



# VX6090 PBIT

SD.DT.G65-0e - December 2016

 VX6090 PBIT User's Guide

Kontron would like to point out that the information contained in this manual may be subject to alteration, particularly as a result of the constant upgrading of Kontron products. This document does not entail any guarantee on the part of Kontron with respect to technical processes described in the manual or any product characteristics set out in the manual. Kontron assumes no responsibility or liability for the use of the described product(s), conveys no license or title under any patent, copyright or mask work rights to these products and makes no representations or warranties that these products are free from patent, copyright or mask work right infringement unless otherwise specified. Applications that are described in this manual are for illustration purposes only. Kontron makes no representation or warranty that such application will be suitable for the specified use without further testing or modification. Kontron expressly informs the user that this manual only contains a general description of processes and instructions which may not be applicable in every individual case. In cases of doubt, please contact Kontron.

This manual is protected by copyright. All rights are reserved by Kontron. No part of this document may be reproduced, transmitted, transcribed, stored in a retrieval system, or translated into any language or computer language, in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise), without the express written permission of Kontron. Kontron points out that the information contained in this manual is constantly being updated in line with the technical alterations and improvements made by Kontron to the products and thus this manual only reflects the technical status of the products by Kontron at the time of publishing.

Brand and product names are trademarks or registered trademarks of their respective owners.

© 2016 by Kontron AG

Lise-Meitner-Str. 3-5

86156 Augsburg

Germany

[www.kontron.com](http://www.kontron.com)

## REVISION HISTORY

PUBLICATION TITLE:		VX6090 PBIT User's Guide
DOC. ID:		SD.DT.G65-0e
Revision	Brief Description of Changes	Date of Issue
0e	Initial Issue	12-2016

## SYMBOLS

The following symbols may be used in this manual:

### DANGER

DANGER indicates a hazardous situation which, if not avoided, will result in death or serious injury.

### WARNING

WARNING indicates a hazardous situation which, if not avoided, could result in death or serious injury.

### CAUTION

CAUTION indicates a hazardous situation which, if not avoided, may result in minor or moderate injury.

### NOTICE

NOTICE indicates a property damage message.



#### **Electric Shock!**

This symbol and title warn of hazards due to electrical shocks (> 60V) when touching products or parts of them. Failure to observe the precautions indicated and/or prescribed by the law may endanger your life/health and/or result in damage to your material. See also to the "High-Voltage Safety Instructions" portion below in this section.



#### **ESD Sensitive Device!**

This symbol and title inform that the electronic boards and their components are sensitive to static electricity. Care must therefore be taken during all handling operations and inspections of this product in order to ensure product integrity at all times.



#### **HOT Surface!**

Do NOT touch! Allow to cool before servicing.



This symbol indicates general information about the product and the user manual.

This symbol also indicates detail information about the specific product configuration.



This symbol precedes helpful hints and tips for daily use.

## FOR YOUR SAFETY

Your new Kontron product was developed and tested carefully to provide all features necessary to ensure its compliance with electrical safety requirements. It was also designed for a long fault-free life. However, the life expectancy of your product can be drastically reduced by improper treatment during unpacking and installation. Therefore, in the interest of your own safety and of the correct operation of your new Kontron product, you are requested to conform with the following guidelines.

### High Voltage Safety Instructions

As a precaution, in case of danger, the power connector is the product's main disconnect device and must be easily accessible.

#### **▲ CAUTION**

##### **Warning!**

All operations on this device must be carried out by sufficiently skilled personnel only.



##### **Caution, Electric Shock!**

Before installing a not hot-swappable Kontron product into a system always ensure that your mains power is switched off. This applies also to the installation of piggybacks. Serious electrical shock hazards can exist during all installation, repair and maintenance operations with this product. Therefore, always unplug the power cable and any other cables which provide external voltages before performing work.

Earth ground connection to vehicle's chassis or a central grounding point shall remain connected. The earth ground cable shall be the last disconnected or the first connected during operations of cabling.

### Special Handling and Unpacking Instructions



##### **ESD Sensitive Device!**

Electronic boards and their components are sensitive to static electricity. Therefore, care must be taken during all handling operations and inspections of this product, in order to ensure product integrity at all times

Do not handle this product out of its protective enclosure while it is not used for operational purposes unless it is otherwise protected.

Whenever possible, unpack or pack this product only at EOS/ESD safe work stations. Where a safe work station is not guaranteed, it is important for the user to be electrically discharged before touching the product with his/her hands or tools. This is most easily done by touching a metal part of your system housing.

It is particularly important to observe standard anti-static precautions when changing piggybacks, ROM devices, jumper settings etc. If the product contains batteries for RTC or memory backup, ensure that the board is not placed on conductive surfaces, including anti-static plastics or sponges. They can cause short circuits and damage the batteries or conductive circuits on the board.

## GENERAL INSTRUCTIONS ON USAGE

In order to maintain Kontron's product warranty, this product must not be altered or modified in any way. Changes or modifications to the device, which are not explicitly approved by Kontron and described in this manual or received from Kontron's Technical Support as a special handling instruction, will void your warranty.

This device should only be installed in or connected to systems that fulfill all necessary technical and specific environmental requirements. This applies also to the operational temperature range of the specific board version, which must not be exceeded. If batteries are present, their temperature restrictions must be taken into account.

In performing all necessary installation and application operations, follow only the instructions supplied by the present manual.

Keep all the original packaging material for future storage or warranty shipments. If it is necessary to store or ship the board, re-pack it as nearly as possible in the manner in which it was delivered.

Special care is necessary when handling or unpacking the product. Consult the special handling and unpacking instruction.

Kontron products may be equipped with parts from Japanese manufacturers. Customers must ensure that final Kontron products destination is not impacted by this condition.

## ENVIRONMENTAL PROTECTION STATEMENT

This product has been manufactured to satisfy environmental protection requirements where possible. Many of the components used (structural parts, printed circuit boards, connectors, batteries, etc.) are capable of being recycled.

Final disposition of this product after its service life must be accomplished in accordance with applicable country, state, or local laws or regulations.




---

Environmental protection is a high priority with Kontron.  
Kontron follows the DEEE/WEEE directive.  
You are encouraged to return our products for proper disposal.

---

The Waste Electrical and Electronic Equipment (WEEE) Directive aims to:

- ▶ reduce waste arising from electrical and electronic equipment (EEE)
- ▶ make producers of EEE responsible for the environmental impact of their products, especially when they become waste
- ▶ encourage separate collection and subsequent treatment, reuse, recovery, recycling and sound environmental disposal of EEE
- ▶ improve the environmental performance of all those involved during the lifecycle of EEE

## TRADEMARKS

This document may include names, company logos and trademarks, which are registered trademarks and, therefore, proprietary to their respective owners.

# Table Of Contents

<b>1 /</b>	<b>PBIT Overview</b> .....	<b>1</b>
1.1	Related Documents .....	1
1.2	PBIT Installation and Activation .....	1
1.3	PBIT Configuration .....	2
1.3.1	Configure the PBIT by command line .....	2
1.3.2	Configure the PBIT to run automatically .....	5
1.4	Synthetic PBIT Result .....	6
1.5	PBIT Tests List .....	7
1.5.1	Selected Tests List .....	7
1.5.2	Not Selected Tests List .....	9
1.6	PBIT Execution Time .....	10
<b>2 /</b>	<b>PBIT Command Line Reference Guide</b> .....	<b>12</b>
2.1	On-line Help .....	12
2.2	Display the List of Selected Tests .....	13
2.3	Execute the PBIT from the Command Line .....	16
2.4	Execute the PBIT in Loop Mode .....	17
2.5	Get the PBIT Results .....	17
2.6	Clear the PBIT Results .....	18
2.7	Configure the PBIT Tests List to Execute .....	18
2.7.1	Run mode parameters .....	18
2.7.2	Adding a Test to the Current Run List .....	18
2.7.3	Removing a Test from the Current Run List .....	19
2.7.4	Set a RUN mode parameter to All the Tests of the Current Run List .....	19
2.7.5	Restore the Default Run List .....	22
2.8	Run the PBIT in Silent Mode .....	22
2.9	Display the PBIT Version .....	23
2.10	PBIT System Test .....	23
2.10.1	Recording the System Configuration .....	23
2.10.2	Testing the Current System Configuration .....	25
2.10.3	Editing the System Configuration .....	26
2.10.3.1	Edit System Items (to Ignore Specific Test) .....	26
2.10.3.2	Print Debug Settings .....	37
2.10.4	Clearing System Stats .....	41
2.11	Bypass the PBIT Tests .....	42
<b>3 /</b>	<b>PBIT and OS Interfaces</b> .....	<b>43</b>
3.1	Linux .....	43
3.1.1	Linux Synthetic PBIT Result .....	43
3.1.2	Linux Detailed PBIT Result .....	43
3.1.3	Linux Kdiag Utility .....	43
<b>4 /</b>	<b>Use Cases</b> .....	<b>52</b>
4.1	PBIT Features and Benefits .....	52
4.1.1	EVAL .....	52
4.1.2	DEVEL .....	52
4.1.3	DEPLOY .....	52
4.1.4	MAINTAIN .....	53
4.1.5	MANUFACTURING .....	53
	<b>Appendix A - List of Acronyms</b> .....	<b>54</b>

# 1 / PBIT Overview

This document describes the PowerOn Built In Test (PBIT) for Kontron VX6090 boards.

In this document, the EFI shell prompt **VX6090>** is a shortcut for **VX6090-A>** or **VX6090-B>**. However the output messages may be slightly different depending on the CPU side.

The PBIT is an optional product available under the VX6090 EFI BIOS shell environment. It is implemented as a binary executable located in the system Flash and included in the BIOS shell application. The PBIT configuration such as tests list and tests result is stored in the VX6090 OS EEPROM.

The PBIT includes among others the following services:

- ▶ It offers a list of tests that can be added or removed from a run list by command according to the desired trade-off between time to boot, coverage rate and system dependent configuration.
- ▶ It also offers a system test that can quickly spot any configuration change.
- ▶ It can be run automatically (when booting firmware) or in an interactive mode (at EFI BIOS Shell firmware prompt).
- ▶ Tests configuration and results are stored in the OS EEPROM and can also be accessed and reconfigured under Operating System such as Linux. See Chapter 3 page 43.
- ▶ Simplified test result is also available in a 8-bit PLD register (register 0x2).

## 1.1 Related Documents

Hardware:

- ▶ VX6090 User's Guide ..... CA.DT.B20

BIOS:

- ▶ VX6090 AMI BIOS User Reference Manual ..... SD.DT.G66

## 1.2 PBIT Installation and Activation

The PBIT software comes pre-installed in the system Flash, along with the EFI BIOS firmware, on the VX6090 boards.

The PBIT can be activated on any VX6090. Please contact Kontron support team for more information.

To install a new BIOS version including a new PBIT version please refer to the VX6090 BIOS User's Reference Manual - SD.DT.G66.

## 1.3 PBIT Configuration

The PBIT must be configured first by an EFI shell command line.

The PBIT is presented as a list of tests to be executed. Each test is focused on a specific device of the VX6090.

The list of tests to be executed can be displayed and modified by using the EFI Shell command "**kdiag**" (see section 1.5 page 7).

### 1.3.1 Configure the PBIT by command line

The following explains how to configure and execute the PBIT by a command line.

- ▶ Select "**Built-in EFI Shell**" as the first boot device:  
Enter the BIOS setup by pressing the <F2> keyboard key and select the Boot menu.  
Select "**Built-in EFI Shell**" as the Boot Option #1 (use key <+> or <->).  
Then, in the Save & Exit menu select «**Saving Changes and Reset**».

```

Aptio Setup Utility - Copyright (C) 2016 American Megatrends, Inc.
Main  Advanced  IntelRCSetup  Kontron  Security  Boot  Event Logs
-----
| Boot Configuration                                     |Number of seconds to
| Setup Prompt Timeout                               1 |wait for setup
| Bootup NumLock State                               [On] |activation key.
| Quiet Boot                                         [Disabled] |65535 (0xFFFF) means
|                                                    |indefinite waiting.
|
| Boot Option Priorities                               |
| Boot Option #1                                     [UEFI: Built-in EFI ...] |
| Boot Option #2                                     [UEFI: KingstonDataT...] |
| Boot Option #3                                     [P1: WDC WD2503ABYZ-...] |
|
| New Boot Option Polic [Default]                    |><: Select Screen
|                                                    |^v: Select Item
| Hard Drive BBS Priorities                          |Enter: Select
|                                                    |+/-: Change Opt.
|                                                    |F1: General Help
|                                                    |F2: Previous Values
|                                                    |F3: Optimized Defaults
|                                                    |F4: Save & Exit
|                                                    |ESC: Exit
-----
Version 2.17.1255. Copyright (C) 2016 American Megatrends, Inc.

```

After reset, the EFI shell prompt is displayed, allowing to enter the PBIT commands.

- ▶ Verify the PBIT version:

```
VX6090> kdiag version
PBIT VERSION 1.4 ID16293
```

- ▶ Launch the PBIT manually for verification:

```
VX6090> kdiag run
PBIT "mem_data" (fast,simple) PASSED
PBIT "mem_addr" (fast,simple) PASSED
PBIT "mem_pattern1" (slow,simple) PASSED
PBIT "mem_pattern2" (slow,simple) PASSED
PBIT "mem_pattern3" (slow,simple) PASSED
PBIT "mem_pattern4" (slow,simple) PASSED
PBIT "core" (fast,simple) AES disabled, 8 core board PASSED
PBIT "tpm" (fast,simple) PASSED
PBIT "m2_bottom" (fast,simple) NO M2S2 on CPUA SIDE, TEST BYPASSED PASSED
PBIT "m2_top" (fast,simple) M2S1A CONNECTED TO SATA, TEST BYPASSED PASSED
PBIT "serial" (fast,simple) PASSED
PBIT "onboardvideo" (fast,simple) PASSED
PBIT "rtc" (fast,simple) PASSED
PBIT "sysflash" (fast,simple) PASSED
PBIT "cpld" (fast,simple) PASSED
PBIT "fram" (fast,simple) PASSED
PBIT "ether_loop0" (fast,simple) PASSED
PBIT "ether_loop1" (fast,simple) PASSED
PBIT "ether_loop2" (fast,simple) PASSED
PBIT "ether_loop3" (fast,simple) PASSED
PBIT "hwmon" (fast,simple) PASSED
PBIT "sata0_controler" (fast,simple) PASSED
PBIT "sata1_controler" (fast,simple) DISABLED PASSED
PBIT "vpd" (fast,simple) PASSED
PBIT "eeprom" (fast,simple) PASSED
PBIT "ehci_controller" (fast,simple) PASSED
PBIT "xhci_controller" (fast,simple) PASSED
PBIT "system" (fast,simple)
SYSTEM INFOS NOT SAVED
PASSED
VX6090>
```

- ▶ Configure the "system" test:

By default the test named "system" is not "ready". The end user should record the system configuration when the system is ready for this. All the equipments must be connected such as other CPU boards, XMC, VPX switch boards, Ethernet connections to external switches, USB devices, SATA devices, BIOS setup configuration. See section 2.10 page 23 "PBIT System Test" for more details.

To record and then activate this test , run the following:

```
VX6090> kdiag learn system
Seg  Bus  Dev  Func
---  ---  ---  ----
00   00   00   00 ==> Bridge Device - Host/PCI bridge
      Vendor 8086 Device 6F00 Prog Interface 0
00   00   01   00 ==> Bridge Device - PCI/PCI bridge
      Vendor 8086 Device 6F02 Prog Interface 0
00   00   02   00 ==> Bridge Device - PCI/PCI bridge
      Vendor 8086 Device 6F04 Prog Interface 0
00   00   02   02 ==> Bridge Device - PCI/PCI bridge
      Vendor 8086 Device 6F06 Prog Interface 0
00   00   03   00 ==> Bridge Device - PCI/PCI bridge
      Vendor 8086 Device 6F08 Prog Interface 0
00   00   03   02 ==> Bridge Device - PCI/PCI bridge
      Vendor 8086 Device 6F0A Prog Interface 0
00   00   03   03 ==> Bridge Device - PCI/PCI bridge
      Vendor 8086 Device 6F0B Prog Interface 0
00   00   05   00 ==> Base System Peripherals - Other system peripheral
      Vendor 8086 Device 6F28 Prog Interface 0
```

```

00 00 05 01 ==> Base System Peripherals - Other system peripheral
Vendor 8086 Device 6F29 Prog Interface 0
00 00 05 02 ==> Base System Peripherals - Other system peripheral
Vendor 8086 Device 6F2A Prog Interface 0
00 00 05 04 ==> Base System Peripherals - PIC
Vendor 8086 Device 6F2C Prog Interface 20
00 00 14 00 ==> Serial Bus Controllers - USB
Vendor 8086 Device 8C31 Prog Interface 30
00 00 1C 00 ==> Bridge Device - PCI/PCI bridge
Vendor 8086 Device 8C10 Prog Interface 0
00 00 1C 02 ==> Bridge Device - PCI/PCI bridge
Vendor 8086 Device 8C14 Prog Interface 0
00 00 1D 00 ==> Serial Bus Controllers - USB
Vendor 8086 Device 8C26 Prog Interface 20
00 00 1F 00 ==> Bridge Device - PCI/ISA bridge
Vendor 8086 Device 8C54 Prog Interface 0
00 00 1F 02 ==> Mass Storage Controller - Serial ATA controller
Vendor 8086 Device 8C02 Prog Interface 1
00 00 1F 03 ==> Serial Bus Controllers - System Management Bus
Vendor 8086 Device 8C22 Prog Interface 0
00 02 00 00 ==> Base System Peripherals - Other system peripheral
Vendor 8086 Device 6F50 Prog Interface 0
00 02 00 01 ==> Base System Peripherals - Other system peripheral
Vendor 8086 Device 6F51 Prog Interface 0
00 02 00 02 ==> Base System Peripherals - Other system peripheral
Vendor 8086 Device 6F52 Prog Interface 0
00 02 00 03 ==> Base System Peripherals - Other system peripheral
Vendor 8086 Device 6F53 Prog Interface 0
00 03 00 00 ==> Network Controller - Ethernet controller
Vendor 8086 Device 15AB Prog Interface 0
00 03 00 01 ==> Network Controller - Ethernet controller
Vendor 8086 Device 15AB Prog Interface 0
00 05 00 00 ==> Network Controller - Ethernet controller
Vendor 8086 Device 1537 Prog Interface 0
00 06 00 00 ==> Network Controller - Ethernet controller
Vendor 8086 Device 1533 Prog Interface 0
00 08 00 00 ==> Display Controller - VGA/8514 controller
Vendor 126F Device 0750 Prog Interface 0

CPLD Version 0x43
DRAM size 8 GB
System Controller
Geographical Address 1
XMC not present
SATA PORT 4 TS32GMTS400 (32.0GB)
Device detected on SMBUS0, address = 0x30
BIOS Setup Checksum : 124299
BIOS Version : ID16293EF
ETH0 connected device speed 10000Mb/s
ETH1 connected device speed 10000Mb/s

Link speed 1000Mb/s ETH3 connected device speed 0Mb/s
1 Drive, 1 Hub
MassStorage Device name (0) : 0.00
USB Port 3 connected

Number of System Test Elements detected : 41
DRAM area [ 0x33CBA9E0 0x33CBB780 ] will be stored in EEPROM
Storing system infos...

Storing system configuration...
VX6090>

```

- ▶ Check the PBIT results:

```
VX6090> kdiag stat
Status of PBITs configured to run from command line :
PASSED : mem_data(1) (fast,simple)
PASSED : mem_addr(2) (fast,simple)
PASSED : mem_pattern1(6) (slow,simple)
PASSED : mem_pattern2(7) (slow,simple)
PASSED : mem_pattern3(8) (slow,simple)
PASSED : mem_pattern4(9) (slow,simple)
PASSED : core(10) (fast,simple)
PASSED : tpm(11) (fast,simple)
PASSED : m2_bottom(14) (fast,simple)
PASSED : m2_top(15) (fast,simple)
PASSED : serial(16) (fast,simple)
PASSED : onboardvideo(17) (fast,simple)
PASSED : rtc(20) (fast,simple)
PASSED : sysflash(22) (fast,simple)
PASSED : cp1d(24) (fast,simple)
PASSED : fram(40) (fast,simple)
PASSED : ether_loop0(55) (fast,simple)
PASSED : ether_loop1(56) (fast,simple)
PASSED : ether_loop2(57) (fast,simple)
PASSED : ether_loop3(58) (fast,simple)
PASSED : hwmon(61) (fast,simple)
PASSED : sata0_controller(68) (fast,simple)
PASSED : sata1_controller(69) (fast,simple)
PASSED : vpd(70) (fast,simple)
PASSED : eeprom(71) (fast,simple)
PASSED : ehci_controller(85) (fast,simple)
PASSED : xhci_controller(86) (fast,simple)
PASSED : system(89) (fast,simple)

RUN      : 28
PASSED  : 28
FAILED  : 0
NOT_RUN : 0

VX6090>
```

### 1.3.2 Configure the PBIT to run automatically

The PBIT uses BIOS environment variables to run automatically at the end of the BIOS boot and before the Operating System boot:

- ▶ Configure the PBIT to be launched at boot time:

The automatic start is activated using the environment variable "**BootCmd**".

```
VX6090> set BootCmd "kdiag run"
```

The delay before executing the **BootCmd** is given by the variable **BootDelay** which is expressed in seconds.

Default value is "5". Value "0" is possible.

```
VX6090> set BootDelay 1
```

- ▶ Verify:

```
VX6090> set
BootCmd = "kdiag run"
BootDelay = 1
    path = .\;FS0:\efi\tools\;FS0:\efi\boot\;FS0:\
debuglasterror = 0x0
lasterror = 0x0
profiles = ;Install;Debug1;Driver1;network1;
uefishellsupport = 3
uefishellversion = 2.1
uefiversion = 2.40
```

- ▶ Then reset the system:

```
VX6090> reset
```

The PBIT will be launched automatically. When finished, the BIOS boots from the next valid device in the boot list.



To stop under the EFI shell after the PBIT execution, define the variable named "StopEfiShell":

```
VX6090> set StopEfiShell 1
```

## 1.4 Synthetic PBIT Result

A 8-bit synthetic PBIT result can be read in the CPLD register 0x2.

This register is accessible under the Operating System using the CPLD OS facility or a direct memory I/O access at address 0x802.

Under the BIOS EFI shell use the following command:

```
VX6090> kpld -r 2
READ : @0x2 = 0x1
```

The 8-bit register 0x2 content is the following (reset value=0):

TEST FAIL NUMBER 1.. 128							RUN
7	6	5	4	3	2	1	0

Bit 0: 0 = NOT RUN                      1 = ALL RUN

Bit 1..7: if All 0 => No FAILED test, if NOT 0 then indicates ID number of first failing test

- ▶ Examples:

0x00 => PBIT not run

0x03 => ALL Tests run and Test 1 FAILED

0x61 => ALL Tests run and Test 48 FAILED (48 = 0x30, 0x30 << 1 = 0x60)

0x01 => ALL Tests run and PASSED



To identify a PBIT test by its number, use the command "kdiag [PBITnumber]":

Example:

```
VX6090> kdiag 16
serial (16) - Checks the serial line COM2
capabilities : fast,simple/complex,allresets
run mode 1   : fast,simple,allresets
```



This register is set to 0 at each hardware reset. It can safely be written to 0 at any time.

## 1.5 PBIT Tests List

The PBIT tests list comes in two parts: a default list of selected tests and a list of additional not selected tests.

This can be changed by the user to fulfill his specific coverage and execution time requirements. The "**kdiag**" command displays the 2 lists. Note: the initial default tests list can be restored with the "**kdiag default**" command

### 1.5.1 Selected Tests List

The default selected tests list contains all the diagnostics that can be run without any specific equipment. All the tests have been designed to be safe for the system containing a VX6090. No signal on any connector will be modified during the default test execution.

The command "**kdiag**" displays the default tests list to run:

```
VX6090> kdiag
PBITs configured to run from command line :
mem_data (1) - Checks Memory/ECC data lines
  capabilities : fast,simple,allresets
  run mode 1  : fast,simple,allresets
mem_addr (2) - Checks Memory/ECC address lines
  capabilities : fast,simple,allresets
  run mode 1  : fast,simple,allresets
mem_pattern1 (6) - Checks Memory/ECC using pattern 0xFFFFFFFF
  capabilities : slow/fast,simple,allresets
  run mode 1  : slow,simple,allresets
mem_pattern2 (7) - Checks Memory/ECC using pattern 0x55555555
  capabilities : slow/fast,simple,allresets
  run mode 1  : slow,simple,allresets
mem_pattern3 (8) - Checks Memory/ECC using pattern 0xAAAAAAAA
  capabilities : slow/fast,simple,allresets
  run mode 1  : slow,simple,allresets
mem_pattern4 (9) - Checks Memory/ECC using pattern 0x00000000
  capabilities : slow/fast,simple,allresets
  run mode 1  : slow,simple,allresets
core (10) - Checks Core count and AES disabled
  capabilities : fast,simple,allresets
  run mode 1  : fast,simple,allresets
tpm (11) - Checks TPM access
  capabilities : fast,simple/complex,allresets
  run mode 1  : fast,simple,allresets
m2_bottom (14) - Checks M.2 bottom module access
  capabilities : fast,simple/complex,allresets
  run mode 1  : fast,simple,allresets
m2_top (15) - Checks M.2 top module access
  capabilities : fast,simple/complex,allresets
  run mode 1  : fast,simple,allresets
serial (16) - Checks the serial line COM2
  capabilities : fast,simple/complex,allresets
  run mode 1  : fast,simple,allresets
onboardvideo (17) - Checks the onboard video component
  capabilities : fast,simple,allresets
  run mode 1  : fast,simple,allresets
```

```

rtc (20) - Checks the RTC time
  capabilities : fast,simple,allresets
  run mode 1   : fast,simple,allresets
sysflash (22) - Checks the BIOS rescue in system flash
  capabilities : fast,simple,allresets
  run mode 1   : fast,simple,allresets
cpld (24) - Checks PLD, GeoAddress, watchdog
  capabilities : fast,simple/complex,allresets
  run mode 1   : fast,simple,allresets
fram (40) - Checks F-RAM device.
  capabilities : fast,simple/complex,allresets
  run mode 1   : fast,simple,allresets
ether_loop0 (55) - Checks ethernet interface 0 in loopback mode
  capabilities : fast,simple/complex,allresets
  run mode 1   : fast,simple,allresets
ether_loop1 (56) - Checks ethernet interface 1 in loopback mode
  capabilities : fast,simple/complex,allresets
  run mode 1   : fast,simple,allresets
ether_loop2 (57) - Checks ethernet interface 2 in loopback mode
  capabilities : fast,simple/complex,allresets
  run mode 1   : fast,simple,allresets
ether_loop3 (58) - Checks ethernet interface 3 in loopback mode
  capabilities : fast,simple/complex,allresets
  run mode 1   : fast,simple,allresets
hwmon (61) - Checks hardware monitoring, temperature and voltage sensors
  capabilities : fast,simple/complex,allresets
  run mode 1   : fast,simple,allresets
sata0_controller (68) - Checks SATA controller 0 (D31:F2)
  capabilities : fast,simple/complex,allresets
  run mode 1   : fast,simple,allresets
sata1_controller (69) - Check SATA controller 1 (D31:F5)
  capabilities : fast,simple/complex,allresets
  run mode 1   : fast,simple,allresets
vpd (70) - Checks VPD data required for board operation.
  capabilities : fast,simple/complex,allresets
  run mode 1   : fast,simple,allresets
eeprom (71) - Checks User EEPROM (0xA2)
  capabilities : fast,simple/complex,allresets
  run mode 1   : fast,simple,allresets
ehci_controller (85) - Check EHCI controller
  capabilities : fast,simple,allresets
  run mode 1   : fast,simple,allresets
xhci_controller (86) - Check xHCI controller
  capabilities : fast,simple,allresets
  run mode 1   : fast,simple,allresets
system (89) - Checks system configuration SETUP,PCIe,SATA,USB,ETH stability
  capabilities : fast,simple/complex,allresets
  run mode 1   : fast,simple,allresets

```

To run the default PBIT, enter the command:

```
VX6090> kdiag run
```

## 1.5.2 Not Selected Tests List

The second part of the list includes all the tests not currently selected for execution. These tests appear at the end of the "kdiag" command after the message "Other PBITs available but not yet configured":

```
VX6090> kdiag
(...)
Other PBITs available but not yet configured :

mem_bitflip (3) - Checks Mem/ECC using bit-flip pattern ((1 << (offset % 64))
  capabilities : slow/fast,simple,allresets
mem_addrpat (4) - Checks Memory/ECC using address pattern (offset)
  capabilities : slow/fast,simple,allresets
mem_addrpat2 (5) - Checks Memory/ECC using address pattern (~offset)
  capabilities : slow/fast,simple,allresets
pcie_vpx_sw (13) - Checks the PCI-Express backplane VPX switch
  capabilities : fast,simple/complex,allresets
smbus0 (26) - Check SMBUS0 between PLD and backplane
  capabilities : fast,simple/complex,allresets
smbus1 (27) - Check SMBUS1 between PLD and backplane
  capabilities : fast,simple/complex,allresets
ether_link0 (50) - Checks the link status on ethernet interface 0.
  capabilities : fast,simple/complex,allresets
ether_link1 (51) - Checks the link status on ethernet interface 1.
  capabilities : fast,simple/complex,allresets
ether_link2 (52) - Checks the link status on ethernet interface 2.
  capabilities : fast,simple/complex,allresets
ether_link3 (53) - Checks the link status on ethernet interface 3.
  capabilities : fast,simple/complex,allresets
sata0_dev_see (62) - Checks if device is present on SATA0
  capabilities : fast,simple,allresets
sata1_dev_see (63) - Checks if device is present on SATA1
  capabilities : fast,simple,allresets
sata2_dev_see (64) - Checks if device is present on SATA2
  capabilities : fast,simple,allresets
sata4_dev_see (66) - Checks if device is present on SATA4 (M.2 Top slot)
  capabilities : fast,simple,allresets
sata5_dev_see (67) - Checks if device is present on SATA5 (M.2 Bottom slot)
  capabilities : fast,simple,allresets
usb0_dev_see (80) - Check USB device presence on USB FP
  capabilities : fast,simple,allresets
usb1_dev_see (81) - Check USB device presence on USB1 (A=RTM USB C,B=RTM USB E)
  capabilities : fast,simple,allresets
usb3_dev_see (83) - Check USB device presence on USB0 (A=RTM USB A-B,B=RTM USB D) or USBSS
(A only)
  capabilities : fast,simple,allresets
faultytest (98) - A dummy test that returns FAIL
  capabilities : fast,simple,allresets
hangtest (99) - A dummy test that will hang
  capabilities : fast,simple,allresets
```

The unselected tests list contains:

- ▶ Memory tests to complete the default memory tests. Long execution time.
- ▶ SMBus tests: I2C tests on the backplane, reserved for complex test with specific external equipment. Do not use it.
- ▶ External equipment dependent tests to check Ethernet link, XMC presence, connected SATA and USB Devices. Only use them when it is necessary to verify a device presence.

- ▶ Utility tests: "**faultytest**" helps you to test the error reporting mechanism. "**hangtest**" is useful for watchdog recovery checking.



All the USB device detection tests are limited to USB Mass storage devices. They will not detect USB keyboard or mouse. Use system test number 89 if you want to check USB mouse or keyboard presence.

## 1.6 PBIT Execution Time

The default PBIT (see section 1.5.1 page 7) runs in about 30 seconds for a 8-GB DRAM board.

Below is the execution time for each test and for a board with 8-GB DRAM.

Tests with a significant duration compared to the other tests are highlighted in red.

TEST NAME	TEST ID NUMBER	EXECUTION TIME
mem_data (fast,simple)	1	< 350 ms
mem_addr (fast,simple)	2	800 ms
mem_bitflip (slow,simple)	3	5.5 s
mem_addrpat (slow,simple)	4	4.5 s
mem_addrpat2 (slow,simple)	5	4.9 s
mem_pattern1 (slow,simple)	6	4.6 s
mem_pattern2 (slow,simple)	7	4.8 s
mem_pattern3 (slow,simple)	8	4.8 s
mem_pattern4 (slow,simple)	9	4.8 s
core (fast,simple)	10	< 200 ms
tpm (fast,simple)	11	<200ms
pcie_vpx_sw (fast,simple)	13	< 200 ms
m2_bottom (fast,simple)	14	<200ms
m2_top (fast,simple)	15	<200ms
serial (fast,simple)	16	300 ms
onboardvideo (fast,simple)	17	<200ms
rtc (fast,simple)	20	< 200 ms
ether_link0 (fast,simple)	50	< 200 ms if PASSED but 3 s if FAILED
ether_link1 (fast,simple)	51	< 200 ms if PASSED but 3 s if FAILED
ether_link2 (fast,simple)	52	< 200 ms if PASSED but 3 s if FAILED
ether_link3 (fast,simple)	53	< 200 ms if PASSED but 3 s if FAILED
ether_loop0 (fast,simple)	55	800 ms (KR)
ether_loop1 (fast,simple)	56	800 ms (KR)
ether_loop2 (fast,simple)	57	2.6 s
ether_loop3 (fast,simple)	58	2.6 s
hwmon (fast,simple)	61	< 300 ms
sata0_dev_see (fast,simple)	62	< 300 ms
sata1_dev_see (fast,simple)	63	< 300 ms
sata2_dev_see (fast,simple)	64	< 300 ms
sata3_dev_see (fast,simple)	65	< 300 ms
sata4_dev_see (fast,simple)	66	< 300 ms
sata5_dev_see (fast,simple)	67	< 300 ms
sata0_controler (fast,simple)	68	< 300 ms
sata1_controler (fast,simple)	69	< 300 ms

TEST NAME	TEST ID NUMBER	EXECUTION TIME
vpd (fast,simple)	70	< 350 ms
usb0_dev_see (fast,simple)	80	< 300 ms
usb1_dev_see (fast,simple)	81	< 300 ms
usb2_dev_see (fast,simple)	82	< 300 ms
usb3_dev_see (fast,simple)	83	< 300 ms
ehci_controller (fast,simple)	85	< 300 ms
xhci_controller (fast,simple)	86	< 300 ms

## 2 / PBIT Command Line Reference Guide

The PBIT are configured and executed using the EFI Shell command «**kdiag**».

The following section describes the various "kdiag" command parameters.

### 2.1 On-line Help

At EFI Shell prompt enter the command "**help kdiag**" to display the usage messages.



The command formats are:

- ▶ [ ] meaning optional parameters
- ▶ | meaning a OR choice between possible parameters
- ▶ ... meaning an undetermined number of repeated previous parameters

```
VX6090> help kdiag
Kontron board diagnostics command
Print list of PBITs and infos about them :
kdiag [<PBITname>|<PBITnum> ...]
  <PBITname>|<PBITnum> ...
    list of PBIT(s) to display. All if the list is empty.
    PBIT(s) are referenced using their name or their number.
Run PBIT(s) from command line :
kdiag run [loop <count>] [<PBITname>|<PBITnum> ...]
  loop <count>
    run PBIT(s) <count> times instead of once
  <PBITname>|<PBITnum> ...
    list of PBIT(s) to run. All if the list is empty.
    PBIT(s) are referenced using their name or their number.
Print PBIT(s) status :
kdiag stat [<PBITname>|<PBITnum> ...]
  <PBITname>|<PBITnum> ...
    list of PBIT(s) to display. All if the list is empty.
    PBIT(s) are referenced using their name or their number.
Clear PBIT(s) status :
kdiag clrstat|clrallstat [<PBITname>|<PBITnum> ...]
  clrstat      : Reset status to NOTRUN
  clrallstat   : Reset status to NOTRUN and clear the (FAILED once) flag
  <PBITname>|<PBITnum> ...
    list of PBIT(s) to clear. All if the list is empty.
    PBIT(s) are referenced using their name or their number.
Restore default PBIT configuration :
kdiag default
Delete all PBITs from configuration :
kdiag deleteall
Configure PBIT(s) :

kdiag cfg <cfgarg> ... [<PBITname>|<PBITnum>] ...
  cfg <cfgarg> : Configure one or several PBIT(s).
                <cfgarg> is either :
                - "delete" to delete PBIT(s) from the list of configured PBITs
                - "default" to configure PBIT(s) with a default run mode
```

```

    - a comma separated list of runflags defining a PBIT
      run mode; for example : fast,complex.
valid runflags are :
- "SPEED" flags (can NOT be mixed)
  slow      : run in slow mode (full testing)
  fast      : run in fast mode (fast testing)
- "CONFIG" flags (can NOT be mixed)
  simple    : run in simple mode (no external hardware)
  complex   : run in complex mode (needs external hardware)
- "HALT" flags (can NOT be mixed)
  haltonfail : halt immediately (hang) if test fails
  promptonfail : halt at Firmware prompt (no OS boot) if test fails
- "RESET" flags (can be mixed together)
  normalreset : run after a normal reset
  poweronreset : run after a power-on reset
  allresets   : run after all resets listed above
[<PBITname>|<PBITnum>] ...
  list of PBIT(s) to configure.
  PBIT(s) are referenced using their name or their number.
  All configured tests if the list is empty
Toggle PBIT running log information:
  kdiag silent
Toggle PBIT Bypassed:
  kdiag bypass
Record System Configuration for system Test:
  kdiag system_learn or kdiag learn system
Edit System Configuration for system Test:
  (cf documentation for details)
  kdiag system_edit or kdiag edit system
Clear System Configuration for system Test:
  kdiag system_clear or kdiag clear system
Display PBIT version :
  kdiag version

```

## 2.2 Display the List of Selected Tests

To display the selected tests list and their configuration use the command:

```
"kdiag [<PBITname>|<PBITnum> ...]".
```

Running the command "kdiag" with no argument prints the list of the tests that are selected to run with the command "kdiag run" and also the other tests not selected by default.

```

VX6090> kdiag
PBITs configured to run from command line :
mem_data (1) - Checks Memory/ECC data lines
  capabilities : fast,simple,allresets
  run mode 1   : fast,simple,allresets
mem_addr (2) - Checks Memory/ECC address lines
  capabilities : fast,simple,allresets
  run mode 1   : fast,simple,allresets
mem_pattern1 (6) - Checks Memory/ECC using pattern 0xFFFFFFFF
  capabilities : slow/fast,simple,allresets
  run mode 1   : slow,simple,allresets
mem_pattern2 (7) - Checks Memory/ECC using pattern 0x55555555
  capabilities : slow/fast,simple,allresets
  run mode 1   : slow,simple,allresets

```

```

mem_pattern3 (8) - Checks Memory/ECC using pattern 0xAAAAAAAA
  capabilities : slow/fast,simple,allresets
  run mode 1   : slow,simple,allresets
mem_pattern4 (9) - Checks Memory/ECC using pattern 0x00000000
  capabilities : slow/fast,simple,allresets
  run mode 1   : slow,simple,allresets
core (10) - Checks Core count and AES disabled
  capabilities : fast,simple,allresets
  run mode 1   : fast,simple,allresets
tpm (11) - Checks TPM access
  capabilities : fast,simple/complex,allresets
  run mode 1   : fast,simple,allresets
m2_bottom (14) - Checks M.2 bottom module access
  capabilities : fast,simple/complex,allresets
  run mode 1   : fast,simple,allresets
m2_top (15) - Checks M.2 top module access
  capabilities : fast,simple/complex,allresets
  run mode 1   : fast,simple,allresets
serial (16) - Checks the serial line COM2
  capabilities : fast,simple/complex,allresets
  run mode 1   : fast,simple,allresets
onboardvideo (17) - Checks the onboard video component
  capabilities : fast,simple,allresets
  run mode 1   : fast,simple,allresets
rtc (20) - Checks the RTC time
  capabilities : fast,simple,allresets
  run mode 1   : fast,simple,allresets
sysflash (22) - Checks the BIOS rescue in system flash
  capabilities : fast,simple,allresets
  run mode 1   : fast,simple,allresets
cpld (24) - Checks PLD, GeoAddress, watchdog
  capabilities : fast,simple/complex,allresets
  run mode 1   : fast,simple,allresets
fram (40) - Checks F-RAM device.
  capabilities : fast,simple/complex,allresets
  run mode 1   : fast,simple,allresets
ether_loop0 (55) - Checks ethernet interface 0 in loopback mode
  capabilities : fast,simple/complex,allresets
  run mode 1   : fast,simple,allresets
ether_loop1 (56) - Checks ethernet interface 1 in loopback mode
  capabilities : fast,simple/complex,allresets
  run mode 1   : fast,simple,allresets
ether_loop2 (57) - Checks ethernet interface 2 in loopback mode
  capabilities : fast,simple/complex,allresets
  run mode 1   : fast,simple,allresets
ether_loop3 (58) - Checks ethernet interface 3 in loopback mode
  capabilities : fast,simple/complex,allresets
  run mode 1   : fast,simple,allresets
hwmon (61) - Checks hardware monitoring, temperature and voltage sensors
  capabilities : fast,simple/complex,allresets
  run mode 1   : fast,simple,allresets
sata0_controller (68) - Checks SATA controller 0 (D31:F2)
  capabilities : fast,simple/complex,allresets
  run mode 1   : fast,simple,allresets
sata1_controller (69) - Check SATA controller 1 (D31:F5)
  capabilities : fast,simple/complex,allresets
  run mode 1   : fast,simple,allresets
vpd (70) - Checks VPD data required for board operation.
  capabilities : fast,simple/complex,allresets
  run mode 1   : fast,simple,allresets
eeprom (71) - Checks User EEPROM (0xA2)
  capabilities : fast,simple/complex,allresets
  run mode 1   : fast,simple,allresets

```

```

ehci_controller (85) - Check EHCI controller
  capabilities : fast,simple,allresets
  run mode 1   : fast,simple,allresets
xhci_controller (86) - Check xHCI controller
  capabilities : fast,simple,allresets
  run mode 1   : fast,simple,allresets
system (89) - Checks system configuration SETUP,PCIe,SATA,USB,ETH stability
  capabilities : fast,simple/complex,allresets
  run mode 1   : fast,simple,allresets

```

Other PBITs available but not yet configured :

```

mem_bitflip (3) - Checks Mem/ECC using bit-flip pattern ((1 << (offset % 64))
  capabilities : slow/fast,simple,allresets
mem_addrpat (4) - Checks Memory/ECC using address pattern (offset)
  capabilities : slow/fast,simple,allresets
mem_addrpat2 (5) - Checks Memory/ECC using address pattern (~offset)
  capabilities : slow/fast,simple,allresets
pcie_vpx_sw (13) - Checks the PCI-Express backplane VPX switch
  capabilities : fast,simple/complex,allresets
smbus0 (26) - Check SMBUS0 between PLD and backplane
  capabilities : fast,simple/complex,allresets
smbus1 (27) - Check SMBUS1 between PLD and backplane
  capabilities : fast,simple/complex,allresets
ether_link0 (50) - Checks the link status on ethernet interface 0.
  capabilities : fast,simple/complex,allresets
ether_link1 (51) - Checks the link status on ethernet interface 1.
  capabilities : fast,simple/complex,allresets
ether_link2 (52) - Checks the link status on ethernet interface 2.
  capabilities : fast,simple/complex,allresets
ether_link3 (53) - Checks the link status on ethernet interface 3.
  capabilities : fast,simple/complex,allresets
sata0_dev_see (62) - Checks if device is present on SATA0
  capabilities : fast,simple,allresets
sata1_dev_see (63) - Checks if device is present on SATA1
  capabilities : fast,simple,allresets
sata2_dev_see (64) - Checks if device is present on SATA2
  capabilities : fast,simple,allresets
sata4_dev_see (66) - Checks if device is present on SATA4 (M.2 Top slot)
  capabilities : fast,simple,allresets
sata5_dev_see (67) - Checks if device is present on SATA5 (M.2 Bottom slot)
  capabilities : fast,simple,allresets
usb0_dev_see (80) - Check USB device presence on USB FP
  capabilities : fast,simple,allresets
usb1_dev_see (81) - Check USB device presence on USB1 (A=RTM USB C,B=RTM USB E)
  capabilities : fast,simple,allresets
usb3_dev_see (83) - Check USB device presence on USB0 (A=RTM USB A-B,B=RTM USB D) or USBSS (A only)
  capabilities : fast,simple,allresets
faultytest (98) - A dummy test that returns FAIL
  capabilities : fast,simple,allresets
hangtest (99) - A dummy test that will hang
  capabilities : fast,simple,allresets

```

Use "help kdiag" to get more info.

VX6090>

## 2.3 Execute the PBIT from the Command Line

To run the PBIT selected tests list from the command line, enter:

```
VX6090> kdiag run
PBIT "mem_data" (fast,simple) PASSED
PBIT "mem_addr" (fast,simple) PASSED
PBIT "mem_pattern1" (slow,simple) PASSED
PBIT "mem_pattern2" (slow,simple) PASSED
PBIT "mem_pattern3" (slow,simple) PASSED
PBIT "mem_pattern4" (slow,simple) PASSED
PBIT "core" (fast,simple) AES disabled, 8 core board PASSED
PBIT "tpm" (fast,simple) PASSED
PBIT "m2_bottom" (fast,simple) NO M2S2 on CPUA SIDE, TEST BYPASSED PASSED
PBIT "m2_top" (fast,simple) M2S1A CONNECTED TO SATA, TEST BYPASSED PASSED
PBIT "serial" (fast,simple) PASSED
PBIT "onboardvideo" (fast,simple) PASSED
PBIT "rtc" (fast,simple) PASSED
PBIT "sysflash" (fast,simple) PASSED
PBIT "cpld" (fast,simple) PASSED
PBIT "fram" (fast,simple) PASSED
PBIT "ether_loop0" (fast,simple) PASSED
PBIT "ether_loop1" (fast,simple) PASSED
PBIT "ether_loop2" (fast,simple) PASSED
PBIT "ether_loop3" (fast,simple) PASSED
PBIT "hwmon" (fast,simple) PASSED
PBIT "sata0_controler" (fast,simple) PASSED
PBIT "sata1_controler" (fast,simple) DISABLED PASSED
PBIT "vpd" (fast,simple) PASSED
PBIT "eeprom" (fast,simple) PASSED
PBIT "ehci_controller" (fast,simple) PASSED
PBIT "xhci_controller" (fast,simple) PASSED
PBIT "system" (fast,simple)
SYSTEM INFOS NOT SAVED
PASSED
VX6090>
```

To run a single test or a limited list of tests, enter:

```
VX6090> kdiag run <PBIT number | PBIT name ...>
```

► For example:

```
VX6090> kdiag run mem_data core
PBIT "mem_data" (fast,simple) PASSED
PBIT "core" (fast,simple) AES disabled, 8 core board PASSED
```

That is equivalent to:

```
VX6090> kdiag run 1 10
PBIT "mem_data" (fast,simple) PASSED
PBIT "core" (fast,simple) AES disabled, 8 core board PASSED
```



The "PBIT number" is displayed by the "kdiag" command (with no parameter) or with the "kdiag <PBIT name>" command.

## 2.4 Execute the PBIT in Loop Mode

To run the PBIT in loop mode, enter:

```
VX6090 > kdiag run loop <count>
```

<count> is the number of loop to execute.

To run a single test in loop mode, enter:

```
VX6090> kdiag run loop <count> <PBIT number | PBIT name ...>
```

- ▶ Example: running 10 times the test ether\_loop0 number 55, enter:

```
VX6090> kdiag run loop 10 55
```

It is equivalent to enter:

```
VX6090> kdiag run loop 10 ether_loop0
```

## 2.5 Get the PBIT Results

To get the PBIT results, use the "kdiag stat" command:

```
VX6090> kdiag stat
Status of PBITs configured to run from command line :
PASSED : mem_data(1) (fast,simple)
PASSED : mem_addr(2) (fast,simple)
PASSED : mem_pattern1(6) (slow,simple)
PASSED : mem_pattern2(7) (slow,simple)
PASSED : mem_pattern3(8) (slow,simple)
PASSED : mem_pattern4(9) (slow,simple)
PASSED : core(10) (fast,simple)
PASSED : tpm(11) (fast,simple)
PASSED : m2_bottom(14) (fast,simple)
PASSED : m2_top(15) (fast,simple)
PASSED : serial(16) (fast,simple)
PASSED : onboardvideo(17) (fast,simple)
PASSED : rtc(20) (fast,simple)
PASSED : sysflash(22) (fast,simple)
PASSED : cp1d(24) (fast,simple)
PASSED : fram(40) (fast,simple)
PASSED : ether_loop0(55) (fast,simple)
PASSED : ether_loop1(56) (fast,simple)
PASSED : ether_loop2(57) (fast,simple)
PASSED : ether_loop3(58) (fast,simple)
PASSED : hwmon(61) (fast,simple)
PASSED : sata0_controler(68) (fast,simple)
PASSED : sata1_controler(69) (fast,simple)
PASSED : vpd(70) (fast,simple)
PASSED : eeprom(71) (fast,simple)
PASSED : ehci_controller(85) (fast,simple)
PASSED : xhci_controller(86) (fast,simple)
PASSED : system(89) (fast,simple)

RUN      : 28
PASSED  : 28
FAILED  : 0
NOT_RUN : 0

VX6090>
```

## 2.6 Clear the PBIT Results

Upon failure of any test, a specific "FAILED ONCE" flag is set. This flag is kept even if this test is successfully PASSED later. This feature has been designed to keep track of intermittent failures.

- ▶ To clear the PBIT results enter:

```
VX6090> kdiag clrstat
```

- ▶ To clear all the PBIT history including the "FAILED ONCE" flags, enter:

```
VX6090> kdiag clral1stat
```

## 2.7 Configure the PBIT Tests List to Execute

The list of tests to execute can be modified with the "kdiag" command.

Each test can be added, removed and configured with a specific run mode. If no run mode is specified then the default run mode (fast,simple) is applied.

### 2.7.1 Run mode parameters

The possible specific run modes are defined with the following test flags:

- ▶ **HALT** flag (can NOT be mixed):
  - haltonfail**: halt immediately (hang) if test fails
  - promptonfail**: halt at BIOS prompt (no OS boot) if test fails

This flag offers the possibility to halt all test execution when an error is detected.
- ▶ **SPEED** flag (can NOT be mixed)
  - slow**: run in slow mode (full testing)
  - fast**: run in fast mode (fast testing)

In the current PBIT version, no test implements a difference between fast and slow modes.
- ▶ **CONFIG** flag (can NOT be mixed)
  - simple**: run in simple mode (no external hardware)
  - complex**: run in complex mode (needs external hardware)

Complex mode requires external devices to the VX6090 and is reserved for the factory tests.
- ▶ **RESET** flag
  - normalreset**: run after a normal reset (means a board warm boot).
  - poweronreset**: run after a power-on reset (PBIT with this flag will run only when board is powered ON or cold boot)
  - allresets**: run after all resets listed above

### 2.7.2 Adding a Test to the Current Run List

To add a test to the run list, enter:

```
VX6090> kdiag cfg <cfgarg> ... <PBITname>|<PBITnum> ...
```

"cfgarg" allows to choose the test run mode.

Use the keyword "default" to set the default mode (typically fast and simple)

To add test 16 (serial test) with default mode, enter:

```
VX6090> kdiag cfg default 16
```

To add this test with the promptonfail flag (and the other default flags), enter:

```
VX6090> kdiag cfg promptonfail 16
```

To add this test with the complex and promptonfail flags enter:

```
VX6090> kdiag cfg complex,promptonfail 16
```

Execute the command "kdiag" or "kdiag 16" to check the configuration:

```
VX6090> kdiag 16
serial (16) - Checks the serial line COM2
capabilities : fast,simple/complex,allresets
run mode 1   : fast,complex,promptonfail,allresets
```

### 2.7.3 Removing a Test from the Current Run List

For example, to remove test number 16, enter:

```
VX6090> kdiag cfg delete 16
```

Verify:

```
VX6090> kdiag 16
PBIT "16" is not configured to run from command line
```

To remove all the tests from the current run list, enter:

```
VX6090> kdiag deleteall
```

Don't forget to add some tests to the current list before running kdiag:

```
VX6090> kdiag run
WARNING: No PBIT will be run because run list is empty
```

### 2.7.4 Set a RUN mode parameter to All the Tests of the Current Run List

The command `kdiag cfg "runflag"` allows to set a "runflag" (HALT, CONFIG, RESET described in section 2.7.1 page 18) to all current run list tests.

For example to set default run mode for all tests of current run list:

```
VX6090> kdiag cfg default
Configuration of current tests in asked mode:
command,fast,simple,allresets
List of test to configure
--> mem_data (1) - Checks Memory/ECC data lines
--> mem_addr (2) - Checks Memory/ECC address lines
--> mem_pattern1 (6) - Checks Memory/ECC using pattern 0xFFFFFFFF
--> mem_pattern2 (7) - Checks Memory/ECC using pattern 0x55555555
--> mem_pattern3 (8) - Checks Memory/ECC using pattern 0xAAAAAAAA
--> mem_pattern4 (9) - Checks Memory/ECC using pattern 0x00000000
--> core (10) - Checks Core count and AES disabled
--> tpm (11) - Checks TPM access
--> m2_bottom (14) - Checks M.2 bottom module access
--> m2_top (15) - Checks M.2 top module access
--> serial (16) - Checks the serial line COM2
--> onboardvideo (17) - Checks the onboard video component
--> rtc (20) - Checks the RTC time
--> sysflash (22) - Checks the BIOS rescue in system flash
```

```

--> cpld (24) - Checks PLD, GeoAddress, watchdog
--> fram (40) - Checks F-RAM device.
--> ether_loop0 (55) - Checks ethernet interface 0 in loopback mode
--> ether_loop1 (56) - Checks ethernet interface 1 in loopback mode
--> ether_loop2 (57) - Checks ethernet interface 2 in loopback mode
--> ether_loop3 (58) - Checks ethernet interface 3 in loopback mode
--> hwmon (61) - Checks hardware monitoring, temperature and voltage sensors
--> sata0_controller (68) - Checks SATA controller 0 (D31:F2)
--> sata1_controller (69) - Check SATA controller 1 (D31:F5)
--> vpd (70) - Checks VPD data required for board operation.
--> eeprom (71) - Checks OS EEPROM (0xA2)
--> ehci_controller (85) - Check EHCI controller
--> xhci_controller (86) - Check xHCI controller
--> system (89) - Checks system configuration SETUP,PCIe,SATA,USB,ETH stability
VX6090>

```

For example to set promptonfail, poweronreset run mode for all tests of current run list:

```

VX6090> kdiag cfg promptonfail,poweronreset
Configuration of current tests in asked mode:
command,fast,simple,promptonfail,poweronreset
List of test to configure
--> mem_data (1) - Checks Memory/ECC data lines
--> mem_addr (2) - Checks Memory/ECC address lines
--> mem_pattern1 (6) - Checks Memory/ECC using pattern 0xFFFFFFFF
--> mem_pattern2 (7) - Checks Memory/ECC using pattern 0x55555555
--> mem_pattern3 (8) - Checks Memory/ECC using pattern 0xAAAAAAAA
--> mem_pattern4 (9) - Checks Memory/ECC using pattern 0x00000000
--> core (10) - Checks Core count and AES disabled
--> tpm (11) - Checks TPM access
--> m2_bottom (14) - Checks M.2 bottom module access
--> m2_top (15) - Checks M.2 top module access
--> serial (16) - Checks the serial line COM2
--> onboardvideo (17) - Checks the onboard video component
--> rtc (20) - Checks the RTC time
--> sysflash (22) - Checks the BIOS rescue in system flash
--> cpld (24) - Checks PLD, GeoAddress, watchdog
--> fram (40) - Checks F-RAM device.
--> ether_loop0 (55) - Checks ethernet interface 0 in loopback mode
--> ether_loop1 (56) - Checks ethernet interface 1 in loopback mode
--> ether_loop2 (57) - Checks ethernet interface 2 in loopback mode
--> ether_loop3 (58) - Checks ethernet interface 3 in loopback mode
--> hwmon (61) - Checks hardware monitoring, temperature and voltage sensors
--> sata0_controller (68) - Checks SATA controller 0 (D31:F2)
--> sata1_controller (69) - Check SATA controller 1 (D31:F5)
--> vpd (70) - Checks VPD data required for board operation.
--> eeprom (71) - Checks OS EEPROM (0xA2)
--> ehci_controller (85) - Check EHCI controller
--> xhci_controller (86) - Check xHCI controller
--> system (89) - Checks system configuration SETUP,PCIe,SATA,USB,ETH stability
VX6090>

```

Verify with **kdiag** command, the run mode is promptonfail, poweronreset for all tests in the run list:

```

VX6090> kdiag
PBITs configured to run from command line :
  mem_data (1) - Checks Memory/ECC data lines
    capabilities : fast,simple,allresets
    run mode 1   : fast,simple,promptonfail,poweronreset
  mem_addr (2) - Checks Memory/ECC address lines
    capabilities : fast,simple,allresets
    run mode 1   : fast,simple,promptonfail,poweronreset

```

```

mem_pattern1 (6) - Checks Memory/ECC using pattern 0xFFFFFFFF
  capabilities : slow/fast,simple,allresets
  run mode 1   : fast,simple,promptonfail,poweronreset
mem_pattern2 (7) - Checks Memory/ECC using pattern 0x55555555
  capabilities : slow/fast,simple,allresets
  run mode 1   : fast,simple,promptonfail,poweronreset
mem_pattern3 (8) - Checks Memory/ECC using pattern 0xAAAAAAAA
  capabilities : slow/fast,simple,allresets
  run mode 1   : fast,simple,promptonfail,poweronreset
mem_pattern4 (9) - Checks Memory/ECC using pattern 0x00000000
  capabilities : slow/fast,simple,allresets
  run mode 1   : fast,simple,promptonfail,poweronreset
core (10) - Checks Core count and AES disabled
  capabilities : fast,simple,allresets
  run mode 1   : fast,simple,promptonfail,poweronreset
tpm (11) - Checks TPM access
  capabilities : fast,simple/complex,allresets
  run mode 1   : fast,simple,promptonfail,poweronreset
m2_bottom (14) - Checks M.2 bottom module access
  capabilities : fast,simple/complex,allresets
  run mode 1   : fast,simple,promptonfail,poweronreset
m2_top (15) - Checks M.2 top module access
  capabilities : fast,simple/complex,allresets
  run mode 1   : fast,simple,promptonfail,poweronreset
serial (16) - Checks the serial line COM2
  capabilities : fast,simple/complex,allresets
  run mode 1   : fast,simple,promptonfail,poweronreset
onboardvideo (17) - Checks the onboard video component
  capabilities : fast,simple,allresets
  run mode 1   : fast,simple,promptonfail,poweronreset
rtc (20) - Checks the RTC time
  capabilities : fast,simple,allresets
  run mode 1   : fast,simple,promptonfail,poweronreset
sysflash (22) - Checks the BIOS rescue in system flash
  capabilities : fast,simple,allresets
  run mode 1   : fast,simple,promptonfail,poweronreset
cpld (24) - Checks PLD, GeoAddress, watchdog
  capabilities : fast,simple/complex,allresets
  run mode 1   : fast,simple,promptonfail,poweronreset
fram (40) - Checks F-RAM device.
  capabilities : fast,simple/complex,allresets
  run mode 1   : fast,simple,promptonfail,poweronreset
ether_loop0 (55) - Checks ethernet interface 0 in loopback mode
  capabilities : fast,simple/complex,allresets
  run mode 1   : fast,simple,promptonfail,poweronreset
ether_loop1 (56) - Checks ethernet interface 1 in loopback mode
  capabilities : fast,simple/complex,allresets
  run mode 1   : fast,simple,promptonfail,poweronreset
ether_loop2 (57) - Checks ethernet interface 2 in loopback mode
  capabilities : fast,simple/complex,allresets
  run mode 1   : fast,simple,promptonfail,poweronreset
ether_loop3 (58) - Checks ethernet interface 3 in loopback mode
  capabilities : fast,simple/complex,allresets
  run mode 1   : fast,simple,promptonfail,poweronreset
hwmon (61) - Checks hardware monitoring, temperature and voltage sensors
  capabilities : fast,simple/complex,allresets
  run mode 1   : fast,simple,promptonfail,poweronreset
sata0_controller (68) - Checks SATA controller 0 (D31:F2)
  capabilities : fast,simple/complex,allresets
  run mode 1   : fast,simple,promptonfail,poweronreset
sata1_controller (69) - Check SATA controller 1 (D31:F5)
  capabilities : fast,simple/complex,allresets
  run mode 1   : fast,simple,promptonfail,poweronreset

```

```

vpd (70) - Checks VPD data required for board operation.
  capabilities : fast,simple/complex,allresets
  run mode 1   : fast,simple,promptonfail,poweronreset
eeprom (71) - Checks OS EEPROM (0xA2)
  capabilities : fast,simple/complex,allresets
  run mode 1   : fast,simple,promptonfail,poweronreset
ehci_controller (85) - Check EHCI controller
  capabilities : fast,simple,allresets
  run mode 1   : fast,simple,promptonfail,poweronreset
xhci_controller (86) - Check xHCI controller
  capabilities : fast,simple,allresets
  run mode 1   : fast,simple,promptonfail,poweronreset
system (89) - Checks system configuration SETUP,PCIe,SATA,USB,ETH stability
  capabilities : fast,simple/complex,allresets
  run mode 1   : fast,simple,promptonfail,poweronreset

```

Verify for test 10:

```

VX6090> kdiag 10
core (10) - Checks Core count and AES disabled
  capabilities : fast,simple,allresets
  run mode 1   : fast,simple,promptonfail,poweronreset

```




---

The command will not include the test numbers greater than 95 (hangtest, faulty) because these tests are tool tests for PBIT validation.

---

## 2.7.5 Restore the Default Run List

To restore the default run tests list, enter:

```
VX6090> kdiag default
```

## 2.8 Run the PBIT in Silent Mode

To avoid the PBIT to display test messages during execution, use the toggle command:

```
VX6090> kdiag silent
PBIT set in silent mode
```

To disable the silent mode, re-execute the same command:

```
VX6090> kdiag silent
PBIT silent mode removed
```



- 
1. In this mode error messages are displayed anyway.
  2. To prevent any output messages to the serial line, use the BIOS setup configuration (Serial Line Console Redirection).
-

## 2.9 Display the PBIT Version

To display the PBIT version, enter:

```
VX6090> kdiag version
PBIT VERSION 1.4 ID16293
```

## 2.10 PBIT System Test

### 2.10.1 Recording the System Configuration

The command "**kdiag system\_learn**" or "**kdiag learn system**" is used to record the current system configuration. It might be run when the system configuration is the correct one to be recorded.

It records:

- ▶ All detected PCI devices (the list is visible with the BIOS shell command "**pci**"),
- ▶ PCIe devices infos (vendorID, deviceID, ClassCode) from detected PCI devices,
- ▶ PCIe link width and speed for PCI/PCI bridge devices,
- ▶ SATA information (port connected, disk name, disk size),
- ▶ USB information (ort connected, keyboard, mice, mass storage device name(s)),
- ▶ Ethernet information (port link up, speed),
- ▶ BIOS information: BIOS ID and checksum (checksum of BIOS setup),
- ▶ Information from CPLD registers: CPLD Version, DRAM size, system controller status, geographical address, XMC presence,
- ▶ SMBUS connected device's addresses on backplane SMBUS0 and SMBUS1.

Here is an example of the command on a VX6090 board:

```
VX6090> kdiag learn system
Seg  Bus  Dev  Func
---  ---  ---  ----
 00  00  00  00 ==> Bridge Device - Host/PCI bridge
      Vendor 8086 Device 6F00 Prog Interface 0
 00  00  01  00 ==> Bridge Device - PCI/PCI bridge
      Vendor 8086 Device 6F02 Prog Interface 0
 00  00  02  00 ==> Bridge Device - PCI/PCI bridge
      Vendor 8086 Device 6F04 Prog Interface 0
 00  00  02  02 ==> Bridge Device - PCI/PCI bridge
      Vendor 8086 Device 6F06 Prog Interface 0
 00  00  03  00 ==> Bridge Device - PCI/PCI bridge
      Vendor 8086 Device 6F08 Prog Interface 0
 00  00  03  02 ==> Bridge Device - PCI/PCI bridge
      Vendor 8086 Device 6F0A Prog Interface 0
 00  00  03  03 ==> Bridge Device - PCI/PCI bridge
      Vendor 8086 Device 6F0B Prog Interface 0
 00  00  05  00 ==> Base System Peripherals - Other system peripheral
      Vendor 8086 Device 6F28 Prog Interface 0
 00  00  05  01 ==> Base System Peripherals - Other system peripheral
      Vendor 8086 Device 6F29 Prog Interface 0
 00  00  05  02 ==> Base System Peripherals - Other system peripheral
      Vendor 8086 Device 6F2A Prog Interface 0
 00  00  05  04 ==> Base System Peripherals - PIC
      Vendor 8086 Device 6F2C Prog Interface 20
```

```

00 00 14 00 ==> Serial Bus Controllers - USB
Vendor 8086 Device 8C31 Prog Interface 30
00 00 1C 00 ==> Bridge Device - PCI/PCI bridge
Vendor 8086 Device 8C10 Prog Interface 0
00 00 1C 02 ==> Bridge Device - PCI/PCI bridge
Vendor 8086 Device 8C14 Prog Interface 0
00 00 1D 00 ==> Serial Bus Controllers - USB
Vendor 8086 Device 8C26 Prog Interface 20
00 00 1F 00 ==> Bridge Device - PCI/ISA bridge
Vendor 8086 Device 8C54 Prog Interface 0
00 00 1F 02 ==> Mass Storage Controller - Serial ATA controller
Vendor 8086 Device 8C02 Prog Interface 1
00 00 1F 03 ==> Serial Bus Controllers - System Management Bus
Vendor 8086 Device 8C22 Prog Interface 0
00 02 00 00 ==> Base System Peripherals - Other system peripheral
Vendor 8086 Device 6F50 Prog Interface 0
00 02 00 01 ==> Base System Peripherals - Other system peripheral
Vendor 8086 Device 6F51 Prog Interface 0
00 02 00 02 ==> Base System Peripherals - Other system peripheral
Vendor 8086 Device 6F52 Prog Interface 0
00 02 00 03 ==> Base System Peripherals - Other system peripheral
Vendor 8086 Device 6F53 Prog Interface 0
00 03 00 00 ==> Network Controller - Ethernet controller
Vendor 8086 Device 15AB Prog Interface 0
00 03 00 01 ==> Network Controller - Ethernet controller
Vendor 8086 Device 15AB Prog Interface 0
00 05 00 00 ==> Network Controller - Ethernet controller
Vendor 8086 Device 1537 Prog Interface 0
00 06 00 00 ==> Network Controller - Ethernet controller
Vendor 8086 Device 1533 Prog Interface 0
00 08 00 00 ==> Display Controller - VGA/8514 controller
Vendor 126F Device 0750 Prog Interface 0

```

CPLD Version 0x43

DRAM size 8 GB

System Controller

Geographical Address 1

XMC not present

SATA PORT 4 TS32GMTS400 (32.0GB)

Device detected on SMBUS0, address = 0x30

BIOS Setup Checksum : 124299

BIOS Version : ID16293EF

ETH0 connected device speed 10000Mb/s

ETH1 connected device speed 10000Mb/s

Link speed 1000Mb/s ETH3 connected device speed 0Mb/s

1 Drive, 1 Hub

MassStorage Device name (0) : 0.00

USB Port 3 connected

Number of System Test Elements detected : 41

DRAM area [ 0x33CBA9E0 0x33CBB780 ] will be stored in EEPROM

Storing system infos...

Storing system configuration...

VX6090>

## 2.10.2 Testing the Current System Configuration

The test named "**system**"(89) is used to read the current system configuration and check it against the recorded configuration. The command used is "**kdiag run system**" or "**kdiag system\_run**". By default, the system test prints errors only, and details level is NORMAL.

The following is an example with several detected failures on a VX6090 board:

```
VX6090> kdiag run system
PBIT "system" (fast,simple)
PCI... CPLD... SATA... SMBUS... BIOS... ETH...
Link speed 1000Mb/s USB...
ERR)PCI Bus:Dev:Func (R)00:1F:02 (D)00:1F:03
ERR)PCI 00:1F:02 DeviceID (R)0x8C02 (D)0x8C22
ERR)PCI 00:1F:02 ClassCode (R)010601 (D)010601
ERR)PCI Bus:Dev:Func (R)00:1F:03 (D)02:00:00
ERR)PCI 00:1F:03 DeviceID (R)0x8C22 (D)0x6F50
ERR)PCI 00:1F:03 ClassCode (R)0C0500 (D)0C0500
ERR)PCI Bus:Dev:Func (R)02:00:00 (D)02:00:01
ERR)PCI 02:00:00 DeviceID (R)0x6F50 (D)0x6F51
ERR)PCI Bus:Dev:Func (R)02:00:01 (D)02:00:02
ERR)PCI 02:00:01 DeviceID (R)0x6F51 (D)0x6F52
ERR)PCI Bus:Dev:Func (R)02:00:02 (D)02:00:03
ERR)PCI 02:00:02 DeviceID (R)0x6F52 (D)0x6F53
ERR)PCI Bus:Dev:Func (R)02:00:03 (D)03:00:00
ERR)PCI 02:00:03 DeviceID (R)0x6F53 (D)0x15AB
ERR)PCI 02:00:03 ClassCode (R)088000 (D)088000
ERR)PCI Bus:Dev:Func (R)03:00:00 (D)03:00:01
ERR)PCI Bus:Dev:Func (R)03:00:01 (D)05:00:00
ERR)PCI 03:00:01 DeviceID (R)0x15AB (D)0x1537
ERR)PCI Bus:Dev:Func (R)05:00:00 (D)06:00:00
ERR)PCI 05:00:00 DeviceID (R)0x1537 (D)0x1533
ERR)PCI Bus:Dev:Func (R)06:00:00 (D)08:00:00
ERR)PCI 06:00:00 VendorID (R)0x8086 (D)0x126F
ERR)PCI 06:00:00 DeviceID (R)0x1533 (D)0x0750
ERR)PCI 06:00:00 ClassCode (R)020000 (D)020000
ERR)PCI Bus:Dev:Func (R)08:00:00 (D)No Device
ERR)PCI 08:00:00 VendorID (R)0x126F (D)No Device
ERR)PCI 08:00:00 DeviceID (R)0x0750 (D)No Device
ERR)PCI 08:00:00 ClassCode (R)030000 (D)030000
ERR)SATA Port 4 (R)TS32GMTS400 (32.0GB) (D)Not Connected (0.0GB)
ERR)BIOS Checksum (R)124299 (D)124267
FAILED
```

(R) is for Recorded

(D) is for Detected

## 2.10.3 Editing the System Configuration

A menu is available to edit the system configuration. This menu allows to IGNORE specific items.

For example it is possible to ignore checking SATA disk size or USB keyboard connected ...

The menu also allows to configure a specific level of debug when the system test is executed. To enter **system edit** menu enter:

```
VX6090> kdiag edit system
```

Equivalent to:

```
VX6090> kdiag system_edit
```

The menu is displayed only if a system configuration has been previously recorded with the "**kdiag learn system**" or "**kdiag system\_learn**" command. On the contrary a message informs that no configuration has been recorded yet:

```
SYSTEM INFOS NOT SAVED
```

Once edited the system edition gives you a prompt and is organized with sublevel menus. Help is available from any submenu by typing '?':

Two mains features can be distinguished:

- ▶ the **Ignore** menu to bypass specific item verification
- ▶ the **print debug setting** menu to modify the debug level and manage the scroll limit for it.

### 2.10.3.1 Edit System Items (to Ignore Specific Test)

By default, all the items of the system test are checked. The system edit menu allows user to choose which items will be ignored during the next system test executions.

It is possible to ignore all or part of specific elements of the system test structures. For example an USB port, a SATA port, an Ethernet interface, SMBUS devices, information from the CPLD as the System Controller Status etc.

The following shows edit/run system sessions ignoring partially or totally some system information:

```
VX6090> kdiag edit system

    << KONTRON SYSTEM PBIT : EDIT MODE >>

Edit by feature, choose: PCI CPLD SATA ETH USB SMBUS BIOS
Other available cmds: `s` for settings, `?` for help, `q` to quit edit mode

--SystemEdit>>cpld

--SystemEdit-CPLD>>?

p : print Recorded values
0,..,4 : edit CPLD element
ignoreall : ignore all CPLD infos
default : restore default CPLD config (remove flags)
? : help
q : go back to main menu

--SystemEdit-CPLD>>p
```

```

Recorded CPLD config:

0. CPLD Version          0x43
1. CPLD Dram Size       8GB
2. CPLD SysCon          System Controller
3. CPLD Geo Address     1
4. CPLD XMC Presence    XMC not present
--SystemEdit-CPLD>>2

--SystemEdit-CPLD-SysCon>>?

p : print Recorded value
i : set ignore flag
r : remove flag
? : help
q : go back to CPLD menu

--SystemEdit-CPLD-SysCon>>i

Ignore flag has been set
*--SystemEdit-CPLD-SysCon>>q

*--SystemEdit-CPLD>>p

Recorded CPLD config:

0. CPLD Version          0x43
1. CPLD Dram Size       8GB
2. CPLD SysCon          System Controller      (i)
3. CPLD Geo Address     1
4. CPLD XMC Presence    XMC not present
*--SystemEdit-CPLD>>

      << KONTRON SYSTEM PBIT : EDIT MODE >>

Edit by feature, choose: PCI CPLD (*) SATA ETH USB SMBUS BIOS
Other available cmds: `s` for settings, `?` for help, `q` to quit edit mode

*--SystemEdit>>sata

*--SystemEdit-SATA>>p

0. SATA Port 0          Not Connected (0.0GB)
1. SATA Port 1          Not Connected (0.0GB)
2. SATA Port 2          Not Connected (0.0GB)
3. SATA Port 3          Not Connected (0.0GB)
4. SATA Port 4          TS32GMTS400 (32.0GB) [FAILED ONCE] [LAST FAILED]
5. SATA Port 5          Not Connected (0.0GB)
*--SystemEdit-SATA>>`

ERROR: ```` isn't a valid command

*--SystemEdit-SATA>>4

*--SystemEdit-SATA-Port 4>>i

Ignore flag has been set
*--SystemEdit-SATA-Port 4>>p

TS32GMTS400 (32.0GB)(i) [FAILED ONCE] [LAST FAILED]

```

```

*-SystemEdit-SATA-Port 4>>

*-SystemEdit-SATA>>

    << KONTRON SYSTEM PBIT : EDIT MODE >>

Edit by feature, choose: PCI CPLD (*) SATA (*) ETH USB SMBUS BIOS
Other available cmds: `s` for settings, `?` for help, `q` to quit edit mode

*-SystemEdit>>smbus

*-SystemEdit-SMBUS>>?

p : print Recorded values
0,..,1 : edit SMBUS element
ignoreall : ignore all SMBUS infos
default : restore default SMBUS config (remove flags)
? : help
q : go back to main menu

*-SystemEdit-SMBUS>>p

0. SMBUS0
SMBUS0 Device 0      0x30
SMBUS0 Device 1      No Device
SMBUS0 Device 2      No Device
SMBUS0 Device 3      No Device
SMBUS0 Device 4      No Device
SMBUS0 Device 5      No Device
SMBUS0 Device 6      No Device
SMBUS0 Device 7      No Device
SMBUS0 Device 8      No Device
SMBUS0 Device 9      No Device
1. SMBUS1
SMBUS1 Device 0      No Device
SMBUS1 Device 1      No Device
SMBUS1 Device 2      No Device
SMBUS1 Device 3      No Device
SMBUS1 Device 4      No Device
SMBUS1 Device 5      No Device
SMBUS1 Device 6      No Device
SMBUS1 Device 7      No Device
SMBUS1 Device 8      No Device
SMBUS1 Device 9      No Device
*-SystemEdit-SMBUS>>1

*-SystemEdit-SMBUS-SMBUS1>>?

p : print Recorded values
0,..,9 : edit SMBUS1 element
ignoreall : ignore all SMBUS1 infos
default : restore default SMBUS1 config (remove flags)
? : help
q : go back to SMBUS menu

*-SystemEdit-SMBUS-SMBUS1>>ignoreall

SMBUS1 entirely ignored
*-SystemEdit-SMBUS-SMBUS1>>

*-SystemEdit-SMBUS>>p

```

```

0. SMBUS0
SMBUS0 Device 0      0x30
SMBUS0 Device 1      No Device
SMBUS0 Device 2      No Device
SMBUS0 Device 3      No Device
SMBUS0 Device 4      No Device
SMBUS0 Device 5      No Device
SMBUS0 Device 6      No Device
SMBUS0 Device 7      No Device
SMBUS0 Device 8      No Device
SMBUS0 Device 9      No Device
1. SMBUS1 (ignored)
SMBUS1 Device 0      No Device          (i)
SMBUS1 Device 1      No Device          (i)
SMBUS1 Device 2      No Device          (i)
SMBUS1 Device 3      No Device          (i)
SMBUS1 Device 4      No Device          (i)
SMBUS1 Device 5      No Device          (i)
SMBUS1 Device 6      No Device          (i)
SMBUS1 Device 7      No Device          (i)
SMBUS1 Device 8      No Device          (i)
SMBUS1 Device 9      No Device          (i)
*-SystemEdit-SMBUS>>

    << KONTRON SYSTEM PBIT : EDIT MODE >>

Edit by feature, choose: PCI CPLD (*) SATA (*) ETH USB SMBUS (*) BIOS
Other available cmds: `s` for settings, `?` for help, `q` to quit edit mode

*-SystemEdit>>pci
*-SystemEdit-PCI>>?
pa : print all Recorded PCI Device infos
pb : print all Recorded PCI/PCI Bridge infos
p1 : print Recorded VendorID only
p2 : print Recorded DeviceID only
p3 : print Recorded ClassCode only
p4 : print Recorded Linkstatus only (only for PCI/PCI bridge)
p5 : print Recorded Link Speed/Width only (only for PCI/PCI bridge)
p  : print Recorded values
0,..,49 : edit PCI element
ignoreall : ignore all PCI infos
default : restore default PCI config (remove flags)
? : help
q : go back to main menu

*-SystemEdit-PCI>>p
0. PCI 00:00:00 Bridge Device/Host/PCI bridge
1. PCI 00:01:00 Bridge Device/PCI/PCI bridge
2. PCI 00:02:00 Bridge Device/PCI/PCI bridge
3. PCI 00:02:02 Bridge Device/PCI/PCI bridge
4. PCI 00:03:00 Bridge Device/PCI/PCI bridge
5. PCI 00:03:02 Bridge Device/PCI/PCI bridge
6. PCI 00:03:03 Bridge Device/PCI/PCI bridge
7. PCI 00:05:00 Base System Peripherals/Other system peripheral
8. PCI 00:05:01 Base System Peripherals/Other system peripheral
9. PCI 00:05:02 Base System Peripherals/Other system peripheral
10. PCI 00:05:04 Base System Peripherals/PIC
11. PCI 00:14:00 Serial Bus Controllers/USB
12. PCI 00:1C:00 Bridge Device/PCI/PCI bridge

```

```

13. PCI 00:1C:02 Bridge Device/PCI/PCI bridge
14. PCI 00:1D:00 Serial Bus Controllers/USB
15. PCI 00:1F:00 Bridge Device/PCI/ISA bridge
16. PCI 00:1F:02 Mass Storage Controller/Serial ATA controller [FAILED ONCE] [LAST FAILED]
17. PCI 00:1F:03 Serial Bus Controllers/System Management Bus [FAILED ONCE] [LAST FAILED]
18. PCI 02:00:00 Base System Peripherals/Other system peripheral [FAILED ONCE] [LAST FAILED]
19. PCI 02:00:01 Base System Peripherals/Other system peripheral [FAILED ONCE] [LAST FAILED]
20. PCI 02:00:02 Base System Peripherals/Other system peripheral [FAILED ONCE] [LAST FAILED]
21. PCI 02:00:03 Base System Peripherals/Other system peripheral [FAILED ONCE] [LAST FAILED]
22. PCI 03:00:00 Network Controller/Ethernet controller [FAILED ONCE] [LAST FAILED]
23. PCI 03:00:01 Network Controller/Ethernet controller [FAILED ONCE] [LAST FAILED]
24. PCI 05:00:00 Network Controller/Ethernet controller [FAILED ONCE] [LAST FAILED]
25. PCI 06:00:00 Network Controller/Ethernet controller [FAILED ONCE] [LAST FAILED]
26. PCI 08:00:00 Display Controller/VGA/8514 controller [FAILED ONCE] [LAST FAILED]
27. PCI No Device No Device/No Device
28. PCI No Device No Device/No Device
29. PCI No Device No Device/No Device
30. PCI No Device No Device/No Device
31. PCI No Device No Device/No Device
32. PCI No Device No Device/No Device
33. PCI No Device No Device/No Device
34. PCI No Device No Device/No Device
35. PCI No Device No Device/No Device
36. PCI No Device No Device/No Device
37. PCI No Device No Device/No Device
38. PCI No Device No Device/No Device
39. PCI No Device No Device/No Device
40. PCI No Device No Device/No Device
41. PCI No Device No Device/No Device
42. PCI No Device No Device/No Device
43. PCI No Device No Device/No Device
44. PCI No Device No Device/No Device
45. PCI No Device No Device/No Device
46. PCI No Device No Device/No Device
47. PCI No Device No Device/No Device
48. PCI No Device No Device/No Device
49. PCI No Device No Device/No Device
*-SystemEdit-PCI>>26
*-SystemEdit-PCI-08:00:00>>ignoreall

PCI 08:00:00 entirely ignored
*-SystemEdit-PCI-08:00:00>>p

0. VendorId          0x126F (i)
1. DeviceId          0x0750 (i)
2. ClassCode         030000 (i)
   Prog Interface    0
   BaseClass         Display Controller
   SubClass          VGA/8514 controller
*-SystemEdit-PCI-08:00:00>>

*-SystemEdit-PCI>>16

*-SystemEdit-PCI-00:1F:02>>?

p : print Recorded values
0,..,4 : edit PCI 00:1F:02 element
ignoreall : ignore all PCI 00:1F:02 infos
default : restore default PCI 00:1F:02 config (remove flags)
? : help
q : go back to main menu

*-SystemEdit-PCI-00:1F:02>>2

```

```

*-SystemEdit-PCI-00:1F:02-ClassCode>>?

p : print Recorded value
i : set ignore flag
r : remove flag
? : help
q : go back to PCI 00:1F:02 menu

*-SystemEdit-PCI-00:1F:02-ClassCode>>i

Ignore flag has been set
*-SystemEdit-PCI-00:1F:02-ClassCode>>p

010601 (i)
Prog Interface 1 (i)
BaseClass Mass Storage Controller (i)
SubClass Serial ATA controller (i)
*-SystemEdit-PCI-00:1F:02-ClassCode>>

*-SystemEdit-PCI-00:1F:02>>

*-SystemEdit-PCI>>

    << KONTRON SYSTEM PBIT : EDIT MODE >>

Edit by feature, choose: PCI (*) CPLD (*) SATA (*) ETH USB SMBUS (*) BIOS
Other available cmds: `s` for settings, `?` for help, `q` to quit edit mode

*-SystemEdit>>bios

*-SystemEdit-BIOS>>?

p : print Recorded values
0,..,1 : edit BIOS element
ignoreall : ignore all BIOS infos
default : restore default BIOS config (remove flags)
? : help
q : go back to main menu

*-SystemEdit-BIOS>>p

0. BIOS Checksum          124299          [FAILED ONCE] [LAST FAILED])
1. BIOS Version          ID16293EF

*-SystemEdit-BIOS>>0
*-SystemEdit-BIOS-Checksum>>i

Ignore flag has been set
*-SystemEdit-BIOS-Checksum>>p

124299          (i) [FAILED ONCE] [LAST FAILED]
*-SystemEdit-BIOS-Checksum>>

*-SystemEdit-BIOS>>p

0. BIOS Checksum          124299          (i) [FAILED ONCE] [LAST FAILED])
1. BIOS Version          ID16293EF
*-SystemEdit-BIOS>>

    << KONTRON SYSTEM PBIT : EDIT MODE >>

Edit by feature, choose: PCI (*) CPLD (*) SATA (*) ETH USB SMBUS (*) BIOS (*)
Other available cmds: `s` for settings, `?` for help, `q` to quit edit mode

```

```

*-SystemEdit>>eth

*-SystemEdit-ETH>>p

0. ETH Port 0          Link UP speed 10000 Mb/
1. ETH Port 1          Link UP speed 10000 Mb/
2. ETH Port 2          Not Connected
3. ETH Port 3          Not Connected
*-SystemEdit-ETH>>2

*-SystemEdit-ETH-Port 2>>i

Ignore flag has been set
*-SystemEdit-ETH-Port 2>>

*-SystemEdit-ETH>>p

0. ETH Port 0          Link UP speed 10000 Mb/
1. ETH Port 1          Link UP speed 10000 Mb/
2. ETH Port 2          Not Connected          (i)
3. ETH Port 3          Not Connected
*-SystemEdit-ETH>>3

*-SystemEdit-ETH-Port 3>>i

Ignore flag has been set
*-SystemEdit-ETH-Port 3>>

*-SystemEdit-ETH>>p

0. ETH Port 0          Link UP speed 10000 Mb/
1. ETH Port 1          Link UP speed 10000 Mb/
2. ETH Port 2          Not Connected          (i)
3. ETH Port 3          Not Connected          (i)
*-SystemEdit-ETH>>

      << KONTRON SYSTEM PBIT : EDIT MODE >>

Edit by feature, choose: PCI (*) CPLD (*) SATA (*) ETH (*) USB SMBUS (*) BIOS (*)
Other available cmds: `s` for settings, `?` for help, `q` to quit edit mode

*-SystemEdit>>USB

*-SystemEdit-USB>>p

0. UsbCounts
USB Drive's Count      1 Drive
USB Kbd's Count        0 Keyboard
USB Mouse's Count      0 Mouse
USB Hub's Count        1 Hub
USB Point's Count      0 Point
USB Ccid's Count       0 SmartCard Reader
1. MassNames
USB MassStorage(0)     0.00
USB MassStorage(1)     No Device
USB MassStorage(2)     No Device
USB MassStorage(3)     No Device
2. UsbPorts
USB Port 0             Not Connected
USB Port 1             Not Connected
USB Port 2             Not Connected
USB Port 3             Connected
USB Port 4             Not Connected
*-SystemEdit-USB>>1

```

```

*-SystemEdit-USB-MassNames>>p
0. 0.00
1. No Device
2. No Device
3. No Device
*-SystemEdit-USB-MassNames>>0

*-SystemEdit-USB-MassNames-MassName 0>>i
Ignore flag has been set
*-SystemEdit-USB-MassNames-MassName 0>>

*-SystemEdit-USB-MassNames>>p
0. 0.00 (i)
1. No Device
2. No Device
3. No Device
*-SystemEdit-USB-MassNames>>

*-SystemEdit-USB>>p
0. UsbCounts
USB Drive's Count      1 Drive
USB Kbd's Count        0 Keyboard
USB Mouse's Count      0 Mouse
USB Hub's Count        1 Hub
USB Point's Count      0 Point
USB Ccid's Count       0 SmartCard Reader
1. MassNames
USB MassStorage(0)     0.00 (i)
USB MassStorage(1)     No Device
USB MassStorage(2)     No Device
USB MassStorage(3)     No Device
2. UsbPorts
USB Port 0             Not Connected
USB Port 1             Not Connected
USB Port 2             Not Connected
USB Port 3             Connected
USB Port 4             Not Connected
*-SystemEdit-USB>>
    << KONTRON SYSTEM PBIT : EDIT MODE >>

Edit by feature, choose: PCI (*) CPLD (*) SATA (*) ETH (*) USB (*) SMBUS (*) BIOS (*)
Other available cmds: `s` for settings, `?` for help, `q` to quit edit mode

*-SystemEdit>>
    << KONTRON SYSTEM PBIT : EDIT MODE >>

Edit by feature, choose: PCI (*) CPLD (*) SATA (*) ETH (*) USB (*) SMBUS (*) BIOS (*)
Other available cmds: `s` for settings, `?` for help, `q` to quit edit mode

*-SystemEdit>>q
Quit Edit menu

Modifications have been made.
Do you want to save your configuration before leaving? (y/n)
*-SystemEdit-Save>>y
VX6090>
VX6090> kdiag run system
PBIT "system" (fast,simple)

```

```

PCI (*) CPLD (*) SATA (*) SMBUS (*) BIOS (*) ETH (*) USB (*)
ERR)PCI Bus:Dev:Func (R)00:1F:02 (D)00:1F:03
ERR)PCI 00:1F:02 DeviceID (R)0x8C02 (D)0x8C22
ERR)PCI Bus:Dev:Func (R)00:1F:03 (D)02:00:00
ERR)PCI 00:1F:03 DeviceID (R)0x8C22 (D)0x6F50
ERR)PCI 00:1F:03 ClassCode (R)0C0500 (D)0C0500
ERR)PCI Bus:Dev:Func (R)02:00:00 (D)02:00:01
ERR)PCI 02:00:00 DeviceID (R)0x6F50 (D)0x6F51
ERR)PCI Bus:Dev:Func (R)02:00:01 (D)02:00:02
ERR)PCI 02:00:01 DeviceID (R)0x6F51 (D)0x6F52
ERR)PCI Bus:Dev:Func (R)02:00:02 (D)02:00:03
ERR)PCI 02:00:02 DeviceID (R)0x6F52 (D)0x6F53
ERR)PCI Bus:Dev:Func (R)02:00:03 (D)03:00:00
ERR)PCI 02:00:03 DeviceID (R)0x6F53 (D)0x15AB
ERR)PCI 02:00:03 ClassCode (R)088000 (D)088000
ERR)PCI Bus:Dev:Func (R)03:00:00 (D)03:00:01
ERR)PCI Bus:Dev:Func (R)03:00:01 (D)05:00:00
ERR)PCI 03:00:01 DeviceID (R)0x15AB (D)0x1537
ERR)PCI Bus:Dev:Func (R)05:00:00 (D)06:00:00
ERR)PCI 05:00:00 DeviceID (R)0x1537 (D)0x1533
ERR)PCI Bus:Dev:Func (R)06:00:00 (D)08:00:00
ERR)PCI 06:00:00 VendorID (R)0x8086 (D)0x126F
ERR)PCI 06:00:00 DeviceID (R)0x1533 (D)0x0750
ERR)PCI 06:00:00 ClassCode (R)020000 (D)020000
ERR)USB MassStorage(1) (R)No Device (D) 0.00
FAILED

```

```
VX6090> kdiag edit system
```

```
<< KONTRON SYSTEM PBIT : EDIT MODE >>
```

```
Edit by feature, choose: PCI (*) CPLD (*) SATA (*) ETH (*) USB (*) SMBUS (*) BIOS (*)
Other available cmds: `s` for settings, `?` for help, `q` to quit edit mode
```

```
--SystemEdit>>pci
```

```
--SystemEdit-PCI>>p
```

```

0. PCI 00:00:00 Bridge Device/Host/PCI bridge
1. PCI 00:01:00 Bridge Device/PCI/PCI bridge
2. PCI 00:02:00 Bridge Device/PCI/PCI bridge
3. PCI 00:02:02 Bridge Device/PCI/PCI bridge
4. PCI 00:03:00 Bridge Device/PCI/PCI bridge
5. PCI 00:03:02 Bridge Device/PCI/PCI bridge
6. PCI 00:03:03 Bridge Device/PCI/PCI bridge
7. PCI 00:05:00 Base System Peripherals/Other system peripheral
8. PCI 00:05:01 Base System Peripherals/Other system peripheral
9. PCI 00:05:02 Base System Peripherals/Other system peripheral
10. PCI 00:05:04 Base System Peripherals/PIC
11. PCI 00:14:00 Serial Bus Controllers/USB
12. PCI 00:1C:00 Bridge Device/PCI/PCI bridge
13. PCI 00:1C:02 Bridge Device/PCI/PCI bridge
14. PCI 00:1D:00 Serial Bus Controllers/USB
15. PCI 00:1F:00 Bridge Device/PCI/ISA bridge
16. PCI 00:1F:02 Mass Storage Controller/Serial ATA controller [FAILED ONCE] [LAST FAILED]
17. PCI 00:1F:03 Serial Bus Controllers/System Management Bus [FAILED ONCE] [LAST FAILED]
18. PCI 02:00:00 Base System Peripherals/Other system peripheral [FAILED ONCE] [LAST FAILED]
19. PCI 02:00:01 Base System Peripherals/Other system peripheral [FAILED ONCE] [LAST FAILED]
--SystemEdit-PCI>>?

```

```

pa : print all Recorded PCI Device infos
pb : print all Recorded PCI/PCI Bridge infos
p1 : print Recorded VendorID only

```

```

p2 : print Recorded DeviceID only
p3 : print Recorded ClassCode only
p4 : print Recorded Linkstatus only (only for PCI/PCI bridge)
p5 : print Recorded Link Speed/Width only (only for PCI/PCI bridge)
p  : print Recorded values
0,..,49 : edit PCI element
ignoreall : ignore all PCI infos
default : restore default PCI config (remove flags)
? : help
q : go back to main menu

--SystemEdit-PCI>>ignoreall

PCI entirely ignored
*-SystemEdit-PCI>>

    << KONTRON SYSTEM PBIT : EDIT MODE >>

Edit by feature, choose: PCI (i) CPLD (*) SATA (*) ETH (*) USB (*) SMBUS (*) BIOS (*)
Other available cmds: `s` for settings, `?` for help, `q` to quit edit mode

*-SystemEdit>>sata

*-SystemEdit-SATA>>ignoreall

SATA entirely ignored
*-SystemEdit-SATA>>

    << KONTRON SYSTEM PBIT : EDIT MODE >>

Edit by feature, choose: PCI (i) CPLD (*) SATA (i) ETH (*) USB (*) SMBUS (*) BIOS (*)
Other available cmds: `s` for settings, `?` for help, `q` to quit edit mode

*-SystemEdit>>q
Quit Edit menu

Modifications have been made.
Do you want to save your configuration before leaving? (y/n)
*-SystemEdit-Save>>y
VX6090>
VX6090> kdiag run system
PBIT "system" (fast,simple)
PCI (i) CPLD (*) SATA (i) SMBUS (*) BIOS (*) ETH (*) USB (*)
ERR)USB MassStorage(1)      (R)No Device          (D) 0.00
FAILED

VX6090> kdiag edit system

    << KONTRON SYSTEM PBIT : EDIT MODE >>

Edit by feature, choose: PCI (i) CPLD (*) SATA (i) ETH (*) USB (*) SMBUS (*) BIOS (*)
Other available cmds: `s` for settings, `?` for help, `q` to quit edit mode

--SystemEdit>>usb

--SystemEdit-USB>>p

0. UsbCounts
USB Drive's Count      1 Drive
USB Kbd's Count        0 Keyboard
USB Mouse's Count      0 Mouse
USB Hub's Count        1 Hub
USB Point's Count      0 Point

```

```

USB Ccid's Count      0 SmartCard Reader
1. MassNames
USB MassStorage(0)    0.00                (i)
USB MassStorage(1)    No Device                [FAILED ONCE] [LAST FAILED]
USB MassStorage(2)    No Device
USB MassStorage(3)    No Device
2. UsbPorts
USB Port 0            Not Connected
USB Port 1            Not Connected
USB Port 2            Not Connected
USB Port 3            Connected
USB Port 4            Not Connected
--SystemEdit-USB>>1

--SystemEdit-USB-MassNames>>1

--SystemEdit-USB-MassNames-MassName 1>>i

Ignore flag has been set
*-SystemEdit-USB-MassNames-MassName 1>>?

p : print Recorded value
i : set ignore flag
r : remove flag
? : help
q : go back to MassNames menu

*-SystemEdit-USB-MassNames-MassName 1>>

*-SystemEdit-USB-MassNames>>ignoreall

MassNames entirely ignored
*-SystemEdit-USB-MassNames>>

*-SystemEdit-USB>>

    << KONTRON SYSTEM PBIT : EDIT MODE >>

Edit by feature, choose: PCI (i) CPLD (*) SATA (i) ETH (*) USB (*) SMBUS (*) BIOS (*)
Other available cmds: `s` for settings, `?` for help, `q` to quit edit mode

*-SystemEdit>>q
Quit Edit menu

Modifications have been made.
Do you want to save your configuration before leaving? (y/n)
*-SystemEdit-Save>>y
VX6090> kdiag run system
PBIT "system" (fast,simple)
PCI (i) CPLD (*) SATA (i) SMBUS (*) BIOS (*) ETH (*) USB (*)
PASSED

VX6090>

```

System stats are displayed when we print recorded values with command 'p' ([LAST FAILED] and/or [FAILED ONCE] labels).

## 2.10.3.2 Print Debug Settings

By default, the system test prints errors only, and detail level is normal. It is possible to modify these general parameters in this menu.

▶ **Three print levels are available:**

- ▶ Errors only,
- ▶ Errors and Infos only,
- ▶ Debug lvl (print all).

Each printed message in system test corresponds to one of these levels and is easily identifiable thanks to 3 labels:

- ▶ Error: **ERR)**
- ▶ Info: **INF)**
- ▶ Debug: **DBG)**

▶ **Scroll limit for Debug 1vl mode ONLY:**

If Debug mode is selected, a scroll limit can be set in order to stop scrolling during system test, depending on the chosen limit: 0 is no limit (do not stop scrolling by default), max for scroll limit is 30 lines.

WARNING: WATCH- DOG WILL BE STOPPED during "system" test execution IF SCROLL LIMIT IS NOT 0. Then, at the next system run, use space (scroll to the limit) or Enter (scroll line by line) to make test results scroll.

▶ **Three detail levels are available (for experts because not very useful for user): synthetic mode, normal mode and detailed mode.**

- ▶ Synthetic mode does not display errors and information results but gives the number of errors found by type and the total.

Example with SYNTHETIC MODE:

```
VX6090> kdiag run system
PBIT "system" (fast,simple)
PCI (*) CPLD (*) SATA (*) SMBUS (i) BIOS (*) ETH (*) USB...
ERR)PCI Results : 51 ERRORS
ERR)USB Results : 3 ERRORS
ERR)TOTAL : 54 ERRORS
FAILED
```

- ▶ Normal mode displays system test results depending on the print level. The classcode value will be printed for PCI devices.
- ▶ Detailed mode allows to print more details for PCI devices:  
It prints classcode value but also prog interface nb, BaseClass string, SubClass string which are deducted from classcode value.

▶ **Clear System Stats: used to clear system stats (see section 2.10.4 page 41).**

The following shows a system\_edit session setting print level to Debug Level, a scroll limit of 10 lines and detail level set to Detailed mode:

```
VX6090> kdiag edit system

<< KONTRON SYSTEM PBIT : EDIT MODE >>

Edit by feature, choose: PCI (i) CPLD (*) SATA (i) ETH (*) USB (*) SMBUS (*) BIOS (*)
Other available cmds: `s` for settings, `?` for help, `q` to quit edit mode

--SystemEdit>>s

p : Set Print Level
Current Print Lvl is: ERRORS ONLY
```

```

d : Set Detail Level
Current Details Lvl is: NORMAL
c : Clear System Stats
--SystemEdit-generalSettings>>p

--SystemEdit-generalSettings-printLvl>>?

p : print current print Level
e : ERRORS ONLY mode
i : ERRORS & INFOS ONLY mode
d : DEBUG LVL mode (print all)
q : return to general settings menu
--SystemEdit-generalSettings-printLvl>>d

Print Lvl DEBUG PRINT ALL set, scroll limit is 24 by default
Set Scroll limit (0 = no limit , max = 30)
*-SystemEdit-generalSettings-printLvl-Scroll>>10

Scroll Limit set to 10 line(s)
*-SystemEdit-generalSettings-printLvl>>q

p : Set Print Level
Current Print Lvl is: DEBUG PRINT ALL
d : Set Detail Level
Current Details Lvl is: NORMAL
c : Clear System Stats
*-SystemEdit-generalSettings>>d

*-SystemEdit-generalSettings-detailsLvl>>?

p : print Current Details Lvl
s : synthethic results
n : normal infos
d : detailed
q : return to general settings menu
*-SystemEdit-generalSettings-detailsLvl>>p

Current Details Lvl is: NORMAL
*-SystemEdit-generalSettings-detailsLvl>>d

Details Lvl DETAILED set
*-SystemEdit-generalSettings-detailsLvl>>p

Current Details Lvl is: DETAILED
*-SystemEdit-generalSettings-detailsLvl>>q

p : Set Print Level
Current Print Lvl is: DEBUG PRINT ALL
d : Set Detail Level
Current Details Lvl is: DETAILED
c : Clear System Stats
*-SystemEdit-generalSettings>>
    << KONTRON SYSTEM PBIT : EDIT MODE >>

Edit by feature, choose: PCI (i) CPLD (*) SATA (i) ETH (*) USB (*) SMBUS (*) BIOS (*)
Other available cmds: `s` for settings, `?` for help, `q` to quit edit mode

*-SystemEdit>>q
Quit Edit menu-

```



```

INF)PCI No Device      ignored
INF)PCI No Device      ignored
INF)PCI No Device      ignored
INF)PCI No Device      ignored
INF)PCI No Device      ignored
INF)PCI No Device      ignored
DBG)CPLD Version       (R)0x43                (D)0x43
DBG)CPLD Dram Size     (R)8GB                 (D)8GB
INF)CPLD SysCon        ignored
DBG)CPLD Geo Address   (R)1                   (D)1
DBG)CPLD XMC Presence  (R)XMC not present    (D)XMC not present
INF)SATA Port 0        ignored
INF)SATA Port 1        ignored
INF)SATA Port 2        ignored
INF)SATA Port 3        ignored
INF)SATA Port 4        ignored
INF)SATA Port 5        ignored
DBG)SMBUSO Device 0    (R)0x30                (D)0x30
INF)BIOS Checksum      ignored
DBG)BIOS Version       (R)ID16293EF          (D)ID16293EF
DBG)ETH Port 0         (R)Link UP speed 10000 Mb/s(D)Link UP speed 10000 Mb/s
DBG)ETH Port 1         (R)Link UP speed 10000 Mb/s(D)Link UP speed 10000 Mb/s
INF)ETH Port 2         ignored
INF)ETH Port 3         ignored
DBG)USB Drive's Count  (R)1 Drive             (D)1 Drive
DBG)USB Kbd's Count    (R)0 Keyboard          (D)0 Keyboard
DBG)USB Mouse's Count  (R)0 Mouse             (D)0 Mouse
DBG)USB Hub's Count    (R)1 Hub               (D)1 Hub
DBG)USB Point's Count  (R)0 Point             (D)0 Point
DBG)USB Ccid's Count   (R)0 SmartCard Reader  (D)0 SmartCard Reader
INF)USB MassStorage(0) ignored
INF)USB MassStorage(1) ignored
INF)USB MassStorage(2) ignored
INF)USB MassStorage(3) ignored
DBG)USB Port 0         (R)Not Connected      (D)Not Connected
DBG)USB Port 1         (R)Not Connected      (D)Not Connected
DBG)USB Port 2         (R)Not Connected      (D)Not Connected
DBG)USB Port 3         (R)Connected          (D)Connected
DBG)USB Port 4         (R)Not Connected      (D)Not Connected
PASSED

```

VX6090>

Use important commands ' ? ' and ' p ' to get help and to see recorded values respectively. Note that return (just Enter) is equivalent to ' q ' (go back to previous menu). Ignored elements are marked with the ( i ) label. Stats are shown through [LAST FAILED] and/or [FAILED ONCE] labels.

If the system configuration is edited again, then the excluded information will be marked **IGNORED** and will not participate to the test. If **IGNORED** items changed after the **system\_learn** phase, the system test will not fail because the tests corresponding to the items are bypassed.

Now, it is possible to remove **IGNORED** flags one by one (browse menu and type ' r ' to remove ignore flag of an element) or remove all flags at once of an object or type (to be placed in the right place in the menu and type ' default ' for a group of elements) in order to test items again.

## 2.10.4 Clearing System Stats

To clear [LAST FAILED] and [FAILED ONCE] labels of system test items, enter the **kdiag edit system** and choose the section for general settings:

```
VX6090> kdiag edit system

    << KONTRON SYSTEM PBIT : EDIT MODE >>

Edit by feature, choose: PCI (i) CPLD (*) SATA (i) ETH (*) USB (*) SMBUS (*) BIOS (*)
Other available cmds: `s` for settings, `?` for help, `q` to quit edit mode

--SystemEdit>>s

p : Set Print Level
Current Print Lvl is: DEBUG PRINT ALL
d : Set Detail Level
Current Details Lvl is: DETAILED
c : Clear System Stats
--SystemEdit-generalSettings>>c

System stats have been cleared

p : Set Print Level
Current Print Lvl is: DEBUG PRINT ALL
d : Set Detail Level
Current Details Lvl is: DETAILED
c : Clear System Stats
--SystemEdit-generalSettings>>q

    << KONTRON SYSTEM PBIT : EDIT MODE >>

Edit by feature, choose: PCI (i) CPLD (*) SATA (i) ETH (*) USB (*) SMBUS (*) BIOS (*)
Other available cmds: `s` for settings, `?` for help, `q` to quit edit mode

--SystemEdit>>q
Quit Edit menu

No modifications, quit without saving..
VX6090>
```

Verify the labels have disappeared, for example for USB and BIOS entries compared with outputs given in section 2.10.3.1 page 26:

```
VX6090> kdiag edit system

    << KONTRON SYSTEM PBIT : EDIT MODE >>

Edit by feature, choose: PCI (i) CPLD (*) SATA (i) ETH (*) USB (*) SMBUS (*) BIOS (*)
Other available cmds: `s` for settings, `?` for help, `q` to quit edit mode

--SystemEdit>>usb

--SystemEdit-USB>>p

0. UsbCounts
USB Drive's Count      1 Drive
USB Kbd's Count        0 Keyboard
```

```

USB Mouse's Count      0 Mouse
USB Hub's Count        1 Hub
USB Point's Count      0 Point
USB Ccid's Count       0 SmartCard Reader
1. MassNames
USB MassStorage(0)     0.00                (i)
USB MassStorage(1)     No Device                (i)
USB MassStorage(2)     No Device                (i)
USB MassStorage(3)     No Device                (i)
2. UsbPorts
USB Port 0             Not Connected
USB Port 1             Not Connected
USB Port 2             Not Connected
USB Port 3             Connected
USB Port 4             Not Connected
--SystemEdit-USB>>

    << KONTRON SYSTEM PBIT : EDIT MODE >>

Edit by feature, choose: PCI (i) CPLD (*) SATA (i) ETH (*) USB (*) SMBUS (*) BIOS (*)
Other available cmds: `s` for settings, `?` for help, `q` to quit edit mode

--SystemEdit>>bios

--SystemEdit-BIOS>>p

0. BIOS Checksum       124299                (i)
1. BIOS Version        ID16293EF
--SystemEdit-BIOS>>

    << KONTRON SYSTEM PBIT : EDIT MODE >>

Edit by feature, choose: PCI (i) CPLD (*) SATA (i) ETH (*) USB (*) SMBUS (*) BIOS (*)
Other available cmds: `s` for settings, `?` for help, `q` to quit edit mode

--SystemEdit>>q
Quit Edit menu

No modifications, quit without saving..
VX6090>

```

You must exit the menu so that settings changes are taken into account.

## 2.11 Bypass the PBIT Tests

The following toggle command can be used to avoid the PBIT to run the tests:

```
VX6090> kdiag bypass
PBIT BYPASS set
```

To disable the bypass mode, re-execute the same command:

```
VX6090> kdiag bypass
PBIT BYPASS removed
```

This is useful to bypass the PBIT tests if they are configured to run automatically.

This feature is also accessible thru the **kdiag** Linux utility to enable/disable the tests under OS at the next boot.

## 3 / PBIT and OS Interfaces

The PBIT synthetic result and the PBIT detailed results are accessible under the Operating System delivered with the VX6090 board.

The following gives you an overview of the facilities available under Linux Board Support Package.

Please refer to the appropriate OS Release Notes document for more details.

### 3.1 Linux

#### 3.1.1 Linux Synthetic PBIT Result

The synthetic PBIT result is stored in the CPLD register 0x2.

If the Linux BSP is available for VX6090, the synthetic PBIT result is accessible by using the "**cp1dtool**" utility:

```
[root@localhost ~]# man cp1dtool
CPLD00L(1)                                CPLD00L(1)

NAME
    cp1dtool - Kontron board cPLD register utility

SYNOPSIS
    -a
    -d register
    -f register fieldname hexvalue ...
    -i register
    -s register hexvalue
    -v
    (...)

```

Use the "**-a**" or "**-d 0x2**" options to get the register 0x2 value.

If the "**cp1dtool**" utility is not available, CPLD registers can be read by using the "**port**" command:

```
[root@localhost ~]# port 0x802
@0x802 = 0x01

```

**0x800** being the CPLD base address (refer to **/proc/ioprots**).

Value **0x1** indicates the PBIT has run (bit0) and there was no error (bit7-1).

#### 3.1.2 Linux Detailed PBIT Result

If the Linux BSP is available for VX6090, the detailed PBIT results (corresponding to the EFI shell command "**kdiag -stat**") are accessible under Linux by using the "**sysvartool**" utility.

Use the "**-A pbit -1**" options to display the PBIT results.

#### 3.1.3 Linux Kdiag Utility

The Linux "**kdiag**" utility can be used to configure the PBIT directly under OS.

It is distributed in the BIT-6-VX6090 product including the PBIT Firmware and the Linux BIT for VX6090.

BIT-6-VX6090 ID16193 includes the Linux BIT V2.3 ID16130 that supports the following features:

- ▶ Listing and modifying the PBIT test list to be run at the next boot
- ▶ Getting and clearing the PBIT status

- ▶ Editing the system menu and choose which items will be ignored during the next system test executions
- ▶ Bypassing the PBIT tests at the next boot

The Linux **kdiag** command syntax is similar to the PBIT **kdiag** command used under the BIOS EFI shell:

```
# kdiag help

kdiag    - perform board diagnostics

----- Usage -----

Print list of PBITs and infos about them :

kdiag <PBITname>|<PBITnum> ...

    <PBITname>|<PBITnum> ...
        list of PBIT(s) to display. All if the list is empty.
        PBIT(s) are referenced using their name or their number.

Print PBIT(s) status :

kdiag stat [oside] [<PBITname>|<PBITnum> ...]

    <PBITname>|<PBITnum> ...
        list of PBIT(s) to display. All if the list is empty.
        PBIT(s) are referenced using their name or their number.

oside (VX6060 board only): print PBIT status of peer CPU side

Clear PBIT(s) status :

kdiag [oside] clrstat|clrallstat [<PBITname>|<PBITnum> ...]

    clrstat    : Reset status to NOTRUN
    clrallstat : Reset status to NOTRUN and clear the (FAILED once) flag

    <PBITname>|<PBITnum> ...
        list of PBIT(s) to clear. All if the list is empty.
        PBIT(s) are referenced using their name or their number.

    oside (VX6060 board only): clear PBIT status of peer CPU side

Delete all PBITs from configuration :

kdiag [oside] deleteall

    oside (VX6060 board only): delete all PBITs for peer CPU side

Configure PBIT(s) :

kdiag [oside] cfg <cfgarg> ... <PBITname>|<PBITnum> ...

    cfg <cfgarg> : Configure one or several PBIT(s).
                  <cfgarg> is either :
                  - "delete" to delete PBIT(s) from the list of configured PBITs
                  - "default" to configure PBIT(s) with a default run mode
                  - a comma separated list of runflags defining a PBIT
                    run mode; for example : fast,complex.
```

```

valid runflags are :
- "SPEED" flags (can NOT be mixed)
  slow      : run in slow mode (full testing)
  fast      : run in fast mode (fast testing)
- "CONFIG" flags (can NOT be mixed)
  simple    : run in simple mode (no external hardware)
  complex   : run in complex mode (needs external hardware)
- "HALT" flags (can NOT be mixed)
  haltonfail : halt immediately (hang) if test fails
  promptonfail : halt at U-Boot prompt (no OS) if test fails
- "RESET" flags (can be mixed together)
  normalreset : run after a normal reset
  poweronreset : run after a power-on reset
  allresets   : run after all resets listed above

```

```

[<PBITname>|<PBITnum>] ...
  list of PBIT(s) to configure.
  PBIT(s) are referenced using their name or their number.
  All missing tests if the list is empty (but the cfgarg is default).

```

oside (VX6060 board only): Configure PBIT(s) for peer CPU side

Edit System Configuration for system Test:  
(cf documentation for details)

```
kdiag system_edit or kdiag edit system
```

Clear System Configuration for system Test:

```
kdiag system_clear or kdiag clear system
```

Display kdiag command version :

```
kdiag version
```

Toggle PBIT Bypassed:

```
kdiag bypass
```

For example:

```
# kdiag version
kdiag command Version: V2.3 16130
```

```
# kdiag
PBITs configured to run from command line :
mem_data (1) - capabilities : fast,simple,allresets
  run mode 1 : fast,simple,allresets
mem_addr (2) - capabilities : fast,simple,allresets
  run mode 1 : fast,simple,allresets
mem_pattern1 (6) - capabilities : slow/fast,simple,allresets
  run mode 1 : slow,simple,allresets
mem_pattern2 (7) - capabilities : slow/fast,simple,allresets
  run mode 1 : slow,simple,allresets
```

```

mem_pattern3 (8) - capabilities : slow/fast,simple,allresets
run mode 1 : slow,simple,allresets
mem_pattern4 (9) - capabilities : slow/fast,simple,allresets
run mode 1 : slow,simple,allresets
core (10) - capabilities : fast,simple,allresets
run mode 1 : fast,simple,allresets
tpm (11) - capabilities : fast,simple/complex,allresets
run mode 1 : fast,simple,allresets
m2_bottom (14) - capabilities : fast,simple/complex,allresets
run mode 1 : fast,simple,allresets
m2_top (15) - capabilities : fast,simple/complex,allresets
run mode 1 : fast,simple,allresets
serial (16) - capabilities : fast,simple/complex,allresets
run mode 1 : fast,simple,allresets
onboardvideo (17) - capabilities : fast,simple,allresets
run mode 1 : fast,simple,allresets
rtc (20) - capabilities : fast,simple,allresets
run mode 1 : fast,simple,allresets
sysflash (22) - capabilities : fast,simple,allresets
run mode 1 : fast,simple,allresets
cpId (24) - capabilities : fast,simple/complex,allresets
run mode 1 : fast,simple,allresets
fram (40) - capabilities : fast,simple/complex,allresets
run mode 1 : fast,simple,allresets
ether_loop0 (55) - capabilities : fast,simple/complex,allresets
run mode 1 : fast,simple,allresets
ether_loop1 (56) - capabilities : fast,simple/complex,allresets
run mode 1 : fast,simple,allresets
ether_loop2 (57) - capabilities : fast,simple/complex,allresets
run mode 1 : fast,simple,allresets
ether_loop3 (58) - capabilities : fast,simple/complex,allresets
run mode 1 : fast,simple,allresets
hwmon (61) - capabilities : fast,simple/complex,allresets
run mode 1 : fast,simple,allresets
sata0_controller (68) - capabilities : fast,simple/complex,allresets
run mode 1 : fast,simple,allresets
sata1_controller (69) - capabilities : fast,simple/complex,allresets
run mode 1 : fast,simple,allresets
vpd (70) - capabilities : fast,simple/complex,allresets
run mode 1 : fast,simple,allresets
eeprom (71) - capabilities : fast,simple/complex,allresets
run mode 1 : fast,simple,allresets
ehci_controller (85) - capabilities : fast,simple,allresets
run mode 1 : fast,simple,allresets
xhci_controller (86) - capabilities : fast,simple,allresets
run mode 1 : fast,simple,allresets
system (89) - capabilities : fast,simple/complex,allresets
run mode 1 : fast,simple,allresets

```

Other PBITs available but not yet configured :

```

mem_bitflip (3) - capabilities : slow/fast,simple,allresets
mem_addrpat (4) - capabilities : slow/fast,simple,allresets
mem_addrpat2 (5) - capabilities : slow/fast,simple,allresets
pcie_vpx_sw (13) - capabilities : fast,simple/complex,allresets
smbus0 (26) - capabilities : fast,simple/complex,allresets
smbus1 (27) - capabilities : fast,simple/complex,allresets
ether_link0 (50) - capabilities : fast,simple/complex,allresets
ether_link1 (51) - capabilities : fast,simple/complex,allresets
ether_link2 (52) - capabilities : fast,simple/complex,allresets
ether_link3 (53) - capabilities : fast,simple/complex,allresets

```

```

sata0_dev_see (62) - capabilities : fast,simple,allresets
sata1_dev_see (63) - capabilities : fast,simple,allresets
sata2_dev_see (64) - capabilities : fast,simple,allresets
sata4_dev_see (66) - capabilities : fast,simple,allresets
sata5_dev_see (67) - capabilities : fast,simple,allresets
usb0_dev_see (80) - capabilities : fast,simple,allresets
usb1_dev_see (81) - capabilities : fast,simple,allresets
usb3_dev_see (83) - capabilities : fast,simple,allresets
faultytest (98) - capabilities : fast,simple,allresets
hangtest (99) - capabilities : fast,simple,allresets

```

Use 'kdiag help' to get more info.

```
#
```

```

# kdiag cfg promptonfail 50 51
# kdiag cfg poweronreset 3 4 5
# kdiag
PBITs configured to run from command line :
mem_data (1) - capabilities : fast,simple,allresets
  run mode 1 : fast,simple,allresets
mem_addr (2) - capabilities : fast,simple,allresets
  run mode 1 : fast,simple,allresets
mem_bitflip (3) - capabilities : slow/fast,simple,allresets
  run mode 1 : fast,simple,poweronreset
mem_addrpat (4) - capabilities : slow/fast,simple,allresets
  run mode 1 : fast,simple,poweronreset
mem_addrpat2 (5) - capabilities : slow/fast,simple,allresets
  run mode 1 : fast,simple,poweronreset
mem_pattern1 (6) - capabilities : slow/fast,simple,allresets
  run mode 1 : slow,simple,allresets
mem_pattern2 (7) - capabilities : slow/fast,simple,allresets
  run mode 1 : slow,simple,allresets
mem_pattern3 (8) - capabilities : slow/fast,simple,allresets
  run mode 1 : slow,simple,allresets
mem_pattern4 (9) - capabilities : slow/fast,simple,allresets
  run mode 1 : slow,simple,allresets
core (10) - capabilities : fast,simple,allresets
  run mode 1 : fast,simple,allresets
tpm (11) - capabilities : fast,simple/complex,allresets
  run mode 1 : fast,simple,allresets
m2_bottom (14) - capabilities : fast,simple/complex,allresets
  run mode 1 : fast,simple,allresets
m2_top (15) - capabilities : fast,simple/complex,allresets
  run mode 1 : fast,simple,allresets
serial (16) - capabilities : fast,simple/complex,allresets
  run mode 1 : fast,simple,allresets
onboardvideo (17) - capabilities : fast,simple,allresets
  run mode 1 : fast,simple,allresets
rtc (20) - capabilities : fast,simple,allresets
  run mode 1 : fast,simple,allresets
sysflash (22) - capabilities : fast,simple,allresets
  run mode 1 : fast,simple,allresets
cp1d (24) - capabilities : fast,simple/complex,allresets
  run mode 1 : fast,simple,allresets
fram (40) - capabilities : fast,simple/complex,allresets
  run mode 1 : fast,simple,allresets
ether_link0 (50) - capabilities : fast,simple/complex,allresets
  run mode 1 : fast,simple,promptonfail,allresets
ether_link1 (51) - capabilities : fast,simple/complex,allresets
  run mode 1 : fast,simple,promptonfail,allresets

```

```

ether_loop0 (55) - capabilities : fast,simple/complex,allresets
  run mode 1 : fast,simple,allresets
ether_loop1 (56) - capabilities : fast,simple/complex,allresets
  run mode 1 : fast,simple,allresets
ether_loop2 (57) - capabilities : fast,simple/complex,allresets
  run mode 1 : fast,simple,allresets
ether_loop3 (58) - capabilities : fast,simple/complex,allresets
  run mode 1 : fast,simple,allresets
hwmon (61) - capabilities : fast,simple/complex,allresets
  run mode 1 : fast,simple,allresets
sata0_controler (68) - capabilities : fast,simple/complex,allresets
  run mode 1 : fast,simple,allresets
sata1_controler (69) - capabilities : fast,simple/complex,allresets
  run mode 1 : fast,simple,allresets
vpd (70) - capabilities : fast,simple/complex,allresets
  run mode 1 : fast,simple,allresets
eeprom (71) - capabilities : fast,simple/complex,allresets
  run mode 1 : fast,simple,allresets
ehci_controller (85) - capabilities : fast,simple,allresets
  run mode 1 : fast,simple,allresets
xhci_controller (86) - capabilities : fast,simple,allresets
  run mode 1 : fast,simple,allresets
system (89) - capabilities : fast,simple/complex,allresets
  run mode 1 : fast,simple,allresets

```

Other PBITs available but not yet configured :

```

pcie_vpx_sw (13) - capabilities : fast,simple/complex,allresets
smbus0 (26) - capabilities : fast,simple/complex,allresets
smbus1 (27) - capabilities : fast,simple/complex,allresets
ether_link2 (52) - capabilities : fast,simple/complex,allresets
ether_link3 (53) - capabilities : fast,simple/complex,allresets
sata0_dev_see (62) - capabilities : fast,simple,allresets
sata1_dev_see (63) - capabilities : fast,simple,allresets
sata2_dev_see (64) - capabilities : fast,simple,allresets
sata4_dev_see (66) - capabilities : fast,simple,allresets
sata5_dev_see (67) - capabilities : fast,simple,allresets
usb0_dev_see (80) - capabilities : fast,simple,allresets
usb1_dev_see (81) - capabilities : fast,simple,allresets
usb3_dev_see (83) - capabilities : fast,simple,allresets
faultytest (98) - capabilities : fast,simple,allresets
hangtest (99) - capabilities : fast,simple,allresets

```

Use 'kdiag help' to get more info.

#

```

# kdiag stat
Display status
Status of PBITs configured to run from command line :
PASSED : mem_data (fast,simple)
PASSED : mem_addr (fast,simple)
PASSED : mem_pattern1 (slow,simple)
PASSED : mem_pattern2 (slow,simple)
PASSED : mem_pattern3 (slow,simple)
PASSED : mem_pattern4 (slow,simple)
PASSED : core (fast,simple)

```

```

PASSED : tpm (fast,simple)
PASSED : m2_bottom (fast,simple)
PASSED : m2_top (fast,simple)
PASSED : serial (fast,simple)
PASSED : onboardvideo (fast,simple)
FAILED : rtc (fast,simple)
PASSED : sysflash (fast,simple)
PASSED : cpld (fast,simple)
PASSED : fram (fast,simple)
PASSED : ether_loop0 (fast,simple)
PASSED : ether_loop1 (fast,simple)
PASSED : ether_loop2 (fast,simple)
PASSED : ether_loop3 (fast,simple)
PASSED : hwmon (fast,simple)
PASSED : sata0_controler (fast,simple)
PASSED : sata1_controler (fast,simple)
PASSED : vpd (fast,simple)
PASSED : eeprom (fast,simple)
PASSED : ehci_controller (fast,simple)
PASSED : xhci_controller (fast,simple)
PASSED : system (fast,simple)

```

```

RUN      : 28
PASSED   : 27
FAILED   : 1
NOT_RUN  : 0
#

```

```

# kdiag edit system
    << KONTRON SYSTEM PBIT : EDIT MODE >>

```

```

Edit by feature, choose:PCI CPLD SATA ETH USB SMBUS BIOS
Other available cmds: `s` for settings, `?` for help, `q` to quit edit mode

```

```
--SystemEdit>>pci
```

```
--SystemEdit-PCI>>?
```

```

pa : print all Recorded PCI device infos
pb : print all Recorded PCI/PCI Bridge infos
p1 : print Recorded VendorID only
p2 : print Recorded DeviceID only
p3 : print Recorded ClassCode only
p4 : print Recorded Linkstatus only (only for PCI/PCI bridge)
p5 : print Recorded Link Speed/Width only (only for PCI/PCI bridge)
p  : print Recorded values
0,..,49 : edit PCI element
ignoreall : ignore all PCI infos
default : restore default PCI config (remove flags)
? : help
q : go back to main menu

```

```
--SystemEdit-PCI>>p1
```

```

0. PCI 00:00:00 VendorID 0x8086
1. PCI 00:01:00 VendorID 0x8086
2. PCI 00:02:00 VendorID 0x8086
3. PCI 00:02:02 VendorID 0x8086
4. PCI 00:03:00 VendorID 0x8086
5. PCI 00:03:02 VendorID 0x8086

```

```
6. PCI 00:03:03 VendorID 0x8086
7. PCI 00:05:00 VendorID 0x8086
8. PCI 00:05:01 VendorID 0x8086
9. PCI 00:05:02 VendorID 0x8086
10. PCI 00:05:04 VendorID 0x8086
11. PCI 00:14:00 VendorID 0x8086
12. PCI 00:1c:00 VendorID 0x8086
13. PCI 00:1c:02 VendorID 0x8086
14. PCI 00:1d:00 VendorID 0x8086
15. PCI 00:1f:00 VendorID 0x8086
16. PCI 00:1f:02 VendorID 0x8086
17. PCI 00:1f:03 VendorID 0x8086
18. PCI 02:00:00 VendorID 0x8086
19. PCI 02:00:01 VendorID 0x8086
20. PCI 02:00:02 VendorID 0x8086
21. PCI 02:00:03 VendorID 0x8086
22. PCI 03:00:00 VendorID 0x8086
23. PCI 03:00:01 VendorID 0x8086
24. PCI 05:00:00 VendorID 0x8086
25. PCI 06:00:00 VendorID 0x8086
26. PCI 08:00:00 VendorID 0x126f
27. PCI 00:00:00 VendorID No Device
28. PCI 00:00:00 VendorID No Device
29. PCI 00:00:00 VendorID No Device
--SystemEdit-PCI>>p2
```

```
0. PCI 00:00:00 DeviceID 0x6f00
1. PCI 00:01:00 DeviceID 0x6f02
2. PCI 00:02:00 DeviceID 0x6f04
3. PCI 00:02:02 DeviceID 0x6f06
4. PCI 00:03:00 DeviceID 0x6f08
5. PCI 00:03:02 DeviceID 0x6f0a
6. PCI 00:03:03 DeviceID 0x6f0b
7. PCI 00:05:00 DeviceID 0x6f28
8. PCI 00:05:01 DeviceID 0x6f29
9. PCI 00:05:02 DeviceID 0x6f2a
10. PCI 00:05:04 DeviceID 0x6f2c
11. PCI 00:14:00 DeviceID 0x8c31
12. PCI 00:1c:00 DeviceID 0x8c10
13. PCI 00:1c:02 DeviceID 0x8c14
14. PCI 00:1d:00 DeviceID 0x8c26
15. PCI 00:1f:00 DeviceID 0x8c54
16. PCI 00:1f:02 DeviceID 0x8c02
17. PCI 00:1f:03 DeviceID 0x8c22
18. PCI 02:00:00 DeviceID 0x6f50
19. PCI 02:00:01 DeviceID 0x6f51
20. PCI 02:00:02 DeviceID 0x6f52
21. PCI 02:00:03 DeviceID 0x6f53
22. PCI 03:00:00 DeviceID 0x15ab
23. PCI 03:00:01 DeviceID 0x15ab
24. PCI 05:00:00 DeviceID 0x1537
25. PCI 06:00:00 DeviceID 0x1533
26. PCI 08:00:00 DeviceID 0x0750
27. PCI 00:00:00 DeviceID No Device
28. PCI 00:00:00 DeviceID No Device
29. PCI 00:00:00 DeviceID No Device
--SystemEdit-PCI>>
```

```

    << KONTRON SYSTEM PBIT : EDIT MODE >>

Edit by feature, choose:PCI CPLD SATA ETH USB SMBUS BIOS
Other available cmds: `s` for settings, `?` for help, `q` to quit edit mode

--SystemEdit>>CPLD

--SystemEdit-CPLD>>p

0. CPLD Version           0x43
1. CPLD Dram Size         8GB
2. CPLD SysCon            System Controller
3. CPLD Geo Address       1
4. CPLD XMC Presence      XMC not present
--SystemEdit-CPLD>>

    << KONTRON SYSTEM PBIT : EDIT MODE >>

Edit by feature, choose:PCI CPLD SATA ETH USB SMBUS BIOS
Other available cmds: `s` for settings, `?` for help, `q` to quit edit mode

--SystemEdit>>BIOS

--SystemEdit-BIOS>>p

0. BIOS Checksum          132459
1. BIOS Version            ID16293EF
--SystemEdit-BIOS>>

```

```

    << KONTRON SYSTEM PBIT : EDIT MODE >>

Edit by feature, choose:PCI CPLD SATA ETH USB SMBUS BIOS
Other available cmds: `s` for settings, `?` for help, `q` to quit edit mode

--SystemEdit>>SATA

--SystemEdit-SATA>>p

0. SATA Port 0            Not Connected (0.0GB)
1. SATA Port 1            Not Connected (0.0GB)
2. SATA Port 2            Not Connected (0.0GB)
3. SATA Port 3            Not Connected (0.0GB)
4. SATA Port 4            TS32GMTS400 (32.0GB)
5. SATA Port 5            Not Connected (0.0GB)
--SystemEdit-SATA>>

    << KONTRON SYSTEM PBIT : EDIT MODE >>

Edit by feature, choose:PCI CPLD SATA ETH USB SMBUS BIOS
Other available cmds: `s` for settings, `?` for help, `q` to quit edit mode

--SystemEdit>>

No modifications, quit without saving..

#

```

## 4 / Use Cases

The following "Use Cases" are covered by PBIT:

- ▶ EVAL (platform EVALuation)
- ▶ DEVELOP (application or system DEVELOPMENT)
- ▶ DEPLOY (system production at the factory and DEPLOYment in the field)
- ▶ MAINTAIN (specific use case to MAINTAIN the system once deployed or to help repairing it)
- ▶ MANUFACTURING (used mostly by Kontron)

### 4.1 PBIT Features and Benefits

The following describes the PBIT features and benefits according to targeted Use Cases.

#### 4.1.1 EVAL

PBIT can easily be evaluated from the BIOS, thanks to its interactive mode. Access to the PBIT operator interface and to the results can be done through the console serial line or through the HDMI screen associated with a USB mouse and keyboard.

Following the detailed sections of this manual, all the PBIT commands can be experimented.

#### 4.1.2 DEVEL

The PBIT tests list can be modified in order to try various coverage/execution time trade-offs.

The "**system**" test can be used at the end of the development, to capture a complex peripheral configuration (PCIe, USB, SATA, CPLD infos, BIOS SETUP) which becomes the validated final system configuration for the deployment.

According to the complete system test policy, various PBIT results collection methods can be selected at application design time. The Operating System access to the PBIT results gives most of the information needed by a control and monitoring application to take a decision on the system following the boot steps.

The PBIT synthetic result register featured by the board control unit (cPLD) can be used from a management unit such as Kontron CMB or from other boards linked to the same SMB bus on the backplane. From this result, the Control Unit can alter the control registers to modify the behaviour of the board under control.

#### 4.1.3 DEPLOY

PBIT software implements unrivalled features which make Kontron product deployment easier and cheaper. The PBIT code is located in the same device as the BIOS code. It is deployed along with the BIOS (see Appendix A of the BIOS User's Guide). The PBIT settings (and results) are located in their dedicated non volatile EEPROM device. To copy these settings, boot the Operating System and record the OS EEPROM content from the offset address 0x2000 up to 0x3FFF and store it in a file. Then use this file to copy back data to the new board OS EEPROM to configure.

One of the most useful features of PBIT for deployments is implemented in the "system" test. As this test scans all the possible I/O ports and devices to compare with a pre-recorded system reference, it can signal any change in the expected I/O devices availability. This can cover I/O device failures, or more subtle system alterations (for example a storage device left unconnected). The "system" test results are organized to pinpoint the faulty I/O port easily, before attempting a complete system boot and having to manage the corresponding cascading failures due to this incorrect peripheral configuration.

## 4.1.4 MAINTAIN

The following PBIT features can be very useful in the context of long term maintenance and troubleshooting.

The PBIT results record the behaviour of the last PBIT run (see section 2.5 page 17). In addition to these results, each test implements a dedicated "FAILED ONCE" flag which has to be reset separately (see section 2.6 page 18). In situations where a system is regularly re-started, possibly under the control of a higher level control device, this long term flag can be used by maintenance teams to search for the root cause of previously aborted boots.

Updating the BIOS and the PBIT code is done at the same time, using binary images containing both software, thanks to the same storage device being used by both.

PBIT extension: Thanks to its modular approach, and to the EFI execution environment. PBIT can easily be modified and expanded to match very specific application use case. Please contact [GSS-MAR-Toulon@kontron.com](mailto:GSS-MAR-Toulon@kontron.com) to open a feature request discussion with Kontron.

## 4.1.5 MANUFACTURING

Thanks to the wide range of services the PBIT software can offer, PBIT is a powerful tool used during the manufacturing process of the boards.

Specific features have been implemented for that purpose.

## Appendix A - List of Acronyms

<b>CPLD</b>	Complex Programmable Logic Device
<b>OS</b>	Operating System
<b>PBIT</b>	Power on Built In Test
<b>PCI</b>	Peripheral Component Interface
<b>PMC</b>	PCI Mezzanine Card
<b>XMC</b>	PCI Express Mezzanine Card



## About Kontron

Kontron, a global leader in embedded computing technology and trusted advisor in IoT, works closely with its customers, allowing them to focus on their core competencies by offering a complete and integrated portfolio of hardware, software and services designed to help them make the most of their applications.

With a significant percentage of employees in research and development, Kontron creates many of the standards that drive the world's embedded computing platforms; bringing to life numerous technologies and applications that touch millions of lives. The result is an accelerated time-to-market, reduced total-cost-of-ownership, product longevity and the best possible overall application with leading-edge, highest reliability embedded technology

Kontron is a listed company. Its shares are traded in the Prime Standard segment of the Frankfurt Stock Exchange and on other exchanges under the symbol "KBC".  
For more information, please visit: [www.kontron.com](http://www.kontron.com)



## CORPORATE OFFICES

### EUROPE, MIDDLE EAST & AFRICA

Lise-Meitner-Str. 3-5  
86156 Augsburg  
Germany  
Tel.: + 49 821 4086-0  
Fax: + 49 821 4086-111  
info@kontron.com

### NORTH AMERICA

14118 Stowe Drive  
Poway, CA 92064-7147  
USA  
Tel.: + 1 888 294 4558  
Fax: + 1 858 677 0898  
info@us.kontron.com

### ASIA PACIFIC

1-2F, 10 Building, No. 8 Liangshuihe 2nd Street,  
Economical & Technological Development Zone,  
Beijing, 100176, P.R. China  
Tel.: + 86 10 63751188  
Fax: + 86 10 83682438  
info@kontron.cn