**ELEC241 Practice Referral Presentation Notes**

Personal notes to accompany the ELEC241 PowerPoint. Unsure if this needs to be submitted or not. Keep the points concise where possible (notes, not an essay).

**SECTION 1: Slide 1: Title**

* Introduction
  + Name and student number
* What I am presenting
  + Evidence of management
  + Evidence of version control
  + Answering some questions regarding VHDL and testbenches

**Slide 2: Relevant Links**

* Relevant links
* Intended to make marking easier
* **Going to talk about management**

**Slide 3: Management**

* Management
  + Kanban
    - Time management
    - Task overview
  + OneNote
    - More detailed
  + Good skills to have for future employment
* **Why these tools? They are good but of course they were suggested...**

**Slide 4: Kanban**

* Kanban
  + Time management aspect
  + See everything at a glance
    - What needs changing, where am I falling behind
      * **Mention extended referrals**
  + Task overview
  + **Example of Kanban in use on my OneNote file (we will see this later)**
* **What about a more detailed explanation of changes?**

**Slide 5: OneNote**

* OneNote
  + Where Kanban fails, OneNote excels
    - Detailed explanation. A diary for the project
    - More than just a basic look. Gives an insight into the meat of the project and how it is really going
  + More suited to big tasks
  + **Useful when they are used in conjunction**
  + **Click the link, show the file**
  + **Show the Kanban board!**
  + **Mention the GitHub at the top?**
  + **GitHub repository mentioned… I am using it for Version Control**

**Slide 6: GitHub**

* Travelling from University to home
  + Useful when I got a chest infection at home; still had my files
* I use it in all my modules
* Backups
  + No more USB drives!
* Previous versions
  + Can see where I go wrong
  + History of the project
* Updates
  + Information on what changed and why
  + Concise and easy to process information; no searching or “what does that mean?” moments

**SECTION 2: Slide 7: Tech Questions**

* In this section, my answers to a few questions on the task sheet
  + VHDL Signals vs Variables
  + Testbench Design
    - **Using my code as an example**
  + Assert Command

**Slide 8: Signals vs Variables**

* Signal
  + Type value and time
  + Defined in architecture
  + <= for assignment
  + Only updates when process stops
  + *Always exists*
* Variable
  + Type and value
  + Defined in process
    - Only reachable by the process it is in
  + := for assignments
  + Updates any time
  + *Only exists in the process block*
* A signal has a past history of values, whereas a variable only exists at that current value.

**Slide 9: Testbench Design**

* Generates inputs to the VHDL design and checks the outputs
* Useful for checking code
* Simple steps as follows
  + Generate entity and architecture
    - Names, types, lengths…
  + Port map
    - What is going where?
  + Clock and reset
  + Process block
    - The actual testbench
    - Action, then waits, action, then waits…
    - Can be improved with automation
      * **Where the assert command comes in handy**

**Slide 10: Assert Command**

* Makes the testbench component quicker
  + Action met by an error or a warning where incorrect
    - Error is the default if the severity is not stated elsewhere
    - Failure will stop the code
* Compare two values
  + C1 and C2 are the same = pass
  + C1 and C2 are different = failure
  + Assert C1 = C2 report “failure text” error (will only warn, not stop the code).