Rubik’s Cube Solver

6.111 Project Checklist

Katharine Daly and Jack Hutchinson

Get Initial State Block (Jack)

Input_Configuration

- Display blank ‘opened out’ cube on screen as shown in project presentation
- Allow user to navigate between cubelet faces using FPGA arrow buttons
- Highlight the currently selected cubelet face to ensure user is aware of which face they are editing
- User can press one of the 6 colour input buttons to set the colour of a cubelet face
- Store configuration in correct format to be passed to the solve cube block
- User presses “ready” button when complete configuration has been entered
- Stretch Goals:
  - Display 3D renderings of the cube from both the front and back to make inputting the cube configuration as easy as possible for the user
  - Auto-fill some of the cubelet faces when possible

Check_and_send

- Check that the cube configuration that has been input is valid
- Warn user if invalid and return to Input_Configuration
- Send configuration word to Solve_Cube block and send ready signal

Solve Cube Block (Katharine)

Find_Cubelet Module

- Accept either an edge or corner cubelet defined by the face colors and return the current position of that cubelet
- Be able to return the position of the cube even if the desired orientation is different from the current orientation of the animation

Rotate Module

- Accept a rotation to perform on the cube configuration representation and return a modified representation
- Be able to modify the representation appropriately when the rotation is defined with respect to a different orientation from the current orientation of the animation
- Send a 5-bit number representing the desired rotation to the Show Animation block

Solution_State_Machine Module

- Track the progression of the solving process with 16 states
- States correspond to trying to place a specific cubelet in the correct location without disturbing the cubelets that have already been correctly placed
- Get the cubelet location from the Find_Cubelet module, get the appropriate rotation sequence from one of the seven modules corresponding to each solution step, and call the Rotate module to perform all these rotations
- Only trigger a rotation when given a signal from the Show Animation block
- Be able to inform the Get Input State module if the starting configuration provided by the user is solvable
- Stretch goal: Be able to recover the last few rotations if users have messed up in the recent past

Top_Face_Edge Module

- Take as input the location of a cubelet and provide the correct sequence of rotations to place this cubelet in the correct location and orientation as an edge on the top face

- Set a flag if the cubelet cannot be correctly positioned or oriented due to an incorrect starting configuration

Top_Face_Corner Module

- Take as input the location of a cubelet and provide the correct sequence of rotations to place this cubelet in the correct location and orientation as a corner on the top face
- Set a flag if the cubelet cannot be correctly positioned or oriented due to an incorrect starting configuration

Middle_Face_Edge Module

- Take as input the location of a cubelet and provide the correct sequence of rotations to place this cubelet in the correct location and orientation as an edge on the middle layer
- Set a flag if the cubelet cannot be correctly positioned or oriented due to an incorrect starting configuration

Bottom_Face_Corner_Position Module

- Take as input the location of a cubelet and provide the correct sequence of rotations to place this cubelet in the correct location as a corner on the bottom layer
- Set a flag if the cubelet cannot be correctly positioned or oriented due to an incorrect starting configuration

Bottom_Face_Corner_Orientation Module

- Take as input the location of a cubelet and provide the correct sequence of rotations to place this cubelet in the correct orientation as a corner on the bottom layer (assuming correct positioning has already been achieved)
- Set a flag if the cubelet cannot be correctly positioned or oriented due to an incorrect starting configuration

Bottom_Face_Edge_Position Module

- Take as input the location of a cubelet and provide the correct sequence of rotations to place this cubelet in the correct location as an edge on the bottom layer
- Set a flag if the cubelet cannot be correctly positioned or oriented due to an incorrect starting configuration

Bottom_Face_Edge_Orientation Module

- Take as input the location of a cubelet and provide the correct sequence of rotations to place this cubelet in the correct orientation as an edge on the bottom layer (assuming correct positioning has already been achieved)
- Set a flag if the cubelet cannot be correctly positioned or oriented due to an incorrect starting configuration

Show Animation (Jack)

- Display cube in 3D on the screen
- Animate solution on the screen, one rotation at a time.
- Send ready signal to the Solve_Cube block when the user presses the enter button
- Perform rotation animation based on current configuration and rotation type signal received from Solve_Cube block
- Stretch goals:
  - Cube can be rotated about its axes so user can better examine the cube configuration
  - User can speed up or slow down simulation
  - User presses button to allow rotation be replayed
  - Zoom in and zoom out on cube