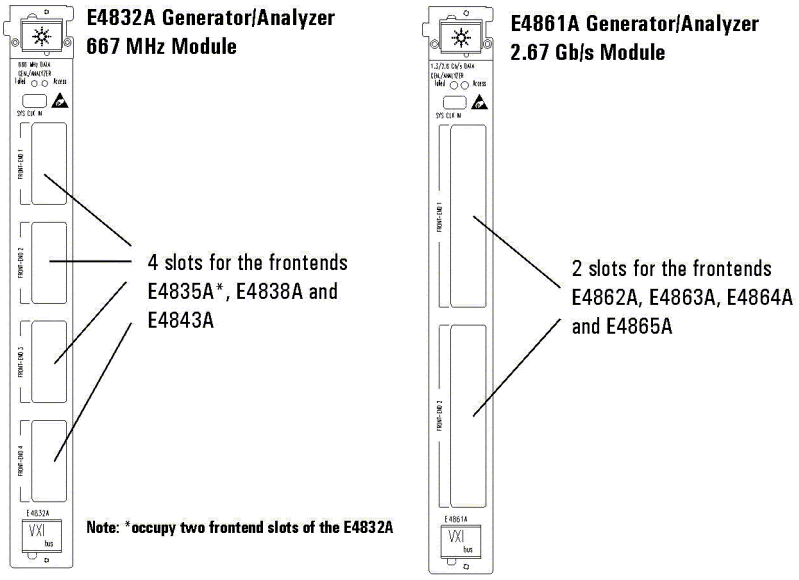


Figure 6 Data Generator/Analyzer Modules

NOTE The Agilent 81200 Data Generator/Analyzer Platform comprises also the

- E4841A data generator/analyzer module, and the
- E4831A central clock and data generator module.

These modules, however, do not provide the automatic analyzer sampling point adjustment which is an important feature of the Agilent 81250 Parallel Bit Error Ratio Tester.

Frontends

The available frontends include data generator and data analyzer frontends.

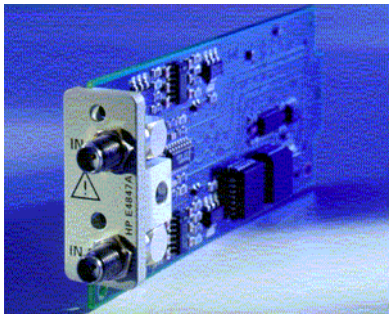


Figure 7 Frontend With Two Connectors

Note that the frontends have built-in protection circuits which automatically disconnect a frontend if an attempt is made to operate the frontend under intolerable conditions.

NOTE If this happens, the user interface is neither informed nor updated. In case of a problem, you should therefore always inspect the green LEDs above the frontend connectors. They clearly indicate the physical connection status.

Once the termination conditions have been corrected, the Connectors On/Off button of the toolbar can be used to re-establish the connection.

Generator Frontends

Generator Frontends for E4861A Modules

The generator frontends for the E4861A 2.67 GHz module are:

- E4862A, 2.67 GHz, differential output frontend
- E4864A, 1.33 GHz, differential output frontend

Generator Frontends for E4832A Modules

The generator frontends for the E4832A 667 MHz module are:

- E4838A, 667 MHz, differential output, low voltage amplitude/offset and variable slopes generator
- E4843A, 667 MHz, NRZ/RZ, differential output frontend

NOTE The Agilent 81200 Data Generator/Analyzer Platform provides additional generator frontends:

- E4842A, 330 Mbit/s, NRZ/RZ, single ended, variable transitions, 3.5 V amplitude
- E4846A, 200 Mbit/s, dual output single-ended frontend

These frontends are not supported by the E4832A or E4861A modules.

For details please refer to the *Agilent 81250 Technical Specifications*.

Analyzer Frontends

Analyzer Frontends for E4861A Modules

The analyzer frontends for the E4861A 2.67 GHz module are:

- E4863A, 2.67 Gsa/s, differential/single-ended input frontend
- E4865A, 1.33 Gsa/s, differential/single-ended input frontend

These analyzer frontends have an AUX OUT connector which provides the input signal as interpreted by the input comparator – either low or high.

This output can for example be used to synchronize a pure analyzing system. If the DUT (such as a deserializer or demultiplexer) generates a recovered clock, this clock signal may be fed into the frontend, and the AUX OUT signal can be used to provide the clock of the analyzing system via the EXT. INPUT connector of the clock module.

Note that the AUX OUT connector has an internal impedance of 50 Ω which must be met by the receiver in order to achieve the specified characteristics. A termination voltage between 0 V and –2 V may be used. If these requirements are not met, the output is disabled.

See also *"How Do I Use the AUX OUT of E4863A/E4865A Frontends?" on page 249*.

Analyzer Frontends for E4832A Modules

The analyzer frontend for the E4832A 667 MHz module is:

- E4835A, 667 MSa/s, differential/single-ended input, high sensitivity analyzer

The E4835A frontends are always installed in pairs. Two E4835A frontends share a common memory plug-in.

NOTE The Agilent 81200 Data Generator/Analyzer Platform provides additional analyzer frontends:

- E4837A, 667 MSa/s, differential input high sensitivity analyzer
- E4844A, 667 MSa/s, single input frontend
- E4845A, 330 MSa/s, dual input frontend
- E4847A, 330 MSa/s, high-Z, dual input frontend

These frontends are not supported by the E4832A or E4861A modules.

For details please refer to the *Agilent 81250 Technical Specifications*.

Trigger Pod

The Agilent 81200 Trigger Pod is an option of the E4805B Central Clock Module. It can be used to detect external events and react on them.

Figure 8 Agilent 81200 Trigger Pod

The ribbon cable has to be connected to E4805B Clock Module.

The Trigger Pod has 8 TTL compatible input lines (input threshold 1.5 V). The input lines are terminated by 4.7 kΩ pull-up resistors to +5 V.

Data acquisition is triggered by the internal sequencer clock. The sequencer clock frequency is:

Sequ. clock = System clock frequency / segment resolution.

The maximum sequencer clock frequency is 41.67 MHz, corresponding to a period of 24 ns.

For technical details please refer to the *Agilent 81250 Technical Specifications*.

The input lines can be used to detect single, asynchronous events. If certain patterns (bit combinations) are to be detected, it is recommended to synchronize the incoming data with the system. This can be done by generating a clock signal at the TRIGGER OUTPUT of the E4805B Central Clock Module and applying that clock to the event source (see also *"How to Set the Characteristics of the Trigger Output" on page 103*).

Identification of Hardware Resources

The hardware resources pool is comparable to a traditional instrument. Here the instrument is seen as a collection of modules, which provide signal connectors. Parameters that can be modified on a connector level, such as levels or timings, are described here.

An Agilent 81250 Parallel Bit Error Ratio Tester can consist of multiple **clockgroups**. Each clockgroup consists of **modules** which in turn have **connectors**.

The following diagram illustrates the numbering system used to address the systems, modules, and connectors.

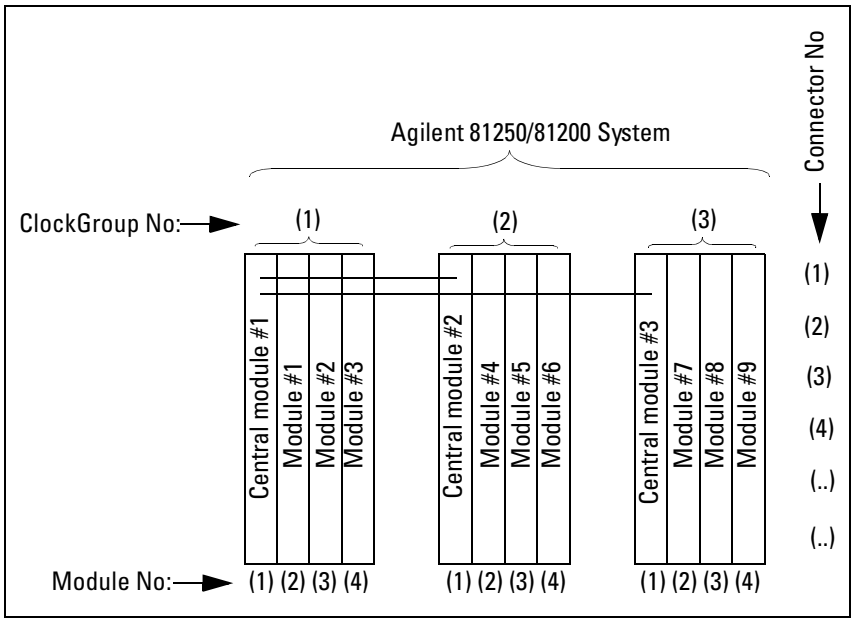


Figure 9 Numbering of Modules and Connectors

The identification of a generator or analyzer channel is:

Cx–My–Cz (ClockgroupNumber–ModuleNumber–ConnectorNumber), such as C1–M3–C5.

Summary of Hardware-Related Terms

Module: One of the following:

- E4805B Central Clock Module
- E4832A 667 MHz Data Generator/Analyzer Module
- E4861A 2.67 GHz Data Generator/Analyzer Module
- E4841A Data Generator/Analyzer Module

Frontend: Generator or analyzer plug-in of a module.

Connector: An output or input connector of a frontend.

Channel: The circuitry behind a connector which includes data generating or analyzing capabilities, data memory, frequency multiplexing and so on.

Trigger Pod: An option of the E4805B Central Clock Module for detecting external events.

Clockgroup: The sum of modules connected to a single clock module.

ClockgroupNumber: Identifies the clock master (= 1) and up to two slaves (2 and 3).

ModuleNumber: Identifies the module within a clockgroup (1 to 11).

ConnectorNumber: Identifies the connector of a module. Is counted from module top to bottom (1 to 8). Differential connectors are counted as one connector.

Operating Principles

The software of the Agilent 81250 Parallel Bit Error Ratio Tester is based on two concepts:

- The idea of **virtual machines**
- The idea of keeping **models** of the real world—a model of the present instrument configuration and a model of the DUT

This section explains the interdependencies and terms. See:

"Virtual Instruments/Systems" on page 24

"Hardware and Setup Models" on page 25

"Software Structure" on page 28

Virtual Instruments/Systems

In practice, one mainframe can house several independent clock modules with associated data generator/analyzer modules. This makes it possible to test a device under asynchronous conditions which require independent clock pulses.

The concept of the Agilent 81250 system is to create so-called virtual instruments from the system's present hardware resources (modules, generator, and analyzer channels).

By editing the *dvtsys.txt* configuration file, new virtual instruments can be created. New modules added to the Agilent 81250 system are automatically entered in the *dvtits.txt* configuration file.

The basic (default) instrument is called DSRA (DSR = digital stimulus and response). Additional instruments with own master clock modules may have suffixes or ascending names, such as DSRB, DSRC, and so on.

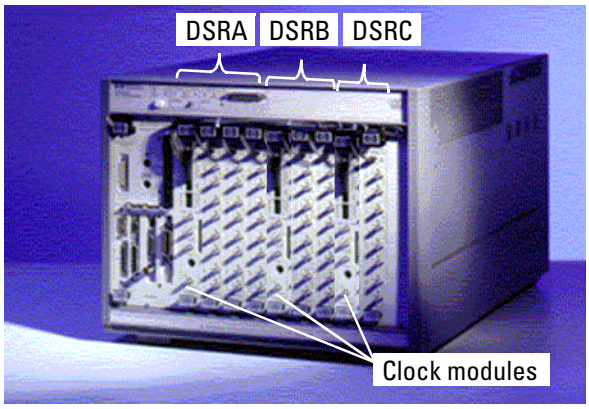


Figure 10 Virtual Instruments in One System

NOTE The user interface and remote control commands enable you to load and operate any of the configured instruments. The Windows NT operating system also enables you to operate several instruments in parallel. The user interface indicates the chosen instrument in the bottom line of the main window.

If the system is operated remotely via SCPI commands, the instrument names are used to construct the **handles** for identifying the respective instrument.

Hardware and Setup Models

On power up, the system automatically checks and identifies the available modules and frontends. It creates an image of the instrument configuration and displays this image in the Connection Editor window.

The image of the DUT needs to be created. This can be done manually with the Connection Editor or by loading a stored setting.

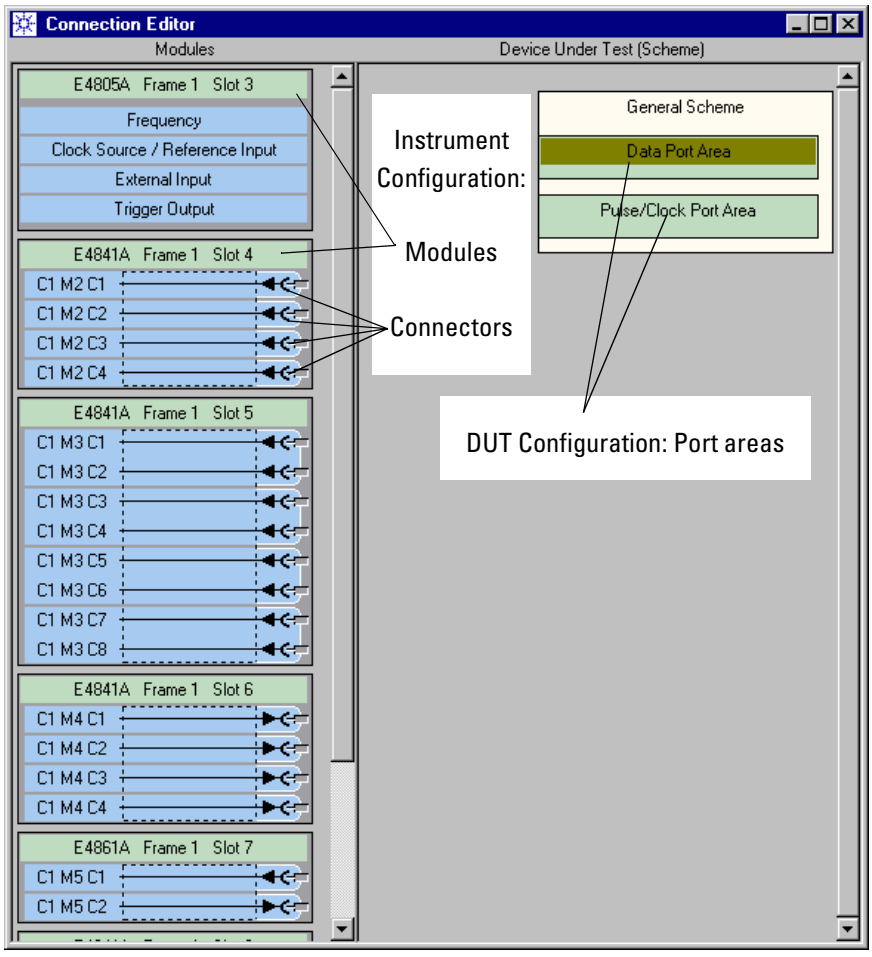


Figure 11 Connection Editor Window

Instrument Configuration

The instrument configuration identifies all the available modules, frontends (there are generator and analyzer frontends), and frontend connectors.

Refer to *"Frontends" on page 19* for the available frontends.

A **connector** represents an output or input connector of the module. Differential outputs are counted as one channel.

A **channel** represents the circuitry behind a connector. It is identified by Cx–My–Cz (ClockgroupNumber–ModuleNumber–ConnectorNumber).

DUT Configuration

The image of the DUT is constructed from a template. The Agilent 81250 system includes a general DUT template.

The general DUT template provides two types of **ports**:

- Data ports

  Data ports provide the ability to define data to be sent or analyzed. Data is handled in the form of segments. Two major data segment types are available: memory-based or PRBS/PRWS. Sequences of segments can be repeated and triggered on events.

  A data port is usually an input or output bus, characterized by one common clock frequency.

- Pulse port

  Like traditional pulse generators, this port provides an easy way to have a pulse generated without the need to set up any data. The terminals of a pulse port can receive different clock signals.

Ports can be added for every group of pins with the same or similar behavior.

Ports consist of **terminals**. These are the DUT pins that must be physically connected to the connectors of the system's generator or analyzer channels.

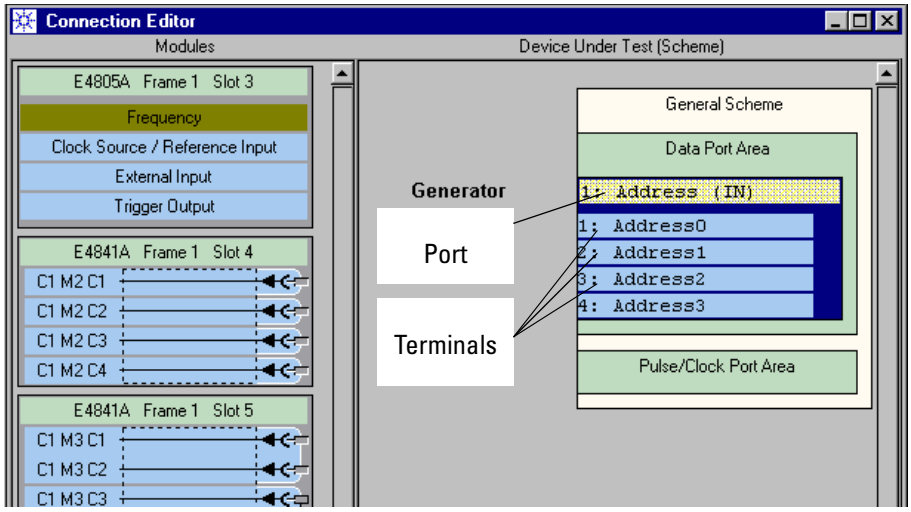


Figure 12 Display of Ports and Terminals

After the virtual DUT has been modeled and connected to the virtual instrument, signal parameters can be set, such as signal timings, pulse delay and pulse width. Signal level parameters can also be defined. Signal parameters may be set up globally for each port or individually for each terminal.

More complex signals can be produced by digital addition of two or four output channels. This allows to generate real-world signals with pulse displacement or width variation.

Analog channel addition allows also to generate signals with glitches, distorted transitions or multiple levels.

Settings

The complete setup for a DUT including all parameters is called **setting**. Settings can be saved in the system's database. It is also possible to export/import a setting as a text file, either manually or in remote control.

Every saved setting can be reloaded at any time.

Every setting also contains references to the signal patterns used for the test. These patterns are stored as segments. If a setting is exported to or imported from another system, the required segments have to be exported/imported as well. Therefore, all the segments required by a setting can be stored in a "local" segment pool which is associated with and only accessible from the setting.

Software Structure

The software is based on a client-server architecture. The Agilent 81250 system can be controlled by any of the following interfaces:

- Graphical User Interface

  This control method uses graphical windows to provide the user by means of setting up a virtual DUT just by pointing and clicking a mouse button.

- VEE Application Programming Interface (API)

  This control method uses VEE for communicating with the Agilent 81250 system remotely.

- C / C++ Application Programming Interface (API)

  This provides a mechanism for communicating with the Agilent 81250 system. An application uses the API to send command strings (based on the SCPI command language) to the instrument.

- GPIB Interface

  The GPIB interface allows remote control of the Agilent 81250 system via the General Purpose Instrument Bus interface.

- Visual Basic Application Programming Interface (API)

  This provides another mechanism for communicating with the Agilent 81250 system. An application uses the API to send command strings (based on the SCPI command language) to the instrument.

All interfaces use an ASCII-driven serial interface protocol to communicate with the firmware. The interface protocol is based on the SCPI command language.

The Agilent 81250 system software includes Agilent 81200 plug and play (pnp) drivers. These drivers provide the APIs for programs created with VEE, C/C++, or Visual Basic and perform the protocol conversion.

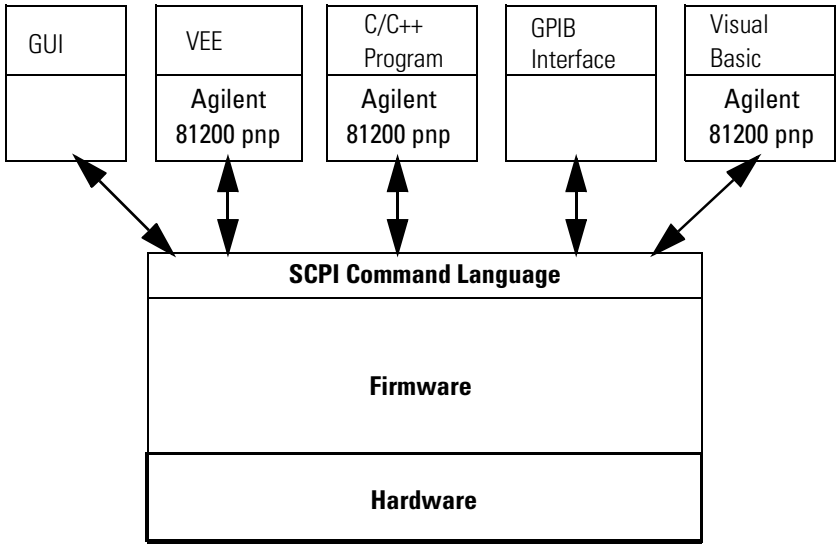


Figure 13 Client-Server Architecture

In remote-controlled operation each virtual instrument needs an individual handle so that the system knows which virtual instrument should receive the command for executing it. All commands and queries have to start with the virtual instrument's specific handle.

By default, the GPIB interface is configured as a listener. If the Agilent 81250 system shall be used as a bus controller, the configuration has to be changed. For details see *Using the Agilent 81250 as a GPIB System Controller* in the *Programming Reference*.

Summary of Setup-Related Terms

Port: A group of DUT input or output pins with identical or similar properties, such as a data bus.

Data port: A DUT port that receives or returns digital data.

Pulse port: A DUT port that receives parametric signals such as a clock pulse.

Terminal: A signal line assigned to a port (a DUT pin).

Setting: The complete setup for a DUT including all parameters and references to the test patterns (segments).

Handle: The identification of an instrument, such as DSRA.

API: Application Programming Interface.

SCPI: Standard Command language for Programmable Instruments.

PNP: 81200 Plug and Play peripheral drivers for VXI components.

Timing Principles

Depending on the frontend it is possible to test devices at frequencies up to 2.67 GHz. Multiple frequencies and memory resolutions are achieved by multiplying the clock frequency. See

"Choice of Clock Sources" on page 31

"FMR and Segment Resolution" on page 31

"Timely Alignment of Generated and Captured Signals" on page 33

"Trigger-Controlled Start and Stop" on page 34

Choice of Clock Sources

The Agilent 81250 system has a built-in 10 MHz reference oscillator. But it can also be locked to an external clock.

The external clock can be used to substitute the built-in reference. It can also be used to drive the system directly.

Additionally, the system can be started by an external signal. As long as the system does not include an E4861A module, it can also be stopped or gated by an external signal.

FMR and Segment Resolution

The internal data handling is based on words. The number of bits which are allocated to one word depends on the **segment resolution**. The word length allocated at a specific channel may be varied by setting the frequency multiplier to a value other than 1.

The available frequencies and resolutions depend on the data generator/analyzer module:

Table 1 Word Length, Memory Depth, and Frequency Range of an E4832A Module

Segment resolution (bits)	Max. memory depth (bits)	Max. data rate (Mbit/s)
1	131,008	41.67
2	262,016	83.83
4	524,032	166.67
8	1,048,064	333.33
16	2,097,152	666.6

The E4832A module provides 128 K of 16-bit words of memory for each channel. Depending on how many bits are used, this results in 128 Kbit to 2 Mbit usable memory. If 16 bits are allocated to a word, it is possible to have up to 667 MHz signals with 2 Mbit maximum memory depth.

NOTE The E4841A module has only half the capacity of the E4832A: Up to 1 Mbit per channel (64 K of 16-bit words).

The E4861A module has a memory capacity of up to 8 MB per channel (see the table below):

Table 2 Word Length, Memory Depth, and Frequency Range of an E4861A Module

Segment resolution (bits)	Max. memory depth (bits)	Max. data rate (Mbit/s)
16	2,097,152	666.6
32	4,194,304	1,333
64	8,388,608	2,666

The data streams are organized in **blocks**. Every block has a certain length. This length has to be a multiple of the segment resolution.

The desired system clock rate determines the minimum segment resolution. The system clock rate is generated by frequency multiplication. As long as the multiplying factor is less than 16 (64 for E4861A modules), memory depth may be traded against segment resolution.

The available multiplying factors and hence the available segment resolutions are expressed by the **frequency multiplier range (FMR)**. The FM factor shows the relationship between segment resolution, memory depth, and maximum system clock rate.

E4832A-Example This example refers to the E4832A module.

If the desired clock rate is 100 MHz, the minimum segment resolution is 4, which leads to 512 Kbit memory depth and a frequency multiplier range of 1/4, 1/2, 1, 2, 4. That means, an individual channel can run at 25 MHz, 50 MHz, 100 MHz, 200 MHz, or 400 MHz.

Other possible FM factors for this clock rate are:

- 8, which leads to 1 Mbit memory depth, segment resolution = 8, FMR = 1/8, 1/4, 1/2, 1, 2.
- 16, which leads to 2 Mbit memory depth, segment resolution = 16, FMR = 1/16, 1/8, 1/4, 1/2, 1.

The relations are shown in the table below:.

Table 3 Matrix of Segment Resolution, FMR, Memory Depth and Clock Frequency

Segment Resolution	Frequency Multiplier Range[a]	Memory Depth[b]	System Clock Rates
1 bit (=1)	1, 2, 4, 8, 16	128 Kbit	≤ 41.67 MHz
2 bits (=2)	1/2, 1, 2, 4, 8	256 Kbit	≤ 83.83 MHz
4 bits (=4)	1/4, 1/2, 1, 2, 4	512 Kbit	≤ 166.67 MHz

Table 3 Matrix of Segment Resolution, FMR, Memory Depth and Clock Frequency

Segment Resolution	Frequency Multiplier Range[a]	Memory Depth[b]	System Clock Rates
8 bits (=8)	1/8, 1/4, 1/2, 1, 2	1 Mbit	≤ 333.33 MHz
16 bits (=16)	1/16, 1/8, 1/4, 1/2, 1	2 Mbit	≤ 666 MHz

[a] This is the range of multiples and fractions that can be used at individual connectors. If you have most of your signals at 40 MHz and your pattern lengths are less then 64 Kbit, then you can choose segment resolution 1. You have the chance to set individual connectors to a multiple of this general setting. For example, selecting 16 as the multiply factor for a connector gives you 1 Mbit memory depth and 640 MHz with a segment resolution of 16.

[b] Subtract 32 x segment resolution, as this memory space is occupied by a 2^5-1 PRxS and the sequencing initialization.

If, for example, most of the signals are at 200 MHz, then the available corresponding segment resolutions are either 8 or 16.

If you have chosen 8 as the general segment resolution, then each data port and each terminal of a pulse port can be set individually to frequencies of 1/8, 1/4, 1/2, 1 or 2 times the system clock frequency.

If the frequency multiplying factor is changed for individual ports or terminals, then the segment resolution, memory depth and frequency also change for these connectors.

NOTE Additional restrictions apply for E4841A modules with generator or analyzer frontends in combination with certain measurement modes. For details see the *Agilent 81250 Technical Specifications*.

Timely Alignment of Generated and Captured Signals

Several features support the timely alignment of signals.

Zero Adjust, Cable, and Propagation Delay Compensation

With the Agilent 81250 system it is possible to align the outputs and inputs of newly installed frontends or new modules.

To assure that all generator output signals are applied at the same time either at the DUT board or even at the DUT input pins it is possible to perform a cable delay and propagation delay compensation.

The procedures are menu driven and semi-automatic. For details see *"How to Compensate for Internal and External Delays" on page 216*.

Negative Delay

Apart from aligning signals it may be interesting and important to have some signals applied in advance to other signals.

Therefore the Agilent 81250 system provides the option to set a general time offset for all connectors, so that individual ones can be set to negative delays and hence start earlier than others.

The delay offset feature can be used in setup and hold time measurements. For details see *"How to Set the General System Frequency" on page 94*.

Trigger-Controlled Start and Stop

The EXT INPUT of the central clock module can be used to start and stop the timing system of the Agilent 81250. The state of this input is sampled once every system period.

The system period is normally derived from the built-in PLL, but if the CLOCK/REF INPUT provides an External Clock Source, this signal defines the system period.

When using the EXT INPUT without an External Clock Source at the CLOCK/REF INPUT, the start/stop signal must be applied for a time greater than the system clock period.

When using the EXT INPUT with an External Clock Source at the CLOCK/REF INPUT, setup and hold time must be considered to achieve a predictable timing.

NOTE Setup and hold time violations may influence the Agilent 81250 System only in the aspect that the system will generate consistent relative timings for data generation and data capture, but the absolute timing, related to the external input, may vary by about ±1 system clock periods.

Trigger-Controlled Start

Starting the system via the EXT INPUT has no restrictions. All internal pipelines are prefilled so that the first signal comes out after

±1 system periods + 45 ns + output delay.

Trigger-Controlled Stop

Trigger-controlled stop or trigger-controlled gating via the EXT INPUT is not available, if the system contains E4861A modules.

When stopping the system via the EXT INPUT some restrictions apply.

The timing system is stopped immediately, even if the period and delay of a bit is not complete. The consequence is, that the word at the output might not be aligned during such a stop. After a restart (in Gated mode) the bits are realigned again.

When stopping the system ensure that the output word is stable for a longer time. Use a PAUSE segment for that purpose. The pause should have a duration of

system period * frequency multiplier * 32 + maximum delay,

where the maximum delay is the maximum delay of all involved channels. Details are documented in the *Agilent 81250 Technical Specifications*.

Summary of Timing-Related Terms

FMR: **Frequency Multiplier Range**. The available factors for multiplying the clock frequency (1/16 to 16, dependent on the clock rate). The actual range is determined by the segment resolution.

FM: Frequency Multiplier. The individual factor by which a channel frequency differs from the system clock. Choices are restricted by the FMR.

Block: A portion of a test sequence which references transmitted or expected data. A block refers to all data ports.

Segment resolution: The available resolution of the block in bits. Depends on the chosen frequency multiplier factor.

Channel addition: Exclusive OR (modulo 2) addition of two or four signal generator channels of one module.

Data Generation Principles

Once the general signal parameters have been set up, it is time to apply patterns. See:

"Emulate Real Pattern and Waveform Conditions" on page 36

"Data Sequences" on page 36

"Data Blocks" on page 37

"Data Segments" on page 38

"Properties of Real Segments" on page 39

"Loops" on page 41

"Hardware Dependencies" on page 42

Emulate Real Pattern and Waveform Conditions

Data patterns can be stored in the system database and output as part of a sequence with or without algorithmic data. A PRBS, for example, is algorithmic data.

Data patterns for the signals sourced to or expected from the DUT can easily be set up in terms of data **segments** that span across several output or input connectors of the Agilent 81250 system.

Captured data or data produced by a simulation can be imported as an ASCII text file.

The Agilent 81250 system can be used to stimulate communication devices using its sequencing capability.

Packets or cells consisting of payload and control data can be produced by creating control segments and using a PRBS segment for the payload. Cell/packet size can be varied and control segments can be stored in the database and used in any number of different packets. A PRBS pattern may be used as the payload to test error rates. Intermittent data with long dead-times between bursts can easily be produced using the pause segment.

For testing multiplexers/demultiplexers it is possible to set up PRWS data and compare segments. Also, it is possible to run different ports at different frequencies.

The overall stream of generated and expected data is called a **sequence**. The Sequence Editor defines the structure of the data streams sent to or expected by the Agilent 81250 system.

A sequence includes all data ports of the device under test. It does not include pulse ports.

Data Sequences

A sequence specifies which data segments are generated or expected, on which ports, and in which timely order.

For details see *"Creating the Stream of Generated and Expected Data" on page 139*.

A sequence consists of **blocks**. Loops are also part of a sequence.

Sequences are independent of data. This is achieved by defining the data to be generated or expected with the Segment Editor or externally as vectors in a text file and referencing these segments in the sequence blocks individually for each data port.

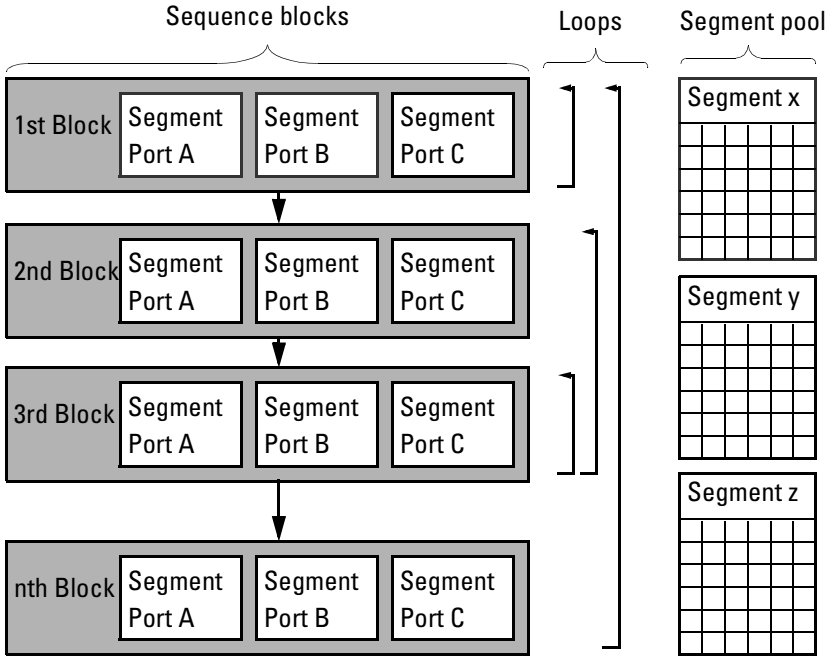


Figure 14 Sequence and Segments

Every block can reference an arbitrary segment for every data port.

Data Blocks

Blocks are portions of the sequence. A block spans across all data ports. For details see *"Contents of the Detail Mode Sequence Editor Window" on page 153*.

Every block references a **segment** for each DUT data port (not pulse ports). The segments contain the patterns of generated and expected data.

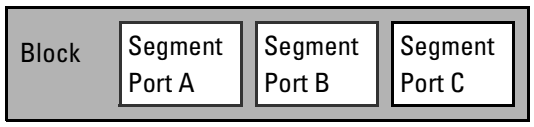


Figure 15 Block Structure

NOTE Segments are not included into but only referenced by the blocks.

The length of a block must be a multiple of the segment resolution.

Single, several or all blocks may be repeated a specified number of times or perpetually.

Such loops have an impact on the minimum blocklength and the allowed number of blocks (see *"Hardware Dependencies" on page 42*).

Trigger pulses can be specified to be generated at the beginning of a block and output by the TRIGGER OUTPUT connector of the master clock module. If certain **events** have been detected while a block is executed, actions can be performed immediately or at the end of the block. See *"Usage of Events" on page 52*.

A single block can also be used for synchronizing the analyzer frontends with the incoming data stream. See *"Analyzer Sampling Point Adjustment" on page 46*.

Data Segments

Segments can be freely created. A segment has a width and a length. The width defines the number of parallel signal lines (traces). The length defines the number of data words (vectors). The length of a segment must not remain under the length of the block into which it is going to be inserted.

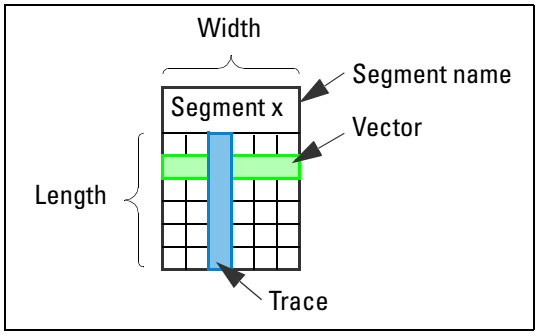


Figure 16 Segment Structure

Segments can be created separately or while editing the sequence. A segment may be larger than the block that references it.

If an existing segment is referenced by a block, the width of the port and the length of the block are automatically considered. Segment data that does not fit into the current block is ignored.

Segment Types There are real segments and pseudo segments.

- Real segments contain either free programmable memorized data, or the specification of PRBS or PRWS data.
- Pseudo segments are commands, such as Pause, Don't care, Expect 0, Acquire, and so on. Pseudo segments can save channel memory.

New segments are created with the Segment Editor or can be imported from vector-formatted text files.

Segment Pools

Segments are stored in a segment pool, which is part of the system database. There is one segment pool with global scope and one segment pool per setting with local scope.

- Segments in the local segment pools can only be accessed if the appropriate setting is loaded.
- Segments in the global segment pool can be accessed from any setting.

Using the "local" segment pool makes it easy to export all the segments required by the current setting if the setting is going to be exported to another system.

Properties of Real Segments

We distinguish between memory segments and PRBS/PRWS segments.

- PRBS/PRWS segments are defined by the polynomial they are calculated from. The width of a PRWS segment is automatically adjusted to the width of the port to which it is assigned.
- Memory segments consist of vectors and traces. A **vector** specifies all the parallel bits of a port. The serial bit stream of a terminal line is called **trace**.

Data Memory Usage

To understand data memory consumption in sequence mode it is best to think in data words. A word usually consists of 1 to 16 bits, depending on the segment resolution and the frequency multiplier setting of the specific channel. A word of an E4861A module, however, consists of 16 to 64 bits.

- One word is reserved for internal use.
- A PRBS/PRWS segment assigned to an E4841A module consumes as many words as its polynomial says. A 2^{15}–1 PRBS, for example, consumes 32767 words.

  A PRBS/PRWS segment assigned to an E4861A or E4832A module consumes memory only, if it is distorted. Pure PRBS/PRWS segment

are not produced by the controller and downloaded but directly generated by the E4832A or E4861A modules.

- Even if you don't use a distorted PRBS, there is a 2^5–1 PRBS allocated internally, which means 31 words are allocated.
- A pseudo segment (Pause0/1, for example) consumes 1 word, if such a segment is used at all channels of the module.
- The remaining memory is used for the programmable memory-type data segments.

Segment Type Combinations

The various data generator/analyzer modules have different capabilities.

E4841A Module An E4841A data generator/analyzer module can only execute either memory type segments or PRBS/PRWS segments at the same time.

PRBS/PRWS can also be combined with Pause0/1, Expected0/1, or Don't Care. That means that such a channel must always use that segment when PRBS segments are used on that module.

Pause0/Pause1, Expected0/Expected1, Don'tCare segments save memory if all channels of the module execute such type of segments at the same time.

On a module with analyzer frontends only, all connectors can either be in Pause mode or they must all be in another mode than Pause, for example, Capture or Acquire.

E4832A Module The E4832A data generator/analyzer module can simultaneously execute memory type segments and pure PRBS/PRWS segments. If distorted PRBS is specified, the same restrictions as for the E4841A module apply.

E4861A Module The E4832A data generator/analyzer module can execute memory type segments and pure PRBS/PRWS segments in parallel.

Data to Connector Assignment

The algorithm how the available segment data is assigned to the connectors is as follows:

- The first terminal within a port gets trace0, the second gets trace1 and so on.

  The assignment to the connectors depends on what connections you have selected from the terminals to the connectors in the Connection Editor.

- If a terminal is connected to a connector where channels are added, the connector that holds the connection gets the first trace. An added channel gets the next trace.

  In this case an exception to the rule “from top to bottom” is made. Added channels are assigned from bottom to top.

Loops

Loops are used to repeat data blocks. For details see *“How to Create and Change Loops” on page 162*.

Looping Example

A sequence looping 1 Kbit portions of a 2^{15}–1 PRBS followed by a pause of 64 bits infinitely might look as follows:

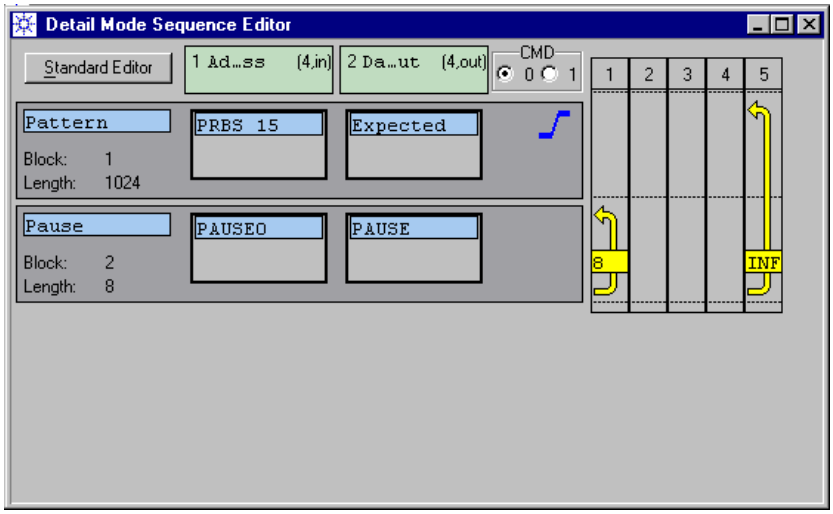


Figure 17 Simple Test Sequence With Loops

If you had chosen a memory-type segment in block one, this segment would start from the beginning with every loop. If the segment is larger than the portion used in the sequence, then there is data that is never generated in this sequence.

However, this is different when looping blocks with PRBS/PRWS segments.

Looping Blocks With PRBS/PRWS Segments

If you do not use the complete PRxS you have chosen for your application but are looping it, then with each new loop cycle the next portion of the PRxS is used, when the looping is going on.

It can happen that the next portion is the rest of the PRxS and a bit of the beginning. The portioning of the PRxS will go on as long as the looping lasts.

The data representation in the Agilent 81250 system memory and the data stream output for one connector may look as follows:

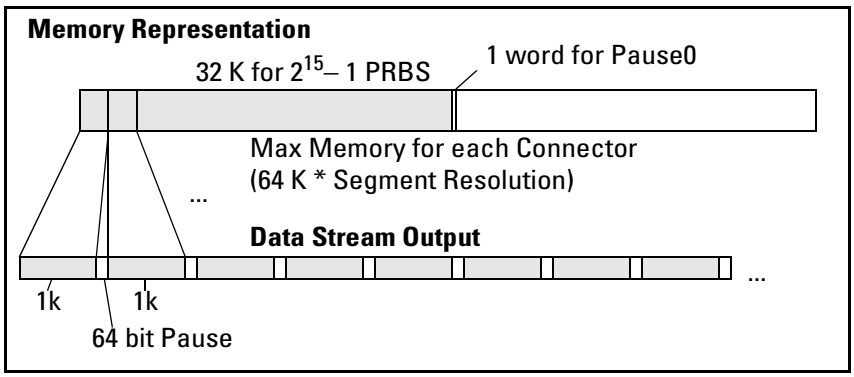


Figure 18 PRBS Memory Representation and Generated Data Stream

Hardware Dependencies

The number of blocks and loops that can be used depends on the selected sequencer type (incorporated in the central clock module).

Generating Loops With the E4805B Clock Module

The E4805B Clock Module provides:

- 4 counted loop levels (1 performed by the data module, 3 by the clock module).
- Loop counts may be up to 2^{20}. Note that for loop level 1 there is a restriction:

  LoopCount1 * (blocklength / segment resolution) ≤ 2^{20}.

  If this restriction is not met, the loop level 1 can not be used. In this case the looping has to be performed on a higher loop level.
- 1 infinite loop in level 5
- Number of blocks + number of loops on level 1 must be ≤ 60.
- Number of blocks + number of loops on levels 2, 3, or 4 must be ≤ 60.
- Minimum blocklength:
  - segment resolution, if no counted loop is used.
  - 2 * segment resolution, if one loop is starting on level 1, 2, 3, or 4.
  - 4 * segment resolution, if two loops are starting on level 2, 3, or 4.
  - 5 * segment resolution, if loops are starting on level 2, 3, and 4.

Summary of Data-Related Terms

Sequence: The overall stream of generated and expected data, based on sequence blocks.

Block: A portion of a sequence that references segments with generated and expected data and actions on events. Comprises all data ports. Single blocks and groups of blocks can be repeated (loops). A trigger pulse can be issued at the beginning of a block.

Segment: Contains the data to be generated or expected: A certain pattern, PRBS, or PRWS. PRxS means algorithmic data. A pattern consists of vectors and traces.

PRBS/PRWS: Pseudo Random Bit/Word data Stream.

Vector: Specifies all the parallel, simultaneous bits of a port within a segment.

Trace: Specifies the serial data transmitted to or expected from a terminal.

Data Capturing and Analysis Principles

The system provides four test and measurement modes. Tests can be preceded by a synchronization procedure during which the analyzer frontends optimize the position of the sampling point. See:

"Functional Tests" on page 44

"Error Analysis and Marginal Tests" on page 45

"Analyzer Sampling Point Adjustment" on page 46

"Display of Test Results" on page 51

Functional Tests

The functional tests are chosen from the Measurement Configuration window. For details see *"Choosing the Kind of Measurement" on page 135*.

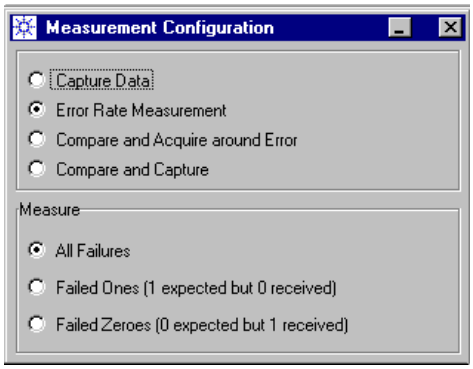


Figure 19 Measurement Configuration Window

The functional tests include:

Capture Data In this mode, the instrument captures data until the memory is filled. The result can be reviewed in a state list and also graphically. Depending on the data generator/analyzer module and the segment resolution, an analyzer can capture up to 1, 2, or up to 8 Mbit of data.

Bit Error Rate Measurement The Bit Error Rate Measurement scans the received data in real time and shows the resulting actual and accumulated number of bits, the actual and accumulated number of errors, and the actual and accumulated bit error rate. The display is updated every second.

Compare and Acquire Around Error The Compare and Acquire around Error mode compares and acquires data in real time. The memory capacity in words/channel and the number of bits per word depend on the type of module and the chosen segment resolution or frequency multiplying factor, respectively.

Table 4 Capture Memory

	E4841A	E4832A	E4861A
Storage capacity	64 K words	128 K words	128 K words
Bits/word	1 to 16	1 to 16	16 to 64

If an error occurs, it is possible to define when the system should stop after the occurrence of the error.

If the system includes E4832A modules, then the minimum value to stop is 976; if E4861A modules are present, the minimum is 3904.

The maximum is the available memory, calculated as

storage capacity * segment resolution.

The captured data including the errors can be viewed as an Error State list and also graphically with the Waveform Viewer.

It is possible to load expected data segments, which may have been captured from a reference device or imported from a simulation.

Real-time compare up to 165 MHz can be achieved with dual-input analyzer frontends. Real-time compare up to 330 MHz can be achieved with single-ended analyzer frontends.

Real-time compare up to 660 MHz is supported by the E4832A module equipped with E4835A frontends.

The E4861A modules support real-time compare up to 2.6 GHz.

Compare and Capture

The Compare and Capture mode compares and acquires data in real time. It continues until the sequence expires or the Stop button is pressed. The same frequency restrictions apply as stated above.

This mode is first of all intended to be used with event handling where the reaction on an error is specified within the sequence.

You can view the result in the Error State Display where errors are highlighted, and also graphically with the Waveform Viewer.

Error Analysis and Marginal Tests

A device can be stimulated with arbitrary input signals using the variable pulse parameters provided by the Agilent 81250 system. These functions are provided by the 667 MHz module and its generator frontends.

Parameters include levels, delay, and width and can be varied independently for each channel or for a DUT port as a whole.

Distorted signals as well as glitches and pulse delay variations can be emulated using the channel addition capabilities of the Agilent 81250 system. Up to four channels can be added to emulate a real-time pulse delay variation with up to four phases.

E4838A generator frontends are additionally supported by the analog channel add function which allows to generate signals with overshot and ringing.

For details see *"How to Set Up a DUT Input Port or Generator Channel" on page 117*.

Analyzer Sampling Point Adjustment

The proper comparison of received data with expected data requires that the analyzer captures the incoming data at the right point of time.

That means first of all that the analyzing ports, modules, or frontends have to be triggered by a suitable clock frequency.

The sampling frequency may be an issue:

- Received data usually arrives with the frequency of the stimulating signal or an integer multiple or fraction thereof. If the sampling frequency is a 2^n-multiple or fraction of the system master clock frequency, this can often be handled within one system. The frequency multiplier provides adequate choices.
- If the sampling frequency is not a 2^n-multiple or fraction of the system master clock, instruments with independent clock modules have to be installed. They can reside in one and the same mainframe, but if they are to be operated manually, each requires that you start its own user interface.
- Separated generating and analyzing instruments and the DUT can be frequency-synchronized to one clock source. The clock source can be the built-in oscillator of the master clock module, an external reference or even an external source.

For adjusting the sampling start delay and phase, the Agilent 81250 Parallel Bit Error Ratio Tester offers three methods. These methods allow to determine and set the delay before the measurement starts as well as the optimum sampling point.

For details see:

"Manual Analyzer Sampling Point Alignment" on page 47

"Automatic Delay Alignment" on page 48

"Automatic Bit Synchronization" on page 49

The automated methods require a special synchronization block within the test sequence. This block should be the first block that specifies expected data.

The synchronization block may be preceded by Pause blocks. Such blocks can be used for establishing a certain delay before the synchronization starts. A delay may be required for warming up the DUT or giving PLLs time to settle. If the synchronization block is embedded somewhere in a sequence, it should be labeled “START” because the sequence execution begins with the START block.

The synchronization block is automatically repeated until the synchronization criteria are met. The sequencer continues after the analyzers are synchronized.

Manual Analyzer Sampling Point Alignment

For every analyzer channel, the Parameter Editor allows to specify the start delay between the start of the system clock (usually the start of the generators) and the start of the analyzer. The start delay can be specified as a certain amount of time plus a multiple or fraction of a clock period.

If one of these parameter is changed while a test is running, the test is aborted and restarted.

Analyzer frontends plugged into an E4861A or E4832A data generator/analyzer module are also supported by a phase vernier. This vernier, implemented as a slider in the Parameter Editor, allows to shift the analyzer sampling point by up to ±1 clock periods without interrupting a running test.

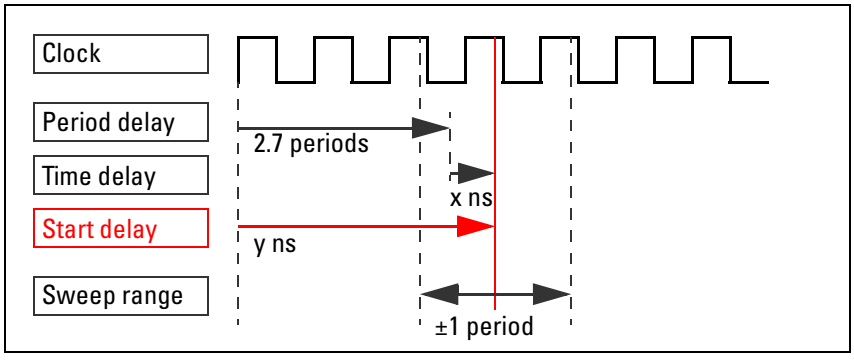


Figure 20 Analyzer Start Delay and Manual Delay Sweep Range

When the sampling point is moved, the bit error rate changes. By observing the bit error rate (BER) one can thus measure the eye opening of a superimposed or differential signal.

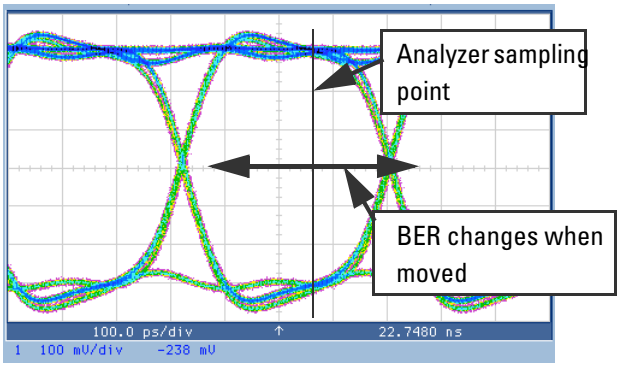


Figure 21 Eye Diagram of a 2.6 GHz Signal

The Parameter Editor always indicates the actual delay. Once the width of the eye opening is known, the analyzer sampling point can be put into optimum position which is in the middle

Automatic Delay Alignment

This function is only available for analyzer frontends plugged into an E4861A or E4832A data generator/analyzer module.

Automatic delay alignment is used if the expected propagation delay can be coarsely specified. The same data must be generated and expected within one sequence block.

Once the start delay has elapsed, the analyzer then searches in both directions for a sampling point at which bit recognition has an adequate, adjustable accuracy. This accuracy is defined by a bit error rate threshold. The search range is limited to ±50 ns for an E4832A data generator/analyzer module and ±10 ns for an E4861A module.

After that, the analyzer shifts the sampling point stepwise in both directions. The width of these steps is adjustable. The analyzer measures the width of the eye diagram and positions the sampling point at the optimum which is in the middle.

If the phase vernier is not in zero position, its setting is now added to or subtracted from the optimum sampling point.

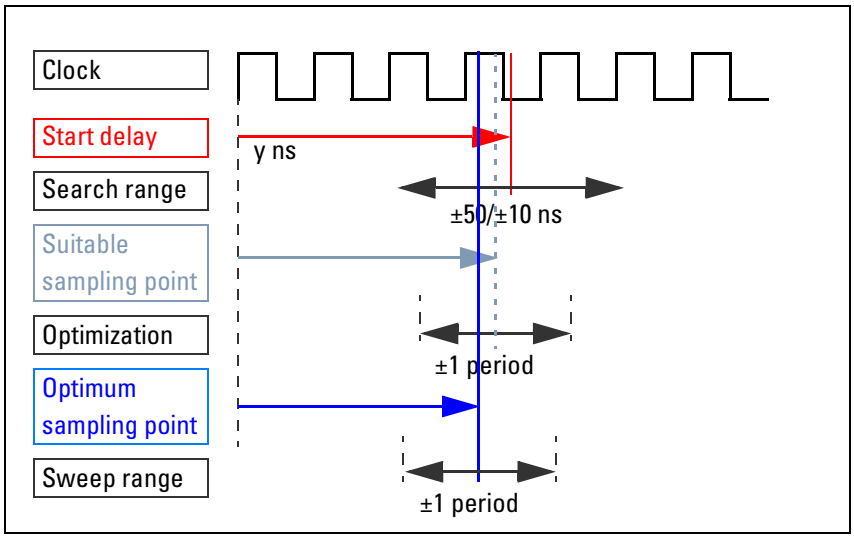


Figure 22 Automatic Delay Alignment

The resulting absolute delay since starting the test is set and indicated in the Parameter Editor window of the DUT output port.

Once the delay has been found, the phase vernier allows to shift the analyzer sampling point by up to ±1 clock periods while the test is running.

Automatic Bit Synchronization

This function is only available for analyzer frontends plugged into an E4861A or E4832A data generator/analyzer module.

Automatic Bit Synchronization is used to align the incoming data pattern with the expected pattern. Automatic Bit Synchronization offers the option to enable or disable Automatic Phase Alignment:

- Automatic Bit Synchronization without Automatic Phase Alignment is used if the total delay from test start is unknown but a certain edge delay relative to the analyzer clock is expected.
- Automatic Bit Synchronization with Automatic Phase Alignment is used if the delay is completely unknown.

Pseudo random data can be sent and expected within one sequence block. A pure analyzing system may also expect memory-type data.

Auto Bit Sync without Auto Phase Alignment

The analyzer uses the start delay that has been specified with the Parameter Editor to determine the sampling point in relation to its clock. It calculates the sampling point position as “start delay modulo periods”. That means, if the start delay includes a number of full periods, these periods are ignored.

The analyzer then samples the incoming data until the incoming data matches the expected pattern with an adequate, adjustable accuracy. This accuracy is defined by a bit error rate threshold.

In order to minimize the time needed for synchronization, the algorithm takes the kind of expected data into account.

If memory-type data is used for Automatic Bit Synchronization, the first 48 expected bits of each analyzer channel have to be unequivocal. They may include don’t care bits.

Once the desired accuracy is reached, the incoming bits are aligned with the expected bits—the analyzer is synchronized with the incoming data.

Auto Bit Sync with Auto Phase Alignment

If Automatic Phase Alignment is enabled, then the analyzer fully automatically adjusts itself to capture the incoming data at the optimum sampling point.

It shifts the sampling point stepwise in both directions until the specified bit error rate is reached. The width of these steps is adjustable. The analyzer then measures the width of the eye diagram and positions the sampling point at the optimum which is in the middle. Actually, the same optimization procedure as for Automatic Delay Alignment is applied.

If the phase vernier is not in zero position, its setting is now added to or subtracted from the optimum sampling point.

The delay found by Automatic Bit Synchronization, which is set and indicated by the Parameter Editor, is relative to the analyzer’s sampling clock. It does not report the absolute delay that would be required between the start of the generator and the analyzer for capturing a complete pattern.

Once the analyzer has been synchronized, the phase vernier allows to shift the analyzer sampling point by up to ±1 clock periods while the test is running.

Display of Test Results

The results of a bit error rate measurement are displayed in the Bit Error Rate Display.

- The Bit Error Rate window shows the current and accumulated results and is continually updated.

  Captured data and the results of real-time compare tests can be investigated with the Error State Display and the Waveform Viewer.

- The Error State Display shows the captured data and errors in tabular form. Auxiliary functions are provided that support quick navigation.
- The Waveform Viewer shows the captured data and errors in graphical form. It provides waveform selection as well as markers and zoom for precise waveform analysis.

For details see *"Viewing Generated and Captured Data" on page 203*.

Summary of Analysis-Related Terms

BER: Bit Error Rate.

Start delay: Analyzer sampling point setting with Parameter Editor.

Delay vernier: A slider provided by the Parameter Editor for analyzer frontends plugged into an E4861A or E4832A data generator/analyzer module.

Automatic Delay Alignment: Analyzer sampling point optimization if time window is known. Sets and shows the full delay since start.

Automatic Bit Synchronization: Analyzer sampling phase adjustment based on BER. Sets and shows the phase delay with respect to the analyzer clock.

Error State Display: Shows captured data and errors in tabular form.

Waveform Viewer: Shows generated and captured data as well as errors in graphical form. Allows to investigate phase relationships.

Event Handling Principles

The Agilent 81250 system can detect a variety of events and react on events.

The reaction may simply be a trigger pulse at the TRIGGER OUTPUT of the clock module, but can also be a change of the test sequence. See:

Usage of Events

"What is an Event?" on page 53

"Actions Upon an Event" on page 53

Usage of Events

Reacting on events provides many capabilities:

- Stop and go:

  This is useful for production tests, where data is sourced to the DUT, a measurement is performed with other equipment, the next data pattern is sourced, and so on.

- Block switching:

  The data sequence is no longer fixed. Based on certain events, certain portions of the overall sequence can be executed.

  This has the advantage that one and the same sequence can be created and downloaded once and then used for several tests. There is no need for re-programming the instrument.

- Trigger external devices:

  The event can generate a trigger signal at the TRIGGER OUTPUT of the clock module. This can be used to trigger an external instrument like a sampling oscilloscope or logic analyzer to sample the data at an error location.

- Bolt on:

  The Agilent 81250 system can be integrated into a large IC test system. The IC tester would issue a trigger to start the Agilent 81250 system for a special measurement. The Agilent 81250 system would perform the test and return pass/fail information that can be examined and evaluated by the IC tester.

- Match loop:

  PLL-based devices typically require an initialization segment that has to be repeated until the device is synchronized. The event that controls repetition would be “an error occurred”.

For setup examples see *“How Do I Use Events?” on page 232*.

What is an Event?

An Agilent 81250 system equipped with the Agilent E4805B clock module has a whole bunch of options. It can react on

- any bit combination of the 8-bit trigger pod (see *“Trigger Pod” on page 22*).
- any bit stream error detected by one of the analyzer frontends,
- the status of the VXI ECL trigger lines T0 and T1,
- an event triggering command issued locally or remotely.

Ten events can be defined—five for immediate actions and five for deferred actions.

Events Causing Immediate Action

Actions on such events occur immediately (although there is an internal delay). They can be used to launch a trigger or to abort the test, for example.

Events Causing Deferred Action

Actions on such events occur at the end of the current sequence block. If the events come asynchronously, this feature ensures that the current block is properly executed and terminated.

These events are associated with priorities. Event number 5 has the highest priority, event number 1 the lowest.

NOTE Events for immediate action override events for deferred action.

Actions Upon an Event

If an event occurs, the system provides the following options:

- Go to:

  Goes to a certain block in the overall sequence and executes that block. The block is identified by its block label. The implicit “End” block (which is automatically assigned and does not need to be defined) terminates the sequence and hence the test.

- Trigger:

  Launches a trigger pulse to the TRIGGER OUTPUT of the central clock module.

- VXI-T01:

  Sets the VXI ECL trigger lines T0 and T1 to 01, 10, or 11.

All these options can be freely combined.

Summary of Event-Related Terms

Event: A signal that must be responded to.

Event causing deferred action: An event that causes an action at the end of the currently executed sequence block.

Event causing immediate action: An event that is serviced as fast as possible, without waiting for the end of the currently executed sequence block.

Action upon an event: Any combination of the following:

- Go to sequence block.
- Set the TRIGGER OUTPUT of the central clock module.
- Set the VXI ECL T0/T1 trigger lines.

Test Development Overview

The development of a device test is an iterative process:

1. Set up the test.
2. Run the test.
3. Check the test results.
4. Modify test parameters.
5. Repeat steps 2 to 4 until the results are adequate.
6. Save the final setting for reuse.

All these steps are supported and simplified by the graphical user interface of the Agilent 81250 Parallel Bit Error Ratio Tester.

This chapter provides an overview:

"Procedure for Setting Up the Test" on page 56

"Procedure for Running the Test" on page 58

"Procedure for Viewing Test Results" on page 59

"Procedure for Saving the Test Setting" on page 60

Procedure for Setting Up the Test

To set up the test for a new device, it is recommended to perform the following steps in the given order:

1 Study the Device Under Test (DUT).

Identify its input, output, clock, and trigger or strobe terminals. Gather information about its electrical, logical, and frequency characteristics.

In fact, this is the most important step of all.

2 Start the Agilent 81250 system. It comes up with the Connection Editor and the default setting which is called “untitled”.

If you had already set up the instrument and DUT, you would now load the appropriate setting.

For details see *“Open Setting” on page 74*.

3 To create a new setting, construct an image of the DUT on the screen and connect the DUT pins to the connectors of the generator and analyzer frontends.

This is supported by the Connection Editor, which by default shows an image of the instrument and an empty template for modeling DUT input and output pins.

For details see *“Connecting the DUT” on page 105*.

4 Set the global system parameters.

These parameters refer to the central clock module and cover items like clock source, clock frequency, use of an external trigger, control of the built-in trigger generator, etc. Data ports as well as pulse port terminals and unconnected channels can be set to fractions or multiples of the system clock rate.

The tool for setting all kinds of parameters is the Parameter Editor. For details see *“Setting Global System Parameters” on page 91*.

5 Set the characteristics of the input and output connections.

The characteristics include parameters like voltages, delays, impedances, binary data representation, and so on. This is also done with the Parameter Editor, which in turn can be run conveniently from the Connection Editor.

For details see *“Setting Up Ports and Channels” on page 115*.

6 Decide what kind of test you wish to perform.

Open the Measurement Configuration window. Choices are bit error rate measurement, capture and compare, or just capture DUT output data.

For details see *"Choosing the Kind of Measurement" on page 135*.

7 Create the stream of generated and expected data.

For this purpose, the software provides three Sequence Editors: the Standard Mode Sequence Editor, the Detail Mode Sequence Editor, and the Data/Sequence Editor. These editors enable you to create and maintain the data blocks that form the test sequence.

The Standard Mode Sequence Editor supports easy setup of bit error rate measurements. The other two editors allow to create an arbitrary sequence.

For details see *"Creating the Stream of Generated and Expected Data" on page 139*.

8 Create the data segments referenced by the blocks.

Each block of a sequence contains data segments that specify the generated and expected data. Stored segments can be chosen from lists. New segments can be created with the Segment Editor.

For details see *"Creating and Editing Segments" on page 175*.

9 Connect the DUT physically to the instrument.

Use the Connection Editor to ensure that all physical connections match the image on the screen.

10 Setting up the test of a new device sometimes requires that you change cables, add modules or frontends, or change the DUT board. In this case you should compensate the setup for different signal propagation and cable delays. This can be done with the Deskew Editor. See *"How to Compensate for Internal and External Delays" on page 216*.

Now you are ready to run the test.

Procedure for Running the Test

After you have finished the setup:

1 Download the test sequence to the modules.

This is done by clicking the Prepare button. The download procedure checks whether the test sequence is formally correct and can be executed.

As downloading a complex sequence can take some time, this is also recommended before running a test that is to be started by a trigger.

For details see *"How to Download the Test Sequence" on page 200*.

2 If the test has been set up for measuring the bit error rate, open the Bit Error Rate Measurement Display window.

For details see *"How to View BER Test Results" on page 200*.

3 Click the Run button.

If the test is set up to be controlled by an external start trigger, it will now wait for this trigger. If not, it starts immediately.

The test will run until the test sequence is executed or the capture memory is full or, if it is controlled by an external stop trigger, until the trigger is set—whichever comes first.

4 If the test sequence includes an infinite loop, stop the test by clicking the Stop button.

Procedure for Viewing Test Results

If you are running a bit error measurement, the Bit Error Rate Measurement Display window shows you actual and accumulated results. The display is updated every second.

For details see *"How to View BER Test Results" on page 200*.

If you have been running one of the other tests:

1 Open the Error State Display.

The Error State Display shows the captured DUT output data. If you have been running one of the compare tests, it shows also the deviations between the captured and expected data. Various address and data formats support the investigation.

For details see *"How to View Captured Test Results" on page 204*.

2 Open the Waveform Viewer.

The Waveform Viewer enables you to display generated and captured data as well as compare results graphically in a variety of formats.

For details see *"How to View Waveforms" on page 209*.

Procedure for Saving the Test Setting

It is recommended to save the test setting repeatedly during test development. This ensures that whatever occurs you can always return to the last saved test configuration.

As the system provides different options for different kinds of measurements, it is recommended to save every measurement configuration, such as bit error rate or compare and capture, in its own file.

To save a new setting:

1 Open the *File* menu.

2 Choose *Save Setting As*.

3 Enter a filename that gives some information about the purpose of the setting.

4 Confirm.

To save the current setting occasionally:

1 Click the Save Setting icon.

NOTE Note that settings can also be exported to another directory or drive and imported from there (see also *"Export/Import of a Setting" on page 222*).

System Start and User Interface

The Agilent 81250 user interface basically consists of a window frame, several editors for test setup, and several windows for displaying the test results.

The menu bar on top of the window frame can be used to access the individual windows and to operate the system. Shortcuts are provided by the tool bar buttons.

This chapter provides basic information on both the windows and the options of the main menu:

"How to Start the System" on page 62

"Overview of the Windows" on page 66

"Operating the User Interface" on page 68

"Items of the Main Menu" on page 71

How to Start the System

At a factory-configured Agilent 81250 system with built-in controller, the Windows NT automatic log-in script is enabled. After power on, you are automatically registered as user DVT and the Windows desktop appears.

How to Start the Agilent 81250 Software

The Agilent 81250 software can be run in one of three modes:

- Local.

  This mode starts both the Agilent 81250 hardware server and the user interface. It is used, if the hardware shall be controlled and the user interface shall be run by the built-in computer.

- Controlled.

  This mode starts the Agilent 81250 hardware server only. It is used, if the hardware shall be controlled by the built-in computer, but the user interface shall run on a remote computer. The system can then be operated via GPIB or LAN.

- Remote.

  This mode starts the Agilent 81250 user interface only. It is used on a remote computer to operate an Agilent 81250 server which is in controlled mode.

How to Start the System in its Last Operating Mode

To start the system in its last operating mode:

1 Double-click the Agilent 81250 User Software icon.

Figure 23 Agilent 81250 Start Icon

If the system is started in Remote mode, the default hostname and network port number are displayed:

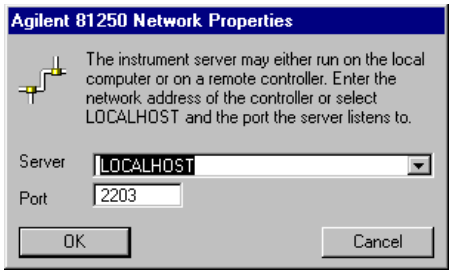


Figure 24 Agilent 81250 Start in Remote Mode

If the defaults have been changed, appropriate parameters have to be entered. For details please refer to the *Agilent 81250 Installation Guide*.

How to Start the System in a New Operating Mode

To change the system's operating mode:

1 Double-click the Agilent 81250 Configuration icon.

Figure 25 Agilent 81250 Configuration Icon

This opens the Startup Configuration window:

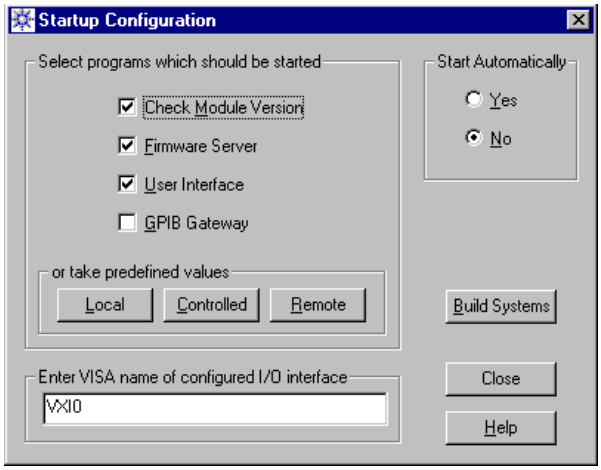


Figure 26 Agilent 81250 Configuration Window

2 Select the operating mode you wish to use from now on. Either use the checkboxes to enable/disable the software components or click one of the predefined mode buttons.

Choices are: *Local*, *Controlled*, or *Remote* (see also *"How to Start the Agilent 81250 Software" on page 62*).

When clicking the mode buttons, the checkboxes indicate which software components will be started.

3 Choose from the following options:

- *GPIB Gateway*: If you intend to start the system in controlled mode and operate it via LAN, you can disable the *GPIB Gateway*.
- *Start Automatically* can be useful for a controlled system where you don't have to care about display properties. The setting takes effect as soon as the DVT user logs in.
- *Build Systems* checks the available modules and creates new configuration files *dvtsys.txt* and *dvtits.txt*. The present files are saved as *dvtsys.bak* and *dvtits.bak*. If the mainframe contains more than one master clock module, additional instruments (DSRB, DSRC, ...) are automatically set up.

  For details please refer to the *Agilent 81250 Installation Guide*.

NOTE The *VISA interface name* refers to the standard installation and should not be changed.

4 Click OK.

5 Double-click the Agilent 81250 User Software icon to start the software in the newly selected mode.

How to Control the GPIB Gateway

If the GPIB gateway has not been disabled, it is automatically activated when the system is configured to be controlled by another computer and then started.

When the GPIB gateway is active, the GPIB to Agilent 81250 Gateway control panel can be displayed from the Windows task bar.

Figure 27 GPIB to Agilent 81250 Gateway Control Panel

The options are:

- *Settings*: Used to specify the termination character for received commands. Transmitted commands and responses are automatically terminated with LF ($0A_{Hex}$), according to IEEE 488.2.

  Choices are *none* or any ASCII character between 0 and 127 (decimal). The current termination character is displayed in hex format and alphabetical notation.

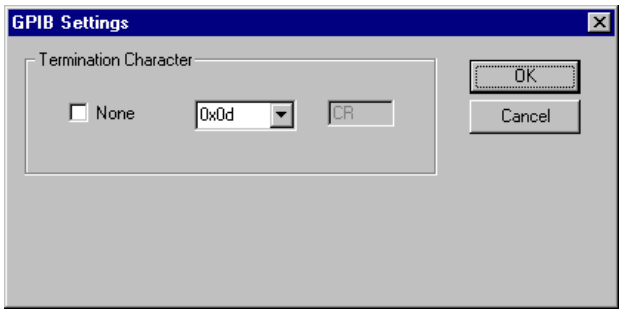


Figure 28 GPIB Settings

The desired character can be entered in decimal, hexadecimal, or octal format. Examples:

Table 5 Termination Characters

Character	Hex	Decimal	Octal
HT	0x09	9	011
LF	0x0a	10	012
FF	0x0c	12	014
CR	0x0d	13	015

These usual characters can also be chosen from the pull-down menu.

- *Monitor*: Used to monitor the command transfer in case of problems.

  Shows the commands passing through the receive and send buffers.

- *Close*: Terminates the GPIB to Agilent 81250 gateway.

  You will be asked whether you wish to terminate the Agilent 81250 server as well.

Overview of the Windows

There are windows for setting up a test and windows for displaying the test results.

Overview of Test Setup Windows

"Overview of Test Result Windows" on page 67

Overview of Test Setup Windows

The test setup windows are:

- Connection Editor—for reproducing (modeling) the properties of and physical connections to the device under test.
- Parameter Editor—for setting test parameters, such as frequencies, delays, voltages, etc.
- Measurement Configuration—for specifying the kind of test, such as bit error rate or capture and compare.
- Standard Mode Sequence Editor—for specifying one infinitely looped block of data containing the segments of generated and expected data for every data port.
- Detail Mode Sequence Editor—for specifying an individual sequence of blocks which contain the generated and expected data segments and loops. Also used for specifying events and reactions on events or triggers to be generated.
- Segment Editor—for creating the data segments to be generated or expected within the data blocks.
- Data/Sequence Editor—a combination of Sequence and Segment Editor.
- Channel Configuration Editor—for adding generator channels to produce a combined signal.
- Event Edit and Branch windows—for specifying events and actions upon events.
- Deskew Editor—for synchronizing the module connectors and compensating for progagation delays caused by cables or the DUT board.

NOTE Most of these windows can be opened by clicking a button of the tool bar, except:

- Channel Configuration Editor—a subfunction of the Connection Editor.
- The Sequence Editors—Standard Mode Sequence Editor, Detail Mode Sequence Editor, Data/Sequence Editor—all accessible from the *Go* menu.
- Parameter Editor and Deskew Editor—accessible from the *Go* menu.
- Event Edit and Branch windows—a subfunction of the Detail Mode Sequence Editor.

Overview of Test Result Windows

A set of windows is provided to present the test results in different views. To access these windows, open *Result Displays* in the *View* menu and select the desired item.

The available test result windows depend on the chosen type of measurement:

- Bit Error Rate—only available, if “Error Rate Measurement” has been selected. Shows the error rate and is updated every second.
- Compared Data—only available, if “Compare and Capture” or “Compare and Acquire around Error” has been selected. Shows captured data and highlights deviations from expected data.
- Captured Data—not available, if “Error Rate Measurement” has been selected. Shows captured data.
- Errored Data—only available, if “Compare and Capture” or “Compare and Acquire around Error” has been selected. Shows zeros (no error) and ones (errors).
- Waveform—not available, if “Error Rate Measurement” has been selected. Shows the captured waveform and indicates errors.

Compared Data, Captured Data, and Errored Data all use the Error State Display for presentation. You can switch between the three views within the window. Alternatively, you can open the window multiple times to inspect the contents concurrently.

Bit Error Rate Display, Error State Display, and Waveform Viewer are accessible by clicking buttons in the tool bar.

Operating the User Interface

The preferred instrument for operating the system is the mouse or the touchpad.

How to Use the Mouse or Touchpad

Some areas of the windows are “active” areas. They can be identified by the cursor changing its shape when placed over these areas.

If you press the right mouse button on an active area, a context menu pops up that lists the actions you can perform on the chosen item. If there is a default action defined for a context menu, this is indicated with bold text.

The default action is automatically executed if you double-click on an active area with the left mouse button.

NOTE If you have highlighted an item by clicking on it once with the left mouse button, you can also select the available actions from the menus of the menu bar on top of the window frame.

How to Navigate With the Keyboard

There are people who hate mice. They may use the keyboard instead:

- Tab and Shift+Tab move the cursor.
- Shift+F10 opens the context menu.
- To close the active window press Ctrl+F4.
- To switch between open windows press Ctrl+F6.
- To terminate the system run press Alt+F4.

If you type something into a data entry field, terminate your input with Enter. This causes the software to check the input and react.

How to Change Units and/or Vernier Steps

Some windows contain data entry fields with vernier buttons and units.

To change the default unit or vernier step size:

1 Click on the unit button to open the *Units and Step Size Adjust* window.

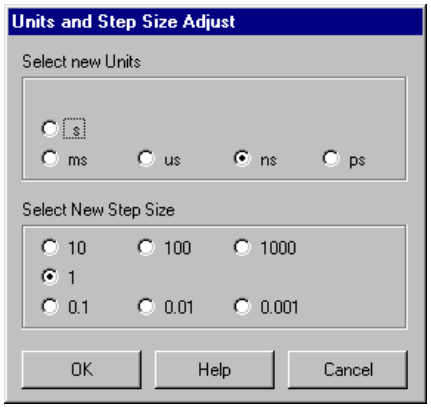


Figure 29 Units and Step Size Adjust Window

2 Change the unit and step size as desired.

3 Click OK.

NOTE You can also click with the right mouse button on the unit of a data entry field and change unit and step size from the context menu.

How to Use the Window Selection Box

This dialog box appears when you are opening the Parameter Editor or the Waveform Viewer from the respective menu item or icon in the main window. It appears also, if you open the Error State Display for a device that has more than one output port.

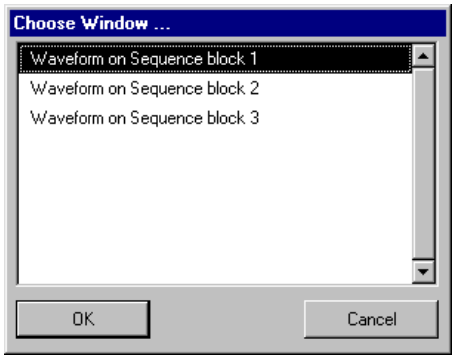


Figure 30 Window Selection Dialog Box Example

1 Click on an item in this dialog box to select this item to be presented in the respective editor/viewer.

2 Click OK.

The respective editor/viewer opens.

For information on the usage of the Parameter Editor, see *"How to Start the Parameter Editor for Global Parameters" on page 92* or *"How to Start the Parameter Editor for Ports/Channels" on page 116*.

For information on the Error State Display, see *"How to View Captured Test Results" on page 204*.

For information on the Waveform Viewer, see *"How to View Waveforms" on page 209*.

Items of the Main Menu

The main window of the Agilent 81250 Graphical User Interface provides easy access to and control of all features and functions of the system.

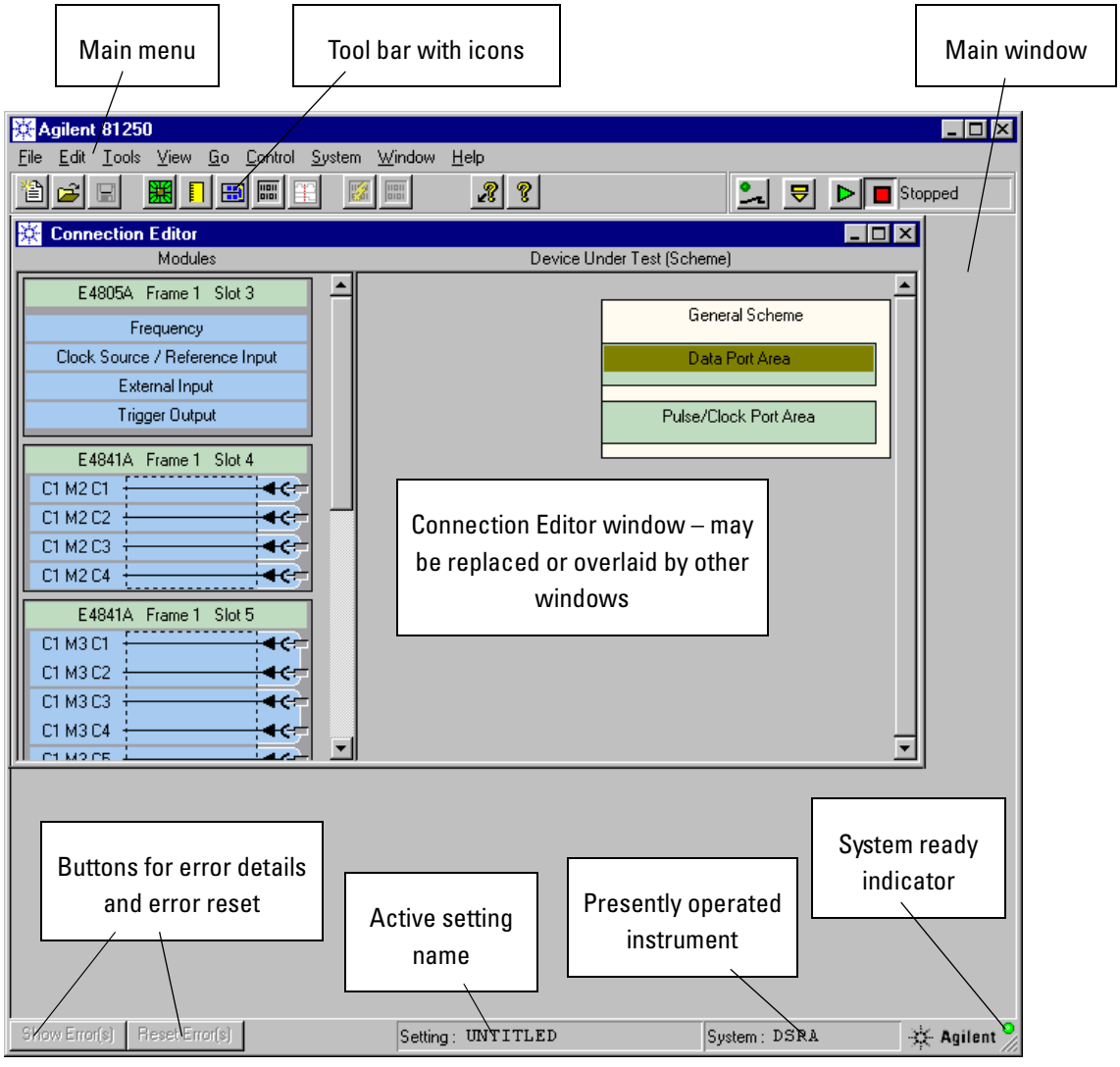


Figure 31 Agilent 81250 Main Window

Main menu The main menu is located at the top of the main window.

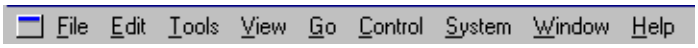


Figure 32 Agilent 81250 Main Menu

It provides access to the following menus:

"File Menu" on page 73

"Edit Menu" on page 78

"Tools Menu" on page 82

"View Menu" on page 83

"Go Menu" on page 85

"Control Menu" on page 86

"System Menu" on page 88

"Window Menu" on page 89

"Help Menu" on page 90

Tool Bar In the main window there is also a tool bar located at the top, which contains icon buttons providing shortcuts to windows and dialogs. They are explained along with the respective menus.

Figure 33 Tool Bar

NOTE Not all items of the menus and the tool bar are available at all times. Whether an item is available or not depends on the active window and the current situation.

The individual windows therefore provide context menus which can be opened by positioning the cursor to the area of interest and clicking the right mouse button.

Run Control Area In the upper right-hand corner of the main window you see the Run Control Area. Here you find a set of icon buttons and a status field showing the current test status.

Figure 34 Run Control Area

For an explanation of these buttons see *"Control Menu" on page 86*.

Status Line At the bottom of the main window you can see the status line.

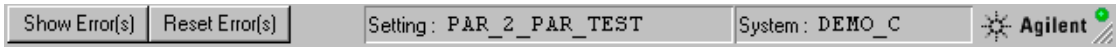


Figure 35 Status Line

Located in the status line are:

- The *Show Error(s)* and the *Reset Error(s)* buttons.

  If you enter invalid data into a text field of an editor window or a dialog box, the software reports an error by highlighting all erroneous fields and the *Show Error(s)* button.

  Clicking this button opens an error window that explains the error. Clicking the *Reset Error(s)* button resets the invalid entry to the last valid value.

- The name of the loaded test setting.
- The name of the presently operated instrument.

File Menu

The *File* menu is primarily used for storing data in files and retrieving data from files.

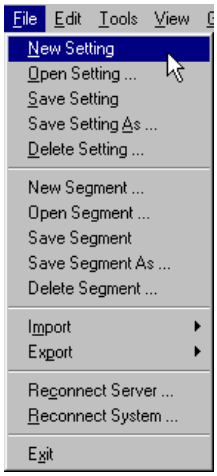


Figure 36 File Menu

New Setting

Closes the current setting and returns you to the default setting. The default setting is used for starting with a new device under test (DUT).

If the current setting has been changed and not saved, you will be asked whether you wish to save the current setting before the default setting is loaded. The current setting is indicated in the bottom line of the window frame.

Shortcut: New Setting icon in the tool bar.

Open Setting

Used to load one of the saved settings.

If the current setting has been changed and not saved, you will be asked whether you wish to save the current setting before the new setting is loaded. The current setting is indicated in the bottom line of the window frame.

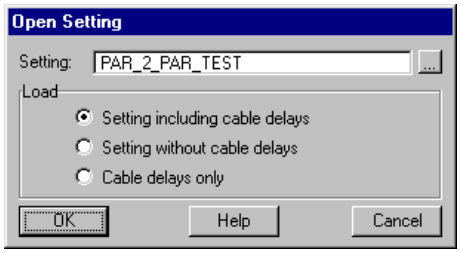


Figure 37 Open Setting Dialog

Default is the last opened setting. Others can be chosen from the list. The options for opening a setting are:

- Complete setting including cable delays
- Setting without cable delays (e.g. if connection cables to the DUT have been changed)
- Cable delays only (e.g. if the same cables are used for connecting to a new DUT)

Shortcut: Open Setting icon in the tool bar.

Save Setting

Saves the currently loaded setting on the disk, including all changes that have been made. Overwrites the previous version.

For saving or creating a new setting or keeping the original setting use *Save Setting As* instead.

Shortcut: Save Setting icon in the tool bar.

Save Setting As

Displays a list of the stored settings and enables you to overwrite one of these or to save the current setting under a new name on the disk.

Delete Setting

Displays a list of the stored settings and enables you to delete one or several of these.

New Segment

Enables you to create a new data segment. You need to select a pool (global or local) and enter a segment name. You can change width and length.

After that you enter the Segment Editor see *"Creating and Editing Segments" on page 175*.

The option *New Segment* can also be invoked from the Standard or Detail Mode Sequence Editor.

Open Segment

Displays a list of the stored segments. After choosing a segment, you enter the Segment Editor for investigating or changing the properties of the chosen segment.

Shortcut: Segment Editor icon in the tool bar.

Save Segment

Only available if the Segment Editor is active and the segment has been changed.

Saves the modified segment under its original name.

Shortcut: If the segment has been changed and you terminate the Segment Editor, you will be asked whether you wish to save your changes.

Save Segment As

Only available if the Segment Editor is active.

Displays a list of the stored segments. The list can be toggled to show the contents of the LocalSegments or GlobalSegments pool. Enables you to overwrite one of the existing segments or to save the current segment under a new name on the disk.

Delete Segment

Displays a list of the stored segments. The list can be toggled to show the contents of the LocalSegments or GlobalSegments pool. Enables you to delete one or several of the segments.

Import

Enables you to import settings or segments that have been exported to files.

Import Setting

Importing a setting replaces the current setting.

If the current setting has been changed and not saved, you will be asked whether you wish to save the current setting before the new setting is imported.

The file containing the setting can be found with the browser.

The imported setting can then be saved with the *Save Setting As* function.

Import Segments

Segments can be imported into the local or global segments pool.

The file containing the segments can be found with the browser.

The file can include more than one segment. You can therefore specify whether you wish to overwrite existing segments or not.

Export

Enables you to export settings or segments as files.

The destination path can be specified with the browser. Enter a new file name or select a file to overwrite.

Export Setting

When exporting a setting, the options are:

- Complete setting including cable delays
- Setting without cable delays (e.g. if connection cables to the DUT have been changed)
- Cable delays only (e.g. if the same cables are used for connecting to a new DUT)

The generated file is an ASCII file which can be investigated with a suitable text editor. A setting file contains all the firmware commands that establish that setting.

Export Segments

When exporting segments, the options are:

- all from GlobalSegments pool
- all from LocalSegments pool
- a single segment which can be chosen from the browser

The generated file is an ASCII file which can be investigated with a suitable text editor. Segment files are Vector Format text files.

Reconnect Server

You may wish to control an Agilent 81250 system from your desk via LAN or via the GPIB board built into your workstation and a GPIB cable. To achieve this:

1. Start the Agilent 81250 system in *Controlled* mode. This activates the Agilent 81250 server software and (if desired) the GPIB gateway software. The system now waits for LAN or GPIB commands.
2. Start the Agilent 81250 software on your workstation in *Remote* mode. This activates the Agilent 81250 graphical user interface.
3. Use *Reconnect Server* to connect the user interface with the LAN or GPIB board.

For system configurations please refer to the *Installation Guide*.

Reconnect System

Your Agilent 81200 Data Generator/Analyzer Platform may include more than one system. A system consists of at least one master clock generator module (additional ones may be added as slaves) and at least one data generator/analyzer module with generator and/or analyzer frontends.

See also *"System Components" on page 13*.

If your mainframe houses more than one master clock generator module, this function enables you to switch to another system.

Reconnect System displays a list of all available system configurations to choose from.

NOTE You can also run two independent systems in parallel by starting the user interface twice.

Exit

Closes the user interface and terminates the system run.

Shortcut: The Close button in the upper right-hand corner of the window frame or Alt+F4.

Edit Menu

The *Edit* menu is used for preparing a test. It contains the editing functions that can be used in the various test setup editors.

The appropriate menu items are enabled when the respective test setup editor window is active.

Figure 38 Edit Menu

Cut

Copies the highlighted item to the clipboard and removes it from the current window.

Shortcut: Ctrl+x

Copy

Copies the highlighted item to the clipboard.

Shortcut: Ctrl+c

Paste

Only available if data has been copied to the clipboard.

Inserts the clipboard contents at the specified location.

Shortcut: Ctrl+v

Paste before

An option of the Detail Mode and Data/Sequence Editors. Only available if a block has been copied to the clipboard.

Inserts the clipboard contents above the currently highlighted block.

Paste after

An option of the Detail Mode and Data/Sequence Editors. Only available if a block has been copied to the clipboard.

Inserts the clipboard contents below the currently highlighted block.

Insert

An option of the Connection Editor and the Segment Editor.

Inserts a new port, terminal, vector or trace at the specified location.

Insert before

An option of the Detail Mode and Data/Sequence Editors.

Inserts a new block above the currently highlighted block.

Insert after

An option of the Detail Mode and Data/Sequence Editors.

Inserts a new block below the currently highlighted block.

Delete

An option of the Connection Editor and the Segment Editor.

Deletes the highlighted object.

Move

An option of the Connection Editor.

Moves the highlighted terminal up or down.

Shortcut: Drag and drop.

Connect

An option of the Connection Editor.

Enables you to connect the highlighted terminal to a frontend connector.

Shortcut: Drag and drop.

Disconnect

An option of the Connection Editor.

Enables you to disconnect terminals from frontend connectors.

Shortcut: Drag the connection (i.e. the blue rectangle next to the terminal, labeled Cx My Cz) to the left somewhere into the modules area but not on a connector and release the mouse button.

Go to

An option of the Segment Editor and the Error State Display.

Enables you to jump to a specified vector address.

Trigger

An option of the Detail Mode and Data/Sequence Editors.

Used to enable or disable the generation of a trigger pulse associated with a block.

Events

An option of the Detail Mode and Data/Sequence Editors.

Enables you to specify events and define actions upon events (such as sequence branches).

Sync

An option of the Detail Mode Sequence Editor.

Enables you to activate, disable, or change the automatic sampling point adjustment of analyzer modules.

For detailed information see *"How to Specify Events and Reactions Upon Events" on page 164*.

Set Start

An option of the Sequence Editors.

Enables you to specify a start block of a sequence. When running the test, the start block will be the first to be executed. Any blocks above the start block are ignored when the test is executed. This option is useful to determine different entry points in a data sequence without the need of major modifications in the sequence content itself.

NOTE The start block is automatically named START. In order to avoid confusion, you should not assign that label manually to a block.

Rename

An option of the Connection Editor.

Enables you to rename the highlighted port or terminal of the DUT.

Properties

An option of several editors.

Shows the individual properties (parameters) of the highlighted object.

Tools Menu

The *Tools* menu contains additional options of the Segment Editor.

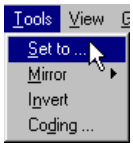


Figure 39 Tools Menu

Set to

With this option you can set a highlighted section in the Segment Editor to a specified value.

Mirror

With this option you can mirror a highlighted section in the Segment Editor either horizontally or vertically.

Invert

With this option you can invert a highlighted section in the Segment Editor—zeros to ones, ones to zeros.

Coding

With this option of the Segment Editor you can change the coding of the data segment. Selecting this option opens the Data Converter window.

View Menu

The options of the *View* menu allow you to change the appearance of data and waveform displays and to choose a suitable result display.

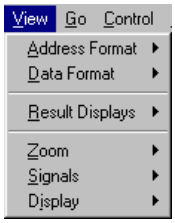


Figure 40 View Menu

Address Format

Enables you to change the displayed address format. Options are:

- Decimal
- Hexadecimal
- Octal

Data Format

Enables you to change the displayed data format. Options are:

- Binary
- Hexadecimal
- Octal

Additional options are available if the Segment Editor has been started from the Sequence Editor for editing or creating a segment within a block:

- Trace View: Shows trace numbers of the data segment
- Port View: Shows the data port to which the segment is assigned
- Terminal View: Shows the terminal names of the data port
- Connector View: Shows the identifications of the connected channels

Result Display

Enables you to choose a suitable display. Available options depend on the kind of test that has been specified in the Measurement Configuration window:

- Bit Error Rate—only available, if “Error Rate Measurement” has been selected. Opens the Bit Error Rate display window.

Shortcut: Bit Error Rate Display icon in the tool bar.

- Compared Data—only available, if “Compare and Capture” or “Compare and Acquire around Error” has been selected. Opens the Error State display window.

Shortcut: Error State Display icon in the tool bar.

- Captured Data—not available, if “Error Rate Measurement” has been selected. Opens the Error State display window.

Shortcut: Error State Display icon in the tool bar.

- Errored Data—only available, if “Compare and Capture” or “Compare and Acquire around Error” has been selected. Opens the Error State display window.

Shortcut: Error State Display icon in the tool bar.

- Waveform—not available, if “Error Rate Measurement” has been selected. Opens the Waveform Viewer.

Shortcut: Waveform Viewer icon in the tool bar.

NOTE Compared Data, Captured Data, and Errored Data all use the Error State Display for presentation. You can switch between the three views within the window. Alternatively, you can open the window multiple times to inspect the contents concurrently.

Zoom

An option of the Waveform Viewer.

Enables you to zoom into or out of the waveform display.

Signals

An option of the Waveform Viewer.

Enables you to change the waveform amplitudes in the display and to rearrange the signal order.

Display

An option of the Waveform Viewer.

Determines the display unit for the Waveform Viewer, either time (for DUT input ports) or number of samples (for output ports).

Go Menu

The options of the *Go* menu can be used to open the various test setup editors. They provide also access to functions which are not represented by tool bar buttons.

Figure 41 Go Menu

Connection Editor

Opens the Connection Editor. For details see *"Connecting the DUT" on page 105*.

Shortcut: Connection Editor icon in the tool bar.

Measurement Configuration

Opens the Measurement Configuration window. For details see *"Choosing the Kind of Measurement" on page 135*.

Shortcut: Measurement Configuration icon in the tool bar.

Parameter Editor

Opens the Parameter Editor. For details see *"How to Start the Parameter Editor for Global Parameters" on page 92* and *"How to Start the Parameter Editor for Ports/Channels" on page 116*.

Sequence Editor (Standard)

Opens the Standard Mode Sequence Editor, if applicable. If not, the Detail Mode Sequence Editor is started.

For details see *"Creating the Stream of Generated and Expected Data" on page 139*.

Shortcut: Sequence Editor icon in the tool bar.

Sequence Editor (Detail)

Opens the Detail Mode Sequence Editor.

This editor is always available. It is used for specifying the sequence of data blocks which reference the generated and expected data segments and specify loops (see *"The Detail Mode Sequence Editor" on page 152*).

Shortcut: Sequence Editor icon in the tool bar.

Data/Sequence Editor

Opens the Data/Sequence Editor (see *"How to Start the Data/Sequence Editor" on page 190*).

Waveform Viewer

Opens the Waveform Viewer (see *"How to View Waveforms" on page 209*).

Shortcut: Waveform Viewer icon in the tool bar.

Deskew Editor

Opens the Deskew Editor. For details see *"How to Compensate for Internal and External Delays" on page 216*.

Command Line

Opens the Command Line input window. For details see *"How to Execute Firmware Commands" on page 225*.

Control Menu

The items in the *Control* menu are used to control the actual test run on the DUT.

Figure 42 Control Menu

All control actions of this menu can be performed by clicking the respective buttons in the upper right-hand corner of the main window. The current status of the test is also displayed at the right-hand side of these buttons.

Run

Starts a test.

Shortcut: Run button in the tool bar.

Stop

Stops a running test.

Shortcut: Stop button in the tool bar.

Prepare Run

Downloads the test sequence to the modules. This is required for immediate reaction on a start trigger.

Shortcut: Prepare button of the tool bar.

Connectors On/Off

Switches relays inside the frontends:

- *Connectors Off* opens all input and output relays. This isolates the DUT electrically from the system.
- *Connectors On* connects all module connectors with activated outputs or inputs electrically with the cables. Connector outputs or inputs can be activated or deactivated with the Parameter Editor.

Shortcut: Connectors On/Off button in the tool bar.

System Menu

The *System* menu has functions for testing the system's integrity. These tests can be performed at any time, as long as no test is running.

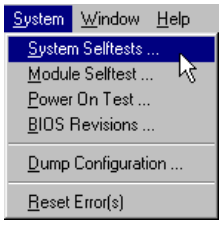


Figure 43 System Menu

Note, that the optional diagnostics software package provides additional tests which in case of problems can identify defective field replaceable units.

System Selftests

Provides a window, from which the complete selftest or subsets can be started. Ensures that all modules respond. Returns the current firmware revisions of the modules and the identification numbers of installed frontends.

Module Selftest

Enables you to check all or single modules. Checks the frontends built into the modules. For technical reasons, this test may take a minute.

Power On Test

Is automatically performed at power on. Checks all modules.

BIOS Revisions

Returns the current firmware revisions of the modules.

Dump Configuration

Enables you to write the current system configuration to a file. You have to specify the path and file name.

The file will receive a contiguous string of ASCII characters, showing the identifications of all modules and frontends.

Reset Error(s)

This menu item corresponds to the *Reset Error(s)* button in the lower left-hand corner of the main window. If invalid data was entered in any of the editor windows (e.g. the Parameter Editor), the software reports this error by highlighting all erroneous fields and the *Show Error(s)* button.

Reset Error(s) then resets the invalid input to the last correct value.

Window Menu

The *Window* menu contains standard functions provided by Microsoft Windows. You can use these items to rearrange the open windows or to switch between them.

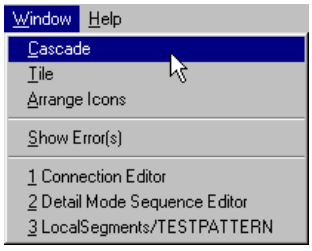


Figure 44 Window Menu

Shortcut for switching between windows: Ctrl+F6

Cascade

Superimposes all open windows. You can click any window title to bring that window to the front.

Tile

Arranges all open windows within the window frame.

Show Error(s)

This menu item corresponds to the *Show Error(s)* button in the lower left-hand corner of the main window. If invalid data was entered in any of the editor windows (e.g. Parameter Editor), the software reports this error by highlighting all erroneous fields and the *Show Error(s)* button.

Show Error(s) then displays an error window describing the error in detail.

Help Menu

The *Help* menu is supposed to be self-explanatory. You can start with the table of contents or search from the alphabetical index.

Figure 45 Help Menu

The options of the *Help* menu are:

- Contents.

  This opens the Help Desk window presenting an introduction to the system.

- Index.

  This opens the Help Desk window presenting an introduction to the system.

- Help on Window.

  This opens the Help Desk window presenting information on the currently active window.

- Help on Item.

  This option corresponds to the Help on Item button in the tool bar. When selected, the cursor changes its shape.

  Place the cursor on the item of interest an click the left mouse button. The Help Desk window presents information on the selected item.

- Help on Help.

  This opens the Help Desk window presenting an explanation how to use help.

- About.

  This opens a window telling the software version and contact and copyright information.

TIP For help on a specific item place the cursor on this item and press F1. The Help Desk window then presents information on this item.

Setting Global System Parameters

Global system parameters refer to the master clock module. If slaves are connected, they will follow the master.

The clock module has

- a built-in 10 MHz reference,
- a PLL-controlled oscillator,
- pulse delay circuits,
- a frequency multiplier/divider,
- two input connectors named CLOCK / REF INPUT and EXT INPUT,
- one output connector named TRIGGER OUTPUT.

The Parameter Editor is used for setting appropriate parameters.

This chapter provides instructions on how to set up the clock module:

"How to Start the Parameter Editor for Global Parameters" on page 92

"How to Set the Clock Frequency" on page 93

"How to Choose the Clock Source" on page 100

"How to Set the Characteristics of the External Input" on page 101

"How to Set the Characteristics of the Trigger Output" on page 103

How to Start the Parameter Editor for Global Parameters

There are global and channel-related system parameters. Both are set with the Parameter Editor. To support all kinds of parameters for the central clock module, data generator/analyzer modules, channels, ports, terminals and so on, the Parameter Editor has several modes of operation.

The global system parameters comprise the setup of:

- System, port, and channel frequencies,
- Clock source,
- External input,
- Trigger output.

The simplest way to access these setups is from the Connection Editor:

1 Double-click the corresponding fields in the *Modules* section.

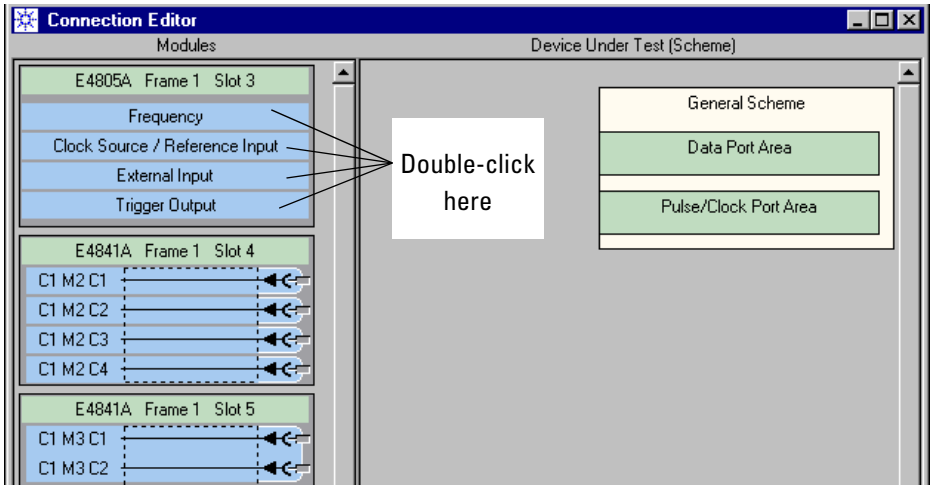


Figure 46 Accessing Global Parameters

If the Connection Editor is not displayed:

1 Choose *Parameter Editor* from the *Go* menu.

You get a list of all the items that have been configured and can have parameters. As long as you have not connected any DUT terminals with the Connection Editor, the list starts with the clock module.

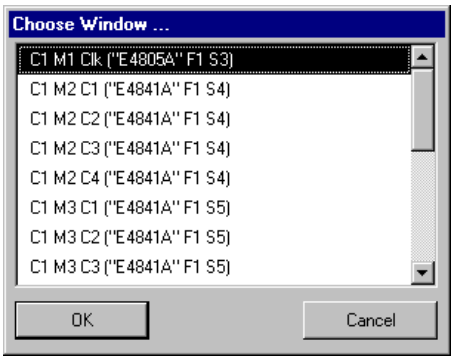


Figure 47 Parameter Editor Selection Window

2 Select the master clock module, identified as "C1 M1 Clk" and click *OK*.

The Frequency setup window appears.

NOTE Once the Parameter Editor has been started, it provides at the top of the window a browser labeled *Resource* and up/down arrows that enable you to switch to another item.

How to Set the Clock Frequency

You have to set a general system clock frequency. Then you may set individual clock frequencies for:

- Unconnected instrument channels
- Connected DUT data ports
- Connected pulse port terminals

Individual clock frequencies can be generated by multiplying or dividing the system clock frequency by factors which are multiples of 2. For details see:

"How to Set the General System Frequency" on page 94

"How to Use Multiple Frequencies" on page 96

How to Set the General System Frequency

1 Open the Parameter Editor for the clock module (see *"How to Start the Parameter Editor for Global Parameters" on page 92*). Choose the *Frequency* window.

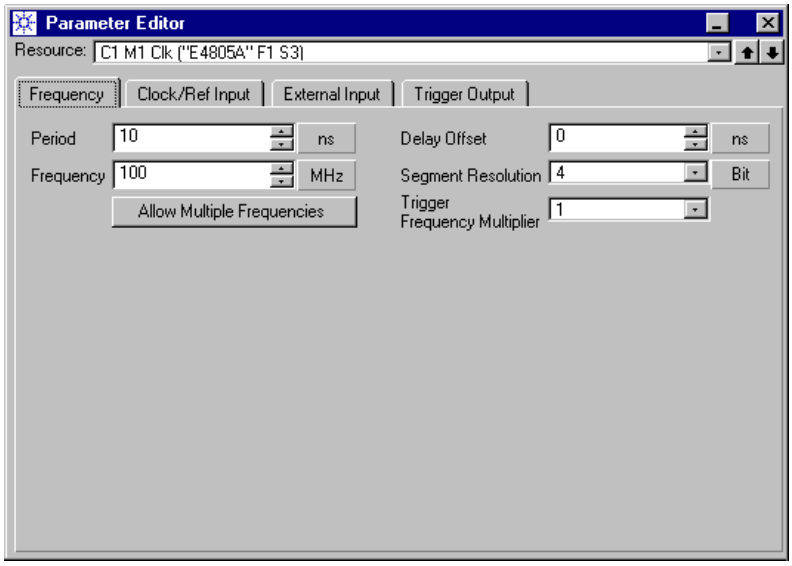


Figure 48 Setting the General System Frequency

You can specify the desired period or the desired frequency. Both are equivalent.

2 Accept or change the units.

The Parameter Editor displays default units (ns/MHz) and has default vernier steps. Both can be changed (see *"How to Change Units and/or Vernier Steps" on page 68*).

3 Set the desired general *Frequency* or *Period*.

If you have typed a number, terminate your input with the Enter or Return key. This updates the other field (Period or Frequency).

4 If desired, set a global *Delay Offset*.

A global delay offset enables you to specify negative time offsets for individual signals. Such signals then appear in advance of the regular clock pulse.

5 Check the *Segment Resolution*.

The segment resolution has an impact on the available memory resources for generating or capturing data. The default setting of the

segment resolution is 4 bits for E4832A modules and 16 bits for E4861A modules.

The segment resolution is coupled with a certain Frequency Multiplier (FM) factor. The available multiplication factors that can be used for multiplying the system clock are called Frequency Multiplier Range (FMR).

Table 6 Clock Rates, Segment Resolution, and Memory Depth for E4832A Modules

System Clock Frequency Mbit/s	Segment Resolution bits	Memory Depth bits	Frequency Multiplier Range
20.834 – 41.666	1	131,008	1, 2, 4, 8, 16
41.667 – 83.333	2	262,016	1/2, 1, 2, 4, 8
83.334 – 166.666	4	524,032	1/4, 1/2, 1, 2, 4
166.667 – 333.333	8	1,048,064	1/8, 1/4, 1/2, 1, 2
333.334 – 666.667	16	2,097,152	1/16, 1/8, 1/4, 1/2, 1

E4841A modules have only half the memory depth (64 Kbit to 1 Mbit).

Table 7 Clock Rates, Segment Resolution, and Memory Depth for E4861A Modules

System Clock Frequency Mbit/s	Segment Resolution bits	Memory Depth bits	Frequency Multiplier Range
333.334 – 666.666	16	2,097,152	1, 2, 4
666.667 – 1,333.333	32	4,294,304	1/2, 1, 2
1,333.334 – 2,666.667	64	8,388,608	1/4, 1/2, 1

NOTE Depending on the desired clock rate it is possible to choose a lower or higher segment resolution as shown in the tables above. If this is done for the analyzer frontends of an E4832A module, then the phase vernier and the functions for automatic sampling point adjustment are not available.

For details please refer also to *"FMR and Segment Resolution" on page 31*.

NOTE Not all frequencies and hence frequency multiplier factors are supported by all modules and frontends. The system will report an error, if the chosen segment resolution or frequency multiplier factor does not fit to all the components. In this case you may wish to change the setup individually for some channels and use multiple frequencies.

6 Check the *Trigger Frequency Multiplier*.

The trigger FM refers to the TRIGGER OUTPUT of the clock module. The TRIGGER OUTPUT can be used to generate a continuous clock signal or single pulses.

By default, the trigger clock frequency equals the master clock frequency. By setting the *Trigger Frequency Multiplier* to a factor other than 1, you can change the trigger clock frequency. The factor takes effect if the TRIGGER OUTPUT is in *Clock Generator* mode.

See also *"How to Set the Characteristics of the Trigger Output" on page 103*.

How to Use Multiple Frequencies

Multiple frequencies can be required by the DUT, if some ports or pins are operated at different clock rates.

Multiple frequencies are also required if your instrument is equipped with frontends which cannot be operated under one and the same frequency setup.

Additional clock frequencies can be generated by multiplying or dividing the system clock frequency by factors which are multiples of 2.

Setting Multiple Frequencies

1 Open the Parameter Editor for the clock module (see *"How to Start the Parameter Editor for Global Parameters" on page 92*). Choose the *Frequency* window.

2 Click *Allow Multiple Frequencies*.

As long as you have not connected any DUT terminals to frontends with the Connection Editor, the list shows all the connectors:

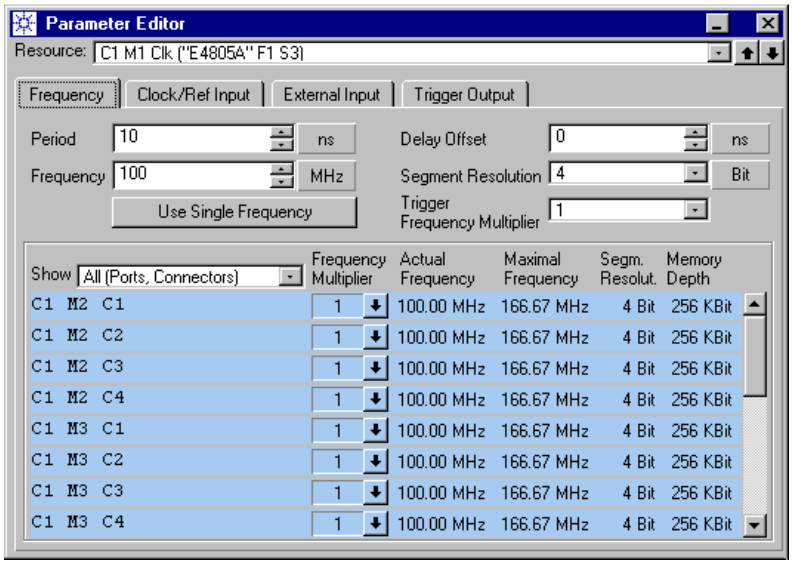


Figure 49 Setting Multiple Frequencies for Unconnected Channels

NOTE This list appears automatically if your system is composed of modules or frontends which require a multiple frequency setup.

If the DUT terminals have already been connected to frontends with the Connection Editor, the list shows all the ports, pulse port terminals, and unconnected channels:

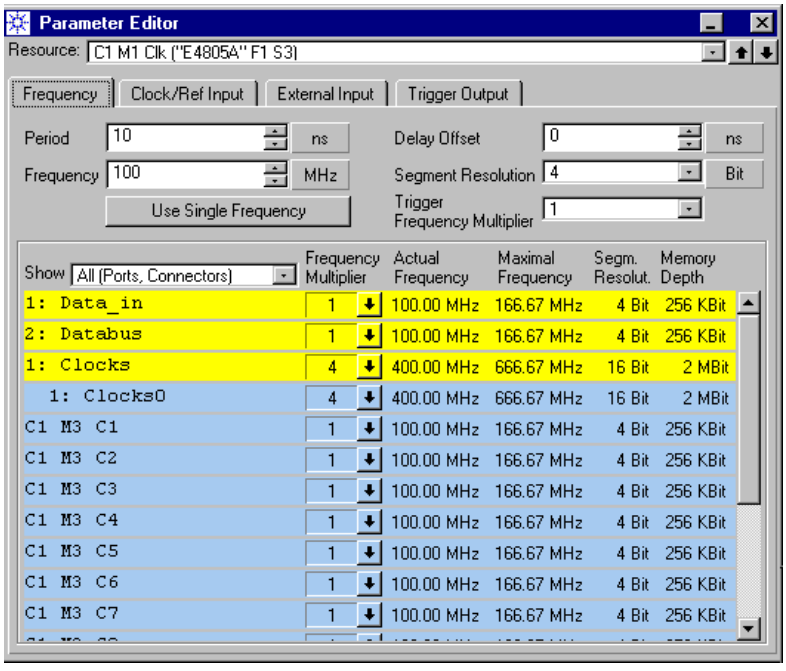


Figure 50 Setting Multiple Frequencies for Connected and Unconnected Channels

Data port terminals are not displayed, because all terminals of a data port must use one and the same frequency. They have to be set up altogether. The terminals of a pulse port, however, can operate at different frequencies.

The *Show* menu allows to restrict the list to certain items.

3 To change the frequency of a port or channel, click the corresponding down-arrow in the *Frequency Multiplier* column. You get a list of the available FM factors:

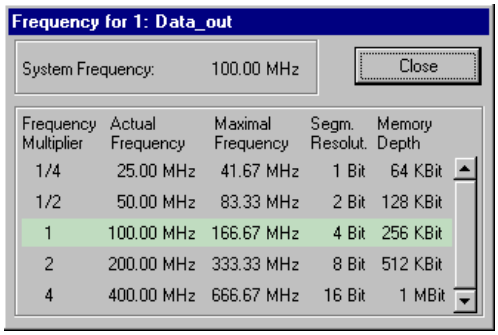


Figure 51 Individual Frequency Selection

4 Choose from the list and click *Close*.

In case of a problem please refer to the tables *"Clock Rates, Segment Resolution, and Memory Depth for E4832A Modules" on page 95* and *"Clock Rates, Segment Resolution, and Memory Depth for E4861A Modules" on page 95*.

Returning to one Single Frequency

If you wish to return to one system frequency for all channels, click *Use Single Frequency* and consider the following warning:

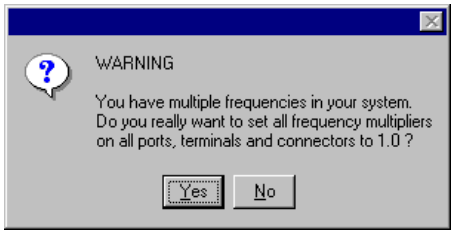


Figure 52 Return to Single Frequency Setup Warning

All frequency multipliers will be reset to one. This may conflict with your system configuration.

If an error occurs:

1 Click *Show Error(s)* and study the error messages.

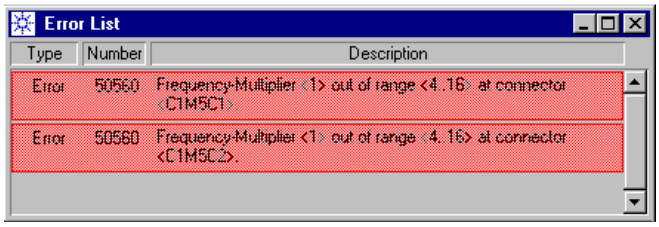


Figure 53 Frequency Multiplier Error Examples

2 Locate the connectors that are causing the problem (in this example C1 M5 C1 and C1 M5 C2):

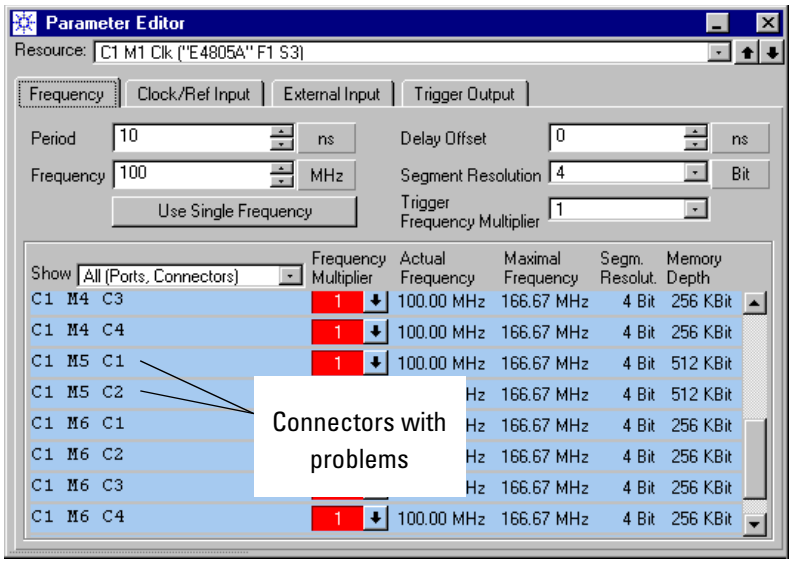


Figure 54 Single Frequency Setup Denied

3 Click the corresponding down-arrow in the *Frequency Multiplier* column. You get a list of the available FM factors:

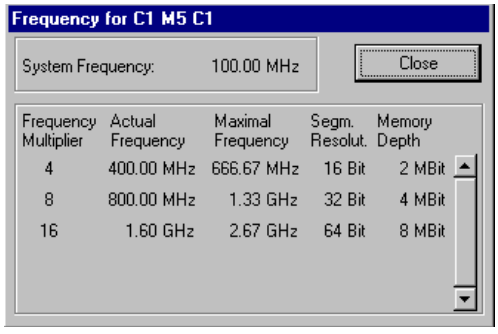


Figure 55 Frequency Multiplier Errors

4 Choose valid FM factors for both connectors, and the error will disappear.

How to Choose the Clock Source

If you wish to use an external clock, this signal has to be connected to the CLOCK / REF INPUT of the central clock module. The external clock can be used as a reference (an input to the built-in PLL-controlled oscillator) or directly.

1 Open the Parameter Editor for the clock module (see *"How to Start the Parameter Editor for Global Parameters" on page 92*). Click the *Clock/Ref Input* tab.

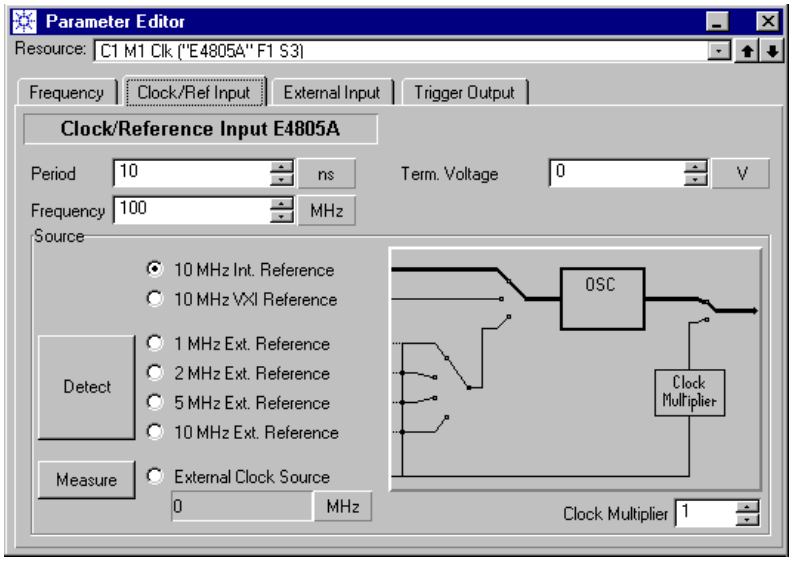


Figure 56 Clock /Reference Window

2 Enable the type of clock.

External reference clock

External reference clocks can have a frequency of 1, 2, 5, or 10 MHz. To use an external clock as a reference to the PLL-oscillator:

1 Connect the clock signal to the CLOCK / REF INPUT .

2 Click the *Detect* button.

The clock is automatically identified and connected.

External source clock

The frequency of the external source clock must exceed 1.302083 MHz. To use an external clock as a direct source:

1 Connect the clock signal to the CLOCK / REF INPUT .

2 Click the *Measure* button.

The clock frequency is measured and the clock is automatically connected.

Once you have connected an external source clock, you can use the *Clock Multiplier* which is controlled from the setting in the lower right-hand corner of the window. The clock multiplier enables you to multiply the applied clock frequency by any integer factor between 1 and 256.

How to Set the Characteristics of the External Input

The EXT INPUT connector of the master clock module can be used to start and stop the system.

1 Open the Parameter Editor for the clock module (see *"How to Start the Parameter Editor for Global Parameters" on page 92*). Choose the *External Input* window.

The External Input window shows the alternatives:

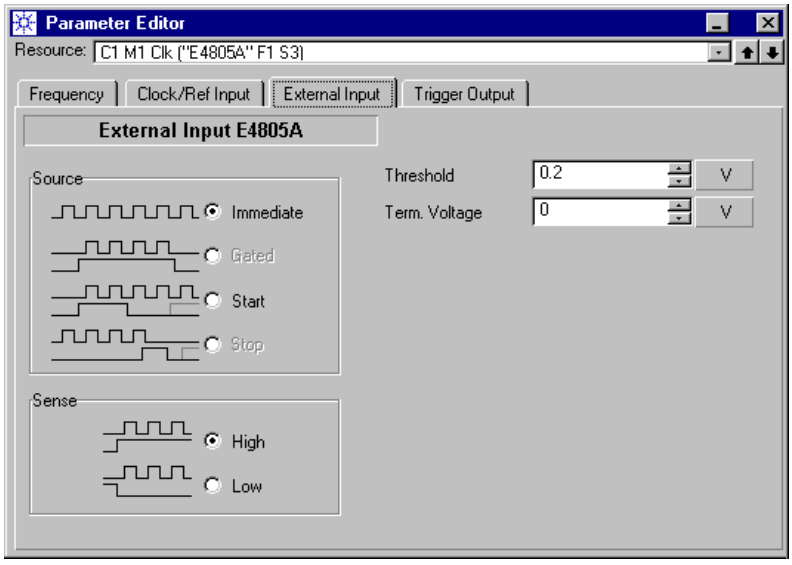


Figure 57 External Input Configuration Window

2 Choose the desired start/stop mode. Options are:

– *Immediate*:

The timing system is not influenced by the EXTERNAL INPUT. It is started either manually by the user with a mouse click or by a software command.

– *Gated*:

The timing system is armed by pressing the *Run* icon. The user interface displays HALTED as long as the system is not started. The timing system is started and stopped according to the chosen *Sense* polarity and *Threshold*.

The system is also stopped, if the sequence terminates. The user interface displays RUNNING, HALTED, or STOPPED. See also *"Trigger-Controlled Stop" on page 34*.

NOTE Gated mode is not supported if the system contains an E4861A "giga" module.

– *Start*:

The timing system is armed by pressing the *Run* icon. The user interface displays HALTED as long as the system is not yet started. The timing system is started with the first edge of the chosen *Sense* polarity that exceeds the *Threshold*. The system is stopped, if the *Stop* icon is pressed or the sequence terminates.

– *Stop*:

The timing system is started by pressing the *Run* icon. The timing system is halted by the first edge of the chosen *Sense* polarity that exceeds the *Threshold*. It is stopped, if the *Stop* icon is pressed or the sequence terminates. See also *"Trigger-Controlled Stop" on page 34*.

NOTE Stop mode is not supported if the system contains an E4861A "giga" module.

3 Specify the trigger *Sense* polarity.

4 Enter an appropriate *threshold* voltage for the external input.

5 Enter an appropriate *termination voltage*, if required.

How to Set the Characteristics of the Trigger Output

The TRIGGER OUTPUT connector of the central clock module can be used to inform other instruments upon an event. This event can be:

- Start of a data block within the overall test sequence
- An event detected by the system (see *"Event Handling Principles" on page 52*).

To set the trigger characteristics:

1 Open the Parameter Editor for the clock module (see *"How to Start the Parameter Editor for Global Parameters" on page 92*). Choose the *Trigger Output* window.

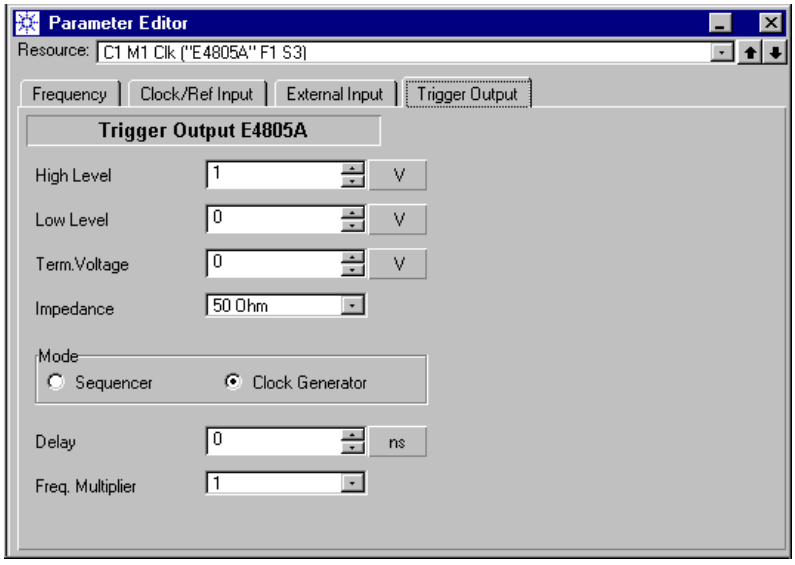


Figure 58 Trigger Output Configuration Window

2 Set the voltages and output impedance.

3 Select the Mode.

- *Clock Generator*: generates a clock signal derived from the system clock.
- *Sequencer*: Generates a transition from high to low or vice versa.

4 Set the *delay*, if desired. The delay refers to the system clock.

5 *Frequency Multiplier* is only available in Clock Generator mode. You can choose one of the factors from the list. The list shows the FMR (see *"How to Set the Clock Frequency" on page 93*).

Connecting the DUT

To connect the device under test (DUT) with the Agilent 81250 system, you need to:

- Create signal ports for the DUT.

  There are data input ports, data output ports, and pulse ports.

- Specify the port characteristics.

  You can add terminals to ports. You can also remove terminals from a port, rename the port or delete it.

NOTE Terminals (individual pins) can only be defined within a port.

- Change terminal characteristics.

  You can connect the terminals with suitable module connectors. You can also move a terminal, rename it, or disconnect it.

All this can be done from the Connection Editor.

After at least one terminal has been connected, the Connection Editor is also used to access the Parameter Editor for setting port and terminal specific properties.

NOTE Of course, physical connections are required as well. But you need only take care that the physical connections match the setup shown by the Connection Editor.

This chapter explains how to use the Connection Editor:

"How to Start the Connection Editor" on page 106

"How to Create a Port" on page 107

"How to Change the Characteristics of a Port" on page 109

"How to Change the Characteristics of a Terminal" on page 110

How to Start the Connection Editor

The Connection Editor is the first window that comes up automatically after starting the user interface.

To start the Connection Editor if you have closed its window:

1 Click the Connection Editor icon in the tool bar.

Alternatively, you can also start the Connection Editor from the *Go* menu.

Contents of the Connection Editor Window

At the left-hand side is the *Modules* section. There the Connection Editor shows an image of the instrument, including all its modules, channels, and connectors. The following figure shows an example.

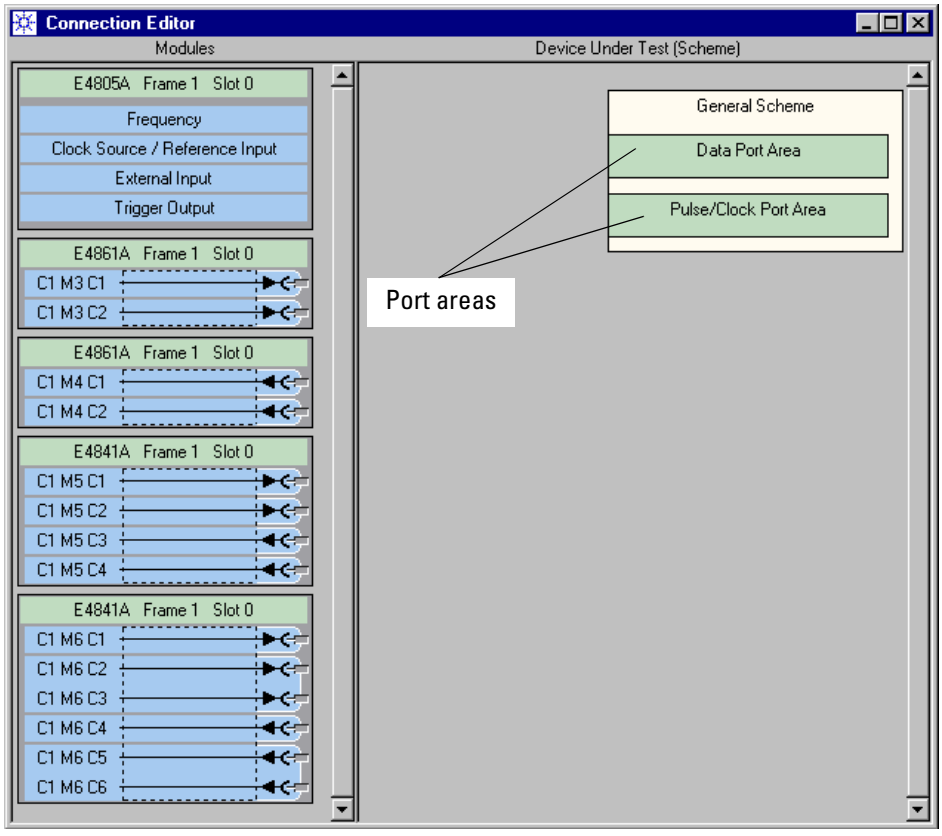


Figure 59 Connection Editor Window

The identification of the channels is:

Cx–My–Cz (ClockgroupNumber–ModuleNumber–ConnectorNumber), as for instance C1 M3 C2.

The connectors are also marked with arrows. So you know at a glance whether a connector belongs to an output channel (data generator) or an input channel (data analyzer).

At the right-hand side there is a template for setting up an image of the DUT. You can add:

- Data input ports
- Data output ports
- Pulse ports
- Port terminals representing the individual pins of the port

How to Create a Port

If you start from scratch, the DUT area of the Connection Editor shows only an empty template. You have to define ports (usually i/o buses, but also ports for clocks, latches, strobes, etc.) and terminals (DUT pins).

DUT data ports receive or return data streams. DUT pulse ports receive clocks and pulses from the Agilent 81250 system.

To create a port:

1 Click on the port area (either Data Port Area or Pulse Port Area) with the right mouse button to open the context menu.

2 For data ports you can choose the type of the port. Select either *Insert Input Port* or *Insert Output Port*. Pulse ports are always DUT input ports.

The Insert Data/Pulse Port dialog box is displayed.

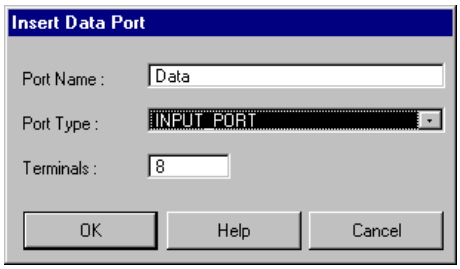


Figure 60 Insert Data Port Dialog

If you have chosen to insert a data port, you can still switch between input and output port.

NOTE The terminals of an input port can only be connected to data generator channels, the terminals of an output port only to analyzer channels.

3 Enter a suitable port name and the number of terminals.

4 Click OK.

The Connection Editor shows the new port. The terminals get the port name and are automatically numbered.

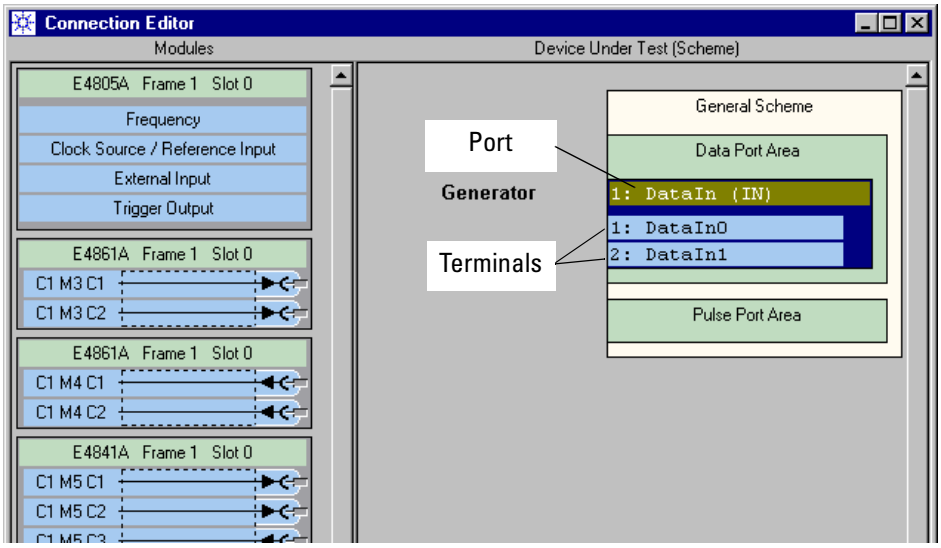


Figure 61 Display of Ports and Terminals

The terminals can be renamed or rearranged if required. See *"How to Rename a Terminal" on page 111* and *"How to Move a Terminal" on page 111*.

How to Change the Characteristics of a Port

Although all actions can also be selected from the menus in the main window, it is a lot more convenient to employ the individual context menu of each item.

To open the context menu of a certain port, place the cursor on the port's header and click the right mouse button.

The available options are to:

- Delete the port
- Rename the port
- Add terminals to the port

After one or several of the port terminals have been connected to instrument connectors, you have the additional options to:

- Disconnect all terminals
- Set or change the port properties

How to Delete a Port

To delete a port:

1 Open the port's context menu by clicking the port header with the right mouse button.

2 Choose *Delete*.

3 Confirm.

How to Rename a Port

To rename a port:

1 Open the port's context menu by clicking the port header with the right mouse button.

2 Choose *Rename*.

3 Specify a new name for the port.

4 Confirm with Enter or by clicking *OK*.

How to Add a Terminal to a Port

To add a terminal to a port:

1 Open the port's context menu by clicking the port header with the right mouse button.

2 Choose *Insert Terminal*.

The window shows the default terminal name (port name plus number) and suggests the position at the end of the terminal list.

3 Specify the desired terminal name.

4 Accept the suggested position or enter one of the occupied positions to insert the new terminal there.

5 Confirm with Enter or by clicking *OK*.

NOTE The terminal's name and position can be changed at any time. Although permitted, you should not use one name for several terminals of a port to avoid confusion.

How to Change the Characteristics of a Terminal

Although all actions can also be selected from the menus in the main window, it is a lot more convenient to employ the individual context menu of each item.

To open the context menu of a terminal, place the cursor on it and click the right mouse button.

The available options are to:

- Rename the terminal
- Delete the terminal
- Connect the terminal to the instrument
- Move the terminal

If a terminal has been connected to an instrument connector, you can also

- Disconnect the terminal
- Set or change the channel properties

How to Rename a Terminal

To rename a terminal:

1. Open the context menu by clicking the terminal with the right mouse button.
2. Choose *Rename*.
3. Enter a new name for the terminal.
4. Confirm with Enter or by clicking *OK*.

How to Delete a Terminal

To delete a terminal:

1. Open the terminal's context menu by clicking it with the right mouse button.
2. Choose *Delete*.
3. Confirm.

How to Move a Terminal

To move a terminal within its port:

1. Open the terminal's context menu by clicking it with the right mouse button.
2. Choose *Move*.
3. Overwrite the present location by the desired location.

   If the terminal is placed at the last position of the port, you can only move it upward.
4. Confirm with Enter or by clicking *OK*.

Shortcut: Click the terminal with the left mouse button and drag it to the desired position.

How to Connect a Terminal

There are several ways to connect the terminals of the DUT to the connectors of the instrument frontends:

- Connecting a Terminal With the Keyboard
- Connecting a Terminal With the Mouse
- Inserting and Connecting a Single Terminal With the Mouse
- Inserting and Connecting Multiple Terminals With the Mouse

NOTE The terminals of a data input port or a pulse port can only be connected to generator channels. The terminals of a data output port can only be connected to analyzer channels.

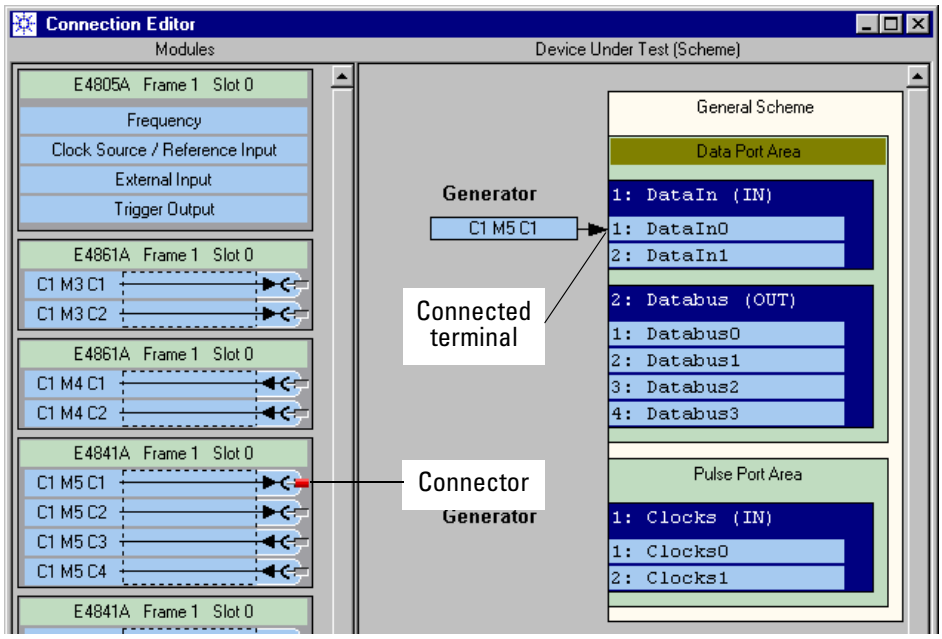


Figure 62 Display of Terminal Connections

Connecting a Terminal With the Keyboard

To connect a terminal with the keyboard:

1. Click the right mouse button to open the terminal's context menu.
2. Choose *Connect*.
3. Enter the identifier of the desired channel: ClockgroupNumber, ModuleNumber, and ConnectorNumber.
4. Confirm with Enter or by clicking *OK*.

Connecting a Terminal With the Mouse

To connect a terminal with the mouse:

1 Position the cursor on the terminal.

2 Press the left mouse button. Hold the mouse button depressed and drag the cursor (which changes its shape) onto an empty connector of appropriate type.

3 Release the mouse button.

Inserting and Connecting a Single Terminal With the Mouse

You can create a new one-terminal port or insert a new terminal into an existing port and connect it in one go:

1 Position the cursor on the frontend connector in the *Modules* section.

2 Press the left mouse button. Hold the mouse button depressed and drag the cursor (which changes its shape) onto the DUT.

- If you place the cursor on the green header of the *Data Port Area*, a new port with a single terminal will be created.
- If you place the cursor on a terminal, this terminal will be connected to the module channel.
- If you place the cursor on the upper border line of a terminal (the border turns red), a new terminal will be inserted at this point.

3 Release the mouse button to establish the connection.

Inserting and Connecting Multiple Terminals With the Mouse

You can create new ports with multiple terminals or add multiple terminals to an existing port and connect them in one go:

1 In the *Modules* section place the cursor on a module label to connect all its connectors at once.

2 Press the left mouse button. Hold the mouse button depressed and drag the cursor (which changes its shape) onto the DUT.

- If you place the cursor on the green header of the *Data Port Area*, one or two new port(s)—an input or output port or both—will be created with all (matching) channel connectors being connected to terminals of this new port.

– If you place the cursor on an existing terminal, this terminal and the following ones will be connected to the module connectors. No new terminals will be created.

– If you place the cursor on the upper border line of an existing terminal (the border turns red), new terminals will be inserted at this point. They will be connected to all matching connectors of the selected module.

3 Release the mouse button to establish the connections.

How to Disconnect a Terminal

To disconnect a terminal:

1 Open the terminal's context menu by clicking the terminal with the right mouse button.

2 Choose *Disconnect*.

3 Confirm.

To disconnect all terminals of a port, choose *Disconnect* from the port's context menu.

Setting Up Ports and Channels

After one or several terminals have been connected to the Agilent 81250 system, you can set up and modify port and channel properties, such as delays, signal formats, voltages, impedances, and so on.

Setting port parameters is an easy way to use the same settings for all connected terminals of that port. Before setting port parameters, **all** the terminals of that port should be connected with the instrument, because later connected terminals do not automatically get the same parameter settings.

Alternatively, you can set individual parameters for individual channels.

NOTE Individual channel parameters override port parameters.

To set and change port and channel parameters the Parameter Editor is used.

NOTE In addition to setting parameters, there is another function that determines channel properties: Two or four data generator channels can be added to produce a combined signal. This is done with the Channel Configuration Editor which can be invoked from the Connection Editor.

This chapter explains how to set up ports and channels:

"How to Start the Parameter Editor for Ports/Channels" on page 116

"How to Set Up a DUT Input Port or Generator Channel" on page 117

"How to Set Up a DUT Output Port or Analyzer Channel" on page 125

"How to Combine Generator Channels" on page 129

How to Start the Parameter Editor for Ports/Channels

The Parameter Editor has several modes of operation. It distinguishes between global system parameters and parameters for generator and analyzer channels. The setup of global parameters is described in *"Setting Global System Parameters" on page 91*.

To open the Parameter Editor you have the following options:

- Use the left mouse button and double-click a port header, a connected terminal of a port, or a channel identifier (Cx–My–Cz).

  The Parameter Editor opens and displays the properties of the selected item.

- Use the right mouse button and click on a port header, a terminal, or a channel identifier (Cx–My–Cz). Then select *Properties* from the context menu.

  The Parameter Editor opens and displays the properties of the selected item.

- Choose *Parameter Editor* from the *Go* menu.

  You get a list of all the items that have been configured so far and can have parameters. Such items are:

  – The clock module

  – Data and Pulse Ports

  – Connected port terminals (= connected channels)

  – Unconnected instrument channels

  Select an item from the list to open the Parameter Editor in the respective view.

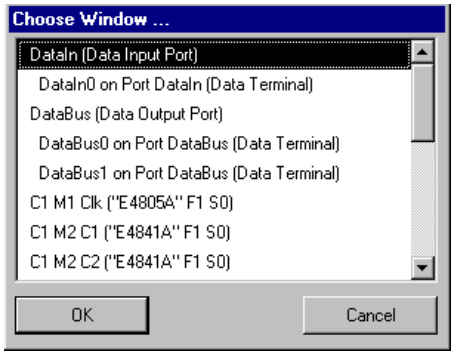


Figure 63 Parameter Editor Selection Window

The display mode of the Parameter Editor depends on the type of port or channel that has been selected. However, a selection list and arrow buttons in the editor window enable you to switch to the other items.

NOTE Port parameters refer to all channels connected to a port. Individual channel parameters override port parameters.

How to Set Up a DUT Input Port or Generator Channel

To set the parameters for a DUT port or terminal that receives generated signals or a channel that generates signals:

1 Open the Parameter Editor for the port or channel (see *"How to Start the Parameter Editor for Ports/Channels" on page 116*).

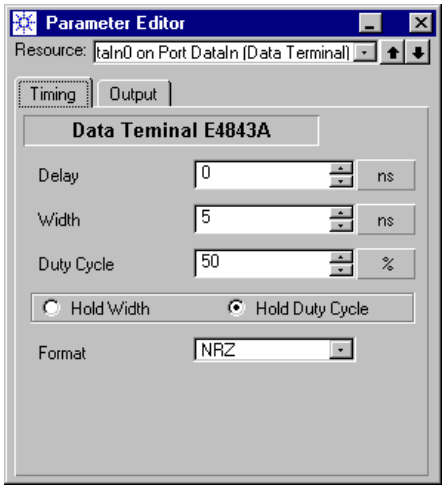


Figure 64 General Timing Parameters for a Data Generator

Note, that the window always displays the properties of the frontend channel. This implies, that the settings for a channel also affect the connected DUT terminal and vice versa.

Depending on the type of the frontend, the editor can contain more parameters than shown in the figure above.

How to Set Generator Timing Parameters

To set the timing parameters for a generator channel:

1 Check the units.

The Parameter Editor displays default units and has default vernier steps. Both can be adjusted according to your needs (see *"How to Change Units and/or Vernier Steps" on page 68*).

2 Set the timing options:

– *Delay*:

The delay is relative to the system clock. A negative delay can be set if the system clock has been delayed.

– *Width* or *Duty Cycle*:

They are mutually dependent. If you have typed a number in one of the two fields, terminate your input with the Return or Enter key. This updates the other field (*Width* or *Duty Cycle*).

– *Hold Width* or *Hold Duty Cycle*:

For the case of a clock frequency change, you can set one of the two values to be fixed.

– *Rise/Fall Time* (with E4838A or E4842A frontends only):

In the Parameter Editor the rise/fall time can be set in the range of 0.5 to 10.0 ns. However, the frontends do not support the complete range of rise/fall times.

The E4838A supports the range from 0.5 to 4.5 ns, the E4842A supports the range from 0.7 to 6.0 ns. See the *Technical Specification* for details.

The rise and fall times are coupled. The resulting signal shape is illustrated below:

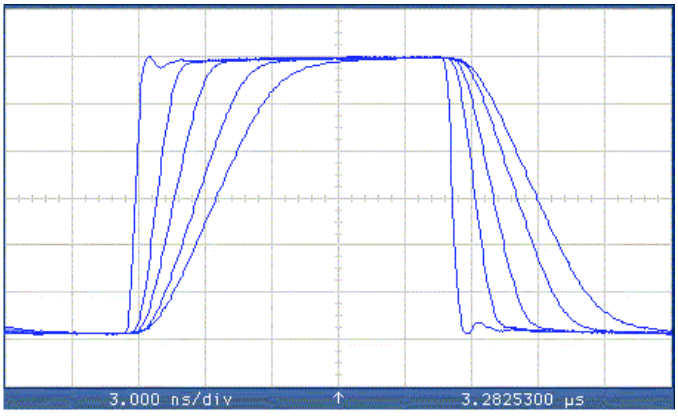


Figure 65 Variable Slopes Setting

– *Format*:

Choose from the list.

The default signal format for a data port and the channels connected to a data port is NRZ. The default format for a pulse port or an unconnected generator frontend is RZ.

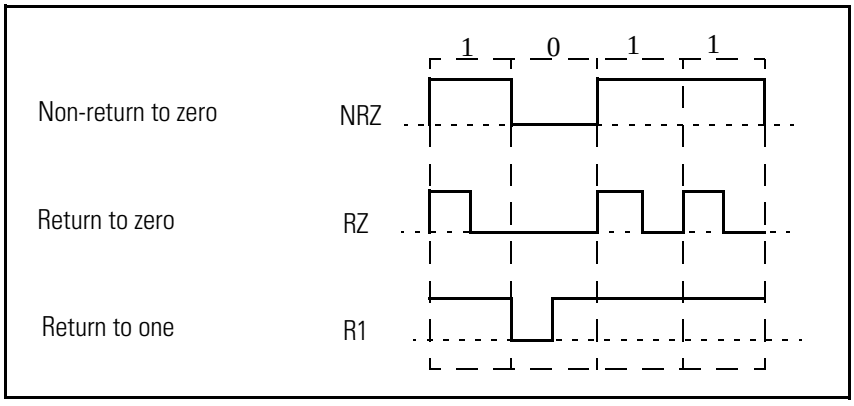


Figure 66 Signal Formats

How to Set Additional Generator Parameters

1 In the Parameter Editor window, click on the second tab which is labeled *Output*.

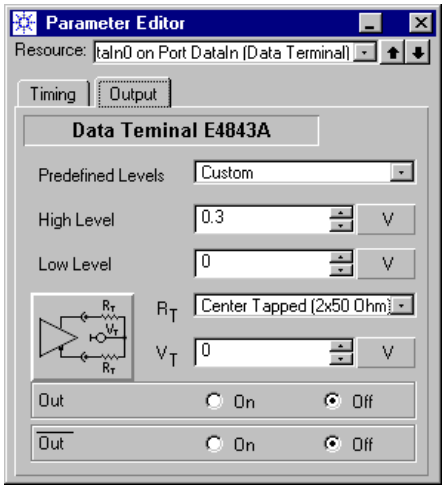


Figure 67 Additional Parameters for a Connection With a Data Generator

2 Check the units.

The Parameter Editor displays default units and has default vernier steps. Both can be adjusted according to your needs (see *"How to Change Units and/or Vernier Steps" on page 68*).

3 Decide on the additional options. The additional options are:

– *Predefined Levels*: Use the levels shown or entered in the window (Custom) or use one of the predefined levels.

The predefined levels for a generator depend on the frontend. They are listed in the table below:

Table 8 Predefined Generator Signal Levels

Name	High Level	Low Level	Termination Voltage	Termination Impedance	E4838A E4842A	E4843A E4846A	E4862A E4864A
TTL (into open)	2.5 V	0.0 V	0.0 V	Open		x	
TTL (into 50 Ω to GND)	2.5 V	0.0 V	0.0 V	2 * 50 Ω	x	x	
CMOS 5V (into open)	5.0 V	0.0 V	0.0 V	Open		x	
CMOS 3.3V (into open)	3.3 V	0.0 V	0.0 V	Open		x	
ECL (into 50 Ω to -2V)	-0.9 V	-1.7 V	-2.0 V	2 * 50 Ω	x	x	x
ECL (into 50 Ω to GND)	-0.9 V	-1.7 V	0.0 V	2 * 50 Ω	x	x	x
PECL (into 50 Ω to +3V)	4.1 V	3.3 V	3.0 V	2 * 50 Ω	x	x	x

– If you are using your own custom setting:

Termination impedance usually is the input impedance of the DUT.

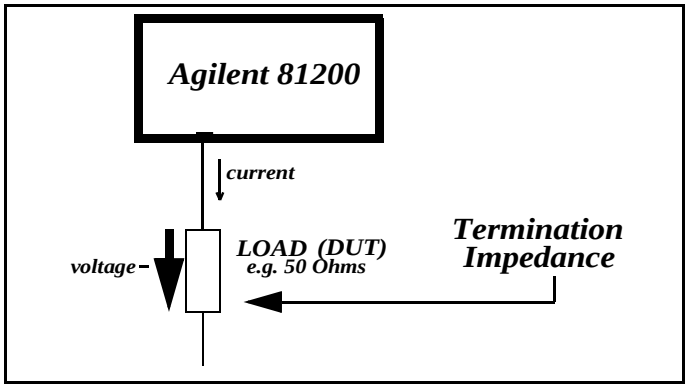


Figure 68 Termination Impedance

The *termination voltage* for a terminal that is not connected to ground must not remain below –2 V.

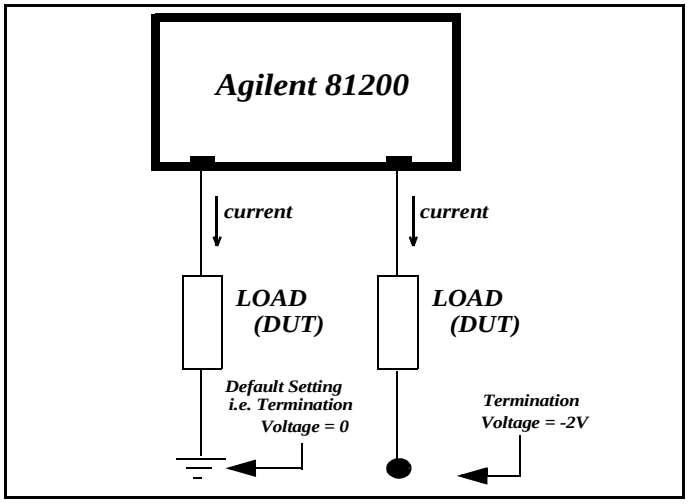


Figure 69 Termination Voltage

4 Special load options are supported by the differential generator frontends E4838A, E4843A, E4862A, and E4864A. For these frontends, you can select between center tapped and differential termination by clicking the graphical button.

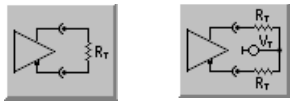


Figure 70 Termination Options for Differential Generators

Center tapped termination uses two 50 Ω resistors. The termination voltage range is –2 to +3 V.

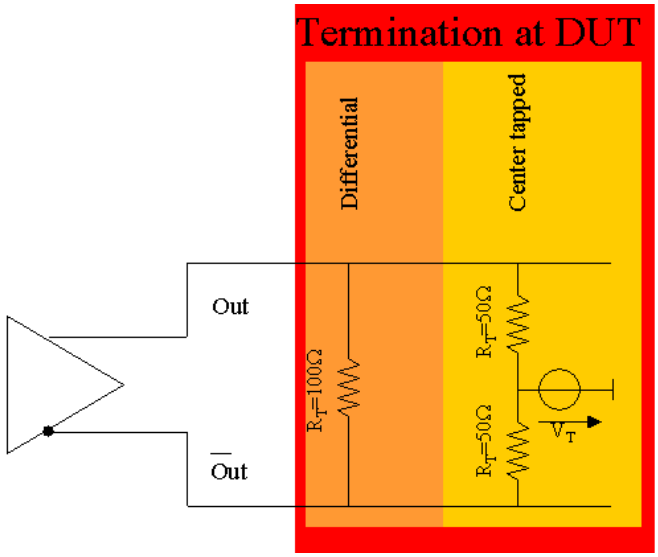


Figure 71 Supported Signal Termination Options at the DUT

5 Choose the polarity (E4838A frontend only): This allows to invert the polarity of the output signal.

6 Activate the connection by switching the output radio button to *On*. This is probably the most important step of all!

TIP When running a test, check the green LEDs on the frontends. They indicate whether the channel is enabled or disabled. If a channel is disconnected due to hardware constraints, correct its physical termination and then use the Connectors On/Off button to re-establish the connection.

7 Close the window.

How to Add Channels in Analog Mode

If you have selected an E4838A frontend, the Parameter Editor shows an additional tab: *Analog Channel Add*. This window provides additional parameters for combining two output channels.

NOTE These parameters are just set with the Parameter Editor. To activate them, you need to combine the channel with the channel above in the Channel Configuration Editor (see *"How to Combine Generator Channels" on page 129*).

An output channel of an E4838A frontend can only be combined with the channel above if this other channel is also an E4843A frontend.

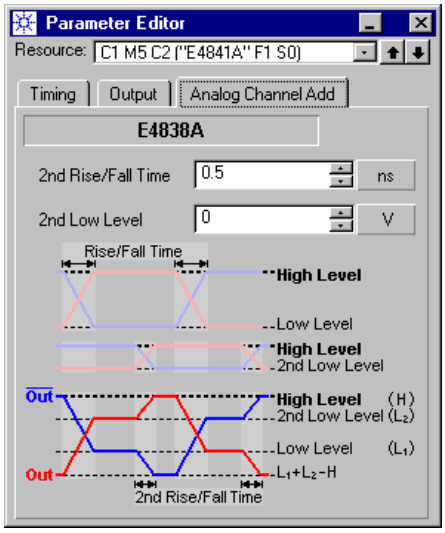


Figure 72 Analog Channel Add Parameters

The window illustrates how the analog channel addition works.

You are adding two differential signals. A differential signal consists of an OUT and a complementary \OUT\ voltage. Both signals can have different slew rates (rise/fall times) and amplitudes (the voltage difference between high and low level).

But the output voltage of the E4838A frontend at 50 Ω load is limited to –2.2 V to +4.4 V, and the output voltage swing must be in the range of 0.05 V to 3.5 V (for details refer to the *Agilent 81250 Technical Specifications*).

So we keep the high level voltage of the frontend that holds the connector. That means, the connected channel's high level voltage is only reached after adding the two OUT channels. The amplitudes Amp1 and Amp2 remain unchanged, and the resulting voltage swing of the signal is the sum of both amplitudes.

A similar scheme applies to the complementary \OUT\ connector.

To change the characteristics of the second channel:

1 Adjust the rise/fall time of the second channel (only E4838A or E4843A frontends, see also *"How to Set Generator Timing Parameters" on page 118*).

2 Set the low level of the second channel. This specifies the amplitude Amp2.

Error messages will be displayed if you try to override the physical limits of the E4838A frontend. If that occurs, you need to correct the amplitudes.

Example Assume the following setting:

High level = 1.5 V

First low level = 0 V

Second low level = 0 V

This results in the following OUT signal:

Table 9 Output Voltage for Analog Channel Addition

Logic at Channel 1	Logic at Channel 2	Output Voltage
1	1	1.5 V
1	0	0 V
0	1	0 V
0	0	–1.5 V

The method is once more illustrated below:

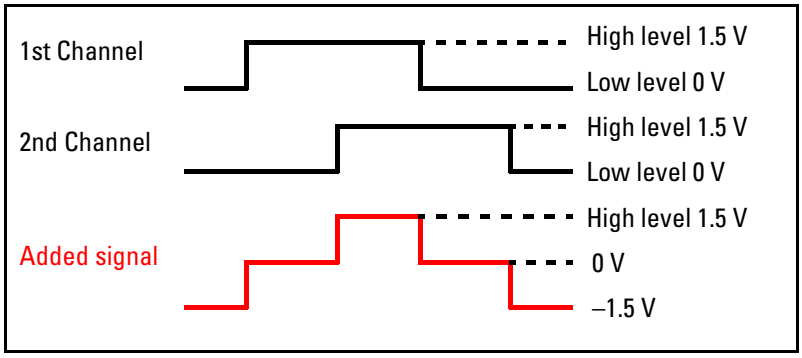


Figure 73 Analog Channel Addition

The above setting is accepted because the resulting amplitude and voltages can be generated by the E4838A frontend.

If you have chosen one of the predefined levels for the connected generator channel, you may need to change the second low level which is set to 0 V by default.

TTL into 50 Ω, for example, would yield a voltage swing of 5 V which is unacceptable. But if you change the second low level to for instance 2 V (for generating spikes of 0.5 V), this is well in the range of the E4838A frontend.

A voltage overshot could hence be generated as illustrated below.

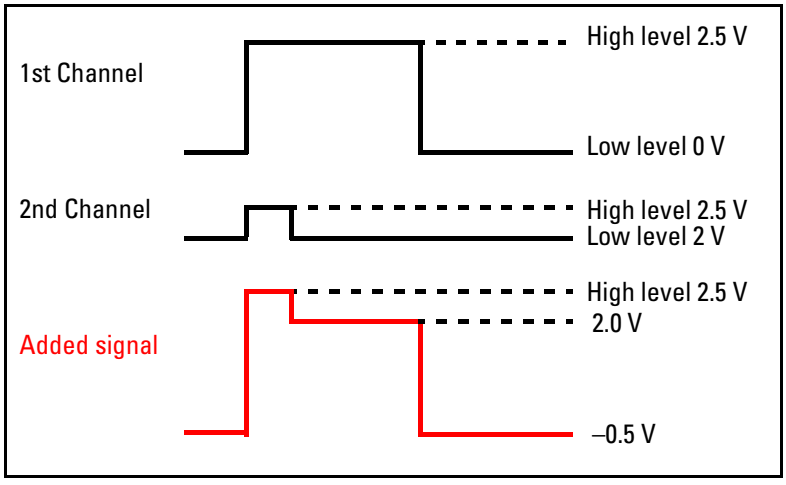


Figure 74 Overshot Generation

To generate real-world signals, you will also have to take the rise/fall times into account.

NOTE The Analog Channel Add setting only takes effect after the two channels have been added by means of the Channel Configuration Editor.

Both channels can have different timing parameters, such as frequency, pulse width and delay. High level and expected load are determined by the channel that holds the connector.

How to Set Up a DUT Output Port or Analyzer Channel

To set the parameters for a port or terminal that returns signals or for a channel that receives signals:

1 Open the Parameter Editor for the port or channel (see *"How to Start the Parameter Editor for Ports/Channels" on page 116*).

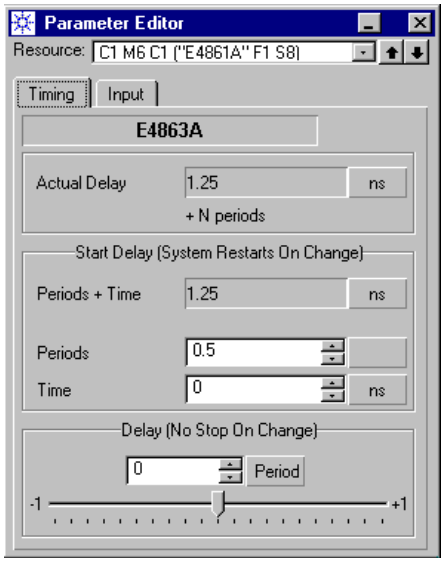


Figure 75 Timing Parameters for a Data Analyzer

NOTE The Parameter Editor always displays the identification of the connected frontend. Thus, changes done in the channel parameters will also appear in the settings of the connected port terminal and vice versa.

How to Set Analyzer Timing Parameters

1 Check the units.

The Parameter Editor displays default units and has default vernier steps. Both can be adjusted according to your needs (see *"How to Change Units and/or Vernier Steps" on page 68*).

2 Set the timing options:

– *Periods and Time*: The total delay for capturing received data is composed of a relative delay (in fractions of system clock cycles) and an absolute delay (independent of the system clock).

If one of these parameters is changed while a test is running, the test is aborted and restarted.

– *Delay (No Stop On Change)*: The sampling point of analyzer frontends plugged into an E4832A or E4861A generator/analyzer module can be fine-tuned within up to ±1 clock periods. The phase can be adjusted with the slider bar without interrupting a running test. See also *"Manual Analyzer Sampling Point Alignment" on page 47*.

How to Set Additional Analyzer Parameters

1 Click the second tab of the analyzer parameter window.

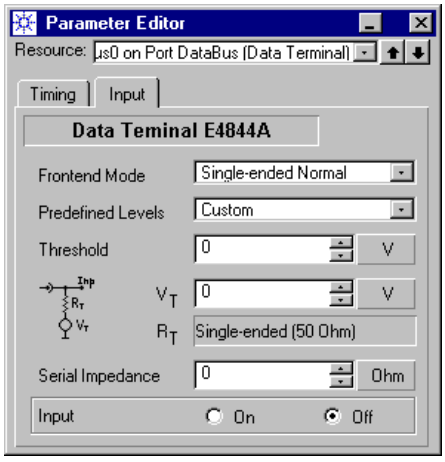
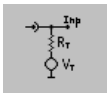


Figure 76 Additional Parameters for Single-ended Data Analyzers

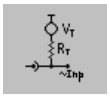
2 Select the frontend mode. Depending on the selected frontend the available options are:

– *Differential* (E4835A, E4837A, E4863A, and E4865A)

– *Single-ended Normal* (E4835A, E4844A, E4845A, E4847A, E4863A, and E4865A)



– *Single-ended Complement* (E4835A, E4863A, and E4865A)



3 Decide on the additional options. The additional options are:

– A set of *Predefined Levels*: Choose one of the predefined levels, use the defaults values or specify your own settings using *Custom* in this field.

The predefined levels for analyzers are listed in the table below:

Table 10 Predefined Analyzer Signal Levels

Name	Threshold	Termination Voltage	Internal Impedance	E4835A E4837A	E4844A E4845A	E4847A	E4863A E4865A
ECL (to -2V)	-1.3 V	-2.0 V	50 Ω	x	x	x	x
ECL (to GND)	-1.3 V	0.0 V	50 Ω	x	x	x	x
PECL (to +3V)	3.7 V	3.0 V	50 Ω	x	x	x	
TTL (to GND)	1.5 V	0.0 V	50 Ω	x	x	x	x
CMOS 5V (to HiZ)	2.5 V		HiZ			x	
CMOS 3.3V (to HiZ)	1.6 V		HiZ			x	

– If you are using your own custom setting:

The minimum *Termination Voltage* for a terminal that is not connected to ground is –2 V.

The standard input *Impedance* of an analyzer frontend is 50 Ω. See the specifications for details.

4 The *Serial impedance* is usually the output impedance of the DUT.

5 Do not forget to activate the connection by switching the *Input* button to *On*.

NOTE When running a test, check the green LEDs on the frontends. They indicate whether the channel is enabled or disabled.

6 Close the window.

Differential Analyzer Frontends

Special options are available for the differential analyzer frontends E4835A, E4837A, E4863A, and E4865A.

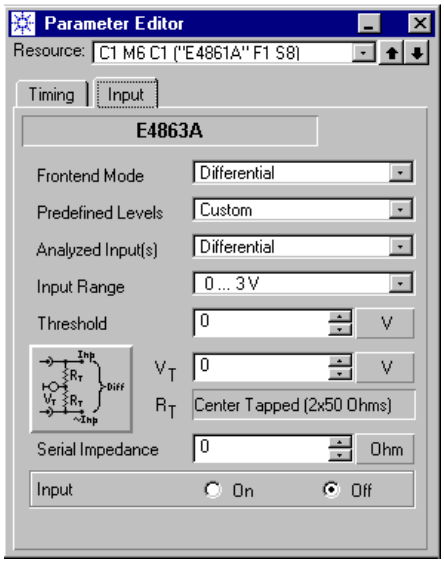


Figure 77 Additional Parameters for a Differential Data Analyzers

For these frontends you can:

- Select the *Analyzed Input(s)*. Choices are: Normal input IN, complementary input \IN\, differential input.
- If the **frontend mode** is set to *Differential* (not the input type!), you can choose between center tapped or differential input signal termination by clicking the graphical button.

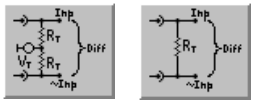


Figure 78 Termination Options for Differential Analyzers

Center tapped termination uses two 50 Ω resistors. The termination voltage range is –2 V to +3 V. Differential termination inserts a 100 Ω resistor.

Hence, these frontends provide the following input and measurement options:

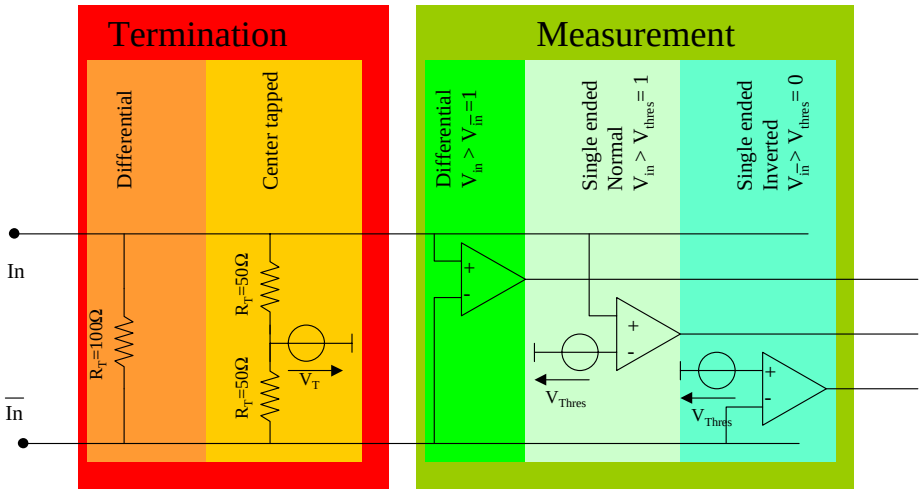


Figure 79 Termination and Measurement Options for Differential Analyzers

- For the analyzer frontends E4835A and E4837A, you can invert the signal polarity with the *Polarity* radio button.

  The settings of *Input Type* and *Polarity* have the following effects on data recognition:

Table 11 Data Acquisition with the E4835A or E4837A Differential Analyzer

Input Type	Input Signal Condition	Acquired Data	
		Normal Polarity	Inverted Polarity
Differential	Signal at IN > \IN\	1	0
	Signal at IN < \IN\	0	1
Normal Input	Signal at IN > Threshold	1	0
	Signal at IN < Threshold	0	1
Complementary Input	Signal at \IN\ > Threshold	0	1
	Signal at \IN\ < Threshold	1	0

How to Combine Generator Channels

Combining generator channels enables you to test devices with multiple edge timings or multi-level signals.

NOTE The channel add function enables you to combine two or four generator channels. The channels to be added must all reside in one module and must be contiguous. Adding channels reduces the number of active connectors.

Digital channel addition

The digital channel addition is an XOR addition (exclusive OR or modulo 2 addition). For details see the figure below. The addition takes place before levels are applied to the signals. The final signal is routed to one output amplifier.

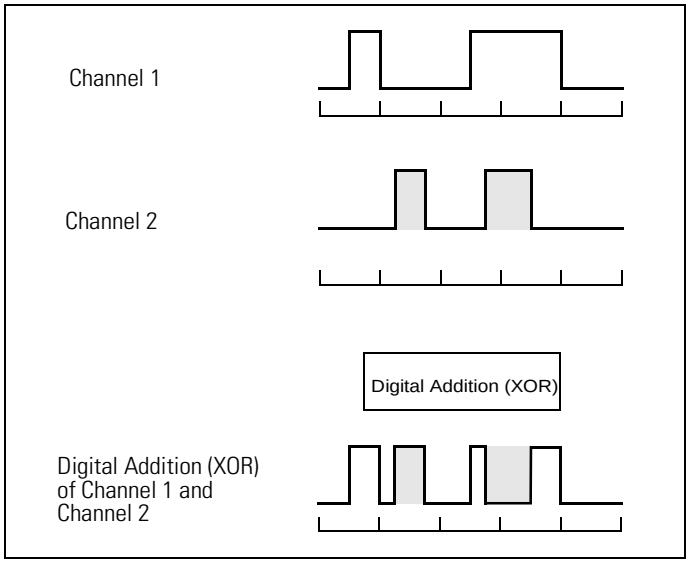


Figure 80 Digital Channel Addition

Analog channel addition

The analog channel addition is available if the module contains at least one E4838A generator module. It allows to add signal voltages. One can thus produce signals or pulses with overshot, ringing, and so on.

Channels are added with the Channel Configuration Editor.

How to Start the Channel Configuration Editor

The Channel Configuration Editor is started from the Connection Editor.

To start the Channel Configuration Editor:

1 Double-click the configuration area of a generator frontend.

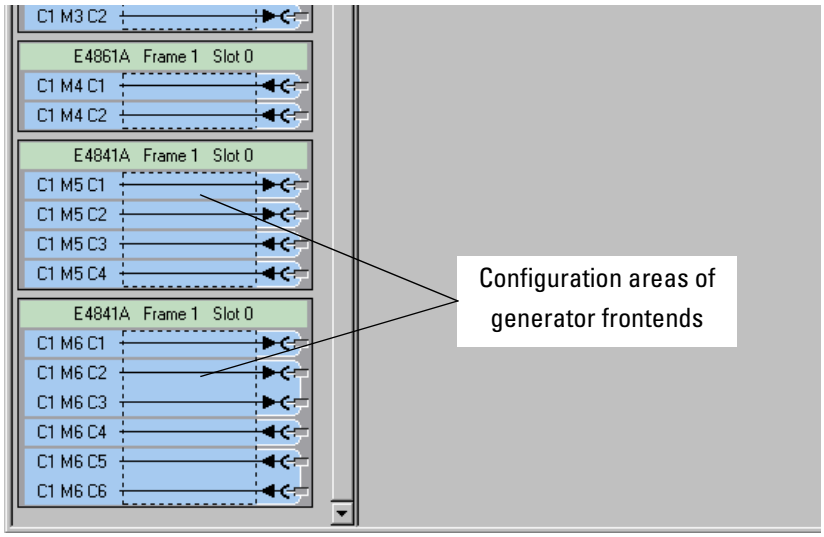


Figure 81 Module Configuration Area

Alternatively, you can also click on the configuration area with the right mouse button and select *Properties*.

How to Use the Channel Configuration Editor

The Channel Configuration Editor comes up with the following window:

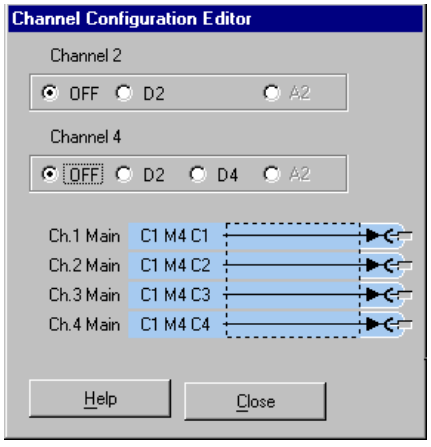


Figure 82 Channel Configuration Editor

With the options of the Channel Configuration Editor, you can affect the properties of channel 2 and/or channel 4.

Combining Two Channels

To add channels 1 and 2 and channels 3 and 4 in digital mode:

1 Activate D2 for channel 2.

2 Activate D2 for channel 4.

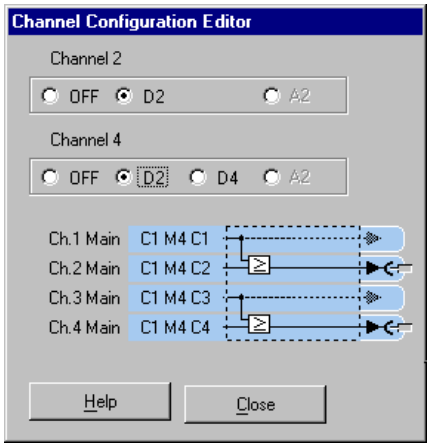


Figure 83 Channel Configuration Editor—Two Added Channels

Note that the connectors of channel 1 and channel 3 are no longer available for connections.

Combining Four Channels

To combine all four channels:

1 Activate D4 for channel 4.

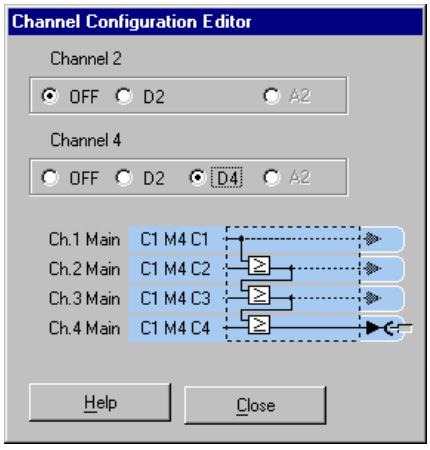


Figure 84 Channel Configuration Editor—Four Added Channels

Note that the connectors of channels 1 to 3 are no longer available for connections.

Analog Channel Adding

If the module contains one or several E4838A generator frontends in even-numbered slots (channel 2 or 4) and identical or E4843A frontends in the slots above, you can also add two channels in analog mode by clicking the A2 radio button.

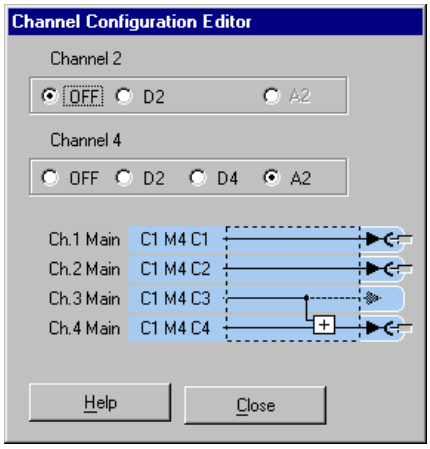


Figure 85 Channel Configuration Editor—Analog Added Channels

If this option is selected, the parameter settings of the E4838A frontend for "Analog Channel Add" take effect (see *"How to Add Channels in Analog Mode" on page 122*).

Choosing the Kind of Measurement

Once the DUT has been modeled, its pins have been connected to instrument channels, and the channel parameters have been specified, it is time to tell the system what kind of measurement is going to be performed.

This has to be done before setting up the stream of generated and expected data, because different tests require different settings. Although the procedure for setting up the data sequence is always the same, the available segment options and result displays depend on the selected kind of measurement.

NOTE It is recommended to save settings repeatedly during test setup. You should take care of the chosen measurement type and save the setting under a file name that indicates the kind of measurement.

The kind of measurement is chosen from the Measurement Configuration window.

How to Access the Measurement Configuration Window

To open the Measurement Configuration window:

1 Click the Measurement Configuration icon in the tool bar.

Alternatively, you can also use the corresponding option of the *Go* menu.

The Measurement Configuration window appears.

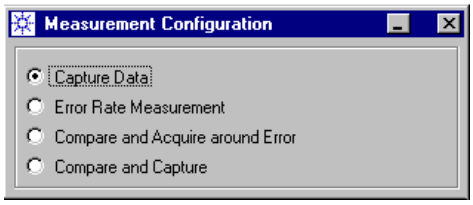


Figure 86 Measurement Configuration Window

How to Set the Measurement Configuration

The Measurement Configuration window provides four options:

- Capture Data
- Error Rate Measurement
- Compare and Acquire Around Error
- Compare and Capture

Capture Data

In this measurement mode the system captures the data received by the analyzer connectors until either the test sequence is finished, or until the memory is full, or the Stop button was clicked.

After the test has finished the recorded data can be viewed in the Error State Display and with the Waveform Viewer.

Error Rate Measurement

In this measurement mode the system continually samples incoming data applied to the analyzer connectors and compares the data in real time with expected data. The errors are counted and the bit error rate is calculated.

The result can be reviewed in the Bit Error Rate Display.

If you have chosen Error Rate Measurement, you have the following additional options:

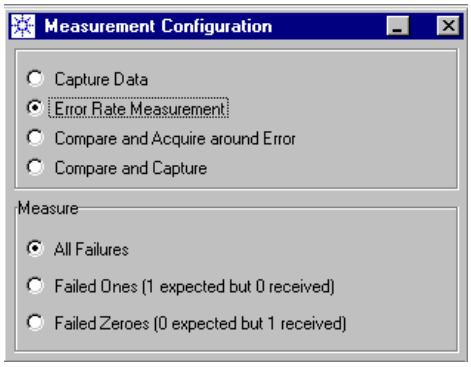


Figure 87 Measurement Configuration Window for Error Rate Measurement

Select the kind of errors you wish to be counted.

Compare and Acquire Around Error

In this measurement mode the system records the data applied to the analyzer connectors until the test sequence is completed. The acquired data is compared in real time with expected data. As soon as an error occurs the system starts a counter.

The advantage of this mode is, that it is possible to define how long data should be recorded after the failure occurred, so that a pre- and post history around the error is captured and can be analyzed. The result can be reviewed in the Error State Display and with the Waveform Viewer.

If you have chosen Compare and Acquire around Error, you have one additional option:

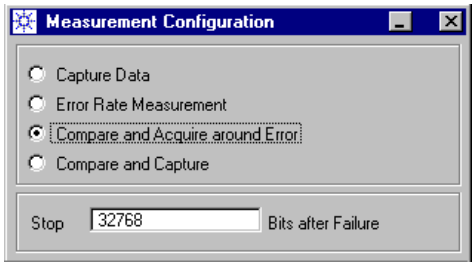


Figure 88 Measurement Configuration Window for Capture Around Error Measurement

You can set the number of bitstream vectors to be stored after the error occurred.

If an error occurs, the test will stop after the specified number of vectors has been acquired. The test will, of course, also stop if the test sequence expires before that number is reached.

Compare and Capture

In this measurement mode the system captures data applied to the analyzer connectors until the sequence is finished or the Stop button is pressed. While capturing, the system also compares the captured data with the expected data in real time.

The result can be reviewed in the Error State Display and with the Waveform Viewer.

Creating the Stream of Generated and Expected Data

After you have chosen the kind of measurement to be performed, you can build the test sequence.

The data to be generated or expected is embedded in a sequence. Three tools are provided for manipulating that sequence:

- Standard Mode Sequence Editor
- Detail Mode Sequence Editor
- Data/Sequence Editor

All the Sequence Editors can be started from the *Go* menu.

You can also click the Sequence Editor icon in the tool bar.

This opens the Standard Mode Sequence Editor, if the sequence conforms to the rules for a BER measurement. If not, the Detail Mode Sequence Editor is started.

For details see:

"The Standard Mode Sequence Editor" on page 140

"The Detail Mode Sequence Editor" on page 152

"Using the Data/Sequence Editor" on page 189

The Standard Mode Sequence Editor

The Standard Mode Sequence Editor is first of all meant for quick and easy setup of tests where generated and expected data are infinitely looped, such as bit error rate measurements.

For a simple and straightforward bit error rate measurement there is no need to worry about all the details of sequence blocks, loops, data segments, triggers, events, and so on.

You need only specify the PRBS polynomial or the data pattern to be used and are ready to run the test. Once it is started, the test will run until the Stop button is clicked.

If you have set up a new device, the Standard Mode Sequence Editor shows a window like the following:

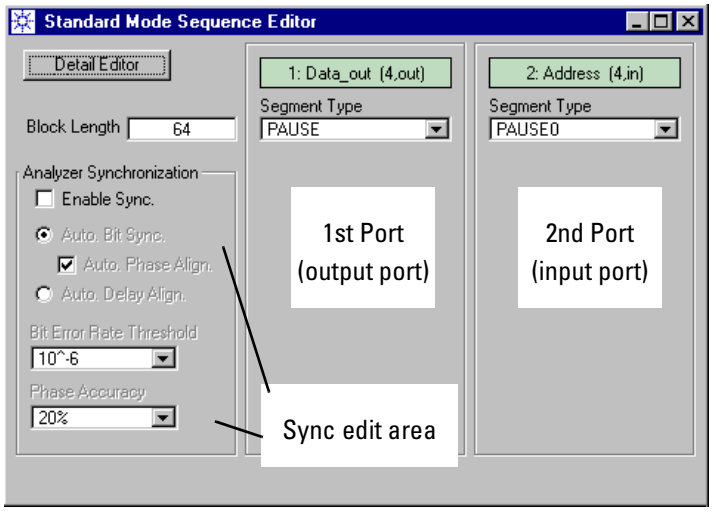


Figure 89 Standard Mode Sequence Editor Window

The window has one panel for each DUT data port. The default segments are PAUSE0 for DUT input ports (connected to generator frontends), and PAUSE for output ports (connected to analyzers).

How to Use the Standard Mode Sequence Editor

You can:

- Replace the current segments by new or existing segments
- Change segment properties
- Switch to the Detail Mode Sequence Editor
- Enable/Disable the automatic analyzer sampling point adjustment
- Edit the synchronization criteria

Replacing a Segment

To replace the current segment of a port by a different segment:

1 Open the *Segment Type* selection box.

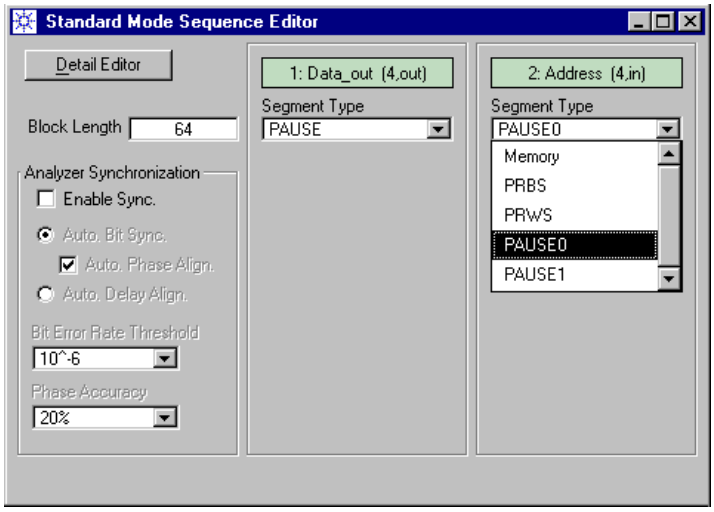


Figure 90 Segment Type Selection

2 Choose the segment type: Memory-based, PRBS, or PRWS.

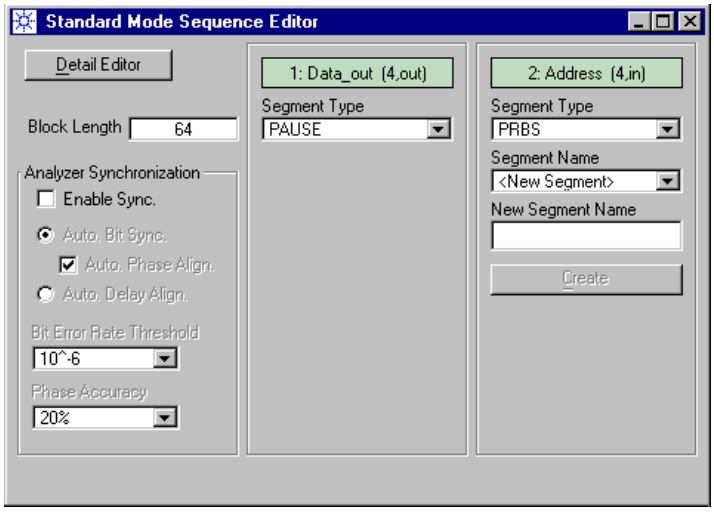


Figure 91 Search Existing or Create New Segment

3 Choose a segment from the *Segment Name* browser, if you wish to use a segment that has been created previously

or

enter a *New Segment Name*, if you wish to create a new segment, and click *Create*.

Changing a Segment

If you have created a new PRBS/PRWS segment or chosen a PRBS/PRWS segment from the **local segment pool** (see also *"Data Segments" on page 38*), you can now change its polynomial and type.

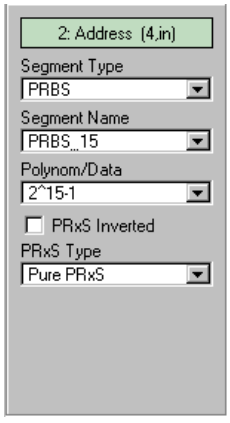


Figure 92 PRBS Parameters

For a description of pure and non-pure PRBS/PRWS please refer to *"Appendix B: PRBS/PRWS Data Segments" on page 251*.

NOTE For 2^{23}–1 or 2^{31}–1 polynomials only *pure PRxS* is supported.

If you are creating a new memory-type segment, you can specify its length. The minimum segment length is equal to the length of the block and indicated on the panel:

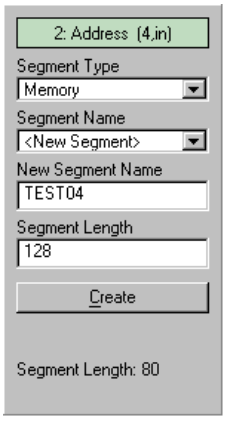


Figure 93 Creating a New Memory Segment

If you have created a new memory-type segment or chosen a memory segment from the local segment pool, you can now click *Edit* and change the contents of the segment.

For details see *"How to Create a New Segment" on page 176*.

A new segment is automatically stored in the **local segment pool** which is associated with the current setting. Local segments can be directly edited with the Standard Mode Sequence Editor.

TIP Use the Segment Editor, if you have chosen a segment from the **global segment pool** and wish to inspect or change its contents. See also *"Data Segments" on page 38*.

Note that the size of a memory-type segment (length and width) may exceed but must not remain under the size of the block. Additional restrictions apply, if *Automatic Bit Synchronization* is enabled (see *"Block Length and Segment Length" on page 150*).

If the segment is too large, only a portion is used. If the segment is too small and is a local segment, its size is automatically increased. You get a message like the following:

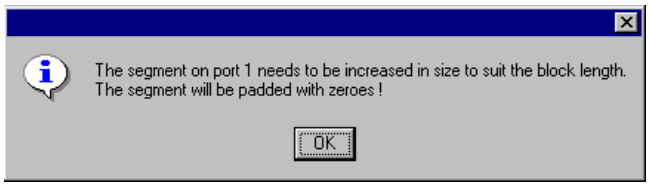


Figure 94 Segment-will-be-increased Message

To use the same segment also for a different port:

1 Select the same *Segment Type*.

2 Choose the segment from the *Segment Name* browser.
The browser shows local as well as global segments.

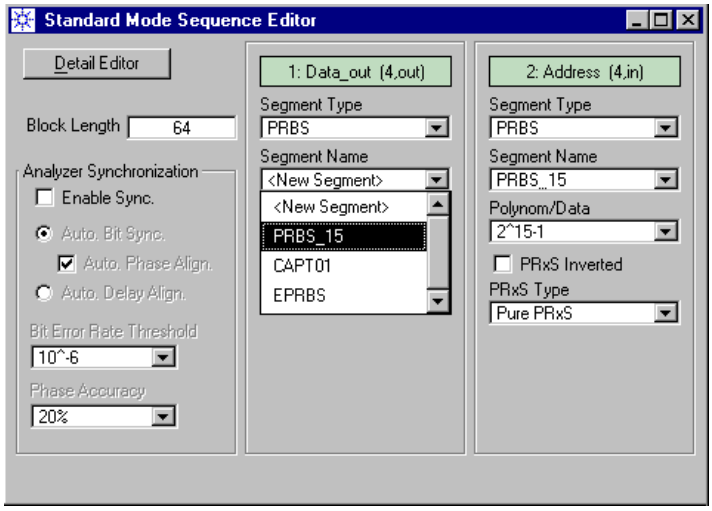


Figure 95 Segment Name Browser

Switching from Standard to Detail Mode Sequence Editor

You can always switch to the Detail Mode Sequence Editor by clicking the *Detail Editor* button.

The Detail Mode Sequence Editor gives you just another view:

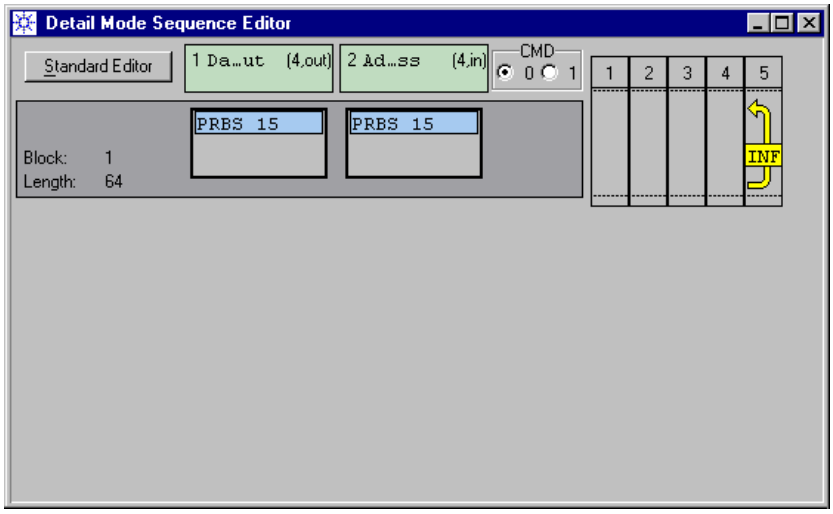


Figure 96 Sequence in Detail View

It shows that the generated and expected data will be infinitely repeated.

As long as you do not change the sequence with the Detail Mode Sequence Editor, you can always return to the Standard Mode Sequence Editor by clicking *Standard Editor*.

How to Synchronize an Analyzer With Incoming Data

The automatic analyzer sampling point adjustment with incoming data is available for both the Standard and the Detail Mode Sequence Editors.

For a detailed description of this feature please refer to *"Analyzer Sampling Point Adjustment" on page 46*.

Analyzer synchronization can be directly enabled/disabled and edited in the Standard Mode Sequence Editor window:

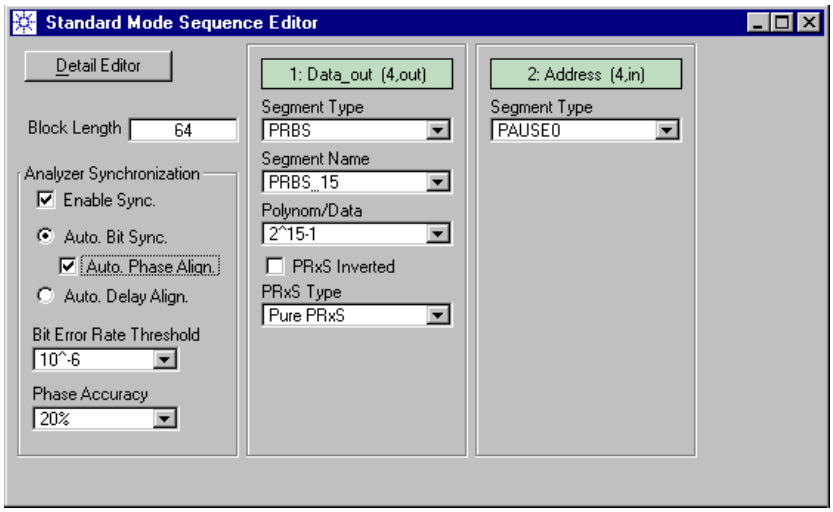


Figure 97 Synchronization Enabled

In the Detail Mode Sequence Editor, the same functions are provided by the *Edit* menu and the context menu of a block:

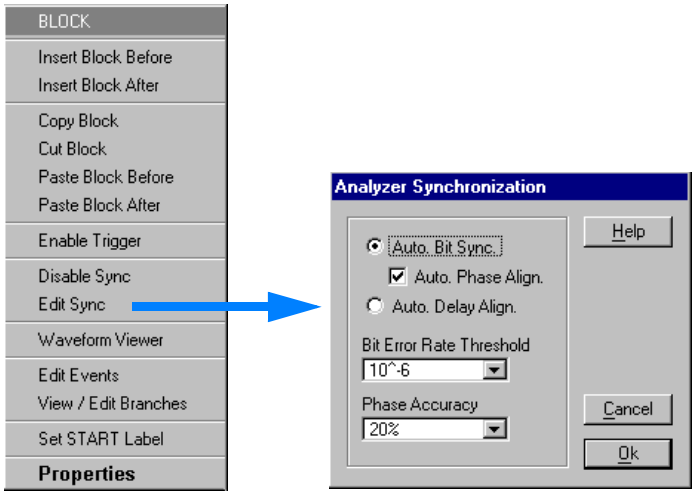


Figure 98 Block Context Menu and Sync Edit Window

1 Click *Enable Sync* to activate the automatic analyzer sampling point adjustment.

If you are using the Standard Mode Sequence Editor, a synchronization block is automatically inserted into the sequence. This block is placed ahead of the infinite loop. It has the same length and contains the same data as the test block.

When the test is run, the synchronization block is automatically repeated until the synchronization criteria are met. Then the sequencer continues with the next block.

2 Choose the kind of synchronization.

Automatic Bit Synchronization

Automatic Bit Synchronization offers the option to enable or disable *Automatic Phase Alignment*:

- Automatic Bit Synchronization without Automatic Phase Alignment is used if the total delay from test start is unknown but a certain edge delay relative to the analyzer clock is expected.
- Automatic Bit Synchronization with Automatic Phase Alignment is used if the delay is completely unknown.

PRBS data may be sent and expected. Memory-based data may also be expected by a pure analyzing system.

The final delay status is indicated by the Parameter Editor.

Automatic Bit Synchronization does not report the number of clock periods that have passed since test start, but only the phase shift relative to the clock.

Auto Bit Sync without Auto Phase Alignment

If Automatic Phase Alignment is disabled, then the analyzer uses the start delay that has been specified with the Parameter Editor to determine the sampling point delay relative to its clock. It then samples the incoming data until the incoming data matches the expected pattern with an adequate accuracy.

Once this accuracy is reached, then the incoming bits are aligned with the expected bits—the analyzer is synchronized with the incoming data.

Auto Bit Sync with Auto Phase Alignment

If Automatic Phase Alignment is enabled, then the analyzer fully automatically adjusts itself to capture the incoming data at the optimum sampling point.

It shifts the sampling point stepwise in both directions until the specified bit error rate is reached. The width of these steps is adjustable. The analyzer then measures the width of the eye diagram and positions the sampling point at the optimum.

Automatic Delay Alignment

Automatic Delay Alignment is used if the delay between the start of the test and the incoming data is coarsely known and set as the start delay with the Parameter Editor. PRBS as well as memory-type data can be generated and expected.

The analyzer starts after the start delay has elapsed. It then shifts the sampling point within a certain range (±50 ns for an E4832A data

generator/analyzer module and ±10 ns for an E4861A module) until it recognizes the expected pattern with an adequate accuracy.

After that, it measures the width of the eye diagram and positions the sampling point in the middle. The final delay status is indicated by the Parameter Editor.

Automatic Delay Alignment reports the full delay since test start.

3 Set the *Bit Error Rate Threshold*.

This threshold defines the "adequate accuracy" which depends on the DUT and the test requirements. The synchronization process is not complete unless the actual BER remains under this threshold.

4 Set the *Phase Accuracy*.

The phase accuracy can be set between 20 % and 1 %. This defines the number of steps performed by the phase optimizing algorithm (5 to 100) and has an impact on the speed of the synchronization process.

The Detail Mode Sequence Editor uses two symbols to highlight the synchronization block:

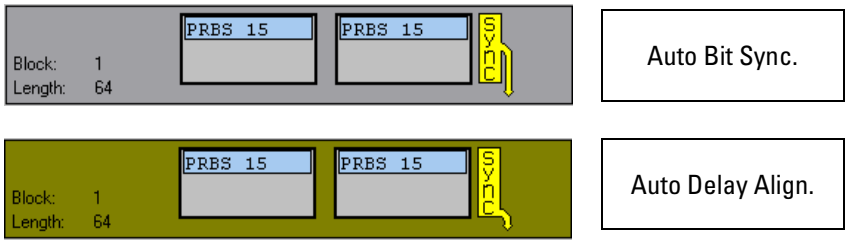


Figure 99 Synchronization Block Indicators

NOTE If you disable the automatic analyzer sampling point adjustment with the Standard Mode Sequence Editor, the synchronization block is removed from the sequence. Only the test block with the infinite loop remains.

If you disable the automatic analyzer sampling point adjustment with the Detail Mode Sequence Editor, only the sync label is removed from the block.

Special Characteristics of the Standard Mode Sequence Editor

If a test sequence was edited with the Detail Mode Sequence Editor or the Data/Sequence Editor, it can happen that the Standard Mode Sequence Editor cannot be opened from the *Go* menu.

The reason is that the sequence does not conform to the rules for a simple bit error rate measurement.

BER Measurement Sequences

A BER measurement requires one single test block which is infinitely repeated. If the automatic analyzer sampling point adjustment is enabled, this block is preceded by a synchronization block.

A valid sequence with Automatic Bit Synchronization may look like this:

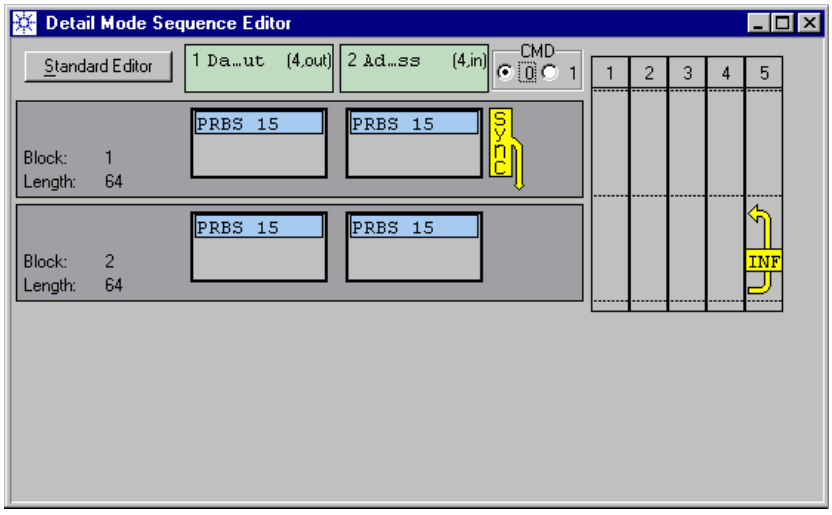


Figure 100 Test Sequence for BER Measurement With Bit Synchronization Enabled

The Standard Mode Sequence Editor can be used, if the test sequence meets the following criteria:

- The sequence contains one test block which is perpetually repeated.
- Different data segments may be used for different ports.
- The test block may be preceded by a synchronization block (a block with sync enabled and without loop). This block must be an unchanged copy of the test block (same size, same segments).
- The sequence does not process any internal or external events.
- A trigger (activated with the Detail Mode Sequence Editor) may be generated but is not shown in the Standard Mode Sequence Editor.
- The blocks have a certain length (which implies that the referenced segments must also have a corresponding minimum length).

Block Length and Segment Length

The Standard Mode Sequence Editor calculates the **block length** automatically. The block length has to be an integer multiple of the system-wide segment resolution (see *"How to Set the General System Frequency" on page 94*).

The **minimum length of a segment** is the product of the segment resolution times the FM factor of the port (SR * FM).

Example:
If the segment resolution is 4 and the FM factor of the port is 2, then the minimum segment length is 8.

If a segment with a user-defined length is entered into a port with the Standard Mode Sequence Editor, its length is automatically adjusted to the next integer multiple of SR * FM.

Example:
If you enter a segment with a length of 120 vectors and the minimum segment length is 16, then the segment length will be increased to 128.

NOTE Two special conditions apply to segments that specify expected data in conjunction with **Automatic Bit Synchronization** (not Auto Delay Alignment). If Automatic Bit Synchronization is enabled, then please observe:

- If a memory-based data segment is used for synchronization, the block length has to be a multiple of 32 * SR and the length of the segment has to be at least 32 * SR * FM.
- If a non-pure PRBS-type segment is used for synchronization, the block length has to be exactly SR * PL (where PL is the length of the polynomial—for example 2^{13}–1 means 8191).

All this is automatically considered by the Standard Mode Sequence Editor, but not by the Detail Mode Sequence Editor which allows all kinds of modifications.

The Standard Mode Sequence Editor automatically sets the correct segment length and adjusts the block length if necessary. This is especially important, if some of the ports are operated at different clock rates.

Switching From Detail to Standard Mode Sequence Editor

If you try to switch from the Detail Mode Sequence Editor to the Standard Mode Sequence Editor, the following question may appear:

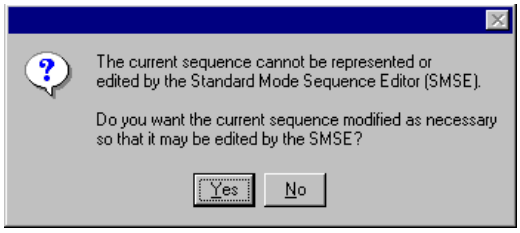


Figure 101 Transition From Detail to Standard Mode Sequence Editor

CAUTION

Click *No*, if you have created a sophisticated sequence.

Remember that the Standard Mode Sequence Editor is first of all meant for setting up a simple BER test, based on one block which is infinitely looped.

If the above question appears and you really wish to use the Standard Mode Sequence Editor, it may be a good idea to first save the current setting. If you have inadvertently clicked *Yes*, the only way to restore the previous sequence is to re-load the last setting.

If you click *Yes*, the following happens:

- All blocks of the sequence except the first one will be deleted.
- All events specified for the sequence are deleted.
- All loops are deleted.
- The remaining block is considered to be the test block. It gets an infinite loop.
- If automatic analyzer sampling point adjustment is enabled for the remaining block, the block is duplicated. The copy is inserted as block one and serves as a synchronization block (a block with sync enabled and without loop).

The Detail Mode Sequence Editor

The Detail Mode Sequence Editor allows to create and maintain individual test sequences.

A sequence consists of blocks. The blocks can be executed one after the other. Blocks and groups of blocks can also be repeated a specified number of times before the sequence continues. In addition, an endless loop can be specified – with the result that the sequence never ends.

If you activate the event recognition feature built into the system, the order of the block execution becomes variable. Based on specified events, you can leave a loop or even a block and continue with another block of the sequence.

The blocks reference data segments. These segments specify the generated and expected data patterns.

For details see:

"Contents of the Detail Mode Sequence Editor Window" on page 153

"How to Add, Move or Delete Blocks" on page 154

"How to Change Block Properties" on page 155

"How to Replace the Current Segment" on page 158

"How to Create and Change Loops" on page 162

"How to Specify Events and Reactions Upon Events" on page 164

"How to Synchronize an Analyzer With Incoming Data" on page 145

TIP After the sequence has been set up, it can also be inspected in detail and changed with the Data/Sequence Editor (see *"Using the Data/Sequence Editor" on page 189*).

Contents of the Detail Mode Sequence Editor Window

For a new device, the Detail Mode Sequence Editor shows one single block:

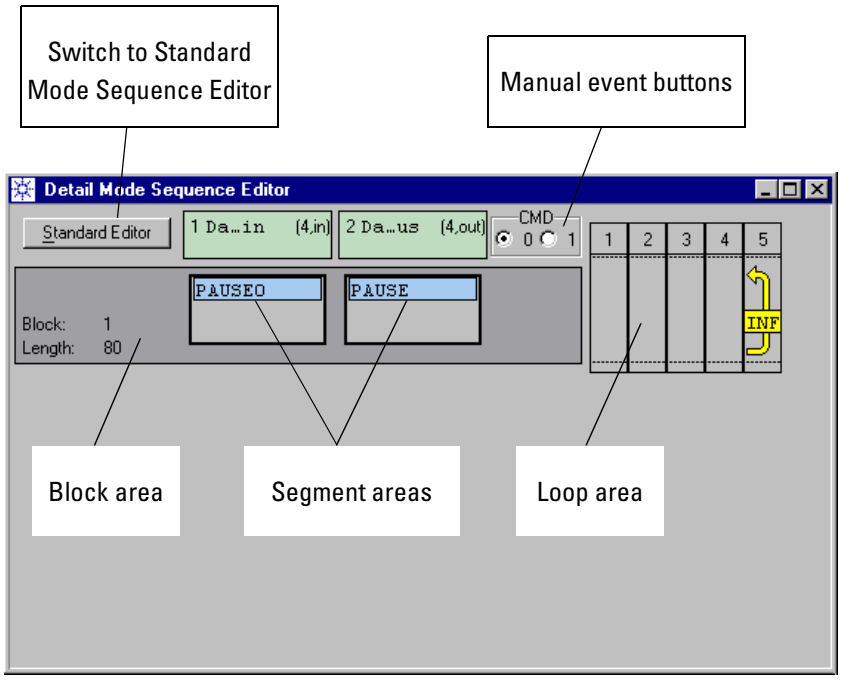


Figure 102 Default Sequence for a New Device With Two Data Ports

The block has a default length and includes default segments for all data input and data output ports that have been configured with the Connection Editor.

If the DUT setup includes more ports than can be shown, then a horizontal scroll bar is displayed to move to the hidden ports.

NOTE The default segments as well as the available pseudo segments depend on the type of port (DUT input or output) and on the kind of measurement.

At the right-hand side of the block is the loop area where repetitions can be specified.

How to Add, Move or Delete Blocks

A sequence often consists of more than one block. A sequence can contain up to 60 blocks. This number decreases if counted loops are used (see *"Hardware Dependencies" on page 42*).

You may wish to add blocks, delete blocks, or move blocks.

To manipulate the overall sequence:

1 Open the context menu of a block (right-click into the block area).

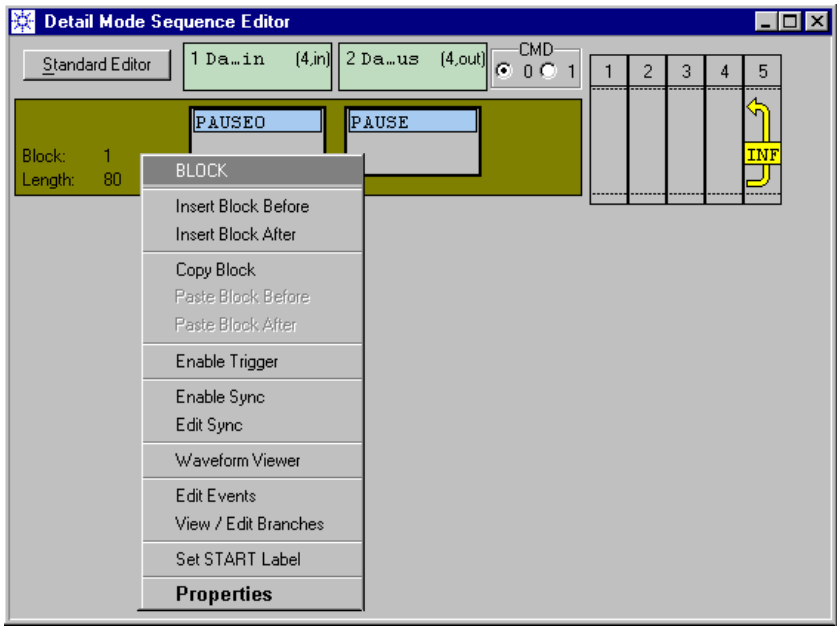


Figure 103 Block Menu

2 Choose the desired action:

- *Insert Block Before*: Inserts a new default block with default segments before the current block.
- *Insert Block After*: Inserts a new default block with default segments below the current block.
- *Copy Block*: Copies the chosen block to the clipboard.
- *Cut Block*: Copies the chosen block to the clipboard and removes it from the sequence (not available if the sequence contains only one block).
- *Paste Block Before*: Available, after a block has been copied or cut. Inserts the block from the clipboard above the current block.
- *Paste Block After*: Available, after a block has been copied or cut. Inserts the block from the clipboard below the current block.

How to Change Block Properties

Block characteristics include block length, block label, and trigger output.

To change the block characteristics:

1 Double-click on the block area.

Alternatively, you can also open the context menu of the block and select *Properties*.

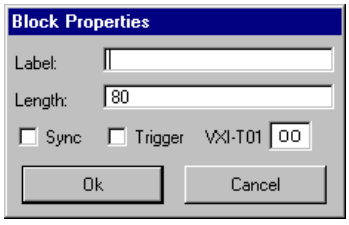


Figure 104 Block Properties Window

2 Enter appropriate data:

– *Label*: The block label should indicate the contents or purpose of the block. If the event recognition feature is used, the block label identifies the block that can be jumped to.

NOTE There are two block labels which have a special meaning:

The label START denotes the first block of a sequence. If this label is present, blocks above may exist, but are not processed when the test is run. The START label can also be assigned from the Block context menu by clicking *Set START label*.

The label END denotes the end of a sequence. This is an implicit label that should not be entered. The END label is used by the event recognition for terminating the test upon an event.

– *Length*: The length of the block must be a multiple of the segment resolution. For details see *"FMR and Segment Resolution" on page 31* and *"Block Length and Segment Length" on page 150*.

– *Sync*: The *Sync* enable button can be used to enable or disable automatic analyzer sampling point adjustment (see *"How to Use a Block for Analyzer Sampling Point Adjustment" on page 157*).

– *Trigger*: If you activate Trigger, a trigger is generated at the TRIGGER OUTPUT connector of the master clock module each time the block is started.

When you set the trigger, you may get the following message:

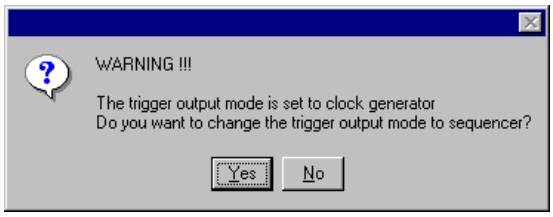


Figure 105 Wrong-Trigger-Mode Warning

By default, the TRIGGER OUTPUT of the master clock module is set up as a clock generator which generates a continuous clock pulse. If you wish to generate single trigger pulses, click *Yes*. See also *"How to Set the Characteristics of the Trigger Output" on page 103*.

The block area of the Sequence Editor indicates, whether a trigger is set.

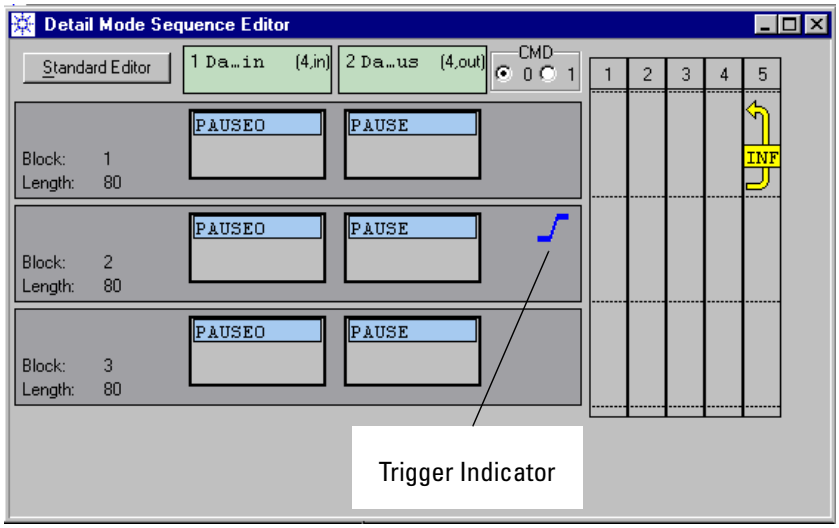


Figure 106 Block Trigger Indicator

– *VXI-T01*: You can also specify the setting of the VXI trigger lines T0/T1 at the beginning of the block.

If you don't wish to change their status, enter xx.

If you wish to activate the VXI triggers for controlling other VXI modules, ensure that you have not defined an event based on the status of the VXI trigger lines. See also *"How to Define Events" on page 168*.

How to Use a Block for Analyzer Sampling Point Adjustment

In principle, any block of the sequence can be used for synchronizing the analyzer channels with incoming data. But if automatic analyzer sampling point adjustment is required, measurements before synchronization usually don't make much sense.

Meaningful options are:

- Place the synchronization block at the beginning of the sequence.
- If a delay is needed before synchronization (for example to allow PLLs to settle), ensure that the synchronization block is only preceded by Pause blocks.
- If you wish to keep an existing sequence, set the START label to start the sequence execution at the synchronization block.

To create a synchronization block:

1 Select the block and open the block's context menu.

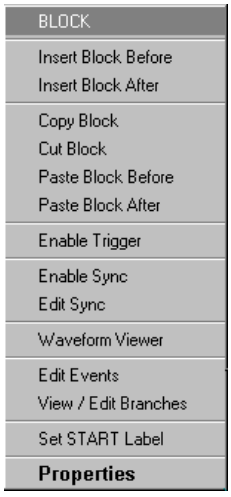


Figure 107 Block Context Menu

2 If it is not the first block, click *Set START Label* (not mandatory, but recommended).

The execution of the sequence will start with this block.

3 Click *Enable Sync*.

4 Click *Edit Sync* to check and eventually change the synchronization criteria. For details see *"How to Synchronize an Analyzer With Incoming Data" on page 145*.

NOTE Especially if you wish to use Automatic Bit Synchronization, you may need special segments and may have to adjust the length of the block and the segments. See *"Block Length and Segment Length" on page 150*.

How to Replace the Current Segment

The segments contained in a block describe the data to be generated or expected.

The default segments that appear in new blocks are pseudo segments. They depend on the type of port (DUT input or output) and on the chosen kind of measurement. They can be replaced by a different pseudo segment or a real segment.

To change a segment:

1 Open the segment's context menu (right-click on the segment name).

You get a menu like the one shown below:

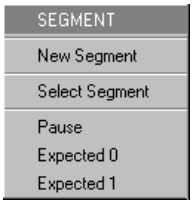


Figure 108 Segment Selection Menu of the Sequence Editor

2 Choose one of the options.

You can create a new segment, select an existing segment, or choose one of the available pseudo segments.

How to Replace a Segment by a New Real Segment

To replace the current segment by a new real segment:

1 Choose a segment and open the segment's context menu.

2 Select *New Segment*.

For details see *"How to Create a New Segment" on page 176*.

How to Replace a Segment by a Stored Segment

PRBS and memory segments that have been previously created are stored in the global or local segment pool.

To select a stored segment:

1 Select a segment and open the segment's context menu.

2 Choose *Select Segment*.

The segment selection window appears.

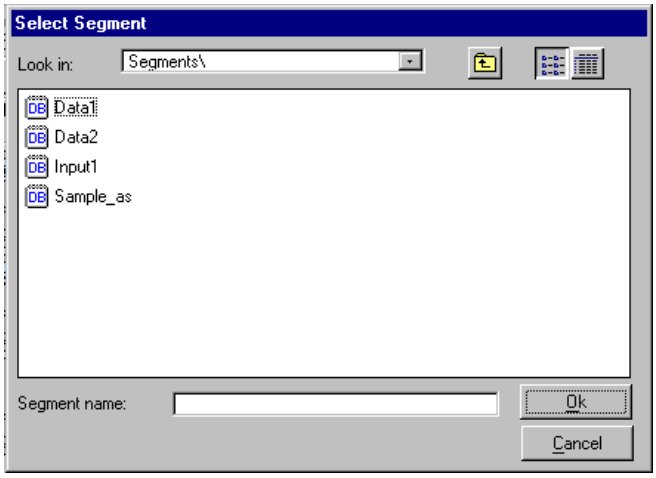


Figure 109 Select Segment Window

Per default, the window shows all accessible segments. You can change the directory to view only the global or local segment pool. Segments in the global segment pool can be accessed from all settings. Segments in the local segment pool can only be accessed from the current setting.

3 Select the segment you wish to insert into the block.

NOTE If you intend to use a stored segment, please note: Not every segment fits to every block.

If the length and/or width of a segment is smaller than the block length or port width, then an error message is displayed when the sequence is downloaded. It is necessary to edit the segment to match the block length and port width.

If the length or width of a segment is larger than the blocklength or port width, then only a portion of the segment will be generated or expected. This portion starts from trace 0 and vector 0 of the segment.

All traces which exceed the width of the port and all vectors which exceed the length of the block are ignored.

The lengths of the blocks have to be a multiple of the segment resolution which is a trade-off between the required system clock rate and the desired memory depth. For details see *"FMR and Segment Resolution" on page 31*.

See also *"Data Memory Usage" on page 39*, *"Segment Type Combinations" on page 40*, and *"How to Set the General System Frequency" on page 94*.

Additional restrictions apply for synchronization blocks if Automatic Bit Synchronization is used. See *"Block Length and Segment Length" on page 150*.

4 Confirm.

How to Replace a Segment by a Pseudo Segment

To select a pseudo segment in the Detail Mode Sequence Editor:

1 Select a segment and open the segment's context menu.

The lower part of the menu lists the available pseudo segments.

The available pseudo segments depend on the type of port (DUT input or output) and on the chosen kind of measurement.

They are listed in the table below.

Table 12 Default and Available Pseudo Segments

Kind of Measurement	DUT Data Input Port	DUT Data Output Port
Capture Data	Pause0 also available: Pause1	Pause also available: Acquire
Error Rate Measurement	Pause0 also available: Pause1	Pause also available: Expected 0 Expected 1

Table 12 Default and Available Pseudo Segments

Kind of Measurement	DUT Data Input Port	DUT Data Output Port
Compare and Acquire Around Error	Pause0 also available: Pause1	Pause also available: Expected 0 Expected 1 Don't Care
Compare and Capture	Pause0 also available: Pause1	Pause also available: Expected 0 Expected 1 Don't Care

Explanation

Pseudo segments for data generator channels:

- *Pause0*: Transmit logical zero (usually low level voltage) for the specified block length.
- *Pause1*: Transmit logical one for the specified block length.

Pseudo segments for data analyzer channels:

- *Pause*: Fall asleep for the specified block length.
- *Acquire*: Capture all DUT output data.
- *Expected 0*: Consider all non-zero data as errors.
- *Expected 1*: Consider all data that are not logical one as errors.
- *Don't Care*: Capture, but don't compare with expected data.

2 Choose from the menu.

How to Create and Change Loops

Loops can be specified in the columns at the right-hand side of the blocks. The right-most column is reserved for infinite loops.

A system equipped with the E4805A/B clock module provides 5 loop levels.

How to Create a Loop

To create a loop:

1 Click with the left mouse button into one of the columns.

Alternatively, you can also open the context menu of an empty column, choose *New Loop* and confirm the default loop properties.

This usually creates a one-block loop with 2 iterations. In the right-hand column, however, it creates an infinite loop.

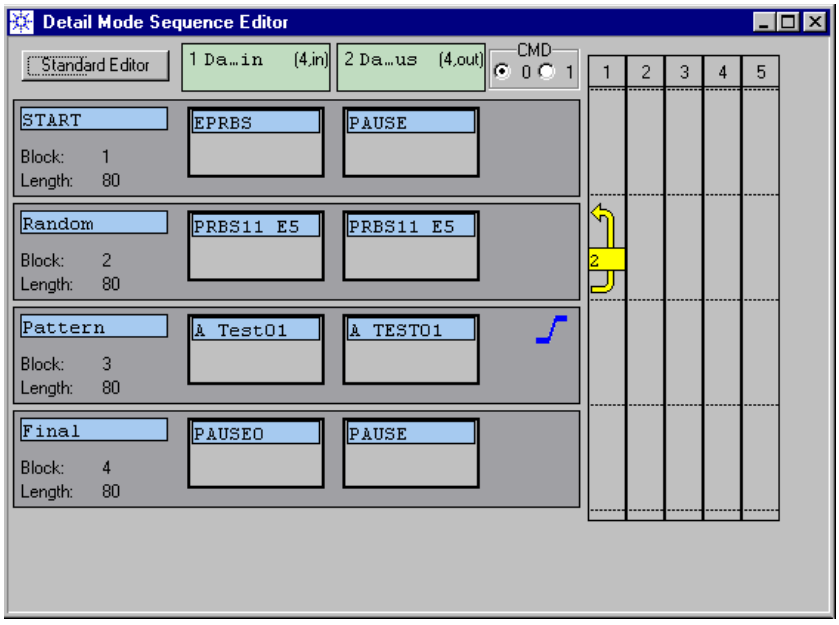


Figure 110 One-Block Loop

How to Change a Loop

Loops can be changed with the mouse and from the loop context menu.

How to Change a Loop With the Mouse

1 To change the length of a loop, click the upper or lower end of the loop and drag vertically.

2 To move a loop to a different level, click the loop and drag horizontally.

3 To change the number of repetitions of a loop, double-click the loop.

This opens the Loop Properties window. Type the desired number of repetitions and confirm.

How to Change Loop Characteristics With the Keyboard

1 Open the context menu of the loop.

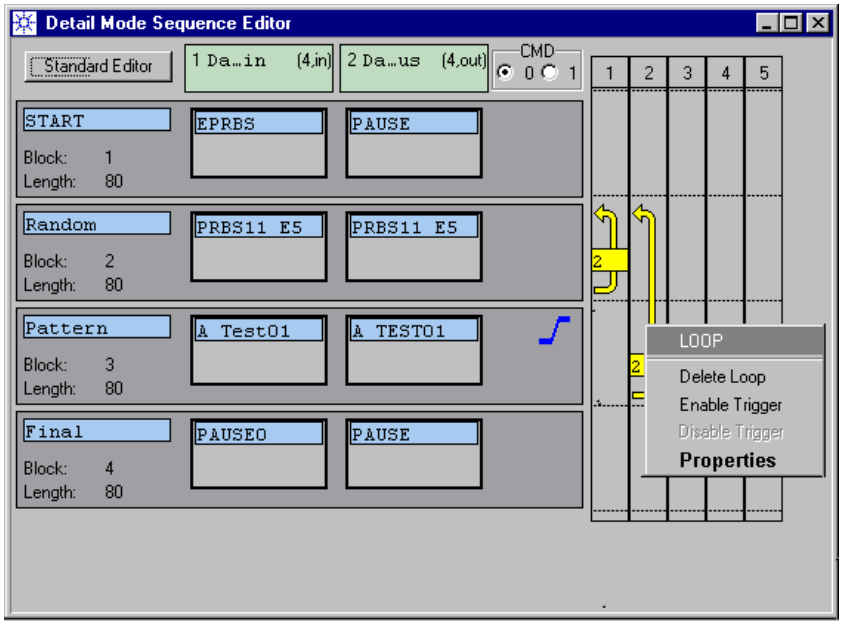


Figure 111 Loop Context Menu

2 Choose from the menu.

You can:

- Delete the loop.
- Enable/disable a trigger to be generated each time the loop is repeated.
- Select *Properties* to change the loop characteristics with the keyboard.

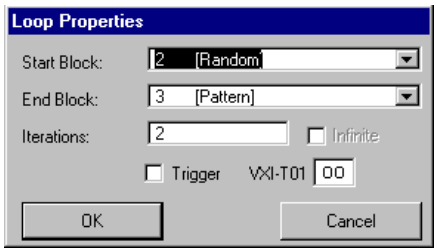


Figure 112 Loop Properties Window

You can change the start block, end block, number of repetitions, set a trigger, or set the VXI trigger lines (see also *"How to Change Block Properties" on page 155*).

How to Specify Events and Reactions Upon Events

The Agilent 81250 system is capable of reacting on events. The reaction can simply be a trigger pulse at the TRIGGER OUTPUT of the clock module, but also a change of the test sequence.

For general information see *"Event Handling Principles" on page 52*.

Events can be defined at any time. The reactions upon events can be specified if the Detail Mode Sequence Editor or the Data/Sequence Editor is active.

Examples can be found in *"How Do I Use Events?" on page 232*.

NOTE Event recognition is disabled when a synchronization block is executed.

Before You Start Using Events

You can define up to 5 events for immediate action and 5 events for deferred action.

Events for immediate action are serviced as soon as they occur. Events for deferred action are serviced at the end of the block. This is in contrast to a trigger associated with a block or loop. Those actions occur when the execution of the block is started.

Events for deferred action are prioritized. The event with the highest number has the highest priority.

What You Need to Find Out Before Using Events

1 Determine what you wish to achieve. Choices are:

- Issue a trigger pulse from the master clock module for starting/synchronizing an external instrument.
- Set the VXI trigger lines T0/T1 for triggering other VXI modules.
- Change the test sequence: Continue with another block, start all over, or terminate the test.

  If you wish to switch to another block, ensure that it is included in the overall sequence and labeled. You can only jump to labeled blocks.

2 Determine the release mechanism. Choices are:

- A command issued locally by clicking a button provided by the Detail Mode and Data/Sequence Editors or remotely.
- One or several bit combinations of the trigger pod (see also *"Trigger Pod" on page 22*).
- The status of the VXI trigger lines TX0 and TX1 which may be changed by an external VXI module.
- A bitstream error detected by one of the data generator/analyzer modules (not available in capture-only or BER mode).

3 Decide on the priorities.

- Do you need immediate reaction?
- In case of deferred reaction: Which event must be serviced under all circumstances? What is the minimum block length to guarantee reaction at the end of the block?

What You Need to Consider Before Using Events

There is of course a delay between the occurrence of an event and its recognition. There is also a delay between the recognition of an event and the reaction on that event.

Detection and Reaction Times

Detection of and reaction on events is controlled by an internal sequencer clock. The port-dependent frequency of that clock is:

Clk_p_freq = system clock frequency / segment resolution.

The maximum sequencer clock frequency is hence 41.67 MHz, corresponding to a period of 24 ns. If you had set a system clock rate of 100 MHz and a segment resolution of 4, the sequencer clock frequency would be 25 MHz, corresponding to a period of 40 ns.

A system equipped with E4832A, E4861A, and eventually E4841A modules has the following delays:

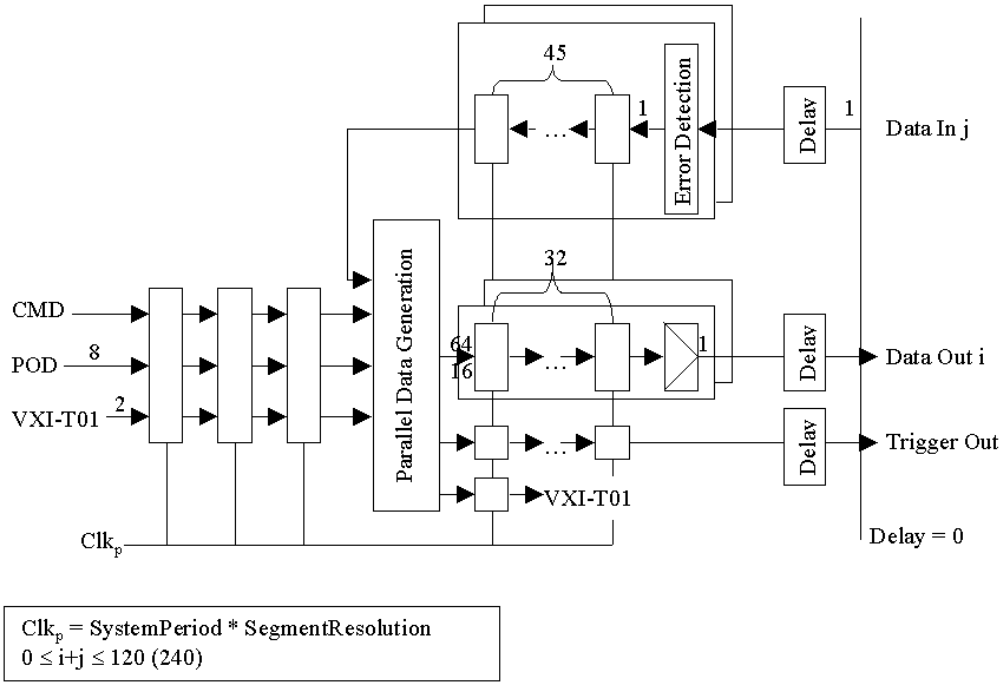


Figure 113 Delays Between Event Occurrence, Detection, and Reaction (System with E4832A, E4861A, eventually E4841A)

Explanation

Event commands as well as changes of trigger pod inputs or VXI trigger lines are processed after three periods of the sequencer clock Clk_p.

If bit stream errors (detected by one of the analyzer frontends) are included into an event definition, event processing takes 22 periods of the sequencer clock Clk_p at a system equipped with E4841A modules only, or 45 periods if the system includes E4832A or E4861A modules.

Once the 11-bit event pattern has been set up, the VXI trigger signals (if specified) are set after one period of the sequencer clock.

Depending on the data module and the current segment resolution, up to 16 or 64 bits can be processed during one period of the sequencer clock.

Launching a trigger at the master clock module's TRIGGER OUTPUT requires additional 32 periods of the sequencer clock Clk_p at a system equipped with E4832A or E4861A modules.

Switching to another sequence block needs 33 periods.

NOTE Especially if you wish to react on errors, this behavior has to be taken into account. For triggering on errors, you should use the Compare and Capture mode, because Compare and Acquire Around Error stops automatically some time after an error and may terminate the test before you get a reaction (see also *"Choosing the Kind of Measurement" on page 135*).

Minimum Block Length

The reactions on events are associated with blocks.

If you wish to react on an error by changing the sequence or setting a trigger and the respective block is **looped**, it must have a length of at least

79 * segment resolution

to ensure that the system can react during the next repetition of the block. 79 is the sum of 46 plus 33.

If the respective block is **not looped**, it must contain more data than compared. Proper triggering or sequence changing on errors during block execution is only ensured, if the block contains

79 * segment resolution

more generated vectors than are compared. Different segments may be needed for generated and expected data.

Example

If you have a system equipped with E4832A modules, a system clock period of 10 ns (100 MHz), a segment resolution of 4, and a block length of 400, then error events within the first 4 vectors can directly lead to a trigger signal or sequence change while or after the block is executed (minimum length for sequence change is 79 * 4 = 316).

As the sequencer clock rate is 100 MHz / 4 = 25 MHz, the delay between error recognition and sequence change is 79 * 4o ns = 3.16 ms. Later error events (resulting from vectors 5 to 400) are only processed if the block is repeated, because the reaction on events is associated with the block.

To react on all possible errors within the execution of 400 vectors, the block must have a length of at least 716 vectors, and the last 316 vectors must not be evaluated.

Triggers

If you intend to issue triggers, please note: Triggers have certain characteristics. The characteristics of the master clock module's TRIGGER OUTPUT are part of the global system parameters (see *"How to Set the Characteristics of the Trigger Output" on page 103*).

The width of the trigger at the clock module or the VXI bus corresponds to the period of the internal sequencer clock:

Clk_p_period = system clock period * segment resolution.

Example: If you had set a system clock rate of 100 MHz and a segment resolution of 4, the trigger width would be 40 ns.

How to Define Events

To specify events:

1 Open the context menu of a block and choose *Edit Events*.

Alternatively, you can also choose *Events* from the Edit menu and select *Edit*. Actually, event definitions are independent of the chosen block.

The Module Events window appears.

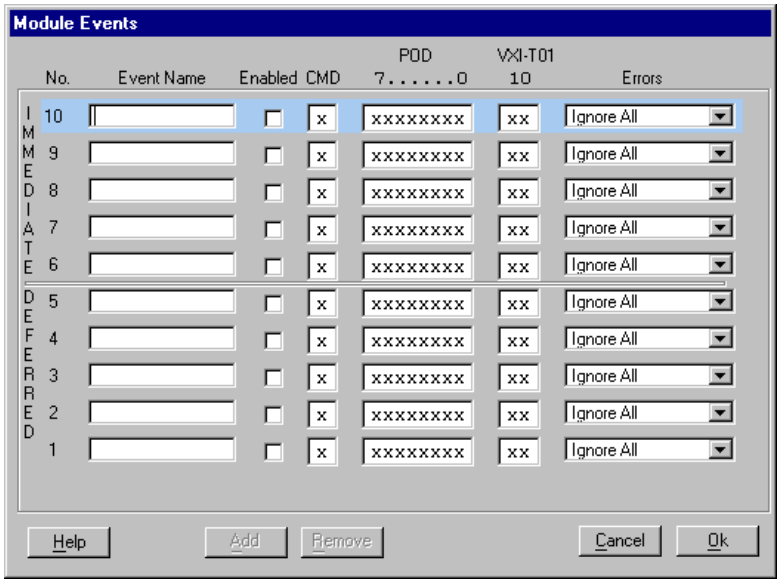


Figure 114 Module Events Window

2 Choose the type (deferred/immediate) and the priority of the event.

The types are explained in *"What is an Event?" on page 53*.

3 Enter the *Event Name*.

Every event requires its own, unique name.

4 Enable the event.

Click the corresponding checkbox or move with Tab and press the space bar.

5 Select and edit the details.

These items are logically ANDed. That means, the combination of whatever is activated and detected will cause an action.

- The *CMD* column refers to the manual or remote command that can cause an interrupt.

  Manual interrupts can be produced from the Detail Mode Sequence Editor and the Data/Sequence Editor windows by clicking *CMD0* or *CMD1*.

  Remote interrupts can be produced by the test program.

  Acceptable input values are x (don't care), 0, or 1.

- The *POD* column refers to the trigger pod (see also *"Trigger Pod" on page 22*). You can set the expected bits to x (don't care), 0, or 1.

- The *VXI* column refers to the VXI trigger lines T0/T1. Acceptable inputs are x (don't care), 01, 11, 10. Note: If you don't wish to react on their status, ensure they are set to xx.

  You can then set the VXI trigger lines as an answer to an event.

- The *Errors* column refers to the built-in analyzer channels. Open the pulldown menu and choose from the list.

A very simple event table which activates just the command-controlled events might look as shown below.

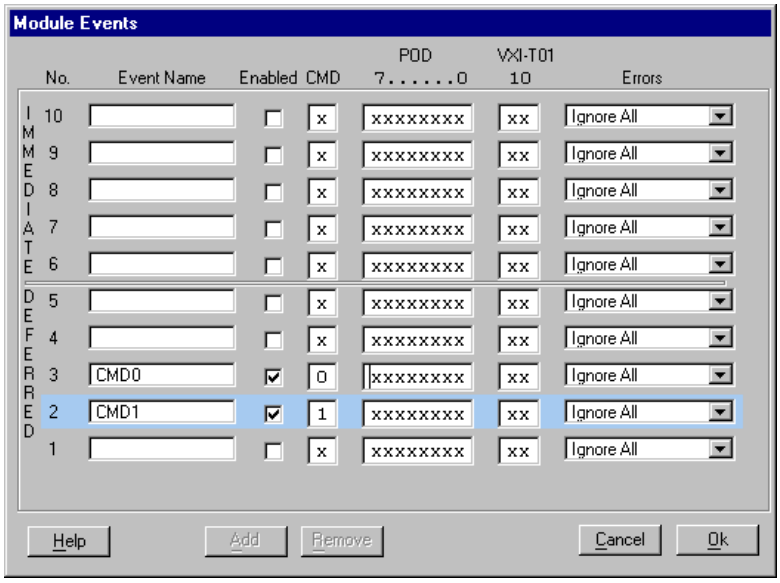


Figure 115 Simple Event Table

The event CMD0 occurs as soon as the *CMD* radio button of the Detail Mode Sequence Editor or the Data/Sequence Editor is identified as zero or after issuing the corresponding firmware command. Similarly, the event CMD1 occurs as soon as the *CMD* radio button of the Detail Mode Sequence Editor or the Data/Sequence Editor is identified as one.

Both are deferred events, which means that the system will react as soon as the presently executed sequence block has come to its end (assuming it is either repeated or long enough, see *"What You Need to Consider Before Using Events" on page 166*).

6 When you are done, click OK.

How to Specify the Reactions on Events

The reactions on events are block-related. You can specify individual reactions for each block of the sequence.

An example may be helpful to understand this procedure. It builds up on the event definition example shown above.

Stop and Go Example

We have set up the sequence shown below:

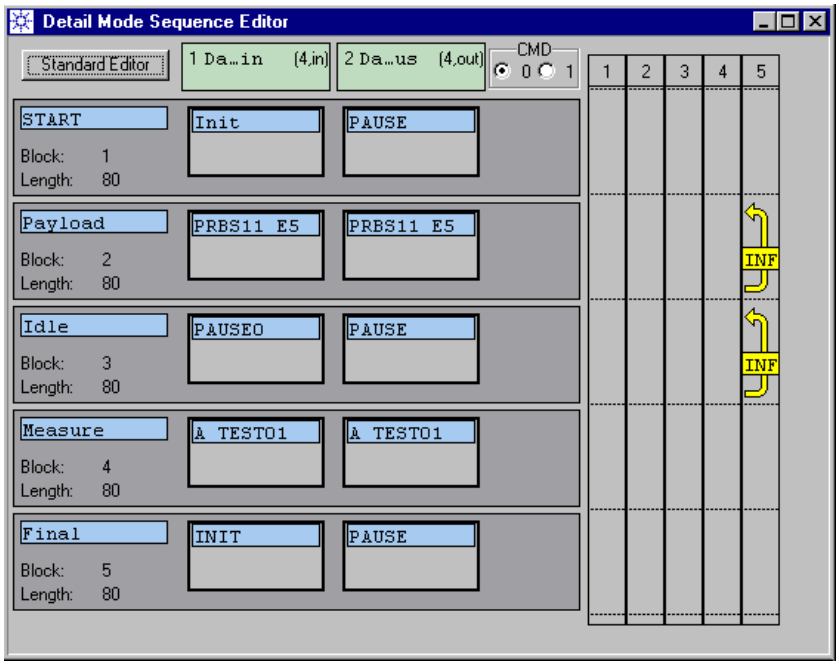


Figure 116 Sequence for the Stop and Go Example

We wish to run the Payload block until the result stabilizes. Then, upon a command, the test shall pause, so that we can examine the results.

A second command shall cause the test to continue with the Measure block and to finish.

We have defined the events CMD0 and CMD1 as shown in the example of *"How to Define Events" on page 168*.

How to Fill In the Branch Table

Each block has its own branch table.

1 Open the context menu of block 2.

2 Choose *View / Edit Branches*.

If no reactions have been specified so far, an empty branch table appears.

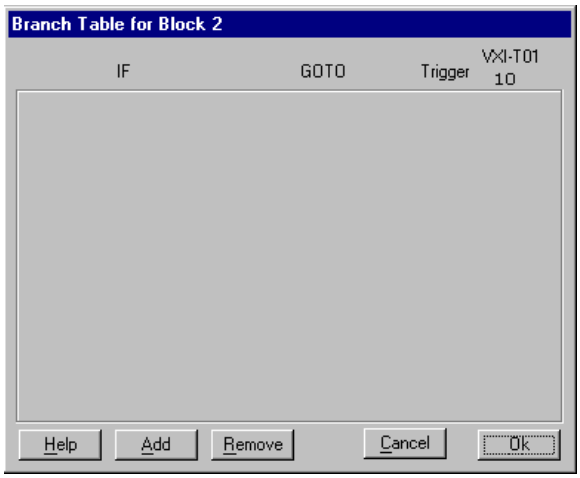


Figure 117 Empty Branch Table

3 Click the *Add* button.

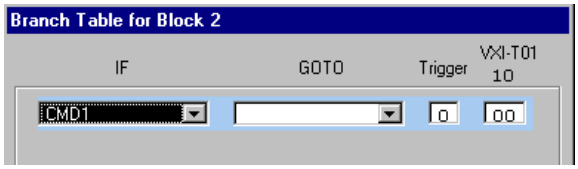


Figure 118 Branch Table Structure

Now you can see how the table is built up: If the specified event has occurred,

– go to a certain block of the sequence and/or

– output a trigger at the clock module and/or

– set the VXI trigger lines.

4 In the *IF* column, select one of the available events. The display starts with the lowest-priority event that has been defined (see also *"How to Define Events" on page 168*).

The pull-down menu offers also the DEFAULT event. This is a deferred event which occurs at the end of a block and must not be defined. It has a lower priority than any user-defined event and can be used to change the normal flow (which is either continuation or iteration).

5 In the *GOTO* column, select one of the available blocks.

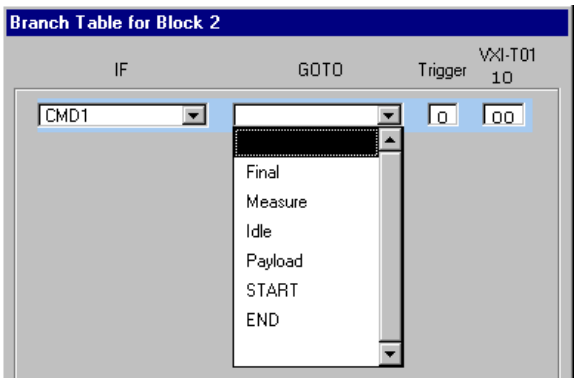


Figure 119 Branch Table – Block Selection

Note that the END block is an implicit block which is always available. It terminates the sequence.

The branch table of block 2 in our example is shown below:

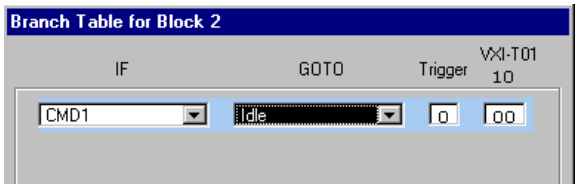


Figure 120 Branch Table – Block 2

As soon as CMD1 occurs, the block will not be repeated any more. The sequence will execute the Idle block.

6 Decide whether you wish to generate a trigger pulse. If you wish to activate the VXI triggers for controlling other VXI modules, ensure that you have not defined an event based on the status of the VXI trigger lines.

7 Click *OK*.

To complete the example, you have to repeat the steps 1 to 6 for block 3.

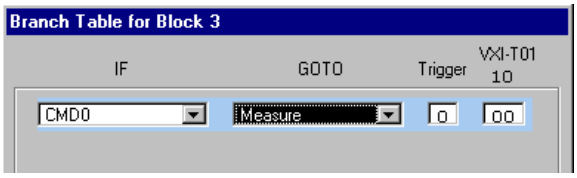


Figure 121 Branch Table – Block 3

As soon as CMD0 occurs, the infinite loop of the Idle block will be left, and the sequence will execute the Measure block

The sequence now shows that branches have been inserted.

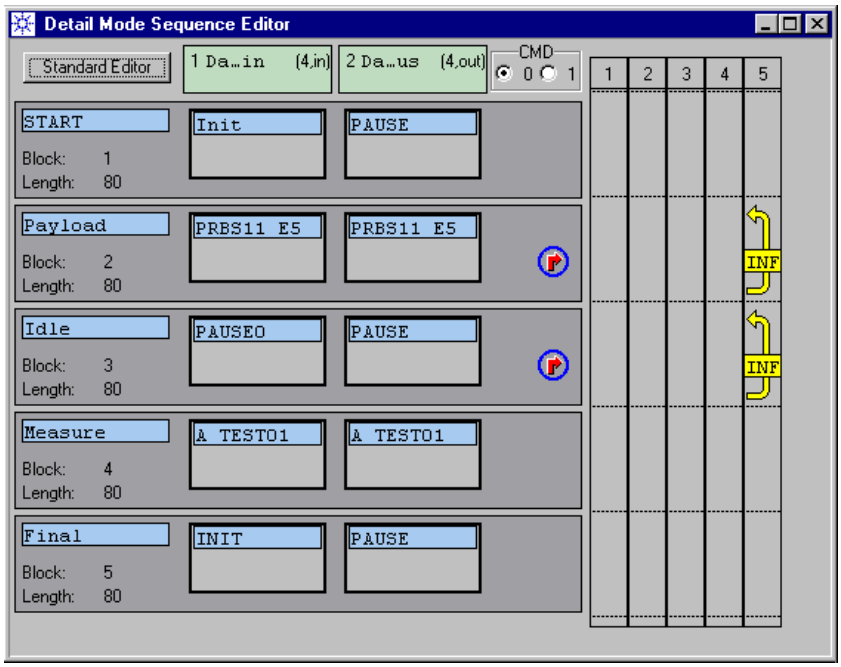


Figure 122 Sequence With Branches on Events

If you now run the test, you can terminate the Payload test by clicking the CMD 1 radio button. You can then resume the test by clicking the CMD 0 radio button. Block 4 and block 5 will be executed, and the test will finish.

More examples can be found in *"How Do I Use Events?" on page 232*.

Creating and Editing Segments

The stream of generated and expected data is defined by the data sequence. A sequence is built up of blocks. Each block references one data segment for each DUT data port.

The two different types of data segments are pseudo segments and real segments (see *"Data Segments" on page 38*).

Real segments can be created and modified manually. This chapter explains how this is done. See:

"How to Create a New Segment" on page 176

"How to Edit a Stored Segment" on page 187

How to Create a New Segment

Two types of real segments can be created and edited with the Segment Editor:

- PRBS/PRWS segments

  PRBS/PRWS segments contain pseudo random data in bit stream (PRBS) or word stream (PRWS) format. Pseudo random data is defined by the generating polynomial.

- Memory segments.

  Memory segments contain a user-defined data pattern.

NOTE Any new or modified segment needs to be saved on disk before it can be referenced in a block.

How to Start Creating a New Segment

To start the New Segment dialog:

1 Open the *File* menu and choose *New Segment*.

You start with the defaults as shown in the figure below.

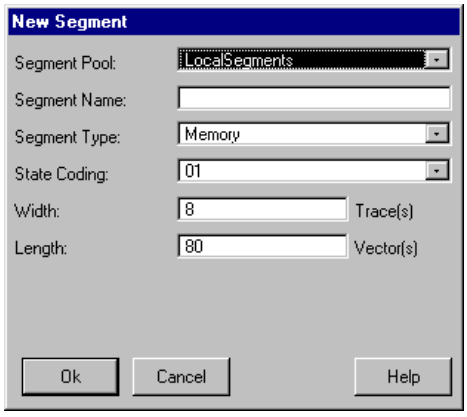


Figure 123 New Segment Window

If the Detail Mode Sequence Editor or the Data/Sequence Editor is displayed, you can also open the context menu of the segment you wish to replace and choose *New Segment*. If you enter the New Segment dialog this way, the block and port characteristics are already filled in.

2 Choose the *Segment Pool*.

Segments in the global segment pool can be accessed from all settings. Segments in the local segment pool can only be accessed from the current setting.

3 Enter or edit the *Segment Name*.

Enter a name that explains the contents or purpose of the segment.

4 Choose the *Segment Type*.

Type Memory means that a freely programmable pattern is stored in the database.

Type PRBS or PRWS means that an algorithm is used for generating a pseudo random bit or word sequence. After downloading to the channel memory pseudo random data of PRxS up to 2^{15}–1 is stored as a pattern. The large 2^{23}–1 and 2^{31}–1 PRxS are generated by the module at runtime.

NOTE PRWS segments are mainly used for testing multiplexers/demultiplexers. The first terminal of the DUT input port (counted from top to bottom as shown in the Connection Editor) gets the first state of the generated random sequence, followed by the next lower pin, and so on. If you had set up a 4-bit port and bit count starts with one, the first terminal would receive bit 1, 5, 9, and so on.

This fashion of sending the data to the channels requires that the connectors used are in the same module. If several modules are addressed, these modules have to be in adjacent slots.

For details see also *"Appendix B: PRBS/PRWS Data Segments" on page 251*.

How to Create a Memory Segment

Start the New Segment dialog (see *"How to Start Creating a New Segment" on page 176*).

Once you have decided to create a new memory segment and where to store it, fill in the remaining fields:

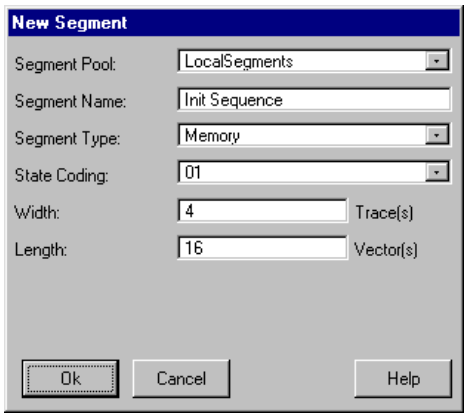


Figure 124 New Segment Window for Creating a Memory Segment

1 Specify the *State Coding*. Choices are 01 or 0x1.

The state coding 01 specifies that every bit of the segment occupies one bit in memory. This is adequate for all data segments to be downloaded to generator channels.

The state coding 0x1 can be used for expected data. This coding enables the x-character used to denote don't care bits. State coding 0x1 specifies that each bit of the segment occupies two bits in memory.

2 Set the *Width* and *Length* of the new data segment.

The width represents the number of pins included in a port and defines the number of traces.

The length is the pattern length and, thus, the number of vectors.

NOTE The length may exceed but must not be shorter than the length of the block where the segment is going to be inserted.

3 Click *OK*.

This opens the Segment Editor for the newly specified memory segment.

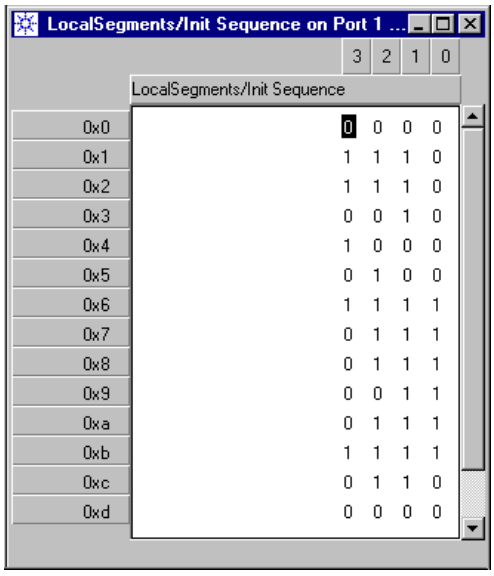


Figure 125 Segment Editor Window

The window shows the specified vectors (horizontal lines) and traces (vertical rows).

Characteristics of the Segment Editor Window

The Segment Editor window has three active areas:

- An area for vector operations
- An area for trace operations
- The data edit area

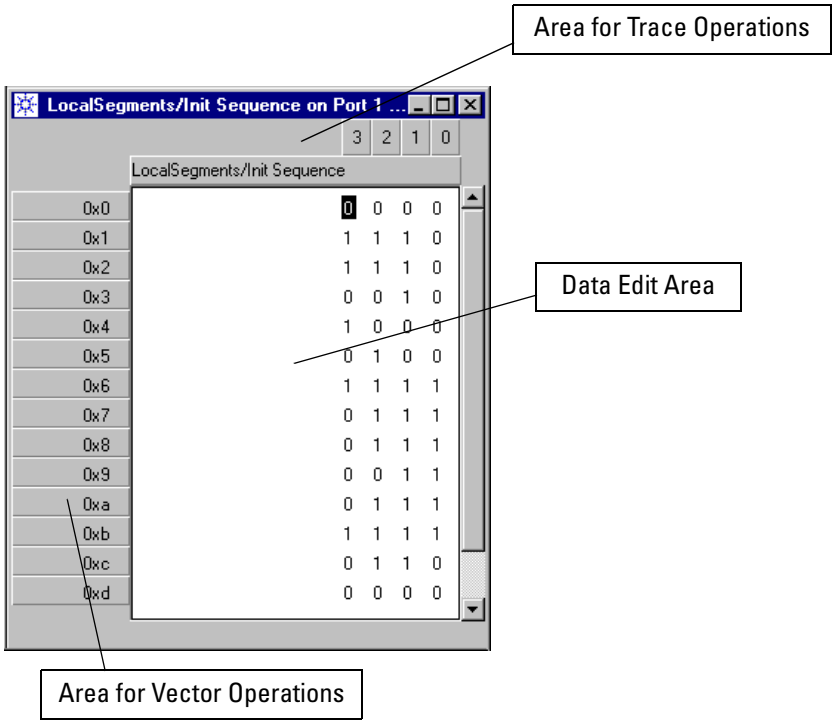


Figure 126 Segment Editor Window Areas

Each of these areas has its own context menu, indicated by the cursor changing its shape when placed over the areas.

How to Use the Segment Editor's Vector Operations Area

Clicking a vector address highlights that vector.

Dragging the cursor across several vector addresses highlights a block of vectors.

The context menu is opened by clicking with the right mouse button. It provides the following options:

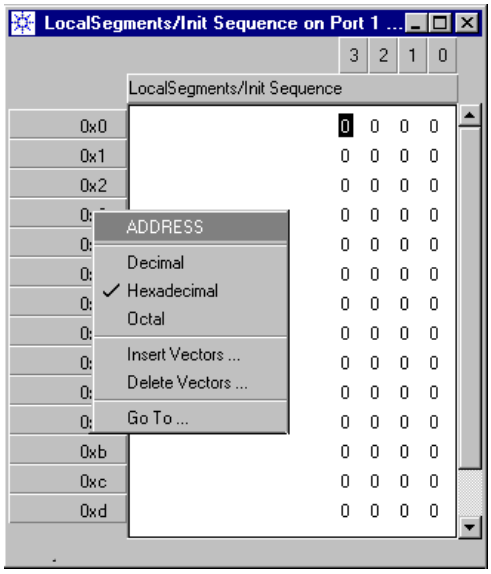


Figure 127 Segment Editor – Context Menu for Vector Operations

The available options are:

- Change the address display format (decimal, hex, or octal)
- Insert or delete highlighted vectors or traces in the table
- Jump to a certain vector address.

How to Use the Segment Editor's Trace Operations Area

Clicking a trace number highlights that trace.

Dragging the cursor across several trace numbers highlights a block of traces.

The context menu is opened by clicking with the right mouse button. It provides the following options:

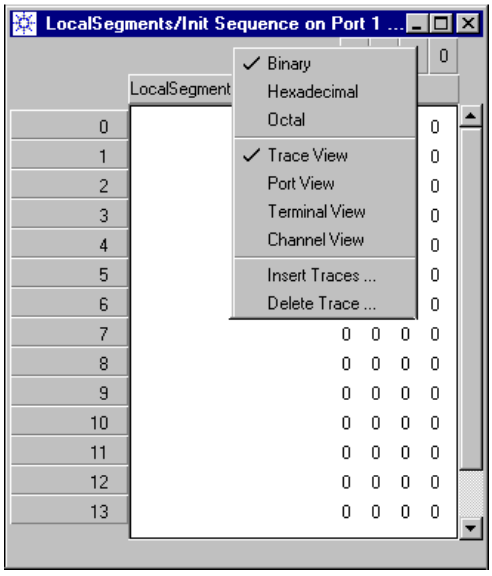


Figure 128 Segment Editor – Context Menu for Trace Operations

The available options are:

- Change the data display format.

  In binary mode, each trace has its own column. In hexadecimal mode, four traces are combined in one column (range 0 to F_{hex}). In octal mode, three traces are combined in one column (range 0 to 7_{oct}).

- Display the port, defined terminals or connected channels instead of trace numbers (available if you have started the Segment Editor by clicking a segment in the Sequence Editor).

- Insert new or delete highlighted traces in the table.

How to Use the Segment Editor's Data Edit Area

Clicking a bit highlights that bit. With the spacebar, you can switch from zero to one and vice versa.

Dragging the cursor across several bits highlights a block.

The context menu initially provides the following options:

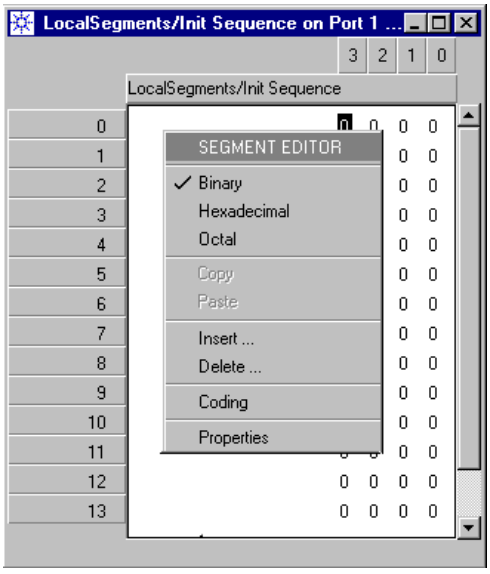


Figure 129 Segment Editor – Initial Context Menu for Editing Data

The available options are:

- Change the data display format (decimal, octal, or hex)
- Insert or delete highlighted vectors or traces
- Change the state coding
- Review the segment's setup information, but don't change it

If you have highlighted a block of data, the context menu provides additional options.

Figure 130 Segment Editor – Extended Context Menu for Editing Data

The additional options are:

- Copy the block to the clipboard (it can then be pasted somewhere else).
- Set the whole block to 1, or 0, or don't care (the latter only if the state coding is 0x1).
- Mirror the block contents horizontally or vertically.
- Invert the bits contained in the block.

NOTE You can also paste captured data from the Error State Display. For details see *"How to Transfer Captured Data Into a Segment" on page 206*.

Segment Editor Shortcuts and Defaults

The cursor movement is from left to right and from top to down. If a block is highlighted, the cursor only moves within that block.

The Page Up and Page Down keys allow to scroll vertically through the data segment. Step size is the number of lines actually visible in the editor window.

The Home key moves the cursor to the vector number 0x0 in hexadecimal (equals decimal 0) and the highest trace number.

The End key moves the cursor to the highest vector number and trace number 0.

The Insert key can be used to insert vectors (rows) or traces (columns) in the table.

The Delete key can be used to delete highlighted vectors or traces.

The Scroll Bar at the right side of the editor window helps to position the cursor in the middle portion of a large data segment.

To deselect a selection, press the Esc key or click outside the highlighted block.

To highlight the whole segment, click the segment label.

How to Create a PRBS/PRWS Segment

Start the New Segment dialog (see *"How to Start Creating a New Segment" on page 176*). Once you have decided to create a PRBS or PRWS segment, the window changes:

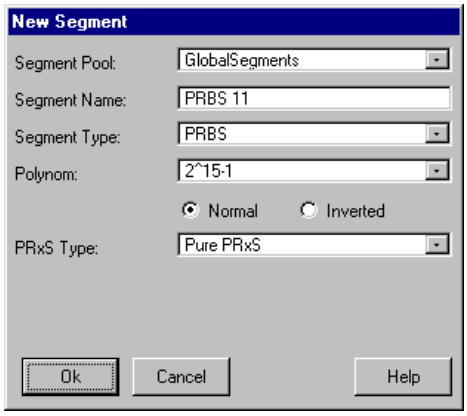


Figure 131 New PRBS/PRWS Segment Window

1 Choose one of the available *Polynomials*.

The polynomial defines the complexity and length of the pseudo random data segment. A 2^{15}–1 PRBS/PRWS, for example, uses 32767 words in memory. The available PRBS/PRWS lengths are 2^5-1 through 2^{15}-1, 2^{23}-1, and 2^{31}-1.

2 Select *Normal* or *Inverted*.

If inverted is selected, the PRBS/PRWS is output in inverse mode.

3 Choose the *PRBS/PRWS Type*.

The options are:

– Pure PRxS

– Errored PRxS

– Variable Mark Density

– Extended Zeros/Ones

For the PRBS/PRWS lengths 2^{23}-1 and 2^{31}-1 only pure PRxS is supported.

If you have chosen an non-pure PRxS, an additional parameter needs to be set. For details see *"Appendix B: PRBS/PRWS Data Segments" on page 251*.

4 Click *OK* to finish creating the PRBS/PRWS segment.

How to Save a New or Changed Segment

To save a new or changed segment:

1 Open the *File* menu

2 Select *Save Segment* to save the segment under its original name.

Alternatively, you can also click *Save Segment As* to save the segment under a new name.

See also *"Save Segment" on page 75* and *"Save Segment As" on page 76*.

How to Edit a Stored Segment

To view or edit a stored segment, you have to select that segment from the pool of segments.

How to Select a Segment

1 Click the Segment Editor icon.

Alternatively, you can also choose *Open Segment* from the *File* menu.

The Open Segment window appears:

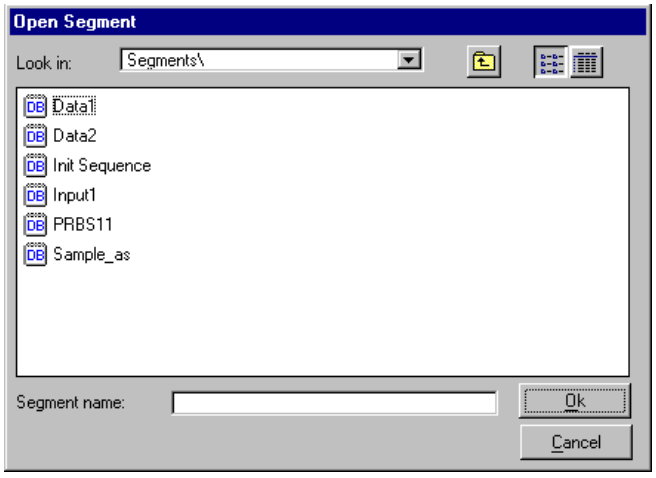


Figure 132 Open Segment Window

2 Select the Segment you wish to view or edit.

3 Click OK.

This opens the Segment Editor.

NOTE You can also open the Segment Editor directly by double-clicking a segment in the Detail Mode Sequence Editor or Data/Sequence Editor.

How to Edit a Memory Segment

After selecting a memory segment, the Segment Editor shows the data pattern:

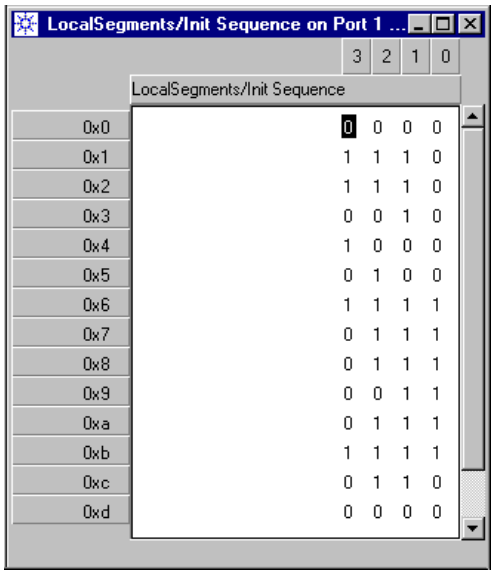


Figure 133 Segment Editor Window

For details see *"Characteristics of the Segment Editor Window" on page 180*.

How to Edit a PRBS/PRWS Segment

After selecting a PRBS/PRWS segment, the Segment Properties window shows the specification:

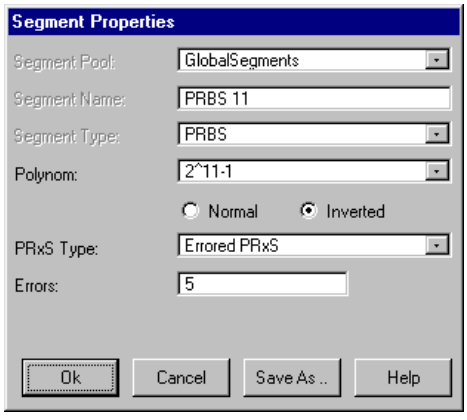


Figure 134 PRBS/PRWS Segment Properties Window

For details see *"How to Create a PRBS/PRWS Segment" on page 185*.

Using the Data/Sequence Editor

The Data/Sequence Editor combines the functions of the Detail Mode Sequence Editor and the Segment Editor.

The Data/Sequence Editor can be used to:

- Inspect all details of the test sequence including data patterns
- Change the test sequence
- Change block characteristics
- Inspect and change the segments contained in the blocks

The Data/Sequence Editor is particularly useful on a system equipped with a high resolution video screen because it eliminates the need for switching between the two standard editors.

The procedures for editing data are basically the same as incorporated in the Sequence Editor and the Segment Editor.

This chapter comprises the sections:

"How to Start the Data/Sequence Editor" on page 190

"How to Customize the Data/Sequence Display" on page 191

"How to Change the Sequence or Edit Segments" on page 195

How to Start the Data/Sequence Editor

To start the Data/Sequence Editor:

1 Select *Data/Sequence Editor* from the *Go* menu.

Contents of the Data/Sequence Editor Window

The Data/Sequence Editor identifies:

- The DUT i/o ports that have been set up
- The terminals that have been set up within the ports
- The blocks that form the test sequence
- The block markers that indicate generated triggers, analyzer synchronization, and sequence changes on events
- The loops that repeat single or groups of blocks
- The segments that have been included in the blocks
- The data patterns stored in the segments

All this is shown in one window.

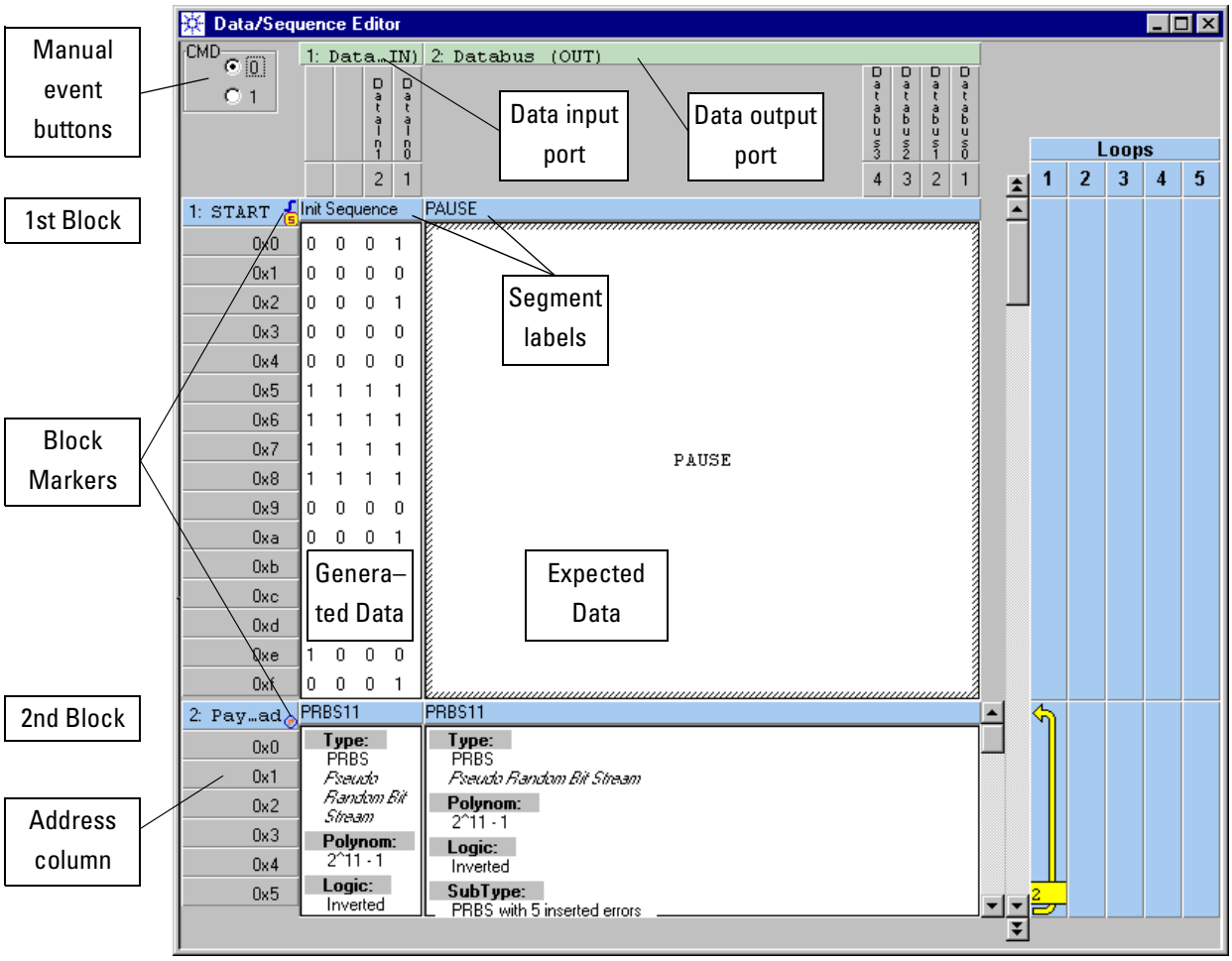


Figure 135 Data/Sequence Editor Window

NOTE Note that the Data/Sequence Editor provides vertical scroll bars for each block and an overall scroll bar for the sequence.

How to Customize the Data/Sequence Display

You can adapt the display to your preferences by changing:

- The width of the columns

- The height of the blocks
- The address display format
- The labeling of the displayed traces

How to Change the Width of the Columns

To view more or less columns (ports):

1 Move the cursor onto the vertical line that marks the column border.
The cursor changes its shape.

2 Hold the mouse button depressed and drag the border line horizontally.

How to Change the Height of a Block

To view more or less blocks:

1 Move the cursor onto the horizontal line that marks the lower block border.

The cursor changes its shape.

2 Hold the mouse button depressed and drag the border line vertically.

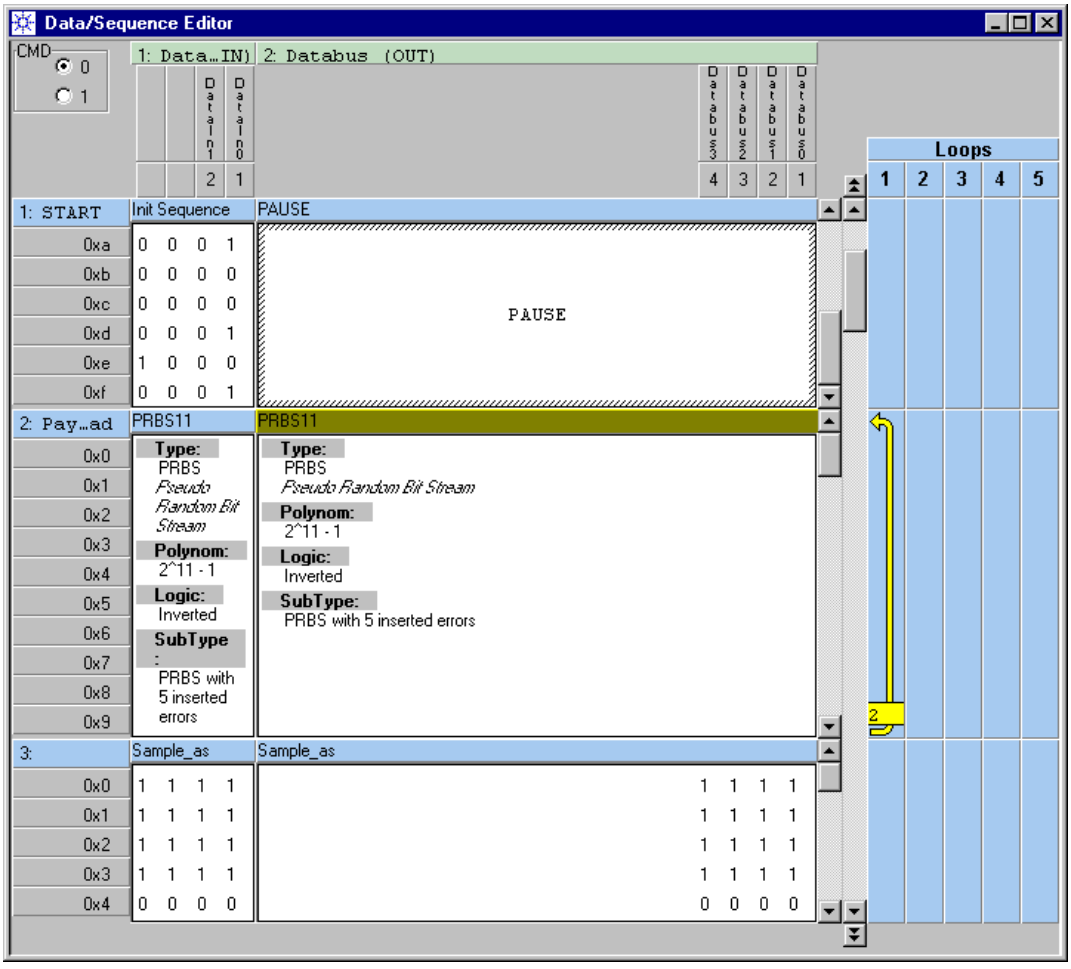


Figure 136 Customized Data/Sequence Editor Window

How to Change the Format of Displayed Addresses

An important difference between the Data/Sequence Editor and the Segment Editor is the meaning of the vector address column at the left-hand side. While the Segment Editor always displays the actual bit vector address in the segment, the Data Sequence Editor displays the clock cycle number at a certain point of time.

If a port uses clock frequency multiplier 1, one vector will be displayed per row (i.e. per cycle number). If another port uses frequency multiplier 4, then four rows of the data segment will occur per cycle for this port. In other words, this port will receive/send four bits for every terminal, while the first one only receives/sends one per terminal.

To change the bit vector/clock cycle number format:

1 Open the context menu of the address column by clicking on it with the right mouse button.

2 Choose from the menu one of the following options:

Decimal, *Hexadecimal*, or *Octal*.

Note, that this menu can also be used to jump to any address within the current block by choosing the *Go to* option.

How to Change the Labels of Displayed Traces

To change the view of traces:

1 Open the context menu of the port/terminal display area by clicking on it with the right mouse button.

2 Choose from the menu.

Choices are: *Trace View*, *Port View*, *Terminal View*, *Channel View* (see *"Data Format" on page 83*).

Note, that this menu can also be used to insert or delete highlighted traces.

How to Change the Sequence or Edit Segments

The Data/Sequence Editor combines all the functions of the

- Sequence Editor
- Segment Editor

That means, it does not only provide the same capabilities, it also works the same way. In fact, the Data/Sequence Editor just invokes procedures that are already known from the other two editors.

How to Change the Sequence Characteristics

To change the sequence:

1 Open the context menu of a block label by clicking on it with the right mouse button.

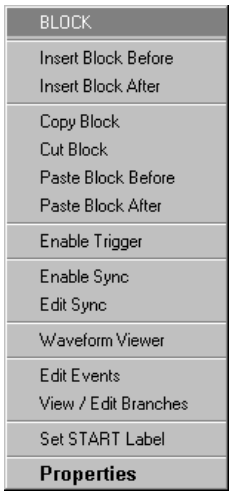


Figure 137 Block Context Menu

You have all the options the Sequence Editor provides.

2 Choose the required action from the menu.

For details please refer to *"How to Add, Move or Delete Blocks" on page 154*.

If you wish to change the loops in your sequence, please refer to *"How to Create and Change Loops" on page 162*.

How to Replace a Segment

To exchange a segment:

1 Open the context menu of a segment label by clicking on it with the right mouse button.

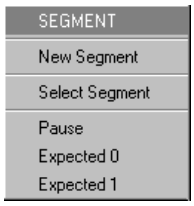


Figure 138 Segment Context Menu

You have all the options the Sequence Editor provides. They depend on the selected type of measurement and on the type of port – data input port or data output port.

2 Choose from the menu.

For details please refer to *"How to Replace the Current Segment" on page 158*.

How to Edit the Contents of a Segment

For changing the contents of a segment, you have all the options the Segment Editor provides.

You can change individual bits or highlight rows, columns, or selections and change their contents using the context menu.

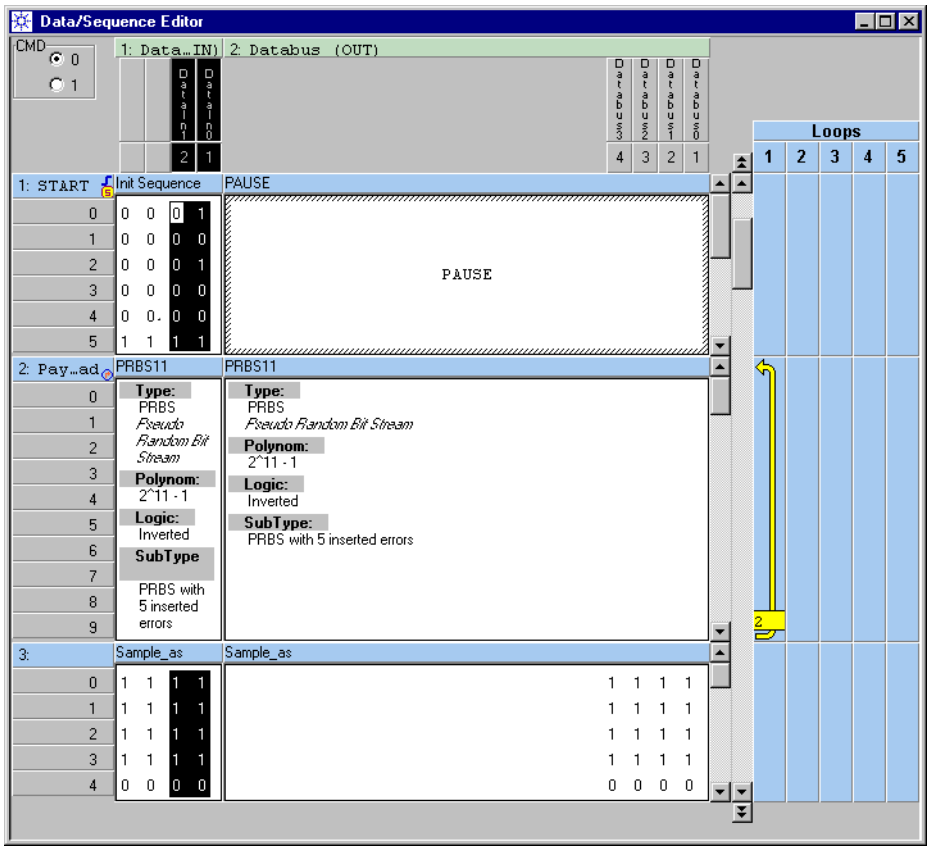


Figure 139 Data/Sequence Editor – Highlighted Column

For details please refer to *"How to Use the Segment Editor's Data Edit Area" on page 183*.

Running the Test

Once the Sequence is complete, you are ready to run the test.

Device Tests can be started and stopped from the user interface. They can also be started and stopped by an external signal applied to the EXT INPUT connector of the master clock module.

NOTE Before starting a test, it is recommended to check the parameter settings of the ports and channels.

To be successful, generated signals must fit to the data formats, voltages and impedances of the DUT. Analyzer sampling point delays must take the signal traveling and processing time of the DUT into account.

This chapter informs you about:

"How to Download the Test Sequence" on page 200

"How to View BER Test Results" on page 200

"How to Start/Stop the Test" on page 201

How to Download the Test Sequence

To download a sequence to the modules:

1 Click the Prepare button of the tool bar.

Downloading ensures that the test sequence is formally correct and can be executed.

NOTE Downloading is especially important, if you have changed the type of measurement, because pseudo segments like Acquire, Expected 0 and so on apply only to certain tests (see *"How to Replace a Segment by a Pseudo Segment" on page 160*).

Downloading also prepares the Agilent 81250 system for immediate start on a trigger event.

The sequence is also downloaded to the modules when the Run button is pressed. But downloading a complex sequence can take some time. In the meantime there is no output signal generated, nor any input signal captured.

How to View BER Test Results

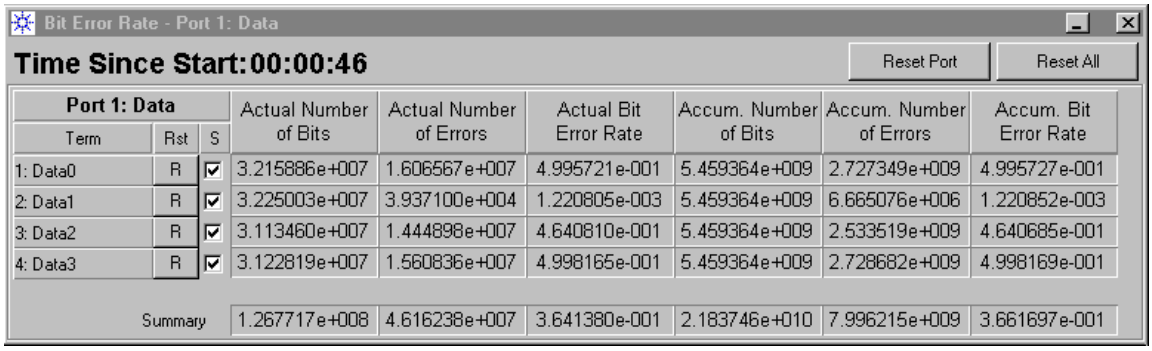
If the test has been set up for measuring the bit error rate:

1 Click the Bit Error Rate Display icon in the tool bar.

This opens the Bit Error Rate window.

2 Drag at the right- or left-hand border to view all the columns.

Once the test is running, the window is continually updated. It is therefore recommended to open this window before starting the test.



Port 1: Data			Actual Number of Bits	Actual Number of Errors	Actual Bit Error Rate	Accum. Number of Bits	Accum. Number of Errors	Accum. Bit Error Rate
Term	Rst	S						
1: Data0	R	☑	3.215886e+007	1.606567e+007	4.995721e-001	5.459364e+009	2.727349e+009	4.995727e-001
2: Data1	R	☑	3.225003e+007	3.937100e+004	1.220805e-003	5.459364e+009	6.665076e+006	1.220852e-003
3: Data2	R	☑	3.113460e+007	1.444898e+007	4.640810e-001	5.459364e+009	2.533519e+009	4.640685e-001
4: Data3	R	☑	3.122819e+007	1.560836e+007	4.998165e-001	5.459364e+009	2.728682e+009	4.998169e-001
Summary			1.267717e+008	4.616238e+007	3.641380e-001	2.183746e+010	7.996215e+009	3.661697e-001

Figure 140 Bit Error Rate Display

The resulting BER is shown as actual and accumulated values per terminal and port. The elapsed time since start of the measurement is also displayed.

All counters can be reset at any time, either individually per terminal (*R* buttons in the *Rst* column), or per port (*Reset Port* button) or all at once (*Reset All* button).

There is a summary line at the bottom. By clicking the marker in the *S* column, terminals can be excluded from or included in the summary line.

TIP The sequence of the columns can be customized. To move a column to a different position, click the column header with the left mouse button and drag the column horizontally to the desired position.

How to Start/Stop the Test

To start the test:

1 Ensure that the frontends are connected to the DUT.

The Connectors On/Off button can be used to disconnect all frontends (by switching relays inside the frontends) and to re-establish the previously specified connections.

2 Click the Run button.

If the test has been set up to be controlled by an external start trigger, the user interface will display HALTED and the system will wait for that trigger. If not, it starts immediately.

The test will run until the test sequence is completely executed or the capture memory is full or, if it is controlled by an external stop trigger, until the trigger is set—whichever comes first.

If the test sequence includes an infinite loop, stop the test by clicking the Stop button.

Viewing Generated and Captured Data

After running one of the tests

- *Capture Data*
- *Compare and Acquire around Error*
- *Compare and Capture*

you can review the captured data.

After running one of the tests *Compare and Acquire around Error* or *Compare and Capture* you can also investigate errors, the pattern around errors, and generated data.

The results of a bit error rate measurement are displayed in the Bit Error Rate window (see *"How to View BER Test Results" on page 200*).

This chapter informs you about:

"How to View Captured Test Results" on page 204

"How to View Waveforms" on page 209

How to View Captured Test Results

Captured data as well as errors can be visually checked in the Error State Display.

How to Start the Error State Display

To open the Error State Display:

1 Click the Error State Display icon of the tool bar.

Alternatively, you can also open the *View* menu and choose *Result Displays*.

The Error State Display identifies the DUT output port and shows the memory contents of the analyzer channels.

How to Operate the Error State Display

The Error State Display has three display modes. It can show:

- Captured Data: Shows what has been captured.
- Compared Data: Shows captured data where errors are highlighted.
- Error Data: Shows errors only.

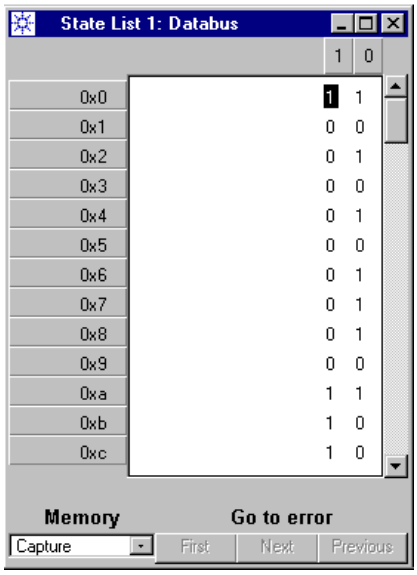


Figure 141 Error State Display in Capture Mode

The window has two active areas with context menus—the address column and the data display box.

In the address column, you can change the address display format (choices are *Decimal*, *Hexadecimal*, or *Octal*) or choose the *Go to* option to specify the start address of the display.

In the data display box, you can change the data display format. Choices are *Binary*, *Hexadecimal*, or *Octal*.

In binary mode, each trace has its own column. In hexadecimal mode, four traces are combined in one column (range 0 to F_{hex}). In octal mode, three traces are combined in one column (range 0 to 7_{oct}).

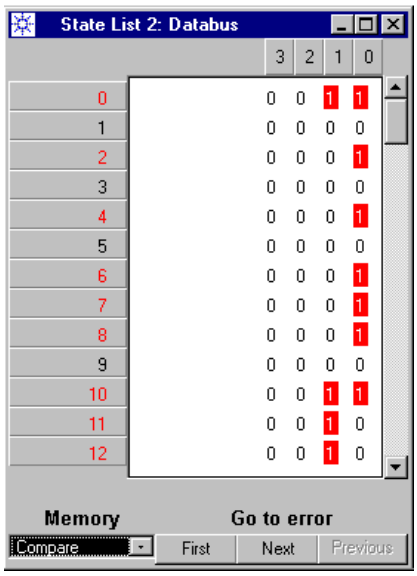


Figure 142 Error State Display in Compare Mode

In Compare mode, the window provides three Go-to-Error buttons to move quickly from one error to the next or previous.

How to Transfer Captured Data Into a Segment

Captured data can be saved in a data segment.

This makes it possible to use the response of a device as a reference for future devices of the same kind. If the segment with the captured data is used to specify the expected data for tests to come, the system precisely measures all deviations from the gold standard.

There are two ways to transfer captured data into a segment. You can:

- Save the captured data as a new segment
- Copy the captured data to the clipboard and then paste that data into a segment

How to Save Captured Data as a New Segment

To convert captured data to a new segment:

1 When the Error State Display is active, open the *File* menu and choose *Save Segment As*.

2 Enter the new segment's file name.

3 Click OK.

NOTE Before using the segment for interpreting received data, check and, if necessary, change the state coding (see *"How to Create a Memory Segment" on page 178*).

How to Copy Captured Data Into a Segment

To copy captured data into a new or existing segment:

1 Put the Error State Display in Capture mode.

2 Drag the cursor across the trace numbers.

This highlights all the captured data. You can of course also highlight a data section.

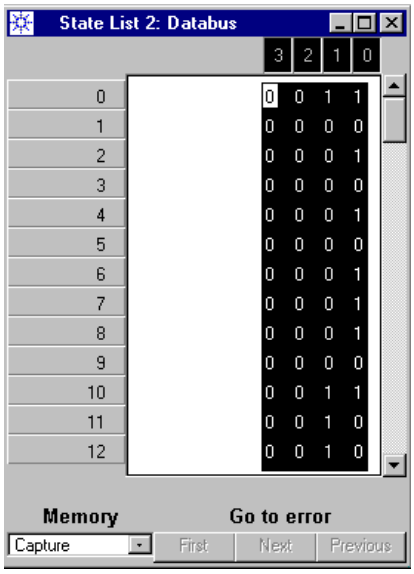


Figure 143 Highlighted Captured Data

3 Open the context menu and choose *Copy*.

This copies the data to the clipboard.

4 Create a new memory segment (see *"How to Create a New Segment" on page 176*).

When creating the segment, ensure that it is long and wide enough to hold the pattern you wish to include.

5 In the Segment Editor, highlight the traces that shall get the data pattern.

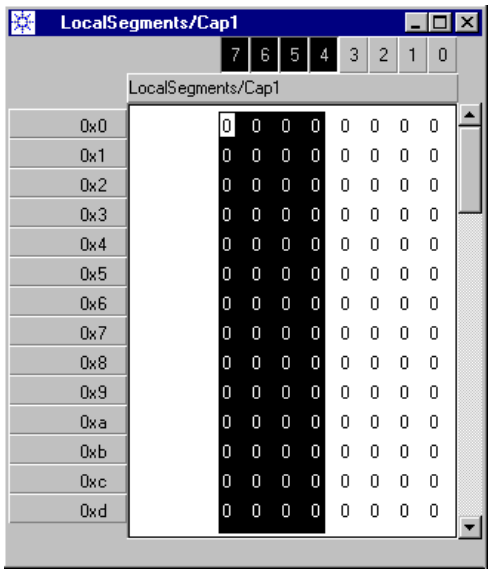


Figure 144 Empty Segment

6 Open the context menu and choose *Paste*.

The result is shown below:

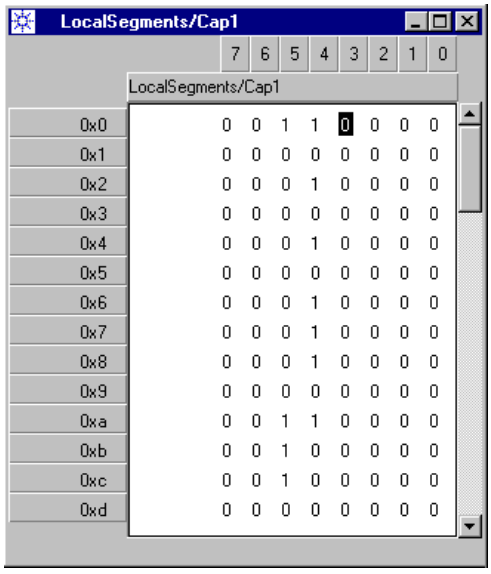


Figure 145 New Segment

Copied data that does not fit into the segment is ignored.

7 Save the segment.

How to View Waveforms

Generated, expected and captured data can be displayed in graphical form with the Waveform Viewer.

How to Start the Waveform Viewer

To open the Waveform Viewer:

1 Click the Waveform Viewer icon of the tool bar.

Alternatively, you can also open the View menu and choose *Result Displays*.

The Waveform Viewer identifies the test sequence and the ports:

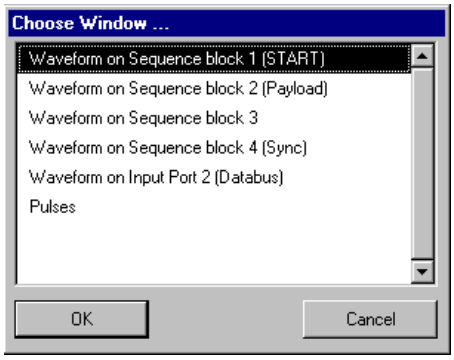


Figure 146 Waveform Viewer Selection Window

If you select a data block of the test sequence or a pulse port, generated and expected data will be displayed in **time mode**. The available resolution is the segment resolution, the unit is nanoseconds (ns). This enables you to check the delays that have been set up. Note that PRBS/PRWS data cannot be displayed in time mode.

If you select a DUT output port (= analyzer input), the data will be displayed in **sample mode**. This corresponds to the way the data has been acquired. The total number of samples is the number of captured vectors times the maximum factor of the FMR (see *"FMR and Segment Resolution" on page 31*).

2 Choose from the menu.

Description of the Waveform Viewer Display

The Waveform Viewer comes up with a default configuration which can be changed at will.

The figure below shows an example of a block which includes one data input port (DataIn) and one data output port (Databus).

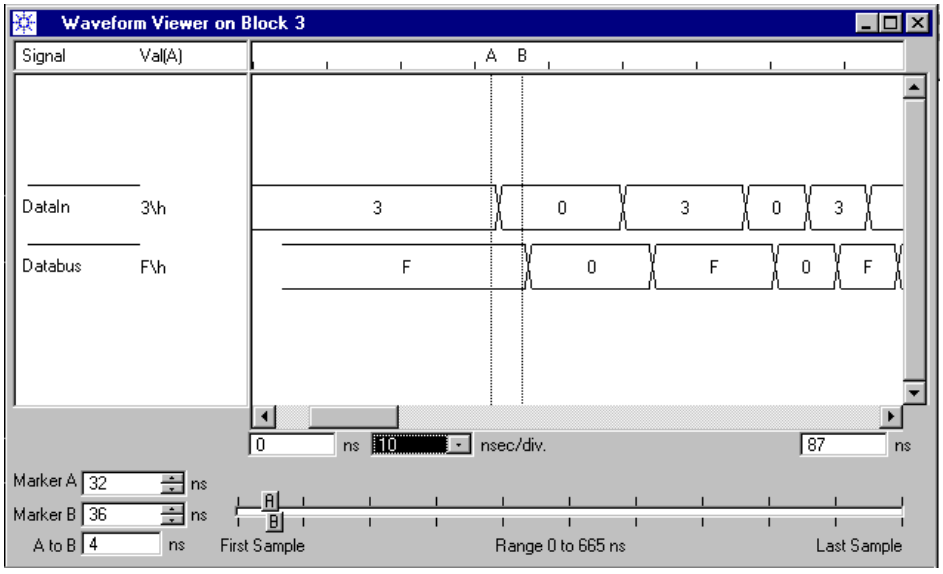


Figure 147 Waveform Viewer in Time Mode

Two traces show the code transitions and the generated and expected codes in hexadecimal format.

The display provides two markers, A and B. Their current position and distance is indicated in the lower left-hand corner. They can be moved with the verniers or by dragging their handles along the ruler.

The column *Val(A)* shows the codes at the position of marker A. The literal \h indicates that these are hex codes.

The current resolution is 10 ns/div but can be changed.

How to Operate the Waveform Viewer

The context menu provides the following options:

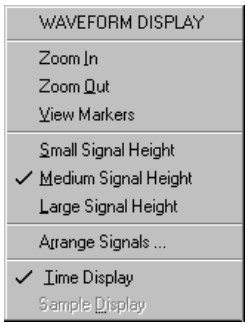


Figure 148 Waveform Viewer Context Menu

You can:

- Zoom in, zoom out, or view the area between the markers.
- Increase or decrease the waveform amplitudes.
- Rearrange the display.

To view additional or different data:

1 Click *Arrange Signals*.

The Arrange Signals Window appears.

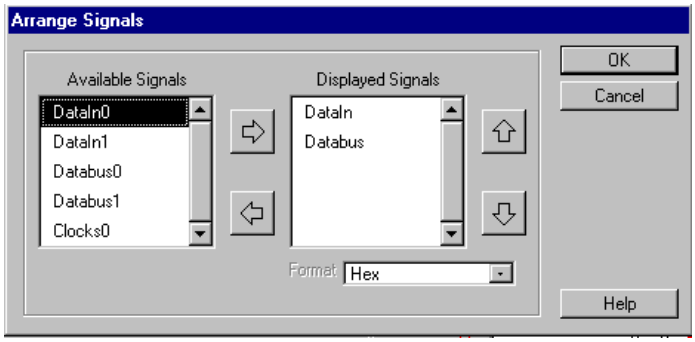


Figure 149 Waveform Viewer Arrange Signals Window

2 To view an item in the display, highlight it in the list of *Available Signals* and click the right-arrow.

3 To remove an item from the display, highlight it in the list of *Displayed Signals* and click the left-arrow.

4 To move an item in the list of *Displayed Signals*, highlight it and click the up- or down-arrow.

The result may look as shown below:

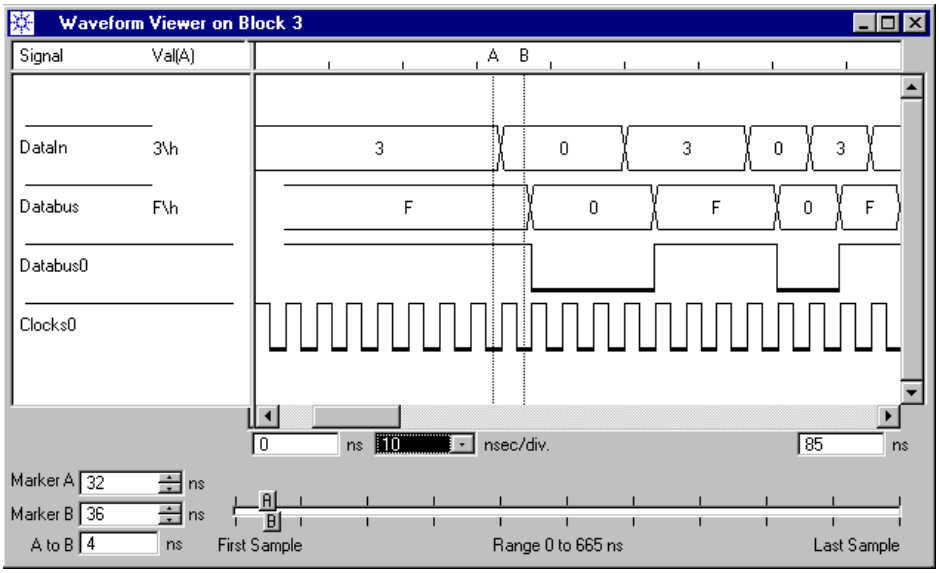


Figure 150 Waveform Viewer in Time Mode—Additional Waveforms

NOTE If you are viewing data input or pulse terminals and have sourced an added channel (see also *"How to Combine Generator Channels" on page 129*): The Waveform Viewer shows only the signal of the channel that has a connector.

If you have opened the Waveform Viewer for a data output port, it shows the summary of the captured data and the individual channels in **sample mode**. If error recognition was enabled, it shows also the deviations from expected data.

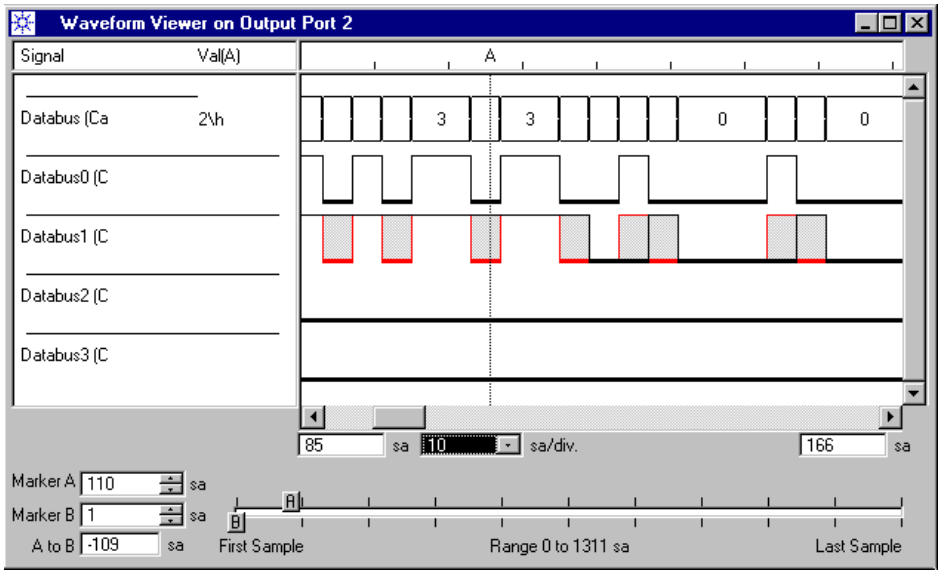


Figure 151 Waveform Viewer—Waveforms in Sample Mode

The signals are identified as

- (Capt)—captured data,
- (Comp)—compared data,
- (Err)—error data.

The display is presently limited to 10 characters.

A zero line tells you that nothing was received or expected.

The *Arrange Signals* menu now offers additional options: You can select between captured, compared, and error data, and, thus, compose an individual display.

The result may look as shown below:

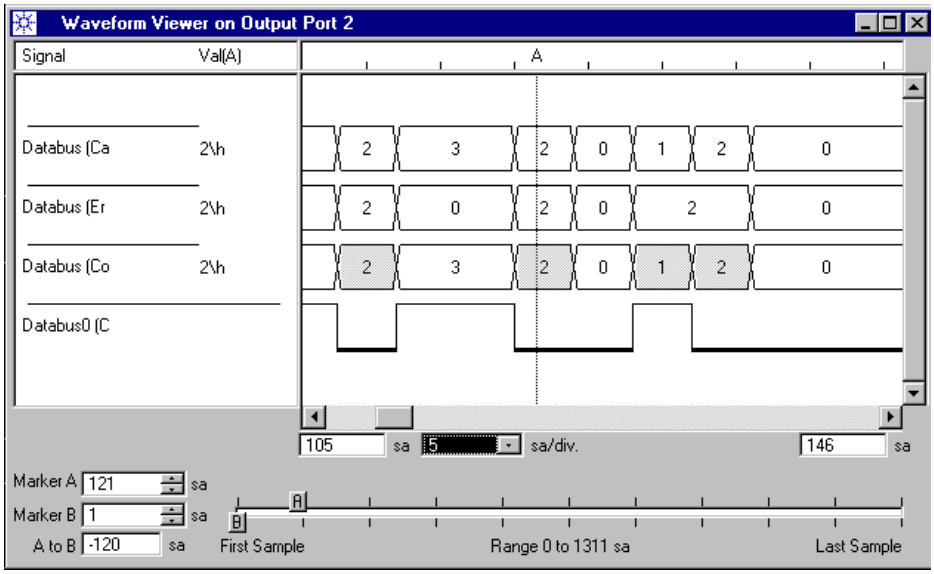


Figure 152 Waveform Viewer in Sample Mode—Additional Waveforms

Using Auxiliary Functions

This chapter provides information on auxiliary functions which are not directly related to test setup and execution. It covers the following topics:

"How to Compensate for Internal and External Delays" on page 216

"How to Export/Import Settings or Segments" on page 222

"How to Execute Firmware Commands" on page 225

How to Compensate for Internal and External Delays

Precise measurements require exact timing. Generated signals must reach the DUT simultaneously, response signals must be captured at the same point of time by all analyzers.

The Agilent 81200 system supports timing adjustments at the generator/analyzer connectors of the system, at the input and output connectors of the DUT board or even at the pins of the DUT.

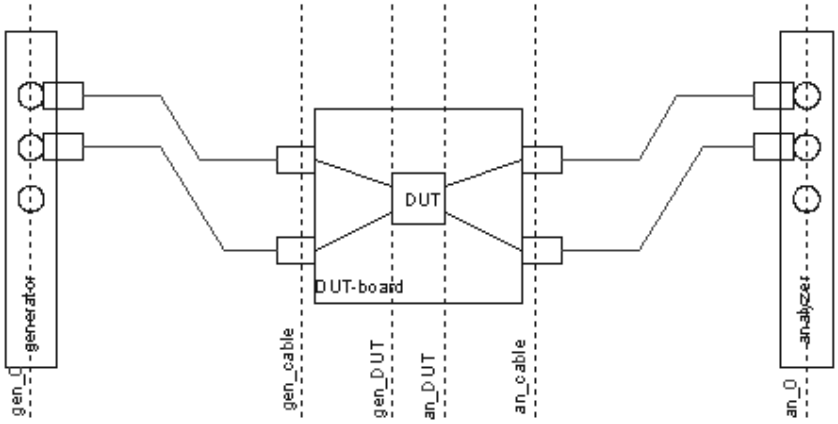


Figure 153 Supported Timing Adjustments

After installing new modules or after replacement of frontends a zero adjust procedure has to be performed to synchronize the new outputs or inputs with the ones already installed.

To compensate for signal delays in the used cables, a cable delay compensation procedure can be performed.

To compensate for both, propagation delays on the DUT board and delays in the cables, a cable and propagation delay compensation procedure can be performed.

Delay compensation is done with the Deskew Editor.

How to Start the Deskew Editor

To start the Deskew Editor:

1 Open the *Go* menu.

2 Click *Deskew Editor*.

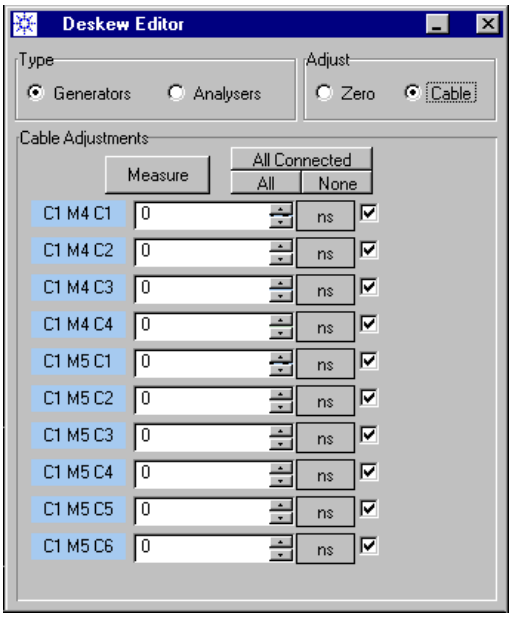


Figure 154 Deskew Editor—Start Window

Per default, the Deskew Editor identifies the available generator channels and suggests a cable delay measurement.

How to Adjust the Instrument Connectors

Zero adjustment ensures that an edge produced by the generator channels appears simultaneously at all generator connectors, and that received data is sampled at all analyzer connectors at the same point of time.

NOTE To perform this procedure, you need a reference SMA cable with known signal delay.

Once the Deskew Editor has been started:

1. Decide whether you wish to measure generator or analyzer channels.
2. Click the *Zero* button in the *Adjust* field to select zero adjust.

   The last measured delay values are displayed.

   Per default all connectors are marked for the zero adjust. With the *None* button you can deselect all and then mark just the new connectors for the zero adjust procedure.
3. Click the *Measure* button.

   For generators, the following window pops up:

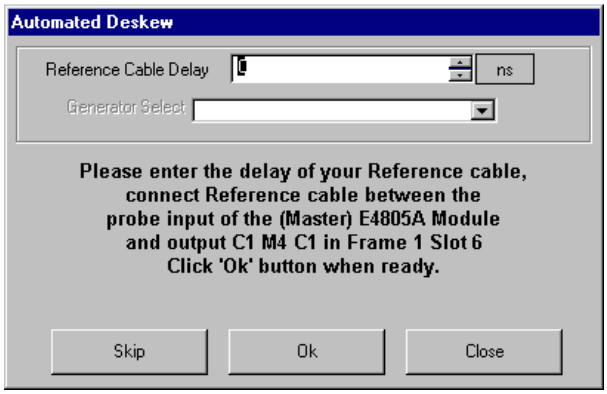


Figure 155 Deskew Editor—Zero Adjust Window for a Generator Channel

4 Follow the instructions given in this window.

Enter the master cable delay in the provided field. The typical delay of a 1 m (3 feet) SMA cable is about 3.5 ns.

The zero adjustment of analyzers requires a signal generator to be used as a reference channel. The system proposes the first of the installed generator channels. If there is no generator installed, the TRIGGER OUTPUT of the clock module is suggested.

The measured zero adjust value is automatically entered in the table and the next instruction to connect the next generator output is displayed. When all marked generators are adjusted the Automated Deskew window disappears.

How to Compensate for Cable Delays

The cable delay compensation assures that the edges of all generator outputs of the Agilent 81200 system appear at the same time at the end of the cables used in the setup. For the Agilent 81200 system analyzer inputs the cable delay compensation assures that all output signals of the DUT are sampled at the same time at the end of the cables, close to the DUT.

NOTE To perform this procedure, you need all the cables which are going to be used.

The cable delay compensation range is ±23 ns.

Perform the cable delay compensation first for the generator outputs, then for the analyzer inputs.

It is recommended to use cables of equal type and length for all inputs and outputs.

To compensate for cable delays:

1 Connect the cables you will use in your application to the Agilent 81200 system's output connectors.

2 Create a scheme of your application in the Connection Editor. Group the signals to ports and make the connections in the scheme.

3 In the Parameter Editor set the levels for all DUT input/output ports.

4 Open the Deskew Editor (see *"How to Start the Deskew Editor" on page 216*).

Per default, generators are selected in the *Type* field and cable delay measurement is enabled.

If you have connected the generator outputs in the Connection Editor, then you can easily mark the outputs for this measurement by clicking on *All connected*.

5 Click the *Measure* button.

The following window pops up.

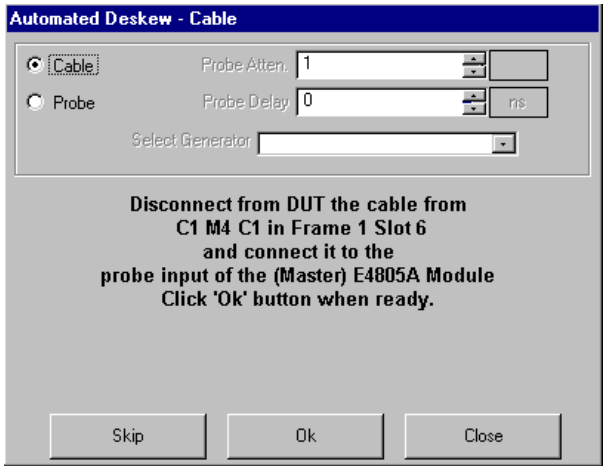


Figure 156 Deskew Editor—Cable Measurement Window for a Generator Channel

6 Follow the instructions given in this window.

The measured cable delay value is automatically entered in the table and the next instruction to connect the next generator output is displayed. After all marked generators have been adjusted the Automated Deskew window disappears.

7 Continue with the analyzer cables.

This is essentially the same procedure, except that you don't use the PROBE connector of the clock module, but one of the generator channels as a reference. The system proposes the first of the installed generator channels.

How to Compensate for Cable and DUT Board Delays

Performing a cable delay and propagation delay compensation assures that the edges of all generator outputs of the Agilent 81200 system are applied to the DUT input pins at the same time. The procedure also assures that all output signals of the DUT are sampled at the same time at the DUT output pins.

NOTE This procedure requires the E4805A central clock module. In an extended system, the master clock module is used.

To perform this procedure, you need all the cables which are going to be used. It is recommended to use cables of the same type and same length at all inputs and outputs. You also need an active probe, e.g. the Agilent 1144A, 800 MHz Active Probe, 10:1.

To compensate for cable and propagation delays:

1. Connect the cables you will use in your application to the Agilent 81200 system's output connectors.
2. Create a scheme of your application in the Connection Editor. Group the signals to ports and make the connections in the scheme.
3. In the Parameter Editor set the levels for all DUT input and output ports.
4. Open the Deskew Editor (see *"How to Start the Deskew Editor" on page 216*).

   Per default, generators are selected in the *Type* field and cable delay measurement is enabled.

   If you have connected the generator outputs in the Connection Editor, then you can easily mark the outputs for this measurement by clicking on *All connected*.
5. Click the *Measure* button.

6 Activate *Probe*.

The following window pops up:

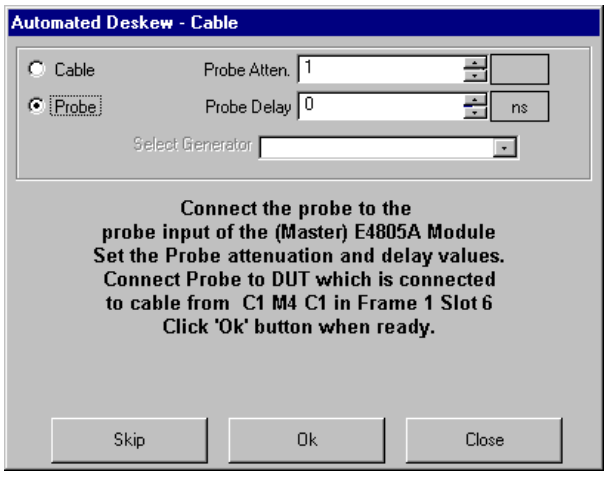


Figure 157 Deskew Editor—Probe Measurement Window for a Generator Channel

7 Enter the *Probe Attenuation* factor

For the Agilent 1144A Active Probe this is 10.

8 Enter the *Probe Delay*.

If you don't know the probe delay, you can make a first measurement by connecting the probe directly to the first generator output. Start the measurement by clicking OK. The result is the probe's propagation delay.

Close the Automated Deskew window. Click *Measure* again. Enter the resulting value as the probe delay and repeat the measurement. Now the resulting value for this output should be 0 ns.

9 Follow the instructions given in the window.

The measured delay value is automatically entered into the table and the next instruction to connect the next generator output is displayed. After all marked generators have been adjusted the Automated Deskew window disappears.

10 Continue with the analyzers.

This is essentially the same procedure, except that you don't use the PROBE connector of the clock module, but one of the generator channels as a reference. The system proposes the first of the installed generator channels.

NOTE If you are using the high speed differential analyzer E4837A, please note:

The analyzer must be operated in differential mode.

Both inputs must be connected to a differential generator (E4838A or E4843A).

Set the cable adjustment parameters according to your test setup (corresponding to the way the analyzer will be connected).

This ensures proper delay compensation.

How to Export/Import Settings or Segments

Settings and segments can be exported as ASCII files. These files can be stored on disk or on diskette, for example.

Export/Import of a Setting

This function serves the following purposes:

- You can create an archive of settings and store it at a secure place.
- If you need one of the settings once again, you can import any of the settings from the archive.
- You can investigate the exported setting with an editor and use it as a template for programming a test.
- If you have downgraded your Agilent 81250 system or frontends have been changed, some of your stored settings may not work any more. Exported settings can be edited and re-imported to fit to the new configuration.

How to Export a Setting

1 Open the *File* menu.

2 Choose *Export Setting* (see *"Export Setting" on page 77*).

How to Import a Setting

1 Open the *File* menu.

2 Choose *Import Setting* (see *"Import Setting" on page 76*).

Contents of a Setting File

A setting file is organized in blocks of firmware commands. The blocks start with a comment line. An example is shown below:

```
// Reset
:MMEM:SETT:NEW


// Create and connect Ports and Terminals
:SGEN:PDAT1:APP "OUTPUT_PORT",2,"Databus"
:SGEN:PDAT1:TERM1:REN "Databus0"
:SGEN:CONN:PDAT1:TERM1:TO (@0102003)
:SGEN:PDAT1:TERM2:REN "Databus1"
:SGEN:CONN:PDAT1:TERM2:TO (@0102004)
:SGEN:PDAT2:APP "INPUT_PORT",1,"Input"
:SGEN:PDAT2:TERM1:REN "Input0"
:SGEN:PPUL1:APP "INPUT_PORT",1,"Clock"
:SGEN:PPUL1:TERM1:REN "Clock0"
:SGEN:CONN:PPUL1:TERM1:TO (@0105001)

// Module type: E4805A
:SGEN:GLOB:TRIG INT10;
:SGEN:GLOB:TRIG:TVOL 0.0E+0;
:SGEN:GLOB:PER 8.333333333333E-9;
:SGEN:GLOB:MUX 4;
:MCL:SOUR ON;
:SGEN:GLOB:DOFF 0.0E+0;
:SGEN:GLOB:ARM IMM;
:SGEN:GLOB:ARM:SENS PLEV;
:SGEN:GLOB:ARM:THR 2.0E-1;
:SGEN:GLOB:ARM:TVOL 0.0E+0;
:TRIG:DEL 0.0E+0;
:TRIG:MUX 1.0E+0;
:TRIG:VOLT 2.5E+0;
:TRIG:VOLT:LOW 0.0E+0;
:TRIG:TVOL 0.0E+0;
:TRIG:IMP 5.0E+1;
:TRIG:MODE CGEN;

// Term type: E4844A
:SGEN:PDAT1:TERM1:MUX 1.0E+0;
:SGEN:PDAT1:TERM1:INP:DEL 4.166667E-9;
:SGEN:PDAT1:TERM1:INP:DEL:CYCL 5.0E-1;
:SGEN:PDAT1:TERM1:INP:DEL:TIME 0.0E+0;
:SGEN:PDAT1:TERM1:INP:THR 0.0E+0;
```

```
:SGEN:PDAT1:TERM1:INP:TVOL 0.0E+0;
:SGEN:PDAT1:TERM1:INP:IMP 5.0E+1;
:SGEN:PDAT1:TERM1:INP:SER 0.0E+0;
:SGEN:PDAT1:TERM1:INP ON;
```

Export/Import of Segments

This function serves the following purposes:

- You can create an archive of segments and store it at a secure place.
- You can create segments with an editor and import them.
- Available data patterns can easily be inserted into the test sequence.
- Exported segments can be edited and re-imported.

How to Export Segments

1 Open the *File* menu.

2 Choose *Export Segments* (see *"Export Segments" on page 77*).

How to Import Segments

1 Open the *File* menu.

2 Choose *Import Segments* (see *"Import Segments" on page 76*).

Contents of a Segment File

A segment file can hold several segments. Each segment has a general structure as shown in the example below:

```
:vectorVariablesDefinitions:
{
   :paraPatternVar:

   {
      :name: Input1
      :statePar: {  {A "01"}  }
      :stateSet: A
      :vectorWidth: 1
      :vectors:
         {
      1
      0
      1
      0
      1
```

```
        0
        1
        1
        1
        0
        1
        0
           }
           :parameters:
           {
              { _Type (MEMORY) }
           }
        }
     }
```

How to Execute Firmware Commands

The Command Line Editor allows control of an instrument through the command string interface. This window is intended as a test editor to test individual commands for a remote program.

How to Start the Command Line Editor

To start the Command Line Editor:

1 Open the *Go* menu.

2 Choose *Command Line*.

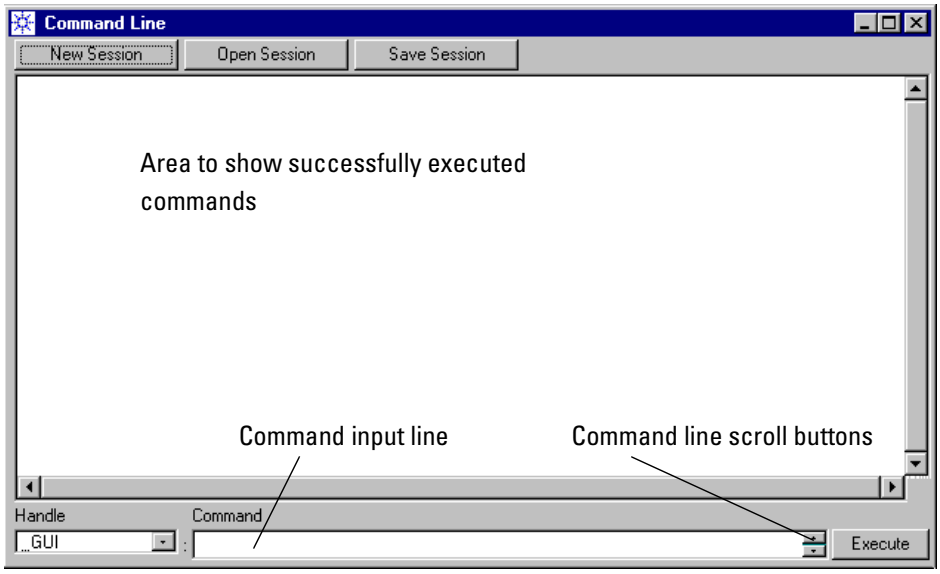


Figure 158 Command Line Editor

The Command Line window is divided into two areas:

- Display area, occupies the upper part of the window.
- Command entry area, occupies the lower part of the window.

Commands can be entered in the command input line. Successful commands and their results are displayed in the display area.

How to Use the Command Line Editor

The commands you enter are transmitted to and executed by the instrument to which the current user interface is connected (see also *"Reconnect System" on page 78*).

All commands available for the Agilent 81250 system can be entered. For details see the *Agilent 81250 Programming Reference*. Note, that the first colon is automatically provided and must not be entered.

Once you have entered a command in the command entry area, click the *Execute* button. This downloads the command to the firmware where it is executed.

Commands which have been successfully executed are moved up to the display area.

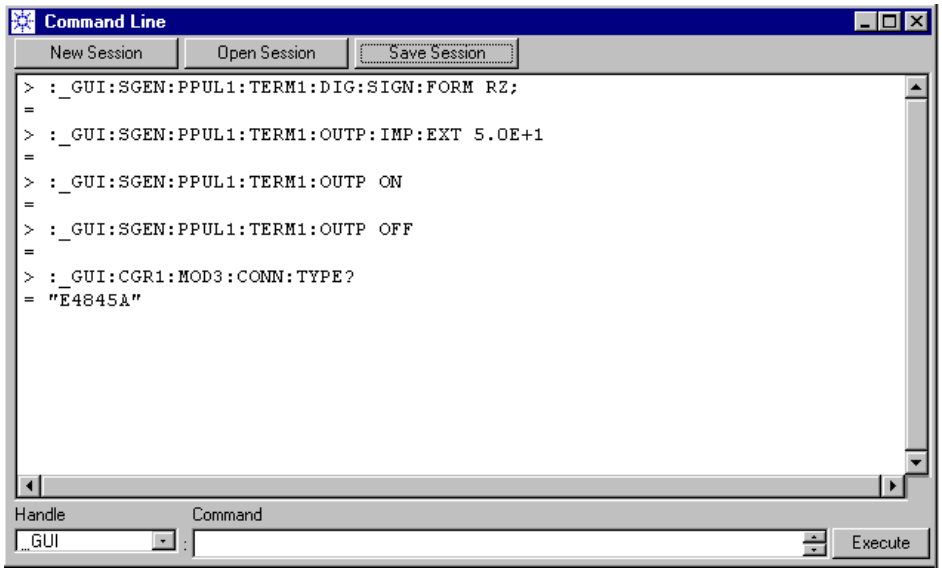


Figure 159 Command Line Editor – Executed Commands

Copy and Paste in the Command Line Editor

Selected text can be cut, copied or pasted from and to the command entry area. The Command Line Editor provides a context menu that supports these operations, and the editor reacts on common keyboard shortcuts (Ctrl+c, Ctrl+x, Ctrl+v). It uses the Windows Clipboard.

You can copy and paste parts of a command or a whole command line. Commands can thus be taken from the display area and entered in the command entry area.

The two arrow buttons at the right-hand side of the command entry line as well as the cursor up/down keys can be used for scrolling through the list of previously successful commands.

TIP You can also paste commands that have been copied to the clipboard from external program files, such as an exported setting file. This works in both directions and enables you also to create program files with any program editor that allows copy and paste.

Buttons of the Command Line Editor

The Command Line Editor provides the following buttons:

- *New Session*: Clears the display area.

  You can start to create a new sequence of commands to be combined in a new session.

- *Save Session*: Saves the commands shown in the display area in a file on disk.

  The files are saved as command session (*.dcs) files in the c:\hp81200\dsr\bin directory.

- *Open Session*: Used to open a previously saved session and execute it immediately.

Select from the list of available command session files.

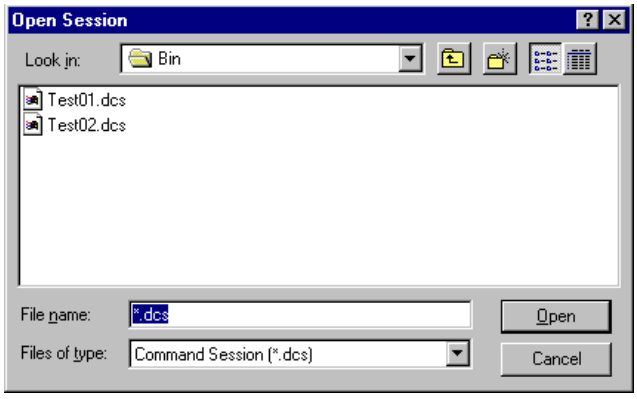


Figure 160 Command Line Editor – Selection of Saved Sessions

Appendix A: How Do I ... ?

This section provides answers to frequently asked questions.

It can well happen that you have a certain problem and just don't know how to solve it quickly and efficiently with the Agilent 81250 Parallel Bit Error Ratio Tester. It is this kind of problems which is addressed in this chapter.

The chapter covers the following topics:

"How Can I Generate a Clock Signal With a Data Module?" on page 230

"How Do I Use Events?" on page 232

"How Can I Change all Traces of a Port to Don't Care?" on page 236

"How Do I Set Up a Multiplexer BER Test?" on page 240

"How Do I Use Automatic Sampling Point Adjustment?" on page 243

"How Do I Use the AUX OUT of E4863A/E4865A Frontends?" on page 249

How Can I Generate a Clock Signal With a Data Module?

The data generator/analyzer module must be equipped with one of the generator frontends:

- E4838A, 660 MHz, differential output, low voltage amplitude/offset and variable slopes generator,
- E4843A, 660 MHz, NRZ/RZ, differential output frontend,
- E4842A, 330 Mbit/s, NRZ/RZ, single ended, variable transitions, 3.5 V amplitude,
- E4846A, 200 Mbit/s, dual output single-ended frontend,
- E4862A, 2.6 Gbit/s, differential NRZ output frontend,
- E4864A, 1.3 Gbit/s, differential NRZ output frontend.

The most comfortable way to apply a clock signal to the DUT is:

1 Use the Connection Editor and create a **pulse port**.

2 Connect the pulse terminal to the generator connector you wish to use.

3 Use the Parameter Editor to adjust the channel properties, such as frequency, pulse width, delay, voltage levels, expected load, and so on. Keep the default RZ (Return to Zero) data format. Switch the generator output on.

On the other hand, you can also apply a clock signal to one or several terminals of a **data port**. This is useful if you need to generate a burst of clock pulses. In this case, the clock frequency is identical with the port frequency and applies to all terminals.

1 Create the sequence and insert the segments.

2 Use the Segment Editor (easily invoked from the Sequence Editor or Data/Sequence Editor) to set the data bits of the desired trace(s) to “1”.

3 Use the Parameter Editor for the respective channel(s) and select “RZ” (Return to Zero) as the format.

NOTE The RZ data format is not supported by the E4862A and E4864A gigabit frontends. For these frontends, you would have to create a 1, 0, 1, 0, 1, 0 ... segment.

In RZ format every logical 1 creates an electrical pulse, as illustrated in the figure below:

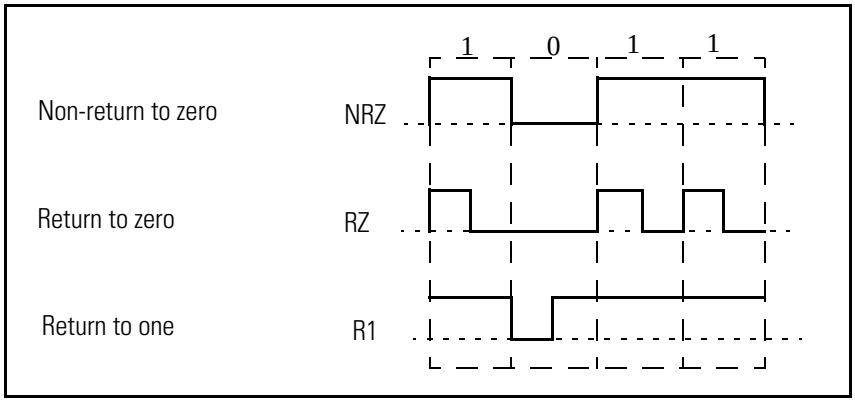


Figure 161 Signal Formats

4 If necessary, adjust the channel properties, such as pulse width, delay, voltage levels, termination resistors, and so on. Switch the generator output on.

NOTE This procedure covers one block of the overall test sequence. If the clock signal shall be output during execution of another block, the steps 2 and 3 have to be repeated for that block.

How Do I Use Events?

The built-in features for detecting events and reacting upon events provide many capabilities. This section shows and explains some examples.

How Do I Select Between Two Different Tests?

With this setup you can switch very fast between two different test sequences using the command control feature. With the trigger pod and external hardware even more blocks could be selected.

1 Define the deferred events CMD0 and CMD1.

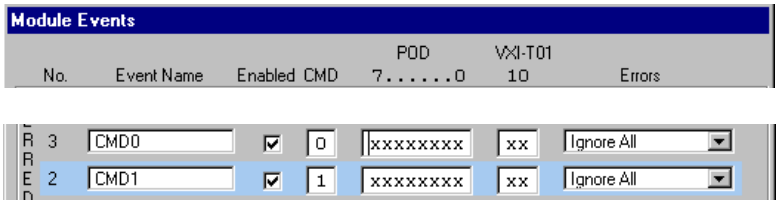
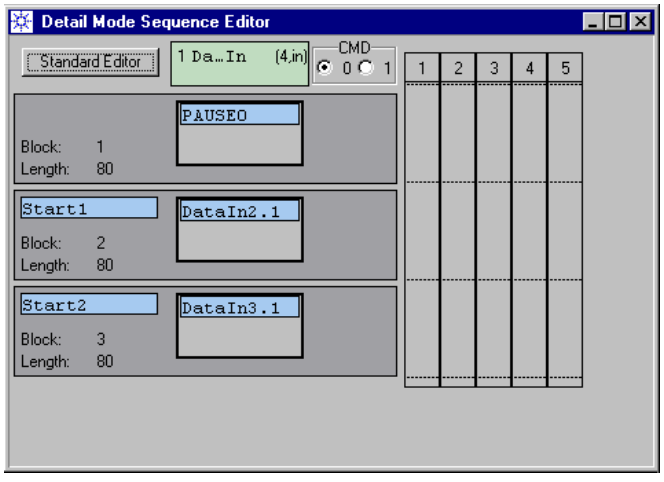


Figure 162 Definition of 2 Command Events

2 Create the sequence and fill in the branch tables of the blocks.



If	Go to	Trig.	VXI-T01
CMD0	Start1	0	00
CMD1	Start2	0	00

CMD0	END	0	00
CMD1	END	0	00

Figure 163 Sequence and Branch Tables for Selecting One of Two Tests

How Do I Set a Trigger on Error?

We simply want to issue a trigger signal out of the TRIGGER OUTPUT whenever an error is detected somewhere in the system. Note that we use an immediate event here.

1 Define the immediate error event.

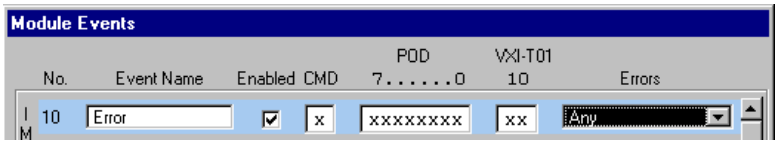
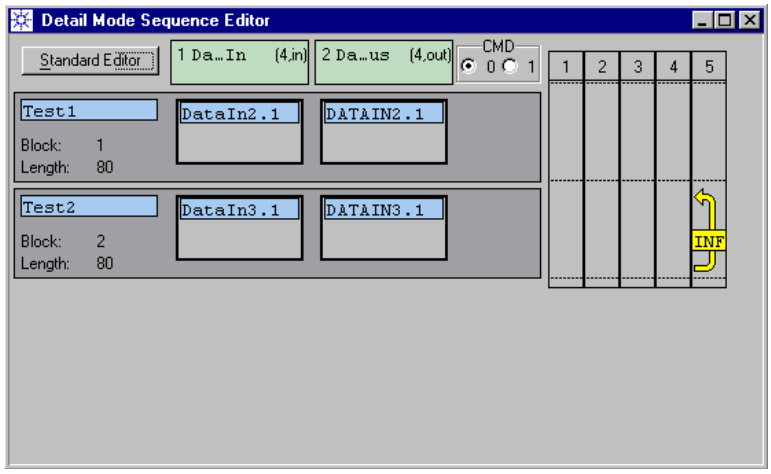


Figure 164 Definition of an Immediate Error Event

2 Create the sequence and fill in the branch tables.



If	Go to	Trig.	VXI-T01
Error		1	00
Error		1	00

Figure 165 Sequence and Branch Tables for Triggering on Error

How Do I Allow the DUT to Stabilize?

Here we wait until we have no error condition for a certain time (determined by the block length).

1 Define a deferred error event.

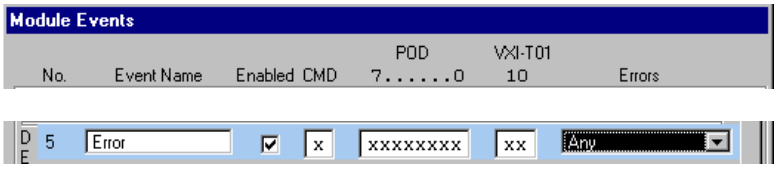
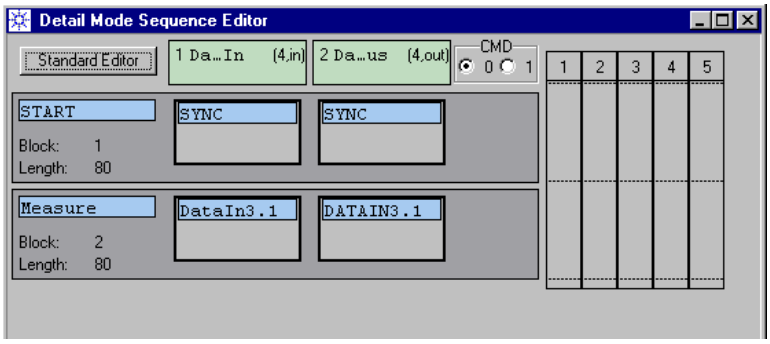


Figure 166 Definition of a Deferred Error Event

2 Create the sequence and fill in the branch table of the START block.



If	Go to	Trig.	VXI-T01
Error	START	0	00

Figure 167 Sequence and Branch Table for a Test With Warm-up

How Can I Return Pass/Fail Information to another Test System?

In this bolt-on example our instrument is integrated into a large IC test system. We shall run a measurement that is selected via the trigger input pod.

The result is a pass/fail signal that is returned to and examined by the large IC tester.

A generator frontend will be used to generate the pass/fail signal. The output of this generator (low or high) is defined by two memory segments. One of these segments contains only zeros, the other only ones.

In order to use the generator, we have specified a one-terminal data input port. This port does actually not belong to the DUT, because the generator's output is physically connected to a sense-pin of the IC test system.

1 Define four events.

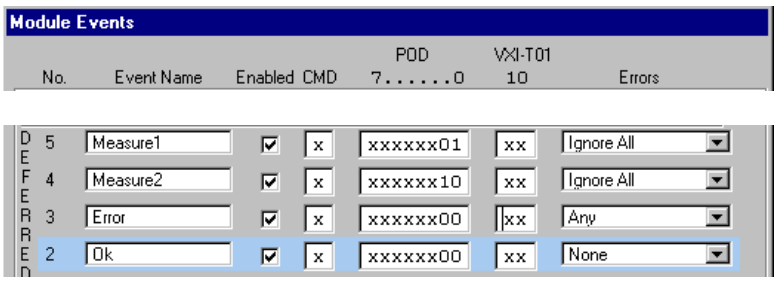
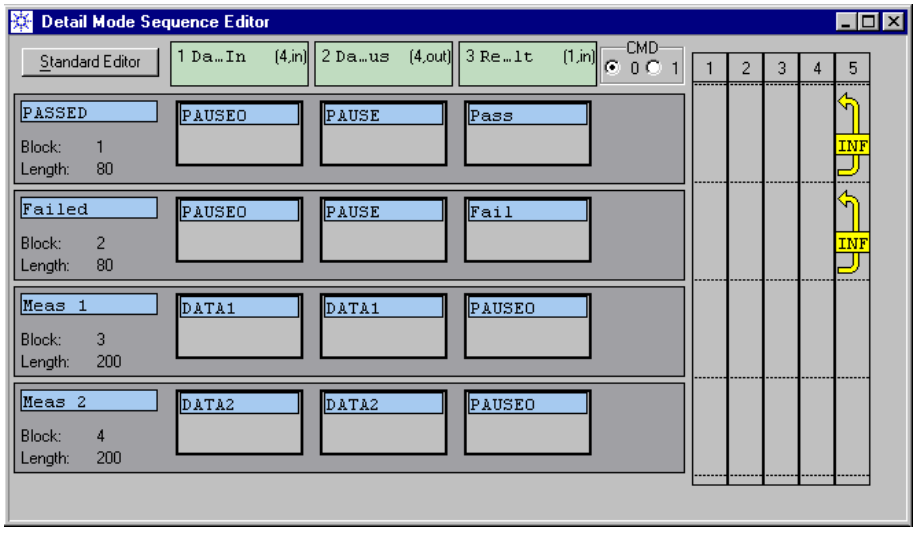


Figure 168 Definition of Pod and Error Events

2 Create the sequence and fill in the branch tables.

Remember that we use a one-terminal data input (= generator output) port and two segments to generate and return the pass/fail signal. The segments are called PASS and FAIL.



If	Go to
Measure1	Meas_1
Measure2	Meas_2

Measure1	Meas_1
Measure2	Meas_2

Error	Failed
Ok	PASSED

Error	Failed
Ok	PASSED

Figure 169 Sequence and Branch Tables

Remarks The MEAS_1 and MEAS_2 blocks need to be longer than the actual measurement so that all analyzer pipelines are toggled through and the events are completely processed.

The event definitions shown above require that the trigger pod inputs are driven with positive pulses to select the measurement blocks. During the measurement the pod input lines must be zero. Otherwise no errors are detected. This should not be a problem because due to the internal pipelining it takes already a couple of sequencer clock periods until the first error can be detected (see also *"What You Need to Consider Before Using Events" on page 166*).

The events that cause the jump to the different measurement blocks can be deferred or immediate. The Ok and Error events, however, must be deferred, and the Error event must have a higher priority than the Ok event.

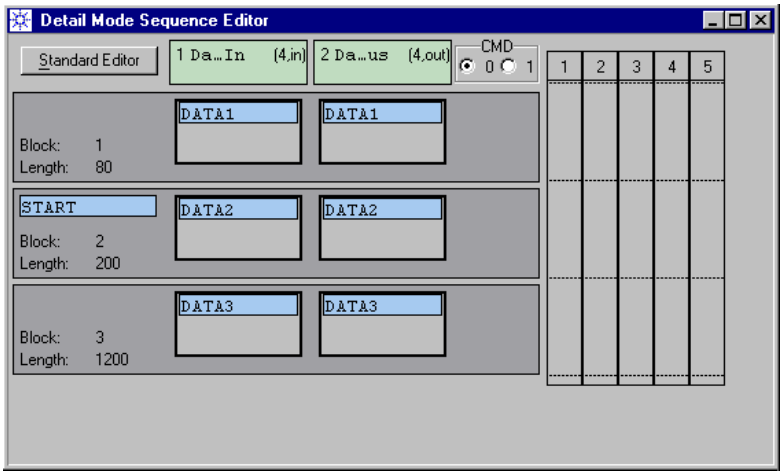
If you want to save events, the Ok event could be replaced by the DEFAULT event.

How Can I Execute Different Tests Embedded in One Sequence?

Starting at a certain block label is fairly simple—the CMD 0/1 command, the VXI trigger lines, or the trigger pod can be used to switch between several start blocks.

The following example shows how the sequence can be terminated after executing a certain block.

We use the DEFAULT event which needs not be defined. It occurs automatically at the end of the block.



If	Go to	Trig.	VXI-T01
DEFAULT	END	0	00
DEFAULT	END	0	00
DEFAULT	END	0	00

Figure 170 Sequence and Branch Tables

In this example just block 2 is executed. By moving the START label, any other block can be run.

How Can I Change all Traces of a Port to Don't Care?

The simplest way to ignore incoming data is to use a Pause segment.

However, don't care setting is available for data output (= analyzer input) ports.

The following procedure can be used if the measurement mode is set to Compare and Capture or to Compare and Acquire around Error.

Use the Detail Mode Sequence Editor or the Data/Sequence Editor.

To replace the present segment by a Don't Care pseudo segment:

1 Open the segment's context menu.

2 Click *Don't Care*.

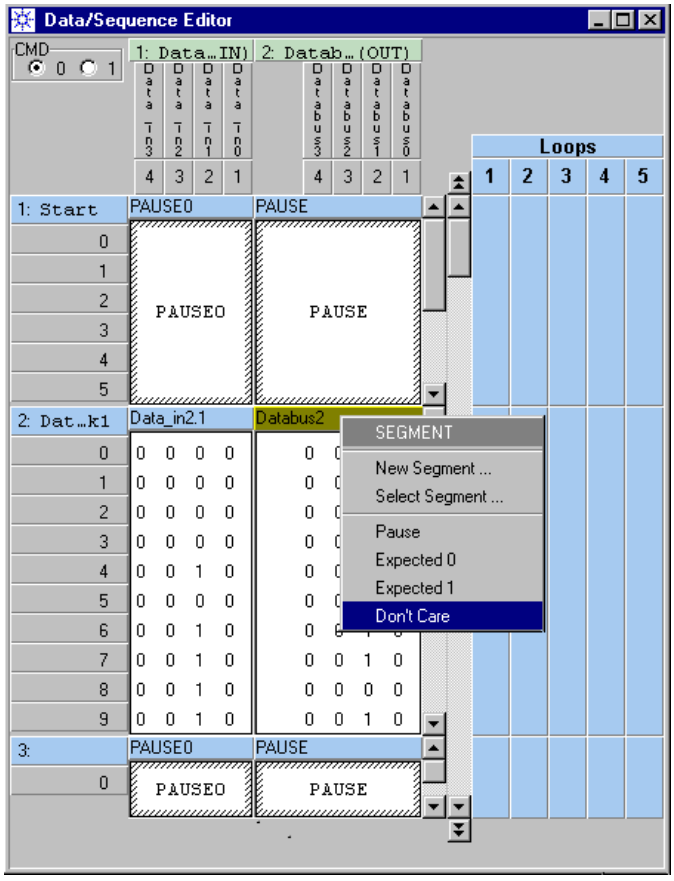


Figure 171 Segment Context Menu

To change individual or all traces:

1 Right-click the segment's display area.

This opens the Segment Editor context menu.

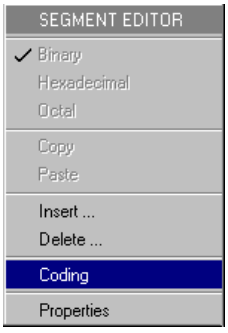


Figure 172 Segment Editor Context Menu

2 Click *Coding* or *Properties* and ensure that the state coding is set to 0x1. Only 0x1-segments can have don't care settings.

3 Highlight the traces or vectors you wish to change.

4 Open the Segment Editor context menu once more and click *Set To*.

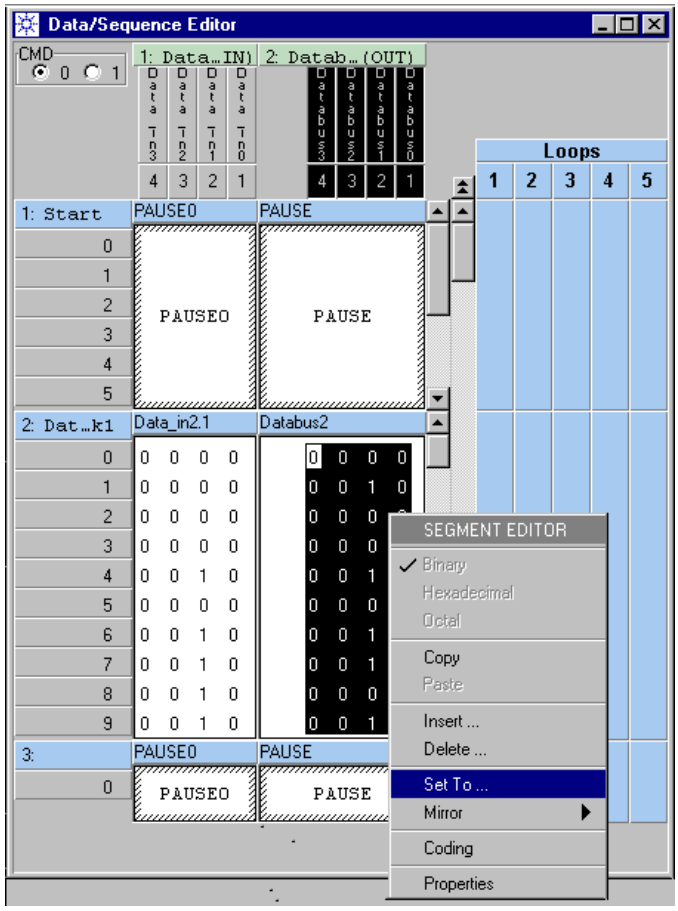


Figure 173 Segment Editor Context Menu after Highlighting Data

5 Choose *x* from the pull down menu.

How Do I Set Up a Multiplexer BER Test?

The bit error rate (BER) of a multiplexer can be measured by sourcing a PRWS (pseudo random word stream) segment to the DUT and comparing the serial output with a PRBS (pseudo random bit stream) segment of the same order.

The connections could be made as shown below:

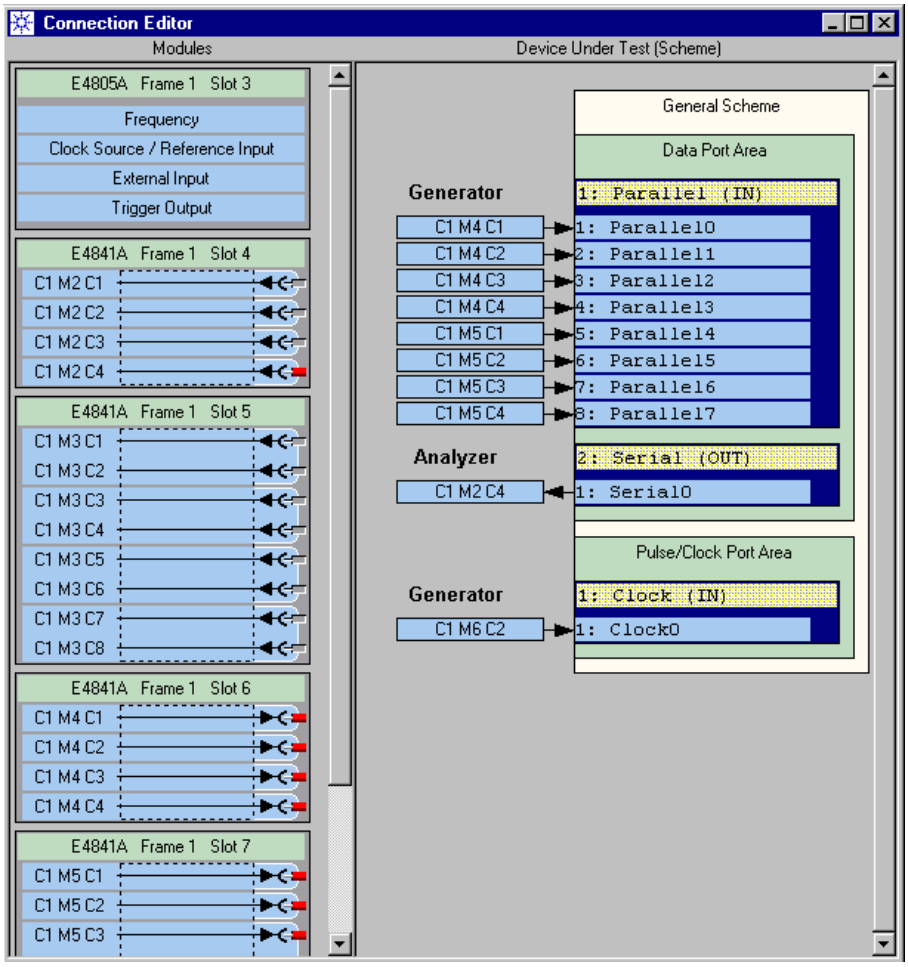


Figure 174 Connections for a Multiplexer Test

1 Create a PRWS segment to be used for the DUT input port.

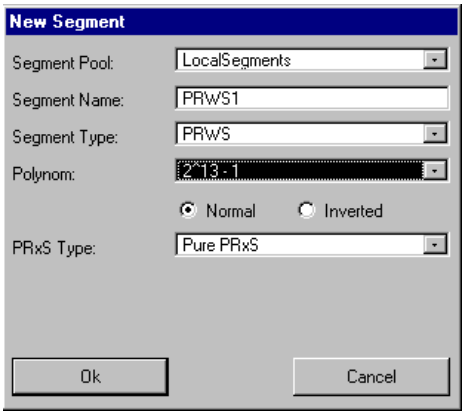


Figure 175 Segment Editor Window

2 Create a PRBS segment of the same order (the same polynomial) to be used for the DUT output port.

3 Use the Sequence Editor and insert the segments into the sequence.

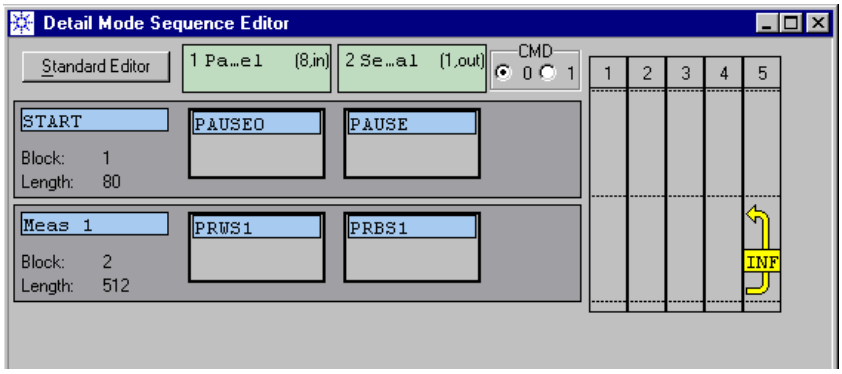


Figure 176 MUX Test Sequence

The correspondence between the generated PRWS and the expected PRBS is as follows:

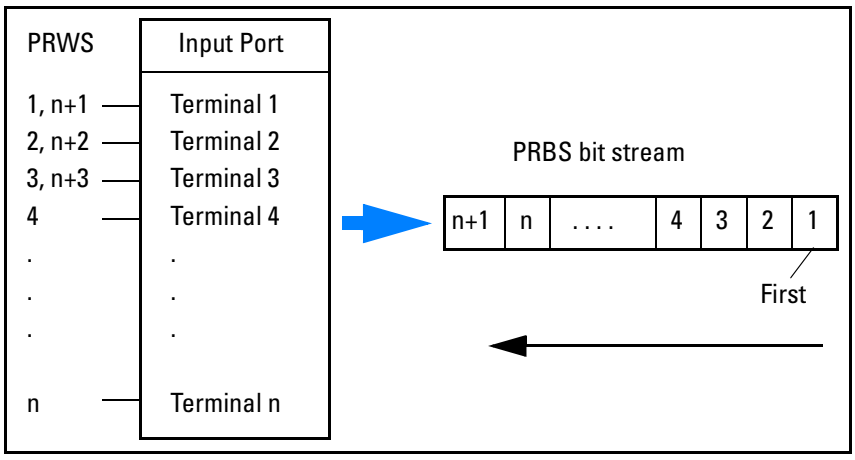


Figure 177 Correspondence Between Random Words and Random Bit Stream

The bits of the PRWS are assigned to the generator channels from top to bottom, as displayed by the Connection Editor.

The PRBS contains the same logical bit sequence just one-directional.

The same principle—only inverted—applies if you are testing a demultiplexer.

NOTE Note that the assignment of bits to generator channels is different, if you apply the PRWS to channels which have been digitally added.

In this case, the system assumes that the word length n is equal to the total number of channels involved. It assigns the first m bits to the connected channels and the remaining n–m bits to the added, unconnected channels.

Example If you had an input port with five terminals and the first two terminals were connected to two added channels, then terminal 1 would receive the XOR of bits 1 and 6 and terminal 2 the XOR of bits 2 and 7, as illustrated below:

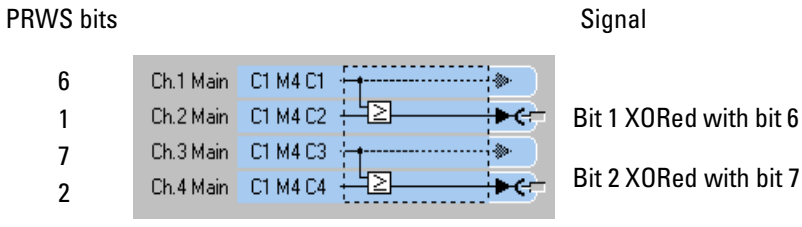


Figure 178 Bit Assignment to Digitally Added Channels

How Do I Use Automatic Sampling Point Adjustment?

The functions Automatic Bit Synchronization and Automatic Delay Alignment can be used to adjust the sampling point of the analyzers of an Agilent 81250 system that sources data to and simultaneously captures data from the DUT.

However, these functions can also be used for synchronizing two separate systems. Separate systems with individual clocks are required if, for example, multiplexers/demultiplexers are to be tested which have a mux-factor other than 2^n.

This section gives some examples of setups and procedures.

How Can I Synchronize a MUX Test With Two Systems?

The functions for automatic analyzer sampling point adjustment require that certain conditions are met.

Requirements for MUX tests with a single system:

Automatic Delay Alignment	Mux-factor = 2^n Delay window is known and can be specified.
Automatic Bit Synchronization	Mux-factor = 2^n Only PRxS data must be sent and expected.

Requirements for MUX tests with two systems:

Automatic Delay Alignment	Delay window is known and can be specified.
Automatic Bit Synchronization	PRxS data may be sent and expected. If memory-type data is used, then the first 48 bits must be unmistakable.

Using Automatic Bit Synchronization

Setup A multiplexer test with two systems can be set up as shown below:

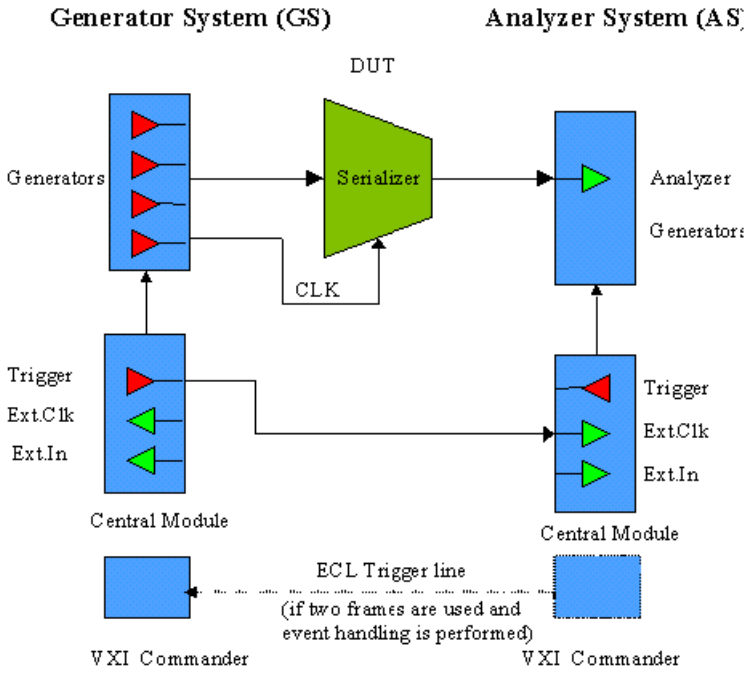


Figure 179 MUX Test Setup With Automatic Bit Synchronization

One system is used to source clock and data to the multiplexer and an additional clock to the analyzer. The other system is used for data analysis.

The external clock frequency for the analyzer is automatically calculated.

The analyzer runs with this “external clock”.

Procedure

1. Connect the DUT in the Connection window.
2. Set the appropriate levels.
3. Set up the data to be sent and expected.
4. Start the generator.
5. Run the test after the PLLs of the DUT and the analyzer have stabilized.

Using Automatic Delay Alignment

Setup The analyzer system is in the external clock mode and starts with an external trigger:

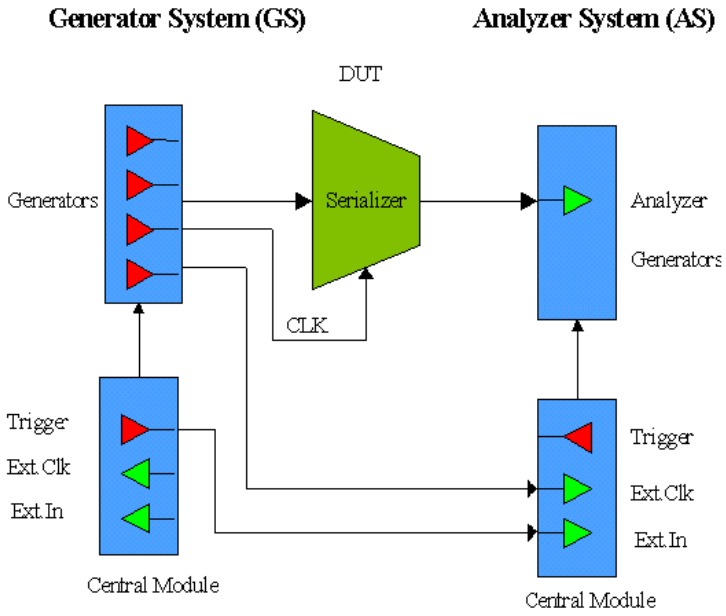


Figure 180 **MUX Test Setup With Automatic Delay Alignment**

Procedure

1. Connect the DUT in the Connection window.
2. Set the appropriate levels.
3. Set up the data to be sent and expected.
4. Enable the Auto Delay Alignment.

   The sequence flow is illustrated below.

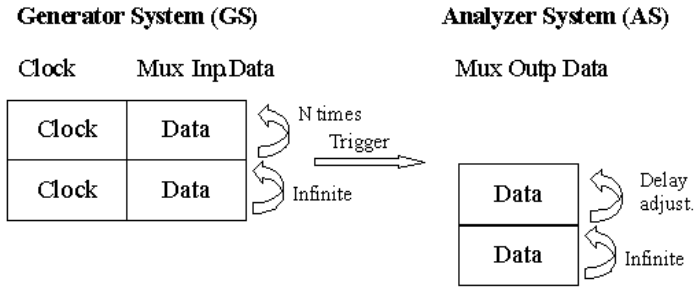


Figure 181 **Sequence Flow for Automatic Delay Alignment**

5. Set the repetition counter of the start block to a number (N) that suffices to allow the PLL of the analyzer to lock on the generator clock (8 ms typically).

The generator sends data and clock to the DUT and an additional clock to the analyzer. After N repetitions of the start block, the generator issues the start trigger and begins an infinite loop.

In the meantime, the analyzer's PLL had time to get locked to the external reference clock. When the trigger arrives, the analyzer becomes active. It waits for the specified start delay and then begins sampling the incoming data. The Automatic Delay Alignment assures that the incoming data is sampled at the optimum point of time.

Now the analyzer enters the second block and performs the measurement.

How Can I Synchronize a DEMUX Test With Two Systems?

The functions for automatic analyzer sampling point adjustment require that certain conditions are met.

Requirements for DEMUX tests with a single system:

Automatic Delay Alignment	Mux-factor = 2^n Delay window is known and can be specified. Word phase delay is known.
Automatic Bit Synchronization	Mux-factor = 2^n Only PRxS data must be sent and expected. If this is not pure PRxS, the word phase delay must be known.

Requirements for DEMUX tests with two systems:

Automatic Delay Alignment	Delay window is known and can be specified. Word phase delay is known.
Automatic Bit Synchronization	PRxS data may be sent and expected. If this is not pure PRxS, the word phase delay must be known. If memory-type data is used, then: • the first 48 bits must be unmistakable • the word phase delay must be known.

Using Automatic Bit Synchronization

Setup A demultiplexer test with two systems can be set up as shown below:

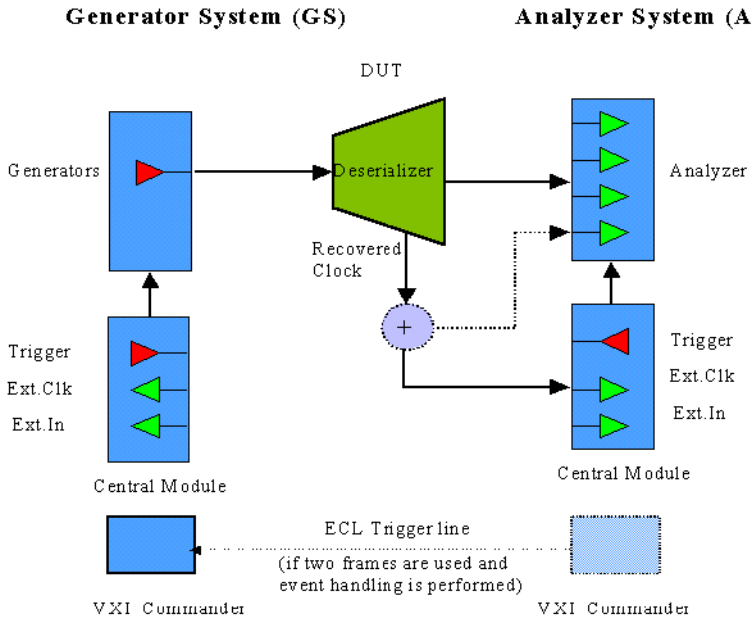


Figure 182 DEMUX Test Setup With Automatic Bit Synchronization

One system is used for data generation. The other system is used for data analysis.

The analyzer runs in “external clock” mode. The demultiplexer provides this clock. The frequency of the external clock has to be programmed or measured by the analyzer system to enable the system to establish the correct sample point delays.

If one wishes to obtain the timing of his measurements relative to the clock of his DUT, then he has to feed the clock of the demultiplexer via a power splitter into the clock module and an additional analyzer channel.

Procedure

1. Connect the DUT in the Connection window.
2. Set the appropriate levels.
3. Set up the data to be sent and expected.
4. Enable the Auto Bit Synchronization with Auto Phase Alignment.
5. Start the generator system.
6. Run the test after the PLLs of the DUT and the analyzer have stabilized.

Using Automatic Delay Alignment

Setup The analyzer system is in the external clock mode and starts with an external trigger:

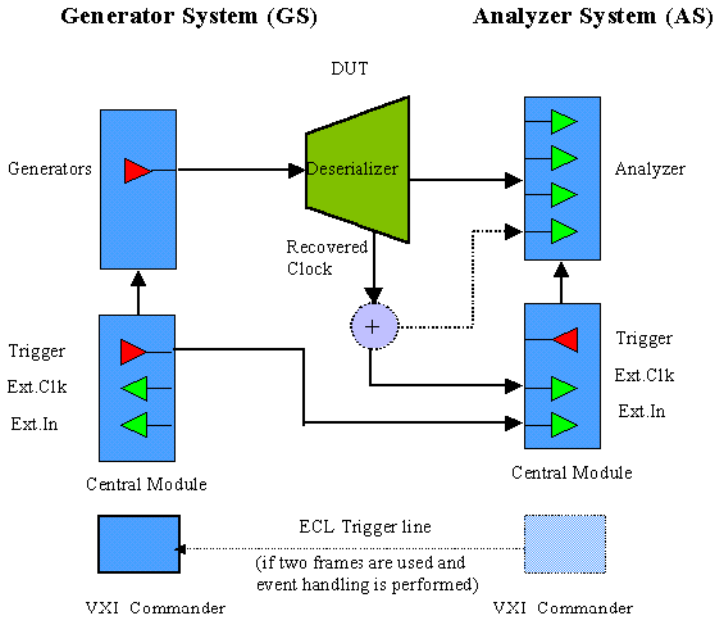


Figure 183 **DEMUX Test Setup With Automatic Delay Alignment**

Procedure

1. Connect the DUT in the Connection window.
2. Set the appropriate levels.
3. Set up the data to be sent and expected.
4. Enable the Auto Delay Alignment.
5. Set the repetition counter of the synchronization block to a number that suffices for the synchronization process.

The generator sends data and clock to the DUT and an additional clock to the analyzer.

The synchronization block is repeated until the DUT's and the analyzer's PLLs have stabilized.

How Do I Use the AUX OUT of E4863A/E4865A Frontends?

The “giga” analyzer frontends E4863A and E4865A have an AUX OUT connector.

Setup Example

This connector, for example, provides the option to route a recovered clock signal to the clock module.

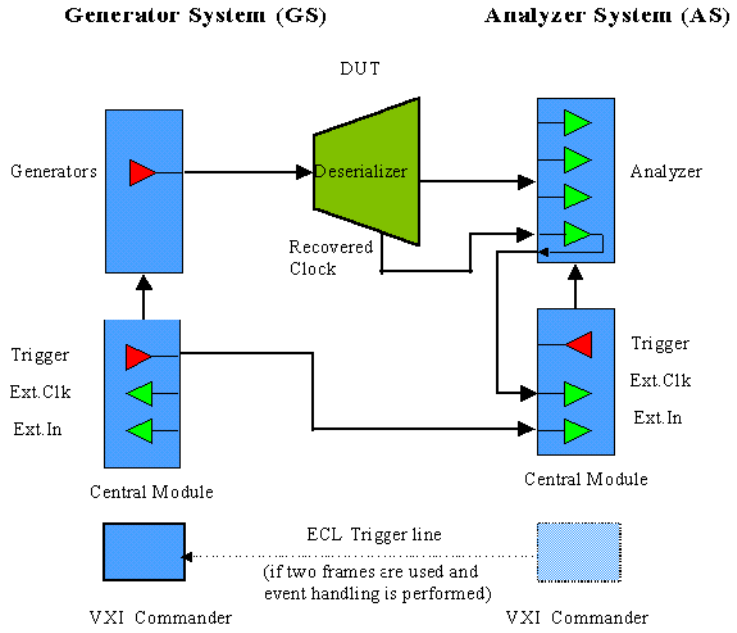


Figure 184 DEMUX Test Setup With Recovered Clock

The AUX OUT connector is connected to the output of the analyzer’s comparators and delivers a unipolar signal.

In the example above, the analyzer system is started by a trigger issued from the generator system. It then uses the recovered clock of the DUT to set its own capturing frequency.

The AUX OUT has an internal impedance of 50 Ω and has to be terminated accordingly.

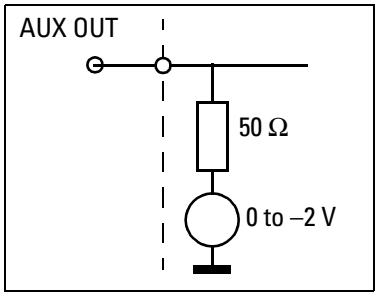


Figure 185 AUX OUT Termination

A termination voltage between 0 V and –2 V may be used.

Appendix B: PRBS/PRWS Data Segments

This appendix contains information how the Agilent 81250 Parallel Bit Error Ratio Tester generates Pseudo Random Bit/Word Streams (PRBS and PRWS). Furthermore, it describes the additional features available with PRBS/PRWS.

Pure and Distorted PRBS

The Agilent 81250 uses a shift register with appropriate feedback to generate the pseudo random data. An example is shown below.

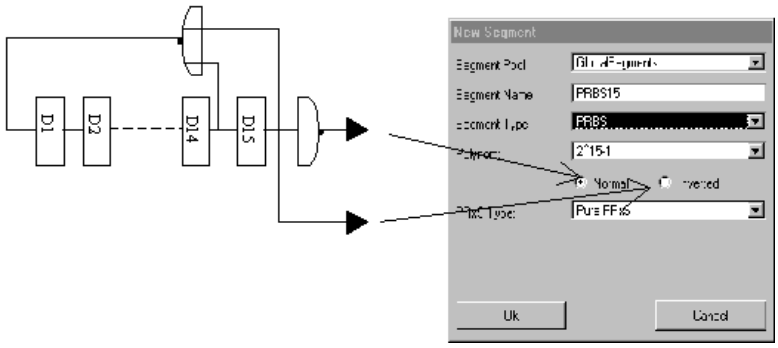


Figure 186 The Generation of a 2^{15}–1 PRBS Using a Shift Register

Here you can see, how a CCITT O.151 compatible 2^{15}– 1 PRBS is generated.

If *Normal* is selected, the output of the inverter is used to generate the bit stream. The shift register starts with all flip-flops loaded to one. This causes the bit stream to start with the longest run of 15 zeros.

If *Inverted* is selected, the bit stream is sent to the DUT without the inverter. The shift register starts with all flip-flops loaded to one. This causes the bit stream to start with the longest run of 15 ones.

A mathematical representation of the shift register with feedback taken from flip-flops 15 and 14 for the 2^{15}–1 PRBS is:

$X^{15} + X^{14} + 1$

A complete list of the available polynomials—using the mathematical notation—is presented in the table below.

Table 13 PRBS Polynomials

PRBS	Polynomial	Comment
2^5– 1	$X^5 + X^4 + X^2 + X^1 + 1$	
2^6– 1	$X^6 + X^5 + X^3 + X^2 + 1$	
2^7 – 1	$X^7 + X^6 + 1$	Compatible with HP 70841A and 70845A
2^8– 1	$X^8 + X^7 + X^3 + X^2 + 1$	
2^9– 1	$X^9 + X^8 + X^5 + X^4 + 1$	
2^{10}– 1	$X^{10} + X^7 + 1$	Compatible with HP 70841A and 70845A

Table 13 PRBS Polynomials

PRBS	Polynomial	Comment
$2^{11}-1$	$X^{11}+X^{8}+X^{5}+X^{2}+1$	
$2^{12}-1$	$X^{12}+X^{9}+X^{8}+X^{5}+1$	
$2^{13}-1$	$X^{13}+X^{12}+X^{10}+X^{9}+1$	
$2^{14}-1$	$X^{14}+X^{13}+X^{10}+X^{9}+1$	
$2^{15}-1$	$X^{15}+X^{14}+1$	CCITT O.151
$2^{23}-1$	$X^{23}+X^{18}+1$	CCITT O.151
		81250 using E4861A or E4832A modules
$2^{31}-1$	$X^{31}+X^{30}+X^{7}+X^{6}+1$	81250 using E4861A or E4832A modules

All bit streams up to 2^{15}–1 are generated by simulating the shift register in the software and finally loading the pattern into the hardware memory of the module. This provides the option to modify the data before downloading it to the modules.

The two longest bit streams (2^{23}–1 and 2^{31}–1) are only available with the E4832A and E4861A modules. These modules employ an internal shift register to generate these bit streams directly in the hardware. For these two bit streams only pure PRxS without any additional options is available.

Variable Mark Density

The ratio of zeros to ones of a pure pseudo random bit stream is approximately 1 to 2. You can modify this ratio by using the option Variable Mark Density.

A PRBS with marker density 1/8, 1/4, 3/4, 7/8 generates bit streams with the respective ratio of zeros to ones. The generation of such a 2^{m}–1 PRBS is done by combining a bit at position n with the bits generated m bits and 2*m bits later. The following formulas are used:

Marker density	Formula
1/8	PRBS[n] = PRBS[n] & PRBS [n+m] & PRBS [n+m*2]
1/4	PRBS[n] = PRBS[n] & PRBS [n+m]
3/4	PRBS[n] = PRBS[n] | PRBS [n+m]
7/8	PRBS[n] = PRBS[n] | PRBS [n+m] | PRBS [n+m*2]

Extended Zeros/Ones

Normal Mode

A pure pseudo random bit stream in normal mode generates the longest run of zeros as the first bits. Thus, for a $2^m - 1$ PRBS the first m bits are zero. With the option Extended Zeros you can extend this sequence of m zeros.

Starting at bit position m+1 in the pattern memory a specified number of bits is set to zero. The following bit is then forced to be a one.

Inverted Mode

The opposite applies for an inverted PRBS. Here you can extend the number of ones sent as the first bits.

Principle of Operation

If ones shall be extended to a normal PRBS, the start phase of the PRBS is modified in such a way that the longest run of ones ends exactly at bit position m (first bit is zero, bit one to m are ones). Therefore, the insert position of the extending ones is again at index m+1.

The same applies for an inverted PRBS that shall be extended with zeros.

Error Insertion

To produce a distinct bit error rate in your test sequence, you can use the Error Insertion option. This option inserts errors into the pseudo random bit stream.

This is achieved by filling a pure PRBS stream into memory and then toggling as many bits as specified at random positions.

For example, 2 errors inserted into a $2^{15}-1$ PRBS yields an error rate of $2 / (2^{15}-1) = 6.1037e^{-5}$.

Pure and Distorted PRWS

A Pseudo Random **W**ord Sequence (PR**W**S) is used to send pseudo random data to a multiplexer (MUX) input port. PRWS is also used to specify the data expected from a demultiplexer (DEMUX) output port.

Pure PRWS

If the signals are connected to the MUX input pins in correct order, a PRBS appears at the output pin of the MUX. This is illustrated in the figure below. The numbers in the figure represent the nth bit of a PRBS data stream.

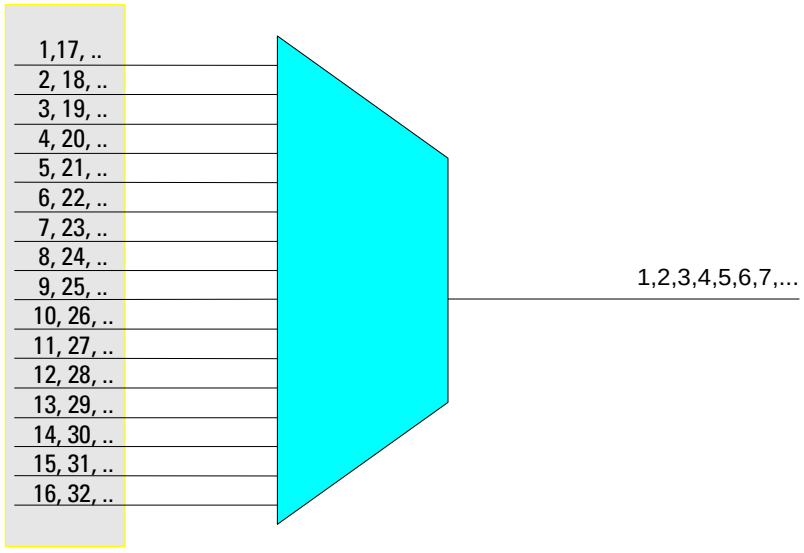


Figure 187 Sending a PRWS to a 16 : 1 MUX

If the MUX-width is a number in power of two, the individual input data streams to the MUX are PRBS streams of the same polynomial as the resulting serial data stream at the MUX output. For a MUX with a width other than a power of two, PRBS using different polynomials must be provided to the MUX. This is done automatically by the software.

Restrictions However, there are a few invalid combinations of PRBS-lengths and MUX-widths. They are listed in the table below.

Table 14 Known Exceptions for MUX-Widths up to 256

PRWS	Invalid MUX-widths
$2^5 - 1$	31, 62, 93, 124, 155, 186, 217, 248
$2^6 - 1$	3, 7, 9, 18, 21, 27, 36, 42, 45, 54, 63, 72, 81, 84, 90, 99, 105, 108, 117, 126, 135, 144, 147, 153, 162, 168, 171, 180, 189, 198, 207, 210, 216, 225, 231, 234, 243, 252
$2^7 - 1$	127, 254
$2^8 - 1$	3, 5, 15, 17, 34, 51, 68, 85, 102, 119, 136, 153, 170, 187, 204, 221, 238, 255
$2^9 - 1$	7, 73, 146, 219
$2^{10} - 1$	3, 11, 31, 33, 66, 93, 99, 132, 165, 198, 231
$2^{11} - 1$	23, 89
$2^{12} - 1$	3, 5, 7, 9, 13, 15, 21, 35, 39, 45, 63, 65, 91, 105, 117, 130, 195
$2^{13} - 1$	No exceptions
$2^{14} - 1$	3, 43, 86, 127, 129, 172, 215
$2^{15} - 1$	7, 31, 151, 217
$2^{23} - 1$	47
$2^{31} - 1$	No exceptions

Some exceptions are obvious: For example, a 2^5–1 PRBS generates a bit stream that is repeated every 31 bits. A multiplexer with 31 inputs would then require constant signals at the input pins to provide a PRBS at the serial output. But as the pure PRBS is generated using a shift register, it is mandatory that the individual inputs to the MUX itself are also PRBS streams of a certain polynomial.

If you need to have a PRWS for a MUX-width that is listed in the exception table, use this workaround: Create a non-pure PRWS. For example, specify it as a PRWS with a variable marker density 1/2.

Distorted PRWS

The different types of non-pure Pseudo Random Words Streams (PRWS) are generated in a similar way as the various PRBS types. The PRWS generation is done in two steps:

1. A PRBS with the specified parameters is generated.
2. The data memories of the related channels are filled in such a way that at the serial side of the MUX the desired PRBS appears.

Index

A

Address Format menu item 83
Allow Multiple Frequencies button 96
Analog Channel Add 122
Analyzer Frontends 20
Analyzer Sampling Point Adjustment 46
Automatic analyzer synchronization 46
Automatic Bit Synchronization 49, 147
Automatic Delay Alignment 48, 147
Automatic Phase Alignment 49, 50, 147

B

BIOS Revisions menu item 88
Bit Error Rate Threshold 148
Block labels 155
Block Length and Sequence Length 150
Block menu 154
Blocks 37
Branch Table 172

C

Cable and DUT board delay compensation 220
Cable delay compensation 218
Capture Data 137
Cascade menu item 89
Changes in ParBERT Rev. 1.0 9
Channel 26
Channel add
  analog 122
  digital 130
Channel Configuration Editor 115, 130
Channel identifier 23
Clock Generator
  trigger output option 103
Clock Modules 17
Clock modules 16
Clock multiplier 101
Clock parameters 93
Clock source 100
Clock Sources 31
Clockgroup 22
Coding 178
Coding menu item 82
Command line editor 225
Command Line menu item 86
Compare and Acquire Around Error 137
Compare and Capture 138
Connect menu item 80
Connection Editor 105, 106
Connection Editor menu item 85
Connector 26
Connectors On/Off button 20, 87, 122, 201
Connectors On/Off menu item 87
Control Menu 86
Controlled operation 62
Copy and paste 227
Copy menu item 79
Cut menu item 79

D

Data Format menu item 83
Data Generator/Analyzer Modules 18
Data ports 12, 27
Data/Sequence Editor 189
Data/Sequence Editor menu item 86
DEFAULT event 172
Delay
  analyzer timing 126
  compensation 216
  generator timing 118
  negative 94
  of system clock 94
  trigger output option 103
Delay Compensation 33
Delay Offset 94
Delete menu item 80
Delete Segment menu item 76
Delete Setting menu item 75
Deskew Editor 216
Deskew Editor menu item 86
Detail Mode Sequence Editor 152
Detail Mode Sequence Editor Window 153
Detect external clock 100
Disconnect menu item 80
Display menu item 84
DSR 24
  DSRA 24
  DSRB 24
Dump Configuration menu item 88
DUT 12
Duty Cycle
  generator timing 118

E

Edit Menu 78
Enable Sync button 146
END label 155
Error Rate Measurement 137
Error State Display 204
Event Handling Principles 52
Events
  Definition 168
  Reactions upon 170
Events menu item 81
Exit menu item 78
Export menu item 76
Export Segment menu item 77
Export Setting menu item 77
Export/Import a Setting 222
Export/Import segments 222
external clock
  detection 100
  frequency measurement 100
External input 101

F

File Menu 73
frequency of system clock 94
Frontends 19
  Analyzers 20
  Generators 20

G

Gated
  external input option 102
Generator Frontends 20
Global System Parameters 92
Go Menu 85
Go to menu item 81

H

handles 25
Hardware Resources
  Channel identifiers 23
  Clockgroup 22
  Identification 22
Help Menu 90
How to
  Add Channels in Analog Mode 122
  Add, Move or Delete Blocks 154
  Change the Characteristics of a Port 109

Change the Characteristics of a Terminal 110
Choose the Clock Source 100
Compensate for Internal and External Delays 216
Create a Memory Segment 178
Create a New Segment 176
Create a Port 107
Create a PRBS/PRWS Segment 185
Define Events 168
Develop a device test 55
Download the Test Sequence 200
Edit a Stored Segment 187
Execute Firmware Commands 225
Export/Import Settings or Segments 222
Move Captured Data Into a Segment 206
Run a test 58
Save the test setting 60
Set the Characteristics of the External Input 101
Set the Characteristics of the Trigger Output 103
Set the System Clock Parameters 93
Set up a test 56
Specify the Reactions To Events 170
Start the Agilent 81200 Software 62
Start the Connection Editor 106
Start the Data/Sequence Editor 190
Start/Stop the Test 201
Synchronize an Analyzer With Incoming Data 145
Use the Standard Mode Sequence Editor 141
View and investigate test results 59
View BER Test Results 200
View Captured Test Results 204
View Waveforms 209

I

Immediate
external input option 102
Impedance
trigger output 103
Import menu item 76
Import Segment menu item 76
Import Setting menu item 76
Input type
analyzer setup 129
Insert after menu item 80
Insert before menu item 80
Insert menu item 80
Instrument Configuration 26
Instrument connector adjustment 217
Invert menu item 82

L

Length
of blocks 150
of segments 150
Levels
analyzer setup 127
generator setup 120
trigger output 103
Local operation 62

M

Main menu 61
Mainframes and Controllers 14
Manual Analyzer Sampling Point Alignment 47
Measure external clock 100
Measurement Configuration menu item 85
Menu items
Address Format 83
BIOS Revisions 88
Cascade 89
Coding 82
Command Line 86
Connect 80
Connection Editor 85
Connectors On/Off 87
Copy 79
Cut 79
Data Format 83
Data/Sequence Editor 86
Delete 80
Delete Segment 76
Delete Setting 75
Deskew Editor 86
Disconnect 80
Display 84
Dump Configuration 88
Events 81
Exit 78
Export 76
Export Segment 77
Export Setting 77
Go to 81
Import 76
Import Segment 76
Import Setting 76
Insert 80
Insert after 80
Insert before 80
Invert 82
Measurement Configuration 85
Mirror 82
Module Selftest 88
Move 80
New Segment 75
New Setting 73
Open Segment 75
Open Setting 74
Parameter Editor 85
Paste 79
Paste after 80
Paste before 79
Power On Test 88
Prepare 87
Properties 82
Reconnect Application 78
Reconnect Server 77
Rename 82
Result Display 83
Run 87
Save Segment 75
Save Segment As 76
Save Setting 74
Save Setting As 75
Set Start 81
Set to 82
Signals 84
Stop 87
System Selftests 88
Tile 89
Trigger 81
Waveform Viewer 86
Zoom 84
Menus
Control 86
Edit 78
File 73
Go 85
Help 90
System 88
Tools 82
View 83
Window 89
Mirror menu item 82
Module frontends 19
Module Selftest menu item 88
Modules 16
clock 17
frontends 19
Generator/Analyzer 18
generator/analyzer 18
Move menu item 80

N

Negative Delay 34
New Segment menu item 75
New Setting menu item 73

O

Open Segment menu item 75
Open Setting menu item 74
Open VXI 15
Operating mode
Controlled 62
Local 62
Remote 62

P

Parameter Editor 105, 115, 116
Parameter Editor menu item 85
Paste after menu item 80
Paste before menu item 79
Paste menu item 79
period of system clock 94
Phase Accuracy 148
Plug and play (pnp) drivers 29
pnp 29
polarity
  analyzer setup 129
  generator setup 122
Port
  data 12
  pulse 12
Ports 12, 27, 107
  types of 12
Power On Test menu item 88
Prepare button 87, 200
Prepare menu item 87
Procedure
  for Running the Test 58
  for Saving the Test Setting 60
  for Setting Up the Test 56
  for Viewing Test Results 59
Properties menu item 82
Pulse ports 12, 27

R

Reconnect Application menu item 78
Reconnect Server menu item 77
Remote operation 62
Rename menu item 82
Result Display menu item 83
rise/fall time
  generator setup 123
Run button 87, 201
Run menu item 87

S

Save Segment As menu item 76
Save Segment menu item 75
Save Setting As menu item 75
Save Setting menu item 74
Segment
  Memory type 39
  PRBS 39
  Properties 39
Segment Editor 178
Segment Pool 177
Segment Pools 39
Segment Resolution 31
Segment resolution 94
Segment Type 177
Segment Types 38
Segments 28, 36
Sense polarity
  external input option 102
Sequence 36
Sequencer
  trigger output option 103
Set Start menu item 81
Set to menu item 82
Setting 28
Signals menu item 84
Software Structure 28
Standard Mode Sequence Editor 140
  Special characteristics 149
Start
  external input option 102
START label 155
State Coding 178
Stop
  external input option 102
Stop button 87
Stop menu item 87
Switching between Sequence Editors 145, 151
Synchronization Block Indicators 148
System Capabilities 12
System Components 13
System Menu 88
System Modules 16
System Selftests menu item 88

T

Terminals 27
termination
  analyzer setup 128
  generator setup 121
termination voltage
  analyzer setup 127
  external input option 102
  generator setup 120
  trigger output 103
threshold voltage
  external input option 102
  for analyzers 127
Tile menu item 89
tool bar 61
Tools Menu 82
traces 38
Trigger menu item 81
Trigger output 103
Trigger Pod
  Description 22
Trigger width 168
Trigger-Controlled Start and Stop 34

U

Use Single Frequency button 98

V

vectors 38
View / Edit Branches menu item 171
View Menu 83
Virtual Instruments/Systems 24
VXI
  Open VXI Configurations 15

W

Waveform Viewer
  Display description 210
  Operation 211
  Sample mode 209
  Start procedure 209
  Time mode 209
Waveform Viewer menu item 86
Width
  generator timing 118
Window Menu 89

Z

Zoom menu item 84