

Embedded Software Testing
Lecture 1
Seer Akademi –NPTEL MOU
Course Introduction

We will see the software testing; we will just read into four session slide what we started with.
(Refer Slide Time: 00:18)

Prerequisites for embedded systems testing

Embedded systems basics :

Prerequisites of embedded systems : Microcontroller target, C language, IDE, debugging, Devices' understanding

We said that the prerequisites required for the embedded system testing are the basic embedded testing knowledge inference development environment debugging and interface divide this etc, also we had look into the marketing drivers.
(Refer Slide Time: 00:41)

Marketing drivers

- Embedded Software systems are one of the prime growing technology areas.
- Embedded Software Testing knowledge is an essential industry need across different verticals.
- Embedded Software Testing accounts more than half the cost of developing the embedded Software.
- At times, the embedded software developers spend more time on testing
- Reliability and predictability requirements are steadily increasing.
- The best way to understand a complex embedded software system is to test it.



For the embedded software one of the keys marketing drivers are embrace of their useful renewable and predictable and testing accounts for more than how we work and the cost of wave development and the legislative understand the complex some of the key drivers.

(Refer Slide Time: 01:14)

Course objectives

- This course covers basic fundamentals of Embedded software testing and life cycle
- Covers about dynamic, model based and coverage testing.
- To study about test management and testing from use cases.
-

Also we had a course objectives embedded software testing and string it will coverage testing is that basic management.

(Refer Slide Time: 01:33)

JOB PROFICIENCY

- Foundation on embedded software testing.
- Perspective of embedded software products from tester point of view.
- Validation and verification.
- Test case design, verification strategy, effective test management
- Testing process.

And job proficiency foundation of the member of the testing disturbed for the embedded various strategy effect this management.

(Refer Slide Time: 01:53)

Learning Outcomes

- Learn about concept of embedded software testing, TEmb method and creating test harness
- Learn about Embedded software test life cycle, V-model
- Learn about different types of testing methods
- Learn static analysis and metrics
- Learn about top-down, bottom-up integration and testing from use cases
- Learn about test management and configuration management



6

Learning outcome also concepts of embedded software testing TEmb different types and practices approach bottom –up integration and testing from use cases learn about test management and configuration.

(Refer Slide Time: 02:24)

Learning Outcomes

- Learn about concept of embedded software testing, TEmb method and creating test harness
- Learn about Embedded software test life cycle, V-model
- Learn about different types of testing methods
- Learn static analysis and metrics
- Learn about top-down, bottom-up integration and testing from use cases
- Learn about test management and configuration management

Then we had introduction on different session how you going to organize we will start continuation of let the special one embedded system basics fundamental about the string of testing method.

(Refer Slide Time: 02:52)

INTRODUCTION contd.

- Session 9 discuss on Code reviews, review guidelines and sample tools.
- Session 10 on Test Metrics and tools usage.
- Session 11 introduces Software integration aspects, to begin with bottom up and topdown approach
- Session 12 discuss on Software integration : Use case based testing, Regression testing
- Session 13 discuss on Test Management, test matrices, defect matrix
- Session 14 and 15 on Test configuration management, control and change. Also learn about testing process.

Embedded software and that is used development of this course.
(Refer Slide Time: 03:11)

Session 1 contd. : Embedded Systems Basics

Okay, so we will continue the session note the written was then.
(Refer Slide Time: 03:28)

Embedded System Definitions

- Similar to the standard 'C' Programming
- Some restrictions because the target hardware has Code/Data memory restrictions and timing restrictions.
- Embedded programming has some special features dedicated for each target hardware (micro-controller / micro-processor) used complaints - (Boris Beizer)
- **Error:** the "mistake" (human, process or machine) that introduces a fault into software

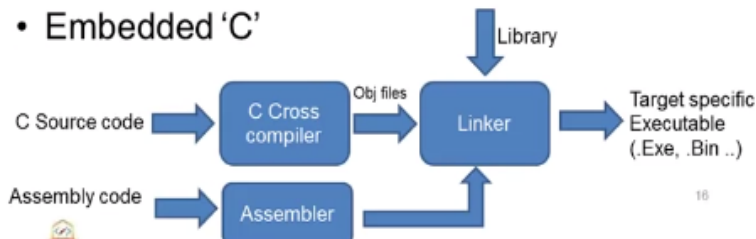


That is software testing.
(Refer Slide Time: 03:28)

14

Embedded 'C'

- Similar to the standard 'C' Programming
- Some restrictions because the target hardware has Code/Data memory restrictions and timing restrictions.
- Embedded programming has some special features dedicated for



16

What is testing? It is high interpret definition had given here this is applicable of any of the testing especially embedded and non embedded of course embedded which play disguising existing or reevaluating different component by manual or automatically to verify that if satisfies this specific requirement ,so what is main thing testing and process it has many things we had many things.

So those are all so what are causes we are explain and evaluate or exist component or different system by different names could be automatically and manually we shall verify to make sure that we satisfy the requirements manual and automated test we will explain later when we take up the

different test methods that mean the hydraulically definition there are other definition fort angle referred software testing.

Software testing is an part of software development life cycle, here software development life cycle includes developed as well as which as planning design, implementation testing so that I what development of life cycle so integral part testing is a causes of including a probability of finding an error, a good test case is one that has a high probability of finding an error. A successful test care is one that detects an as-yet undiscovered error that case aim is too any and disorder for that particular function we see the some of the definition role of testing okay.

(Refer Slide Time: 05:59)

Role of Testing

- V & V
Verification: Are we doing things right?
Validation: Are we doing the right things?
- Determine whether system meets the specification
- Impart confidence of the system under test
- Provide insight into the software development process



19

Role of testing here testing has two primary aspects one is verification and other one is validations typically we called an industrial verification Are we doing things right? Validation is we doing right things? The basic verification the things like secondary roles of testing determine whether system meets the specification behavior of the system is being implicate a general system that is the random system I the specification what we generated then impart confidence of the system under test.

When in a components of the system of there working a expectation can be employed on to the foe the user of it can be effect to failed or it can be delivered the testing impart from the confidence of the system under test, it also provide insight into the software development process maintaining the testing.

We will amuse the some of the embrace how is been development I mean the various analysis test failures or test part will uncover some of the issues of the problems that would have come

across using the software development process would be a planning period ,implementation reduce etc okay.
(Refer Slide Time: 08:17)

Why Testing

- Developers' oversight
- Certain bugs easier to find in testing
- Post release debugging is expensive
- Bugs in development tools such as compilers, library
- Best way to understand and present the product is to test it and report as PASS !
- Avoid customers detecting the defects
- Testing is challenging and rewarding



20

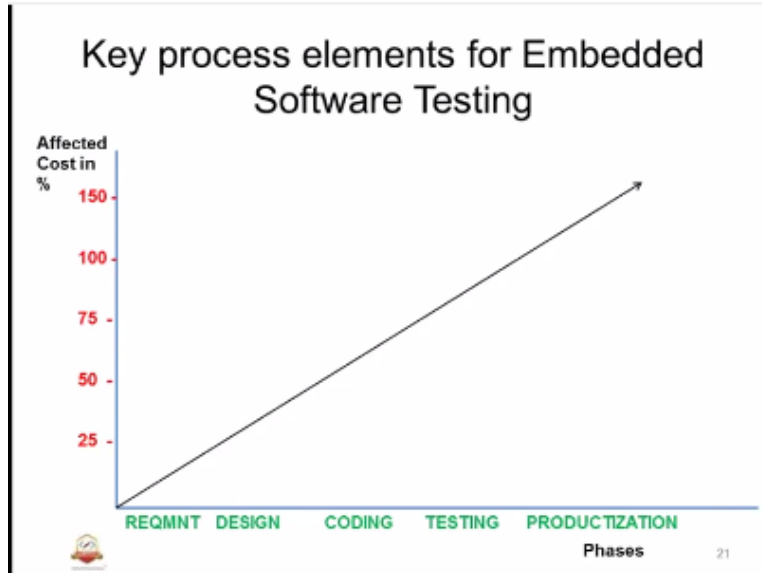
So, why we need to testing? The equation is proper planning design implementation is done for an embedded system why the question of testing of course testing is randomly I understood why practice why this there are developers could be oversight of the implementation of design or common the testing will uncover the easy to find the interesting which we need not be amount dealing a implementation, implementation may not be enough till find out the problem so certain bugs easier to find out during testing.

Post release debugging in expensive we cannot deliver the implementation data base these would be definitely specified this cannot there are change development terms such as complier, library or many of the implementation or coding phrase that we use could have some issues so that is to be tested that will come out of the issue while and best way to understand and the product is to test it and report as pass! Report testing.

So basically or so this is going to be test and we have afford have customers regarding problem solving so doing the very much then just and we cannot afford to have customers requiring the deference problems are the for all provide customers detecting the defects testing is challenging and rewarding will equally has good as testing implementations so In the understand and present the product.

By good defects and can be an implement rewarding so can be improved so that is why we needed process.

(Refer Slide Time: 11:11)

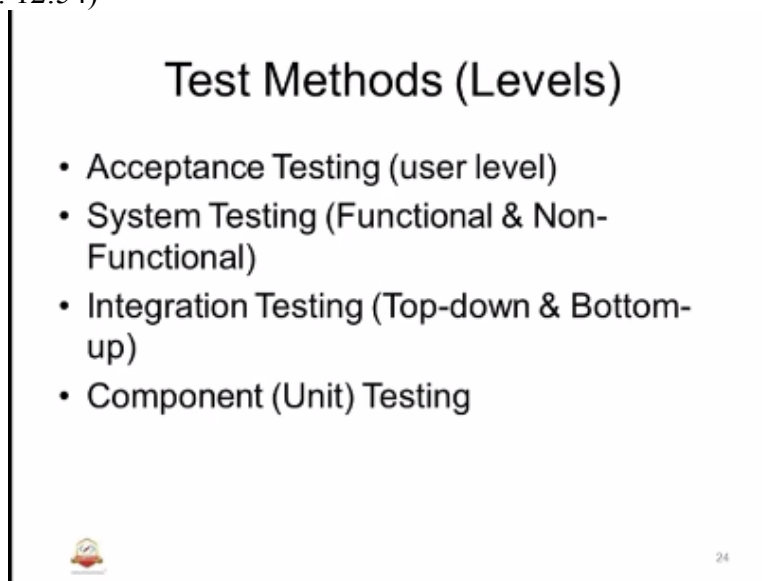


The form in that is, that is used in and what is the cost in so cost of embedded software defects and then we can see the cost in % and a axis and horizontal this different phases for software defects and the requirements have to design coding testing production we can massion complete I mean that is why so that was affected cost in requirement so we find a problem they requirement before we actually avail for the cost. According the design will have a collect the less in column compare the coding we can do this so as the phases those by cost of the defect will be higher. (Refer Slide Time: 12:23)

- ### Key process elements for Embedded Software Testing
- Plan Documents
 - Development, Verification and configuration plans
 - Technical Inputs
 - SRD, SRS, SDD, Code, TC & TP, Scripts, Results
 - Guidelines and standards
 - Software testing standards and guidelines, review guidelines

Some of the key process elements for embedded software testing we have a input and the elements we required we know have to do a software testing basically will be the upon the process, so have a plan documents for development so documents are development verification

and configuration plans we just talk about what is the plan documents we have to verification plan how about the configuration plan.
(Refer Slide Time: 12:54)



Test Methods (Levels)

- Acceptance Testing (user level)
- System Testing (Functional & Non-Functional)
- Integration Testing (Top-down & Bottom-up)
- Component (Unit) Testing

24

So this are the normal time going to use so all these there are the technical inputs one of the very important process software testing and the embedded software testing that has been then next is the technical inputs this is requirement are demand and requirement specification and guidelines why had key process is complete main testing we had everything going good code and hen how you going to report as a report using the process.

So the technical inputs we say that the product is sorted I think we use software testing and standards these are all of course that is are all inputs for embedded testing they are all process standard and guidelines so what are the specific standard you have for example if you take a standard you have been fallowed for space we fallowed srd2671 for example then we have a which one remind a order for standard.

So review and is a, or I mean a, using have been how the different phases on this testing we used so these are one of the important element of the embedded guidelines will be used for the process testing.

(Refer Slide Time: 17:15)

Test Process

- Testing Process Definition
 - Test Planning
 - Test Specification (TC, TP, TEnv.)
 - Test Execution (Manual, Automated)
 - Test Coverage and reporting (Test Log and evidencing)



25

We used a touch up on the test process, and they are test plan and the specification and test execution and test coverage and reporting so this is the best definition of the I mean the process will define first various specifications will be we define a definition of a testing process for element those these element the test planning whatever it is we come back from the embedded software is used and coverage.

And we have what should of a test cases are I am going to tell how test cases will process and what will be execution and normal, so what is the only coded cost is this process this is the specification and the specification we will have different arguments already in this specifications and this process the next process will be the specifications so what are the matters that going to be used in software testing execution the manual it could be automated based that the execution prospective.

So how I going to executed we will recollect the all these process final process, definition are user coding I want to be done this definition so and done of the interns of process and function of requirements are functional I mean every plan and they require allocations that is assembly is to hide a coverage those executions are important So input in used to the we have used that false and a execution outputs this is for embedded systems,
(Refer Slide Time: 17:48)

Test Methods (Levels)

- Testing Process Definition
- System Testing (Functional & Non-Functional)
- Integration Testing (Top-down & Bottom-up)
- Component (Unit) Testing



24

Coming to test methods the complete a embedded software testing are into different levels and different methods so they are acceptance testing it is a user level this a required for a user from functional the requirement basically there are top down and up so that will be user testing there will be also called always functional and non functional system testing the functional and non functional means.

The functional, functional means the functionalities particular product we issued on the functionality software non functional aspects something like how much memory we using the form on this speed some of the background the foundation and there we used so under functional for here so we using similar, and the small the head I will explain a integrated testing top down this mission.

I think only we which defining the some of the key elements of the component. So next one is a integration testing are I explain the when the three of term and all these integration code the integration of different software module these are the definition of the software module onwards and that is hardware and whether are a, various of this we did generally solve the problem as we as top down and bottom up so the last one is the component unit testing this be so, we using in the in all testing.

That means any in the software there are a different of the functionality there could be implemented those remains for a will be under focus during this component the component could be have a one function of the non functional so all the functions, all the remains we called it as testing all we showing the tested we showed only for exit level for method in the used in the embedded software testing.

(Refer Slide Time: 21:16)

Test Methods (Levels) contd.

- System Testing
 - TBD...



26

(Refer Slide Time: 21:20)

Test Methods (Levels) contd.

- Integration Testing
 - TBD...



26

(Refer Slide Time: 21:22)

Test Methods (Levels) contd.

- Component Testing
 - TBD...



And during the next section here will about the acceptance testing and system testing integration testing component testing.