Assignment 3: STATE MACHINE DIAGRAM

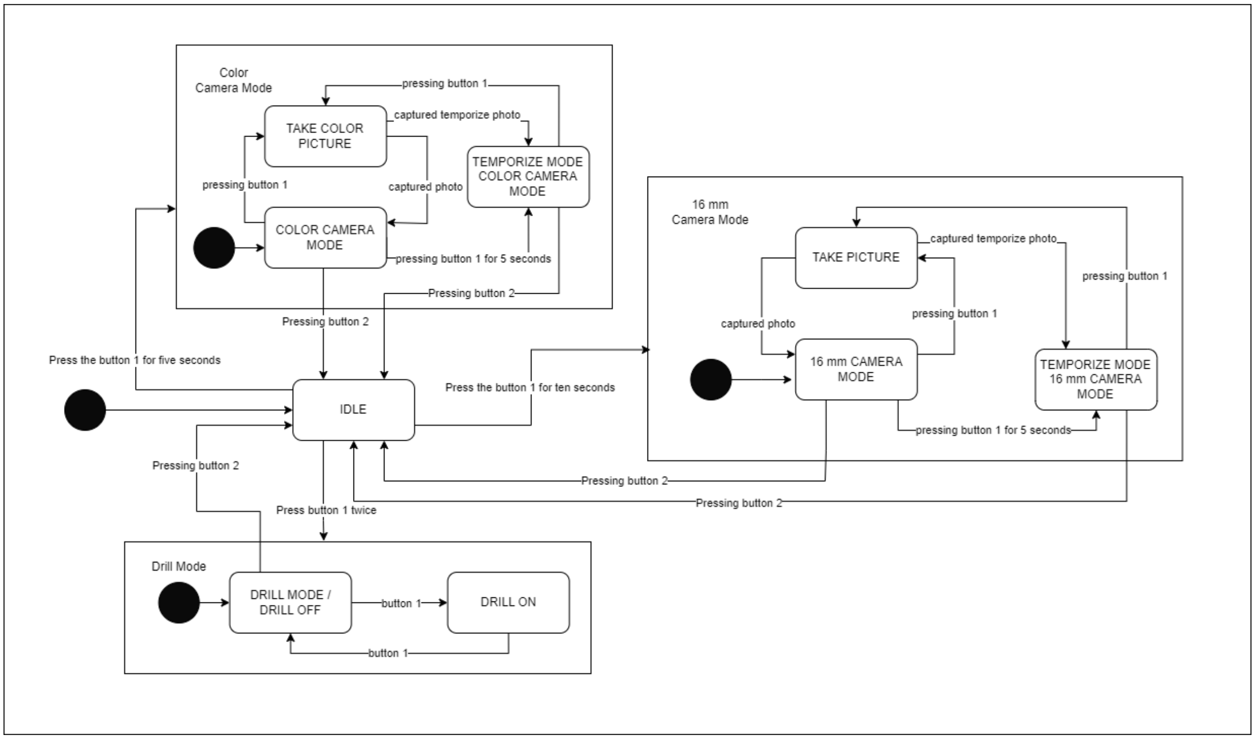
Part 2 - Camera & Drill (Bonus – 4 points):

The coolest feature of the buggy was not its speed but rather the color television camera, the 16-mm camera, and the drill. The control was packed inside a television control unit. The only problem was that the cameras and the lunar rover drill were also controlled by the same control unit. So again engineers used the two-button controller to manipulate all devices:

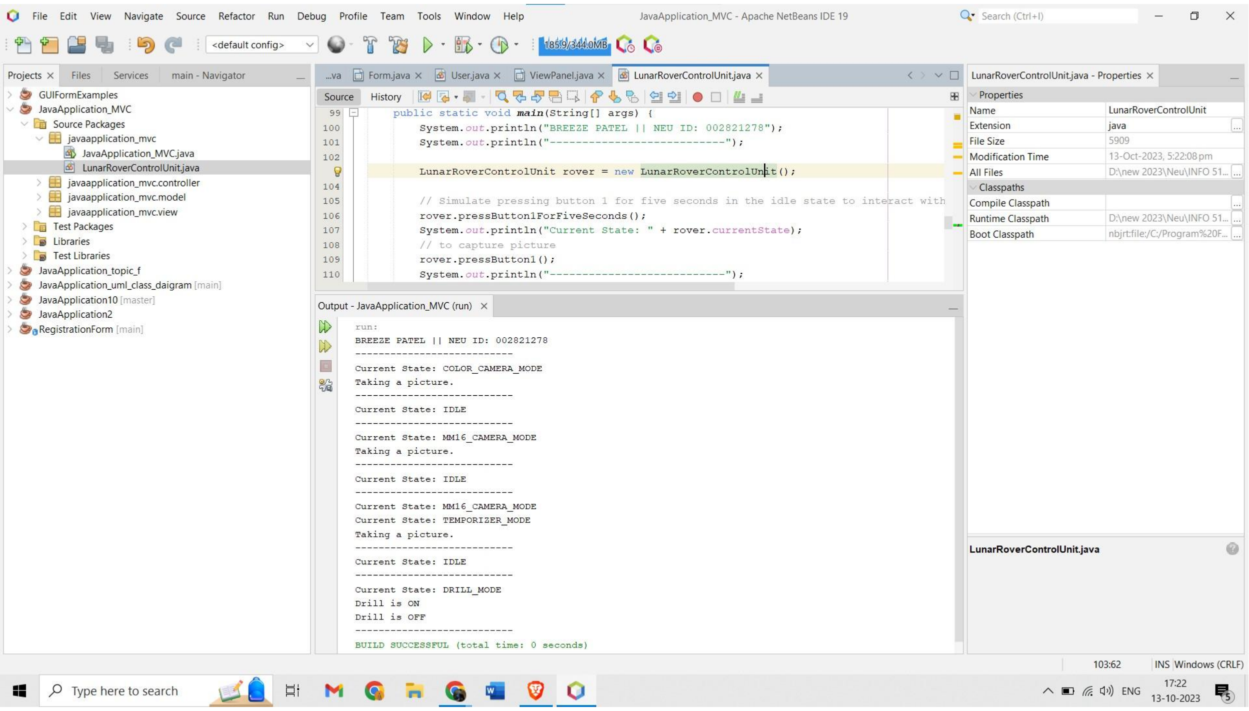
- Press the button 1 for five seconds to interact with the color camera.
- Press the button 1 for ten seconds to interact with the 16-mm camera.
- Press button 1 twice to interact with the drill.
- Pressing button 2 at any of these modes returns to an idle state.
- In any of the two camera modes, pressing button 1 takes a picture, while if pressed for 5 seconds activates the temporizer (moon selfies).
- Last inside the drill mode button 1 only acted as an on-off switch.

1. Design a state machine to model the Camera and Drill. Extend the model you developed for the movement control (2 points).
2. Provide an implementation for this extended state machine in Java (2 points).

Every state should implement an entry action that prints on the console the current state.



JAVA CODE:



```
    public static void main(String[] args) {
        System.out.println("BREEZE PATEL || NEU ID: 002821278");
        System.out.println("--------------------------");

        LunarRoverControlUnit rover = new LunarRoverControlUnit();

        // Simulate pressing button 1 for five seconds in the idle state to interact with
        rover.pressButton1ForFiveSeconds();
        System.out.println("Current State: " + rover.currentState);
        // to capture picture
        rover.pressButton1();
        System.out.println("--------------------------");
```

```
run:
BREEZE PATEL || NEU ID: 002821278
--------------------------
Current State: COLOR_CAMERA_MODE
Taking a picture.
--------------------------
Current State: IDLE
--------------------------
Current State: MM16_CAMERA_MODE
Taking a picture.
--------------------------
Current State: IDLE
--------------------------
Current State: MM16_CAMERA_MODE
Current State: TEMPORIZER_MODE
Taking a picture.
--------------------------
Current State: IDLE
--------------------------
Current State: DRILL_MODE
Drill is ON
Drill is OFF
--------------------------
BUILD SUCCESSFUL (total time: 0 seconds)
```

Properties	
Name	LunarRoverControlUnit
Extension	java
File Size	5909
Modification Time	13-Oct-2023, 5:22:08 pm
All Files	D:\new 2023\Neu\INFO 51...
Classpaths	
Compile Classpath	
Runtime Classpath	D:\new 2023\Neu\INFO 51...
Boot Classpath	nbjrt:file:/C:/Program%20F...

```
/*

 * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license

 * Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit this template

 */

package javaapplication_mvc;

/**

 *

 * @author Breeze

 */

public class LunarRoverControlUnit {



    // Define the states

    public enum State {

        IDLE, COLOR_CAMERA_MODE, MM16_CAMERA_MODE, DRILL_MODE, TEMPORIZER_MODE

    }
```

```
private State currentState = State.IDLE;

// Variables to track button presses and timers
private long button1PressTime = 0;
private long temporizerTimer = 0;
private boolean drillOn = false;

public void pressButton1twice() {
    if (currentState == State.IDLE) {
        currentState = State.DRILL_MODE;
        // Toggle the drill on/off in drill mode
    }
}
public void pressButton1() {
    if (currentState == State.DRILL_MODE) {
        // Toggle the drill on/off in drill mode
        toggleDrill();
    } else if (currentState == State.COLOR_CAMERA_MODE) {
        // Take a picture in color camera mode
        takePicture();
    } else if (currentState == State.MM16_CAMERA_MODE) {
        // Take a picture in 16-mm camera mode
        takePicture();
    } else if (currentState == State.TEMPORIZER_MODE) {
        // Take a picture in TEMPORIZER_MODE
        takePicture();
    }
}
```

```
    public void pressButton1ForFiveSeconds() {
        if (currentState == State.IDLE) {
            // Transition to color camera mode
            currentState = State.COLOR_CAMERA_MODE;
            button1PressTime = System.currentTimeMillis();
        } else if (currentState == State.COLOR_CAMERA_MODE) {
            // activateTemporizer moon selfie in color camera mode
            activateTemporizer();
        } else if (currentState == State.MM16_CAMERA_MODE) {
            // activateTemporizer moon selfie in 16-mm camera mode
            activateTemporizer();
        }
    }

    public void pressButton1ForTenSeconds() {
        if (currentState == State.IDLE) {
            // Transition to 16-mm camera mode
            currentState = State.MM16_CAMERA_MODE;
            button1PressTime = System.currentTimeMillis();
        }
    }

    public void pressButton2() {
        if (currentState == State.COLOR_CAMERA_MODE || currentState == State.MM16_CAMERA_MODE ||
currentState == State.TEMPORIZER_MODE) {
            // Return to idle state
            currentState = State.IDLE;
            button1PressTime = 0;
        }
    }

    private void takePicture() {
        if (currentState == State.COLOR_CAMERA_MODE || currentState == State.MM16_CAMERA_MODE ||
currentState == State.TEMPORIZER_MODE) {
            System.out.println("Taking a picture.");
        }
    }

    private void toggleDrill() {
        if (currentState == State.DRILL_MODE) {
            drillOn = !drillOn;
            System.out.println("Drill is " + (drillOn ? "ON" : "OFF"));
        }
    }

    private void activateTemporizer() {
        if (currentState == State.COLOR_CAMERA_MODE || currentState == State.MM16_CAMERA_MODE) {
            temporizerTimer = System.currentTimeMillis();
            currentState = State.TEMPORIZER_MODE;
        }
    }

    public static void main(String[] args) {
        System.out.println("BREEZE PATEL || NEU ID: 002821278");
        System.out.println("--------------------------");

        LunarRoverControlUnit rover = new LunarRoverControlUnit();

        // Simulate pressing button 1 for five seconds in the idle state to interact with the color camera
        rover.pressButton1ForFiveSeconds();
        System.out.println("Current State: " + rover.currentState);
        // to capture picture
        rover.pressButton1();
        System.out.println("---------------------------");

        // Simulate pressing button 2 to return to idle state from color camera mode
        rover.pressButton2();
        System.out.println("Current State: " + rover.currentState);
        System.out.println("---------------------------");

        // Simulate pressing button 1 for ten seconds in the idle state to interact with the 16-mm camera
        rover.pressButton1ForTenSeconds();
        System.out.println("Current State: " + rover.currentState);
        // to capture picture
        rover.pressButton1();
        System.out.println("---------------------------");

        // Simulate pressing button 2 to return to idle state from 16-mm camera mode
        rover.pressButton2();
        System.out.println("Current State: " + rover.currentState);
        System.out.println("---------------------------");

        rover.pressButton1ForTenSeconds();
        System.out.println("Current State: " + rover.currentState);
        // TEMPORIZER_MODE
        rover.pressButton1ForFiveSeconds();
        System.out.println("Current State: " + rover.currentState);
// to capture picture
```

```
        rover.pressButton1();
        System.out.println("---------------------------");

        // Simulate pressing button 2 to return to idle state from 16-mm camera mode
        rover.pressButton2();
        System.out.println("Current State: " + rover.currentState);
        System.out.println("---------------------------");

        // Simulate pressing button 1 twice in the idle state to enter the drill mode and toggle the drill on/off
        rover.pressButton1twice();
        System.out.println("Current State: " + rover.currentState);
        // Simulate pressing button 1 to toggle drill mode
        rover.pressButton1();
        // Simulate pressing button 1 to toggle drill mode
        rover.pressButton1();
        System.out.println("---------------------------");

    }
}
```