Test Report for: High-Security Linux FPGA Embedded Computer

Single Board Computer act as webserver and REST API(client)

200Mhz ARM9 with MMU and 32 MB SDRAM

Altera 2C8 Cyclone II FPGA

Reference Computer science by Allen B. Tucker

www.techeidot.com

Fast boot to Linux prompt-shell in 1.69 seconds

Software Testing principle and methods

- When a program is implemented to provide a concrete representation of an algorithm, the developers of this program are naturally concerned with the correctness and performance of the implementation.Software engineers must ensure that their software systems achieve an appropriate level of quality. Software verification is the process of ensuring that a program meets its intended specification
- The IEEE standard defines a failure as the external, incorrect behavior of a program [IEEE, 1996].Traditionally, the anomalous behavior of a program is observed when incorrect output is produced or a runtime failure occurs
- The competent programmer hypothesis assumes that competent programmers create programs that compile and very nearly meet their specification
- A software system is considered to be robust if it can handle inappropriate inputs in a graceful fashion. Robustness testing is a type of software testing that attempts to ensure that a software system performs in an acceptable fashion when it is provided with anomalous input or placed in an inappropriate execution environment. Robustness testing is directly related to the process of hardware and software fault injection
- Regression test prioritization approaches assist with regression testing in a fashion that is distinctly different from test selection methods. Test case prioritization techniques allow testers to order the execution of a regression test suite in an attempt to increase the probability that the suite might detect a fault at early testing stages
- Separation of concerns allows us to deal with different aspects of a problem to dominate its complexity, so that we can concentrate on each aspect individually. Separation of concerns is a commonsense practice that we try to follow in our everyday lives to overcome the difficulties we encounter. The principle shouldalso be applied to software development, to master its inherent complexity.
- Even simple software applications have complicated and ever-changing operating environments that increase the number of interfaces and the interface interactions that must be tested. Device drivers, op-erating systems, and databases are all aspects of a software systems environment that are often ignored during testing

- Over the past 50 years, computer systems have increased rapidly in terms of both size and complexity. As a result, it is both naive and dangerous to expect a development team to undertake a project without stating clearly and precisely what is required of the system. This is done as part of the requirements specification phase of the software life cycle, the aim of which is to describe what the system is to do, rather than how it will do it
- Anticipation of change is perhaps the one principle that distinguishes software the most from other types of industrial productions. In fact, software undergoes changes constantly, and anticipation of change is a principle that we can use to achieve evolvability

I. Description of the device

The small embedded computer is a multipurpose board designed specifically for customers needing extreme design security, flexibility, and reliability in applications such as gaming machines, building security equipment, or critical network infrastructure services such as network gateways or firewalls. A 8256 LUT Cyclone II FPGA is included on the board. The Embedded FPGA Linux system is reconfigurable on-the-fly by the 200MHz ARM9 CPU running Debian Linux when an additional real-time soft-coprocessor(s), DSP, or specific additional peripheral logic is needed

II. Software test results

Test	**Modules**			
	Device drivers	**Applications**	**Libraries**	**System**
Structurally-Based Criterion				
Control Flow-Based Criterion				
Fault-Based Criterion				
Error-Based Criterion				
Database-Driven Testing				
Testing Graphical User Interface				
Specification testing				
Network-specific testing				
Security testing				

Analyse

1. Testing is an important technique for the improvement and measurement of a software systems quality. Any approach to testing software faces essential and accidental difficulties and, as noted by Edsger Dijkstra [1968], the construction of the needed test programs is a major intellectual effort. While software testing is not a silver bullet that can guarantee the production of high-quality applications, theoretical and empirical investigations have shown that the rigorous, consistent, and intelligent application of testing techniques can improve software quality. Software testing normally involves the stages of test case selection, test case generation, test execution, test adequacy evaluation, and regression testing. Each of these stages in our model of the software testing process plays an important role in the production of programs that meet their intended specification. The body of theoretical and practical knowledge about software testing continues to grow as research expands the applicability

of existing techniques and pro- poses new testing techniques for an everwidening range of programming languages and application domains