|  |  |  |  |
| --- | --- | --- | --- |
| **Category:**  Development platform  **Name:**  O3.1.1  **Components:**  All components(if changed)  **Stage:**  Design(unless changed),else Architecture | **Description:**  Run on standard computers | **Flexibility:**  None  **Changeability:**  A change of environment from standard computers to different computer types(not likely) | **Impact:**  If a change of environment from standard computers to a different type occurs then the whole architecture would likely have to be rewritten. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Category:**  Functional Features  **Name:**  P1.1.1  **Components:**  One component  **Stage:**  Architecture | **Description:**  Simulating various inputs types | **Flexibility:**  None  **Changeability:**  None | **Impact:**  None |

|  |  |  |  |
| --- | --- | --- | --- |
| **Category:**  Functional Features  **Name:**  P1.1.2  **Components:**  One component  **Stage:**  Architecture | **Description:**  identify output data and compare it to expected output | **Flexibility:**  None  **Changeability:**  None | **Impact:**  None |

|  |  |  |  |
| --- | --- | --- | --- |
| **Category:**  Functional Features  **Name:**  P1.1.3  **Components:**  One component  **Stage:**  Architecture | **Description:**  Emulate various different hardware applications | **Flexibility:**  None  **Changeability:**  Hardware emulation could be removed(not likely) | **Impact:**  Components handling the emulation of hardware would have to be removed. Limited change to the rest of the system(if any) |

|  |  |  |  |
| --- | --- | --- | --- |
| **Category:**  Functional Features  **Name:**  P1.1.4  **Components:**  One component  **Stage:**  Architecture | **Description:**  Emulate various different software applications, platforms and protocols. | **Flexibility:**  None  **Changeability:**  Software emulation could be removed(not likely) | **Impact:**  Components handling the emulation of software would have to be removed. Limited change to the rest of the system(if any) |

|  |  |  |  |
| --- | --- | --- | --- |
| **Category:**  Reliability  **Name:**  P4.2.1  **Components:**  Two components  **Stage:**  Implementation | **Description:**  Input and output components should gracefully recover if the tested system crashed | **Flexibility:**  None  **Changeability:**  None | **Impact:**  None |

|  |  |  |  |
| --- | --- | --- | --- |
| **Category:**  Reliability  **Name:**  P4.2.2  **Components:**  One component  **Stage:**  Architecture | **Description:**  Restarting the tested system from a certain point after a system crash | **Flexibility:**  Yes, latency for restarting the test could be increased to allow different kinds of recovery  **Changeability:**  No need for restarting from a certain point(not likely), No need for restart at all(not likely) | **Impact:**  Changes to this factor would only affect the component that is responsible for restarting the tested system |

|  |  |  |  |
| --- | --- | --- | --- |
| **Category:**  Availability  **Name:**  P4.1.1  **Components:**  Several components  **Stage:**  Architecture | **Description:**  Adding new input mechanisms | **Flexibility:**  None  **Changeability:**  None | **Impact:**  None |

|  |  |  |  |
| --- | --- | --- | --- |
| **Category:**  Availability  **Name:**  P4.1.2  **Components:**  One component  **Stage:**  Architecture | **Description:**  Adding new hardware emulation mechanisms | **Flexibility:**  None  **Changeability:**  None | **Impact:**  None |

|  |  |  |  |
| --- | --- | --- | --- |
| **Category:**  Availability  **Name:**  P4.1.3  **Components:**  One component  **Stage:**  Architecture | **Description:**  Adding new software emulation mechanisms | **Flexibility:**  None  **Changeability:**  None | **Impact:**  None |

|  |  |  |  |
| --- | --- | --- | --- |
| **Category:**  Availability  **Name:**  P4.1.4  **Components:**  One component  **Stage:**  Architecture | **Description:**  Adding new testing techniques | **Flexibility:**  None  **Changeability:**  None | **Impact:**  None |

|  |  |  |  |
| --- | --- | --- | --- |
| **Category:**  Schedule vs Functionality  **Name:**  O1.2.1  **Components:**  All components  **Stage:**  Architecture | **Description:**  Keeping cost and time to implement as low as possible | **Flexibility:**  The balance between cost and time to implement could be altered to better fit the team, making one more important than the other  **Changeability:**  Increase in budget or schedule | **Impact:**  A higher budget or a longer schedule would enable the implementation of better software/hardware |

|  |  |  |  |
| --- | --- | --- | --- |
| **Category:**  Acquisition performance  **Name:**  P3.1.1  **Components:**  One component  **Stage:**  Architecture | **Description:**  Handling large throughput of data | **Flexibility:**  None  **Changeability:**  The required throughput could be made even higher then at this moment | **Impact:**  Worst case the hardware would have to be updated. Software changes would be kept withing a minimal amount of components |

|  |  |  |  |
| --- | --- | --- | --- |
| **Category:**  Functional Features  **Name:**  P1.1.5  **Components:**  One component  **Stage:**  Architecture | **Description:**  Logging all the test data | **Flexibility:**  None  **Changeability:**  None | **Impact:**  None |