Code Design and Explanation:

This code is written in C++ using the Arduino framework, and it is intended to be run on an Arduino MKR IoT Carrier board. The code defines several variables to keep track of button presses, durations, counts, and recording statuses. It also defines two arrays: one for storing a list of emotions and the other for storing color codes to be used in displaying the emotion on the board.

In the setup() function, the code initializes the board by calling carrier.begin() after a 1.5-second delay and sets the CARRIER\_CASE variable to false, as we are not using the outer case in this prototype.

The loop() function is called repeatedly, and it performs several tasks. It first updates the state of all the buttons by calling carrier.Buttons.update(). Then, it checks each button, in turn, to see if it has been pressed or released. If a button has been pressed, the code calculates the duration of the press and checks whether it has exceeded a threshold of 1500 milliseconds (1.5s). If it has, the code checks whether another recording is currently in progress. If there is no other recording, the code sets the recording status of the corresponding emotion to 1, plays a notification sound, and displays the emotion on the screen in the corresponding color. On the other hand, if the button is holding for 1.5s and there is already a recording in progress for the corresponding emotion, the code stops the recording by setting the recording status to 0, plays a notification sound, and changes the display to show that the recording has stopped. The code does nothing if the button has not been held down long enough.

If a button has been released, and If there is a corresponding emotion recording in progress, the code allows the button press to count for intensity, plays a beep sound, and changes the display of the intensity count to show that the button has been pressed and adding one each time.

Overall, the code implements a simple emotion recording system, where each button corresponds to a different emotion. The board starts recording audio for that emotion when a button is pressed and held for more than 1.5 seconds. When the same button is pressed and released, the board increments a count for that emotion and plays a beep sound. The display on the board shows the emotion in the corresponding color.