Group – 7 Project 3

TIMELINE MILESTONES AND FALLBACK PLAN

Hardware Timeline:

**Milestone #1:** April 14, 2003. – Basic Design.
- The hardware team should have a working design to bring to the table.
- Consider limitations and restrictions of sensors and how to compensate for this deficiency.

*** Sync_1 ***
*** Sync_2 ***

**FALLBACK PLAN:** If the basic robot design is not agreed upon or has not been completed by this date then we will hold meetings on April 15 and will be completed by then so that the project can continue.

**Milestone #2:** April 16, 2003. – Robot construction completed.
- Robot construction is completely finished along with all of the sensor layouts and design. The robot will have been tested with the software to ensure that all of the sensors have readings and the basic motor control of the robot is functional. This does not require that the robot achieve the task of the project.

**FALLBACK PLAN:** If the robot does not work properly then all 4 of the group Members will meet Thursday and Friday of the week to have the robot working.

*** Sync_3 ***

**Milestone #3:** April 23, 2003. – Robot design is finalized.
- In this milestone the robot design is completely finalized. The co-ordination between the hardware the software is completed and the robot functions as planed. All the tests are successful with no errors or mishaps.

**FALLBACK PLAN:** If the robot does not work by this time then all the group members will work together to resolve the issue whether it is a hardware or a software issue. If it is a software problem then the hardware people will research the problem to come up with a solution and vice versa for the software people. We should do this by April 21st.

*** Sync_4 ***

**Milestone #4:** April 29, 2003. – Documentation Completed.
- All documentation on the robot hardware will be completed and compiled against the software documentation and presented in the final report to be turned in.

Software Timeline:

**Milestone #1:** April 14, 2003. - Choose development computer (Laptop?).
- First of all, we need to decide on a centralized computer to house the team's software. This is done to ensure version integrity by allowing only one person or group to work on the code at a time. For each project the senior software member should volunteer their personal laptop. If for any reason this is not possible, the junior member can offer an alternative by providing another laptop.
- Install/Check Interactive C (if necessary):
  Download Interactive C to the development computer. This can be accomplished through
  the class web page. Once installed, run simple command line testing to test basic
  functionality of the IC.

*** Sync_1 ***

Milestone #2: April 16, 2003 - Initial software development:
- This is where the software team first attempts to materialize the custom code. This code
  should perform limited movement instructions to hash out the basic range of movement
  of the robot: turning radius, speed, skid steering, etc. The team should eventually come
  up with an abstract motor library, comprised of a number of functions to allow basic
  movement of the robot. If possible functions such as turn(double degrees) and
  move(double feet) should be included.

*** Sync_2 ***

Milestone #3: April 23, 2003 - Full robot behavior, code testing:
- This stage marks the point at which the sensor code has been successfully integrated
  with the motor interface to provide limited capability to the robot. Once this has been
  accomplished, the team should enhance the code by adding error checking, exception
  handling, limited adaptability and any other features necessary to help the robot
  perform at a higher level. This stage should be considered somewhat recursive and could
  be visited multiple times.

*** Sync_4 ***

Milestone #4: April 29, 2003 - Code documentation:
- At the end of the project the code should be documented. However, this does not mean
  that the code will not have any documentation until this point. Throughout the
  development process it should be necessary to keep adequate documentation of the code
  to promote clarity between team member as well as clarity to the author.

Synchronization Definitions:

Sync_1: Initial spin the wheel test – This test will determine whether the software and the
  hardware is set up correctly. The robot design does not have to be completed but the
  communication between the software and the hardware must be successful.

Sync_2: Basic Movement Test – The basic robot design will be completed. Although the
deadline for the robot design to be finished is Mar 17, there should still be a basic prototype to
test the movement that the software designers have developed to be tested on the hardware.

Sync_3: Movement and Sensor Test – This time the robot design should be complete. All of
the sensors should be tested and functional and movement should be almost completed. All the
readings should be accounted for in the software from the sensors in the hardware, although the
final logic in the software does not have to be fully functional.

Sync_4: Test run in real world environment – This should be the completed version of the
robot. This gives us 2 days to refine any bugs that may occur, but any development of the robot
should be completed or finalized by this time.
Timeline Milestones with Fallback Plan:

<table>
<thead>
<tr>
<th>Timeline Milestone</th>
<th>Hardware team</th>
<th>Software team</th>
</tr>
</thead>
<tbody>
<tr>
<td>April 14</td>
<td>Milestone #1</td>
<td>Milestone #1</td>
</tr>
<tr>
<td>April 16</td>
<td>Milestone #2</td>
<td>Milestone #2</td>
</tr>
<tr>
<td>April 17-18</td>
<td>FALLBACK PLAN</td>
<td>FALLBACK PLAN</td>
</tr>
<tr>
<td>April 23</td>
<td>Milestone #3</td>
<td>Milestone #3</td>
</tr>
<tr>
<td>April 24</td>
<td>FALLBACK PLAN</td>
<td>FALLBACK PLAN</td>
</tr>
<tr>
<td>April 28</td>
<td>Milestone #4</td>
<td>Milestone #4</td>
</tr>
<tr>
<td>April 29</td>
<td>FALLBACK PLAN</td>
<td>FALLBACK PLAN</td>
</tr>
</tbody>
</table>