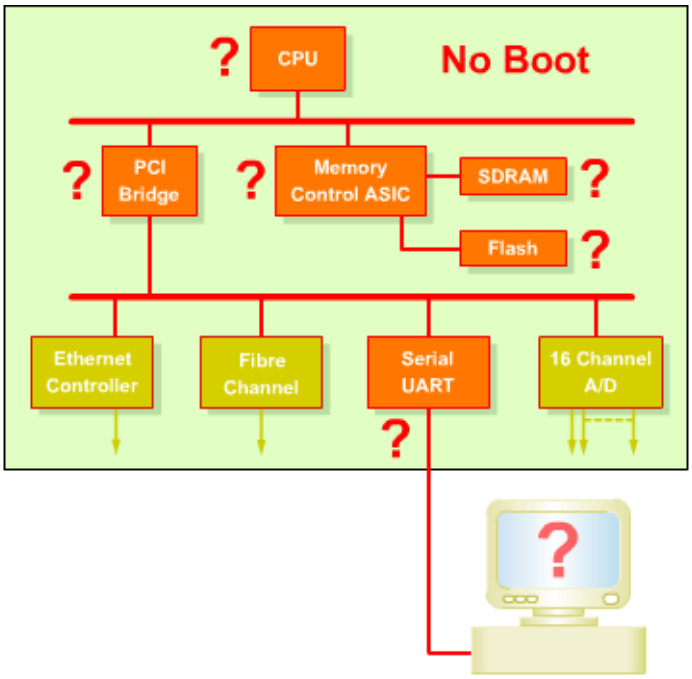
WHITEPAPER ON DEBUGGING DEAD BOARDS WITH PROCESSOR-CONTROLLED TEST (PCT)

WHAT IS A DEAD BOARD?

Consider the following block diagram.



From the user's perspective, this board is dead because no diagnostic feedback from built-in system tests is displayed on the terminal connected to the UART. The cause of the failure could be any of the buses or components shown in red: more than 50% of the board! The board may have partially booted, but no OS-level diagnostic tests can be run.

This principle applies to virtually any board, whether it is a PC motherboard, a telecoms board, an industrial or automotive controller board, and so on. Lack of any feedback equates to a dead board, the cause of which could be due to a large proportion of the board's components.

Debugging the failures on these dead boards requires an alternative approach.

NO BOARD IS DEAD TO PROCESSOR-CONTROLLED TEST!

Processor-Controlled Test uses a totally different approach from other debug solutions. Tests are performed without running either the board's boot code or its operating system. All PCT requires to debug a board is a functioning CPU, and even that can be verified by PCT. Components and buses are sequentially tested, working progressively "down" from the processor. This "top-down" test approach is

very fast because there is no boot cycle. PCT can reach any failing device or bus, and can provide diagnostic information on that failure.

THE PCT DEBUG SOLUTION

PCT board debug solutions consist of:

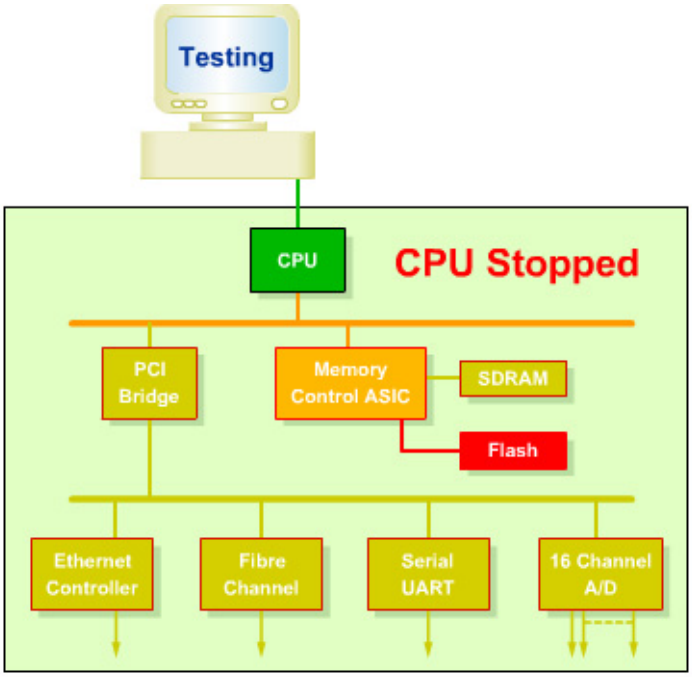
1. A processor control POD that connects to the board under test via its processor's debug port. An optional I/O Emulation Unit containing cards that measure output signals from the board's I/O ports, and generate signals to feed into the board's I/O ports. This allows PCT to compare stimulus with response to verify the correct operation of I/O devices and external connectors.

   The I/O Emulation Unit replaces the need for peripheral devices to check I/O functions, increasing test speeds and giving far greater diagnostic feedback.
2. A host PC containing the PCT test controller card (PCI), and running the PCT test software (Microsoft® Windows®-based).

HOW DOES PCT DEBUG BOARDS?

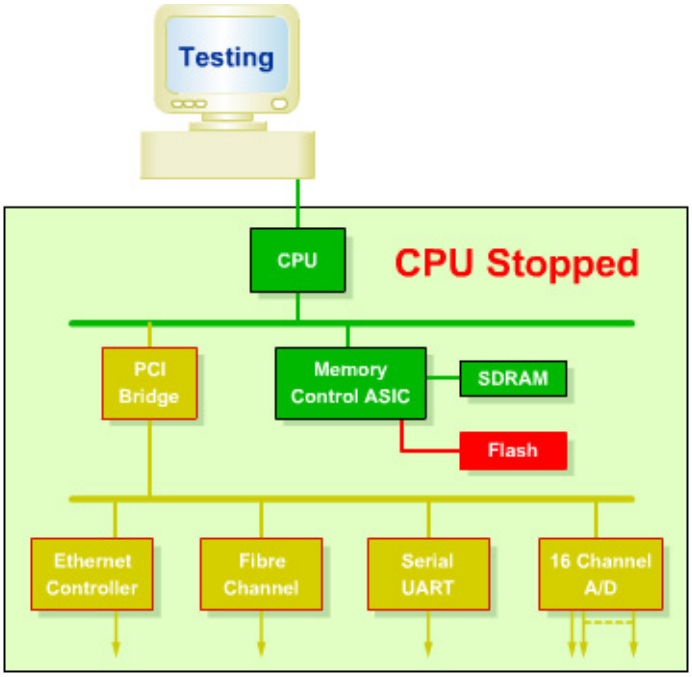
After switching on power to the board via its "Power On" connector, PCT prevents the CPU from running boot code. This control is achieved via the CPU's built-in debug port and debug command set. Most processor manufacturers now include a debug port in their processors.

PCT then instructs the processor to sequentially access each device, initializing it in the same way that the boot code or OS-level drivers would do under normal board operation. Having initialized the devices, where necessary, PCT then exercises them to verify all their functions. A failure in this sequence identifies the cause of the dead board.



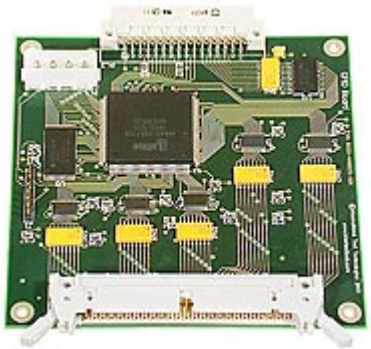
In the above diagram, PCT has already performed the following actions:

1. Powered on the board.
2. Issued a “Stop” command via the CPU’s debug port to prevent it from running boot code.
3. Checked CPU ID code (correct CPU present)
4. Checked CPU pins for correct levels.
5. Run a “Boot ROM Bus” test, reading data from the Boot ROM flash memory to verify its operation and also the operation of all buses between the CPU and the Boot ROM. The test has diagnosed that specific Boot ROM bus lines are stuck high (shown in red). This failure does not prevent PCT from testing the rest of the board.

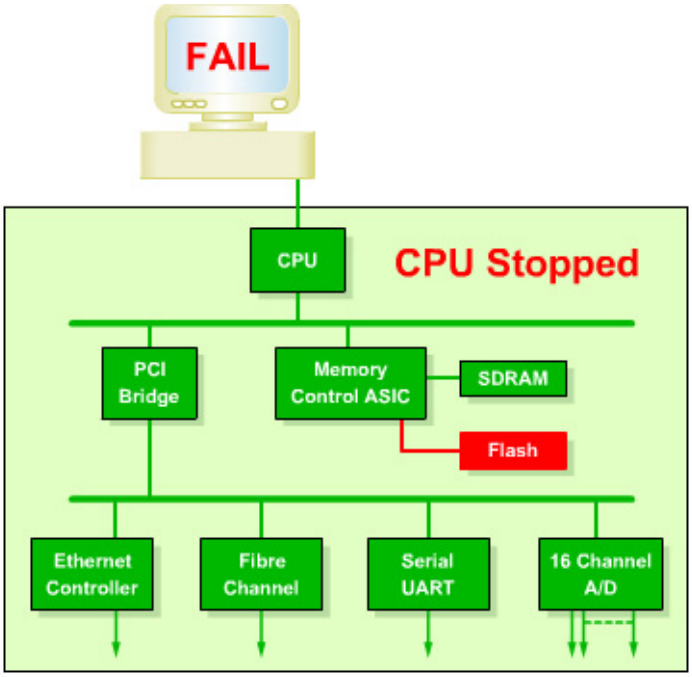


1. Testing continues by initializing the Memory Controller registers for normal operation.
2. Data is written to the SDRAM, which is then read back to verify it. This test will isolate any faults in the SDRAM, the Memory Bus and the back-end of the memory controller. No faults were found in the diagram above (components shown in green).
3. PCI Bridge registers are initialized for normal operation.
4. I/O device registers are initialized for normal operation and data is transferred in / out of the devices. The I/O stimulus generated by PCT is checked against the resulting input or output to verify that the complete I/O path is functioning correctly, including external connectors.

   The optional I/O Emulation Unit contains a wide range of I/O cards for testing standard I/O ports and also programmable cards for non-standard analogue and digital ports.



5. PCT continues to sequentially initialize and test every addressable device on the board that it can access. In the diagram below, the failing Boot ROM bus has not prevented PCT from testing all other buses and devices on the board.



GUIDED FAULT ISOLATION

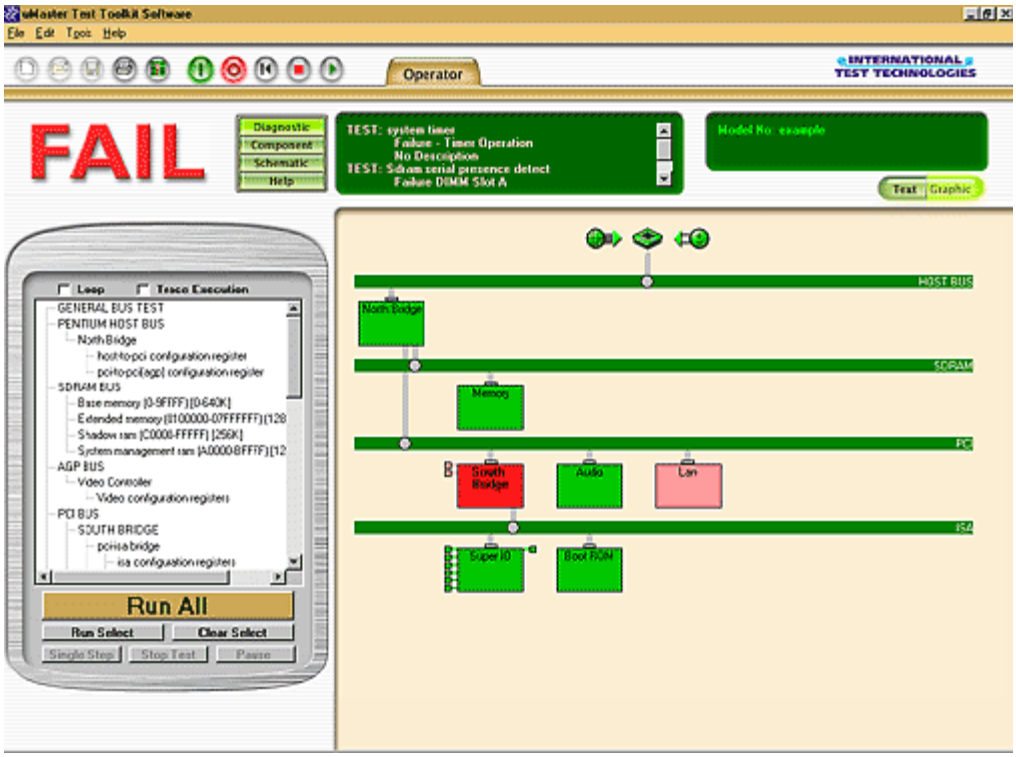
PCT's built-in Guided Fault Isolation interface greatly simplifies the diagnosis of board failures and also gives guidance for repair procedures. The screen dump on the following page shows the PCT interface in Operator Mode. This mode is intended for "one-click" operation in debug environments. The operator simply runs a pre-written test script for the board under test, and the results are displayed on a block diagram of the board.

Components and buses that are found to be functioning correctly are displayed in green. The primary cause of board failure is displayed in red and secondary failures are displayed in pink.

Clicking on any of the failing blocks displays the cause of the failure in the Results window at the top of the screen. Results indicate specific device failures and any address or data bus lines that are stuck high, stuck low, or are open circuit.

The Results window also provides further guidance, including the names of the components that may be related to the failure, a sequence of actions to take in making the repair, and clickable hyperlinks to manuals and board schematics files.

For more detailed diagnoses, the full results of all performed tests can be displayed in text mode.



Board failures can be automatically logged to a database. PCT includes tools to analyze the contents of this database, including the ability to display yields and also the most common failures for a specific board type.

PCT is the ultimate solution for dead board debug.