**Task 4: Conceptual Design Review**

*Due: September 26, 2014 by 11:59 p.m.*

The Conceptual Design Review (CoDR) is a design proposal (team assignment) that should include the following components:

1. Introductory information
2. User needs
3. Use case
4. System-level requirements
5. Functional architecture
6. System-level trade studies
7. Physical architecture
8. Subsystem descriptions
9. System validation experiments
10. Work Breakdown Structure (WBS)
11. Schedule
12. Provisional parts list and budget
13. Responsibilities of each of the team members
14. Risk management

Component descriptions

1. Introductory information. This includes a title page with the team name, team members’ names, date, project title, and a table of contents. See the “CoDR format” document for details.
2. User needs. This is a one-paragraph description of user needs focused on end results, not the technology.
3. Use case. This has two integrated parts:
4. A narrative (1 page) of the system’s operation in the real world (see example attached to this assignment). It should tell a story about how an end user will employ and benefit from the system and describe a typical scenario in which the system will be used. The use case should focus on the core capabilities you expect to achieve this year, though it may include ideal system features that you don’t expect to achieve this year. You should write the use case after completing the rest of the document: it gives the final system concept, concept of operations, and operating environment whose derivation the rest of the document details and supports.
5. One or more system graphical representations (e.g., Photoshop, Google sketch, CAD, hand-drawn) in its use case/mission environment. The narrative should refer to the figure, which should depict the full system operational concept including the robot’s interaction with the environment. For example, if you are creating an Explosive Ordnance Disposal (EOD) robot, the figure should show not only the robot, but the likely bomb site, the area to be traversed to reach that site, and the user interacting with the robot.
6. System-level requirements
7. These should be divided into two categories: mandatory and desirable. Each of the mandatory and desirable categories should be divided into two further categories: functional and non-functional.
8. Use some numbering scheme for your requirements for consistency and ease of tracking their fulfillment as you progress throughout the year. For example:
   1. Mandatory functional: M.F.1, M.F.2, …
   2. Mandatory non-functional: M.N.1, M.N.2, …
   3. Desirable functional: D.F.1, D.F.2, …
   4. Desirable non-functional: D.N.1, D.N.2, …

The requirements and their numbering will carry over into the test plan you will develop later.

1. Where possible, associate a quantitative technical performance measure with a requirement.
2. Functional architecture
   1. The primary content in this section is a block diagram showing your system’s major functions and the flow (information, energy, material) between them.
   2. Don’t use generic subsystem names. The subsystems should be as specific to your system as possible. So, for example, instead of Sensing, Planning, Acting, for a planetary rover you might have Rock Detection, Path Planning, Rock Abrading, Sample Storing.
   3. If the information flow requires additional detail not possible to fit into the main functional architecture block diagram, you should include a software functional architecture block diagram.
   4. Unless your functional architecture block diagram is entirely self-explanatory, you should include some text highlighting its principal features.
3. System-level trade studies
4. The functional architecture should have identified several functional subsystems. Use some rigorous (quantitative if possible) comparison method to evaluate and choose between multiple design or purchase options to realize the function of each of these subsystems.
5. Use figures to illustrate your design concepts.
6. Physical architecture
7. The trade studies will have identified basic choices for realizing subsystem functions. This section fleshes those choices out, connects them, and adds system components that may not yet have been considered (e.g. user interface, communications).
8. The primary content in this section is a block diagram showing your system’s major physical components and the flow (information, energy, material) between them.
9. The physical elements detailed here (hardware & software) are a particular realization of your functional architecture, so the two block diagrams should be strongly parallel. In particular, there should be a strong match between the subsystems in both.
10. Unless your physical architecture block diagram is entirely self-explanatory, you should include some text highlighting its principal features.
11. Subsystem descriptions
    1. Concisely describe and, if appropriate, depict each major subsystem.
    2. Provide a fallback or alternative design for each major subsystem, referring to the trade studies if appropriate.
12. System validation experiments
13. Describe the system validation experiments that will show that your system requirements have been met. These experiments will be presented in lab demonstrations lasting no more than 30 minutes per team at the end of each semester.
14. The validation experiment will become a subset of the test plan that you will develop in later assignments. The test plan will be exhaustive; some or many of its tests will be performed away from lab demonstrations and the results summarized in ILRs or final reports. Only the most important high-level system functions need to be demonstrated in the validation experiments.
15. A validation experiment has the following essential elements:
    1. The test conditions: location, needed equipment, size and nature of operating area, etc.
    2. A list of steps your system will be put through written in a sufficiently clear way for someone with no knowledge of your project to be able to test the robot.
    3. A set of quantitative performance metrics that your system will be measured against during the validation experiment. Typically, these metrics will be written into the list of steps in the previous item.
16. A validation experiment is not a simple list of requirements that are not connected to one another. It is much more like a tightly scripted and quantitatively evaluated version of the use case (or some subset thereof) described at the beginning of the report.
17. Work Breakdown Structure (WBS)
18. The WBS has three levels. The top level is the system itself (e.g. Paint-Stripping Robot). The second level consists of the subsystems you have identified above, along with auxiliary tasks such as management, system integration, testing, etc. The third level breaks each of the subsystems/tasks in the second level into multiple subtasks.
19. Show the WBS as a hierarchical block diagram with three levels. It may be vertical or horizontal. You may show exploded views of parts of it for clarity if needed.
20. Schedule
21. Put the three levels identified in the WBS into a schedule using a Gantt Chart with no less than biweekly resolution. Schedule each of the third-level tasks from your WBS.
22. Show work package dependencies.
23. Spring scheduling may be less detailed than Fall scheduling. You will have the opportunity to refine the schedule in future assignments.
24. Briefly state in text your goals for Progress Reviews 1 (Oct. 15) and 2 (Oct. 23). Your later test plan will give details on goals for subsequent Progress Reviews.
25. Provisional parts list and budget
26. Responsibilities of each of the team members
27. Risk management. Evaluate risk from the standpoints of design, schedule, and resources (budget & personnel) and identify mitigation strategies.

Label and caption figures and refer to them in the text. Follow the formatting guidelines (CoDR format) that are attached to this assignment, which specify a 25-page limit including figures, tables, flowcharts, etc.  
  
All of the Conceptual Design Review (CoDR) documents from the past two years are attached to the assignment. The top two CoDRs in 2011-12 were from Teams B & E, and the top two in 2012-13 were from Teams A & E. Note that the 2011-12 reports are responding to a somewhat different set of content requirements. Don't treat any report as a perfect example. Make your ultimate guide the content requirements above and the explanations given in the MRSD Project and Systems Engineering courses.