



COURAGE



Presented by Station#25

Keychain Virtual Pet - Digimon 1999version

Prepared for: ECE241 Final Project
Tutorial #: 0106
Station #: 25
Mentoring TA: Xander Chin
Prepared by: Shuohe James Zhang
Yi Ashley Yuan
Arfa Tehreem

11 November 2017

EXECUTIVE SUMMARY

Objective

The final project for ECE241 of Station#25 is to designed a Keychain Virtual Pet program, using FPGA to develop in Verilog.

Goal

- Redesign the cultivating-mode game program and implement it all in the FPGA hardware
- Use the video graphics adapter(VGA) as the output
- Use the controller of Playstation4 Game Console, using IP core, as the input
- Design a Finite State Machine to control the front-end interface
- Design a Finite State Machine to store the back-end database
- Design a Finite State Machine to implement an animation effect of object's movement
- Other techniques including random access memory(RAM), register, shifter, counter, arithmetic logical unit(ALU), multiplexer, and etc, all the key concepts had been taught in ECE241.

Brief Description

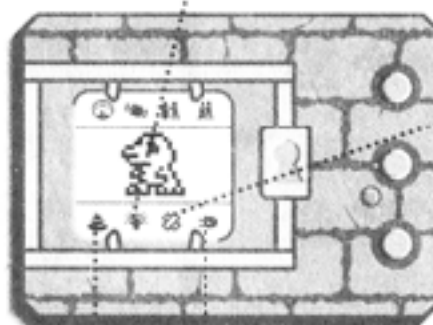
The game starts at hatching an egg. After the hatch, the baby monster will be cultivated by user, in evolution from phase to phase. User can feed the monster and check the status of growing value and age, or some other informations.

MONSTER MATCH NOTES

- You can separate the DigiMon after "OK" appears on the screens
- Each DigiMon needs to be set for monster match mode for the battle connect feature to work.
- Your DigiMon will engage in the monster match using his own individual techniques, which you trained him to master.
- When the "HIT" icon blinks onto screen, the champion and defeated DigiMon will appear.
- A skull image blinks on the screen if your DigiMon is injured during the monster match. Press Button "A" to toggle to the Medical icon, then press Button "B"
- How your DigiMon does in battle is based on how well you train him. The better the training, the better the battler!

FLUSH:

- 1) Press Button "A" repeatedly until you have highlighted the Flush icon.
- 2) Press Button "B" to Flush.
- 3) Press Button "C" to return to the Main Screen.



LIGHTS:

- 1) Press Button "A" repeatedly until you have highlighted the Lights icon.
- 2) Press Button "B" to access the Light On/Lights Off screen.
- 3) Press Button "A" to toggle between Lights On and Lights Off.
- 4) Press Button "B" to apply the selected Lights setting.
- 5) Press Button "C" to return to the Main Screen.



MEDICAL:

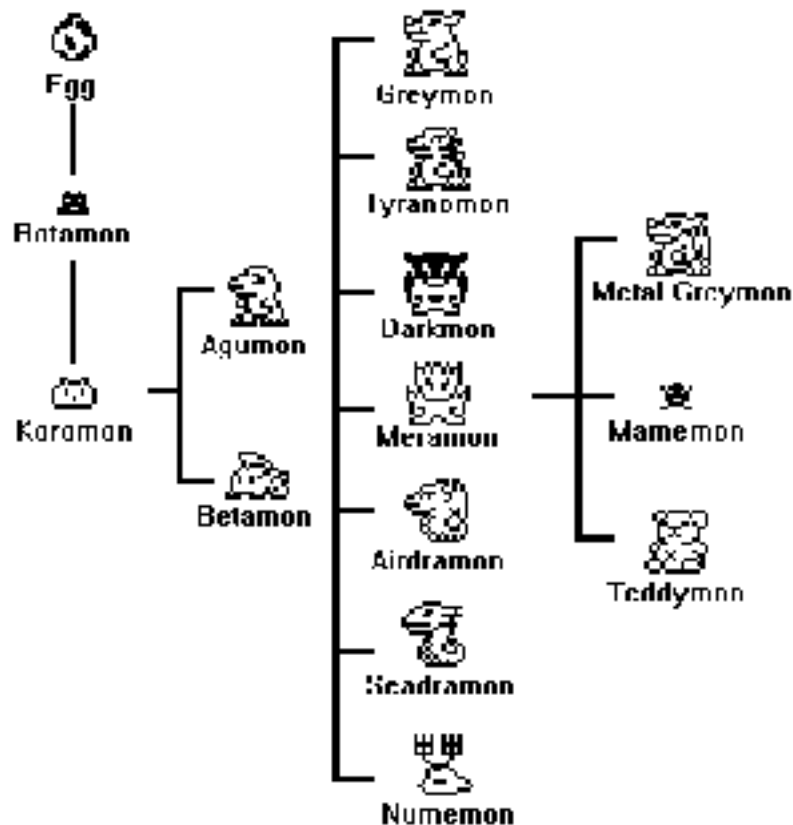
- If your DigiMon sustains injuries during a monster match, be sure to give him Medical attention.
- 1) Press Button "A" repeatedly until you have highlighted the Medical icon.
 - 2) Press Button "B" several times until the skull image disappears.



- 3) Press Button "C" to return to the Main Screen.

ALERT: The Alert icon highlights and beeps when DigiMon is hungry or requires training.





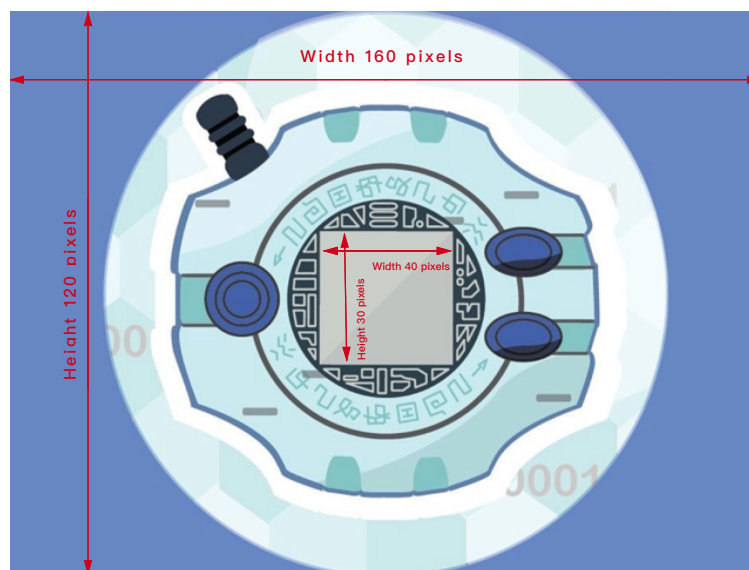
TIMELINE

Work	Object	Week
<ul style="list-style-type: none"> • Kick-off meeting with mentoring TA • Design the flow chart of functions • Work assignment and plan 	1. Product Document 2. Verilog Skeleton 3. Block Diagram	Nov 6 - 7
<ul style="list-style-type: none"> • Implement the simple animation of object's movement • Implement the basic page navigation 	1. Front-end Interface 2. 3 seconds animation	Nov 8 - 13
<ul style="list-style-type: none"> • Refine the program logic • Add audio and visual effect to the program 	1. Object's movement algorithm 2. Animation of evolution 3. Sync the sound track	Nov 14 - 20
<ul style="list-style-type: none"> • Debug • Add extra controller • Consolidate all footage for final report 	1. Design the fully functions 2. Extra input console	Nov 21 - 27
<ul style="list-style-type: none"> • Write the Final Report 	1. Paper statement handed-in 2. PDF by email	Nov 27 - Dec 4

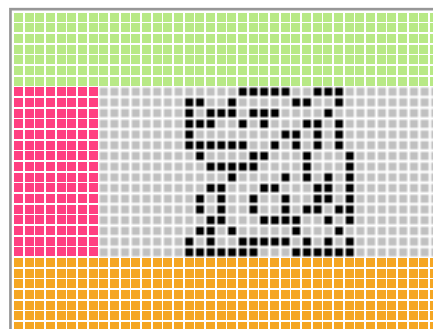
PROJECT SPECIFICATION

Front-end Interface

- We are using VGA to display the program based on 160*120(160 represents the columns and 120 represents the rows) resolution.
- And we are using the area of 40*30 in the middle to display the object, Digimon, and the surrounding area are static background, with Digivice contour and status bar.



- For the dynamic display, consisting of 40*30 pixels.
- As shown below, here are four main areas.
- The Object shown in the middle with 32*16 pixels, allowing object moving horizontally and several actions.
- The Top Bar shown in green, containing 40*7 pixels, will display the cursor and function icons.
- The Side Bar shown in pink, containing 8*16 pixels, will display other objects, other than the Digimon, such as the food feeding to the pet, and the status of the pet.
- The Bottom Bar shown in orange, containing 8*16 pixels, will display the time in Digiworld, 1 minutes in the world, 1 day in the Digiworld.



Function Specification

· We design THREE main functions for the program: 1. FEEDING, 2. TRAINING, 3. STATUS.

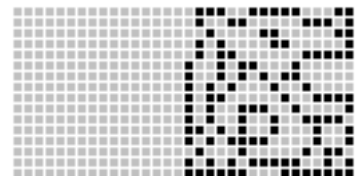
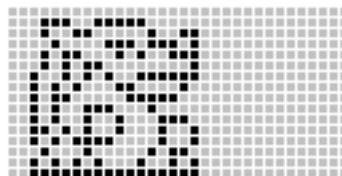
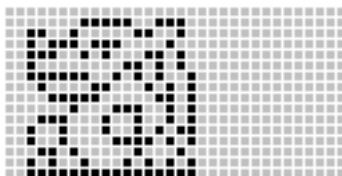
1. FEEDING

- When select the FEEDING function, the Side Bar area will show a barbecue.
- TWO animation will be displayed:
 1. Pet will open mouth and shot, showing by TWO pictures repeatedly .
 2. Meat will decrease, showing by THREE pictures in series.



2. TRAINING

- When select the TRAINING function, the Side Bar area will show a static barbell.
- And the object will move horizontally left and right, and stretch in a quicker, repeatedly to the end.
- And the barbell will be static in Side Bar.



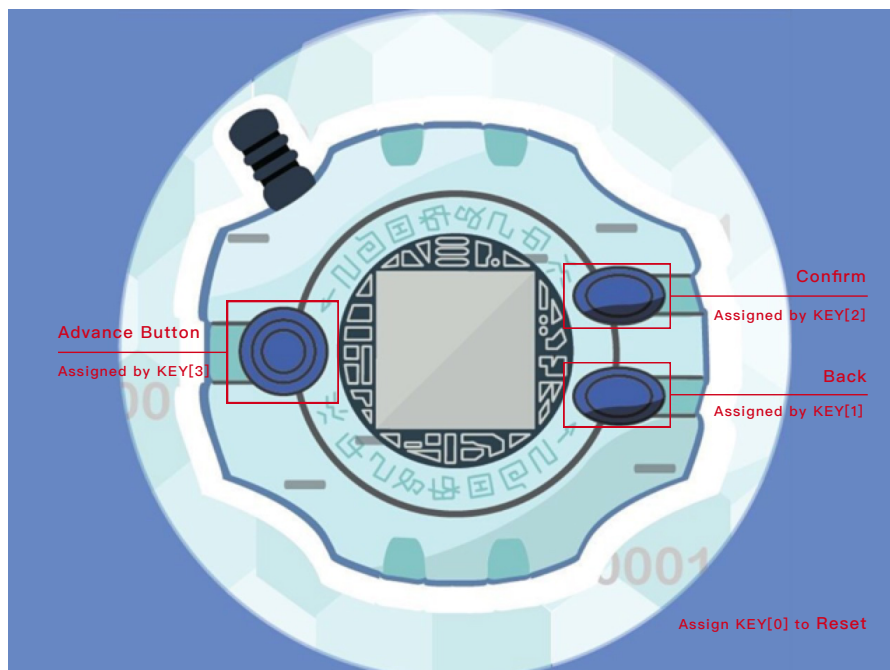
3. STATUS

- When select the STATUS function, the Side Bar area will show the HUNGRY and the ENERGY.
- HUNGRY and ENERGY both are progress bar, different from the example - ours is in vertical.



Control System

- Control system is based the Mealy Model Finite States Machine, both input triggered and status triggered.
- INPUT are three buttons:
 1. ADVANCE, moving the cursor to the next.
 2. CONFIRM, select the function.
 3. BACK, back to the default mode.
 4. RESET, reboot the program.



- Counter is another trigger:
 1. When FEEDING, HUNGRY will add by 1 grid per time, till the progress bar is full.
 2. When TRAINING, ENERGY will add by 1 grid per time, till the progress bar is full.
 3. When a day is past, the system will check the status of HUNGRY and ENERGY:
 - If both HUNGRY and ENERGY are full, then the Digimon will evolve to next stage.