Failure Mode Analysis

Possible failures have been categorized into the main components of the project.

System Hardware (PIC Microcontroller, SRAM, Display etc.)

* Chips placed backwards
* Incorrect Pins / Wiring
* I2C bus connections
* RS232 noise
* RS232 connection problems
* Bad user input
* Mechanical failure (weak solder job, keyboard broken, display broken, etc.)

RFID Reader / Writer

* Invalid card type read
* Incorrect Pins / Wiring
* Interference from other RFID reader
* Incorrect baud rate with RS232 Connection
* Given misformatted data to write
* Card cracked / broken inside – RFID tag open circuit (ex. Broken Husky cards)

XBee Wireless Communication

* Interference from other wireless devices, especially Xbee
* Poor connection between two players – connection lost
* At a high level, the duel disk RFID system allows users to wirelessly play card games with another through XBee wireless. Cards that have RFID tags in them are placed on an RFID scanner to choose the next move. There are 3 different modes for playing: single player, multiplayer, and deck building. Players are able to see game data through a display on the device and enter their own data through the keyboard.

We will be testing the system for its ability to read multiple RFID tags and send data to another player’s device. Communication should not lag and the data received/sent should be accurate. We will also be testing the user interface for its ability to read user input and output data on a display. Again, the data received and sent between the user and the system should be accurate, and the display should consistently update as new data is received about the game.

In this test plan we cover testing of the following items:

1. RFID Reader / Writer
   1. Reading a single / multiple RFID tags
   2. Overwriting current RFID tag data with new data
2. Xbee Wireless Communication
   1. Sending data to another duel disk
   2. Receiving data from another duel disk
3. System Hardware (SRAM, PIC, etc.)
   1. Run different game modes:
   2. Reading game database
   3. Sending commands to RFID reader/writer
4. System Peripherals (keyboard and display)
   1. Accepts user input
   2. Displays data on the current game status
   3. Updates display
5. Game Mode
   1. Single Player
   2. Multiplayer
   3. Card Builder
6. Power On/Off
   1. Ability to reset system

* test spec of what is to be tested. this can be in a list and/or table form

- title and description of what to be tested (1-5 sentences)

- test environment and initial setup or conditions

- expected inputs, expected outputs

- what it is to be verified (1-5 sentences or a list of what is

to be verified)

***Example:***

1.0 Programming of Injection filter:

Verify that the injection filter is correctly programmed

into the DSP so that the radio can be setup with the

filter value.

Test Setup:

DSP is initialized and in idle state. Radio must be initialized

and on before the injection filter can be programmed.

Inputs/Outputs:

Input is filter value in hexadecimal from injection filter

command from the application software. Output is a SUCCESS response from the filter command.

Test Specification:

Verify that the injection filter is successfully programmed

into the DSP by checking the DSP injection filter register

when command is completed.