

## **CSPC61, EMBEDDED SYSTEMS AND ARCHITECTURE**

### **CHAPTER-12: The Final Phases of Embedded Design: Implementation and Testing**

1. What is the difference between a host and a target?

A development environment is typically made up of a target (the embedded system being designed) and a host (a PC, Sparc Station, or some other computer system where the code is actually developed).

2. What high-level categories do development tools typically fall under?

The key development tools are:

Utility tool : Example- editors (for writing source code), VCS (Version Control Software) that manages software files, ROM burners that allow software to be put onto ROMs

Translation tool : Convert code (that a developer intends for the target) into a form the target can execute

Debugging tools : Can be used to track down and correct bugs in the system

3. [T/F] An IDE is used on the target to interface with the host system.

False. Source code is typically written with a tool such as a standard ASCII text editor, or an Integrated Development Environment (IDE) located on the host platform.

4. What is CAD?

It is a software commonly used by hardware engineers to simulate circuits at the electrical level in order to study a circuit's behavior under various conditions before they actually build the circuit.

5. In addition to CAD, what other techniques are used to design complex circuits?

SPICE(Simulation Program with Integrated Circuit Emphasis), PSpice

6. A) What is a preprocessor?

Preprocessing is an optional step that occurs either before the translation or interpretation of source code. Its functionality is commonly implemented by a preprocessor. The preprocessor's role is to organize and restructure the source code to make translation or interpretation of this code easier.

B) Provide a real-world example of how a preprocessor is used in relation to a programming language.

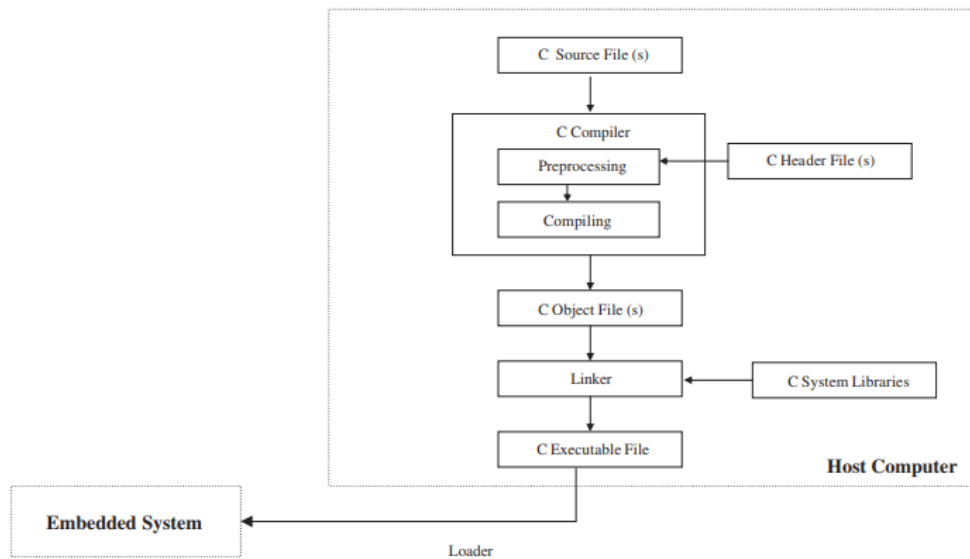


Figure 12-5: C example compilation/linking steps and object file results

7. [T/F] A compiler can reside on a host or a target, depending on the language.  
True. Most compilers are located on the programmer's host machine. Some languages, such as Java and scripting languages, have compilers or interpreters located on the target.
8. What are some features that differentiate compiling needs in embedded systems versus in other types of computer systems?
  - (1) embedded designs can contain several different types of physical memory; (2) they typically have a limited amount of memory compared to other types of computer systems;
  - (3) memory can typically become very fragmented and defragmentation functionality is not available out-of-the-box or too expensive; and
  - (4) certain types of embedded software may need to be executed from a particular memory location.
9. A) What is an object file?

After all the compilation on the programmer's host machine is completed, the remaining target code file is commonly referred to as an object file. It can contain anything from machine code to Java byte code, depending on the programming language used.
- B) What is the difference between a loader and a linker?

Linker integrates this object file with any other required system libraries, creating what is commonly referred to as an executable binary file. This executable is either directly transferred onto the board's memory or ready to be transferred to the target embedded system's memory by a loader.

## **CSPC61, EMBEDDED SYSTEMS AND ARCHITECTURE**

### **CHAPTER-12: The Final Phases of Embedded Design: Implementation and Testing**

10. A) What is an interpreter?

An interpreter translates source code into object code one instruction at a time. The resulting object code is then executed immediately. The process is called interpretation.

B) Name three real-world languages that require an interpreter.

Python, Javascript, PHP

11. An interpreter resides on: B) both host and target

12. A) What is debugging?

Task of locating and fixing errors within the system

B) What are the main types of debugging tools?

Hardware, Software and Manual

C) List and describe four real-world examples of each type of debugging tool.

Refer table pg 552 in book.

13. What are five of the cheapest techniques to use in debugging?

Manual Tools are Readily available, free or cheaper than other solutions, effective, simpler to use.

5 of them are: Print Statements, Dump files, Counters/Timers, Fast Display, Output ports

14. Boot code is: C) Software that starts-up the board.

When power is applied to an embedded board (because of a reset), start-up code, also referred to as boot code, bootloader, bootstrap code, or BIOS (basic input output system) depending on the architecture, in the system's ROM is loaded and executed by the master processor.

15. What is the difference between debugging and testing?

Goals of debugging are to actually fix discovered bugs. Debugging typically occurs when the developer encounters a problem in trying to complete a portion of the design, and then typically tests to-pass the bug fix.

With testing, on the other hand, bugs are discovered as a result of trying to break the system, including both testing-to-pass and testing to-fail, where weaknesses in the system are probed.

16. A) List and define the four models under which testing techniques fall.

static black box testing

static white box testing

dynamic black box testing

dynamic white box testing

## CSPC61, EMBEDDED SYSTEMS AND ARCHITECTURE

### CHAPTER-12: The Final Phases of Embedded Design: Implementation and Testing

Black box testing occurs with a tester that has no visibility into the internal workings of the system (no schematics, no source code, etc.). Black box testing is based on general product requirements documentation, as opposed to white box testing (also referred to as clear box or glass box testing) in which the tester has access to source code, schematics, and so on. Static testing is done while the system is not running, whereas dynamic testing is done when the system is running.

B) Within each of these models, what are five types of testing that can occur?  
unit/module testing (incremental testing of individual elements within the system)  
compatibility testing (testing that the element doesn't cause problems with other elements in the system)  
integration testing (incremental testing of integrated elements), system testing (testing the entire embedded system with all elements integrated)  
regression testing (rerunning previously passed tests after system modification),  
manufacturing testing (testing to ensure that manufacturing of system didn't introduce bugs)

17. [T/F] Testing-to-pass is testing to insure that system minimally works under normal circumstances.

True

18. What is the difference between testing-to-pass and testing-to-fail?  
tests-to-pass tests only to ensure the system minimally works under normal circumstances whereas testing-to-fail tries to break the system and weaknesses in the system are probed.

19. Name and describe four general areas of law under which a customer can sue for product problems.

Breach of Contract (i.e., if bug fixes stated in contract are not forthcoming in timely manner)

Breach of Warranty and Implied Warranty (i.e., delivering system without promised features)

Strict and Negligence liability for personal injury or damage to property (i.e., bug causes injury or death to user)

Malpractice (i.e., customer purchases defective product)

20. [T/F] Once the embedded system enters the manufacturing process, the design and development team's job is done.

False

The responsibilities of the engineering team last throughout the lifecycle of the device, and do not end when the embedded system has been deployed to the field.