SOLUTION NOTES

Structured Hardware Design 2002 Paper 2 Question 10 (DJG)

(a) The sketch for this should use scan multiplexing for the LEDs and the switches and note that a 100x100 grid perhaps needs 300 wires owing to one row wire and two column wires for the led and a contact at each cross point. This is too many wires for a single chip, and so a partition of sorts is needed. There are various partitions. For instance, we could use six chips driving 50 lines each. Thus each chip will have fewer than 100 pins after other connections are added beyond the array wires.

Other parts of the final sketch for part c should show the other buttons and a chip containing a DAC to drive the loudspeaker and probably a microprocessor for the program code. The microprocessor is best embedded in one of the chips but use of a standard microcontroller is also viable.

(b) Owing to the complexity of the game, using a microcontroller embedded in one of the chips is a must, I think. The rules will be held in its ROM in some form. Possible the rules will be hardcoded in the controlling program which itself may be written in a bytecode, like Java or Forth and so a bytecode interpreter will also be present. I chose this method rather than generating a hardware finite state machine because speed of execution is not critical and software is easier than hardware with todays tools.

(c) Audio is stored in ROM and rendered through the DAC to the speaker. MP3 encoding or speech synthesis could be used to reduce ROM requirements. The microcontroller will need program ROM and some scratchpad RAM for the game state. It is cost-beneficial if only one chip is designed and used in several ways, even if certain functions are not used on all instances. Advanced answers will make each chip identical except for the ROM contents and allow the code and voice ROM to pooled among the chips to become a central resource serving one chip that is master, but full marks should be obtained without this.

(d) For Bluetooth we can expect a fairly high-level of protocol processing to be present on the module whereas IRDA will require more code to be added to the microcontroller. In either case, apart from the connector for the module, the thing that is needed is probably a serial port on one of the chips to connect to the module so that the processor can send and receive packets. The main modifications will then be in the software.