1. [10 points] Reaction time is the delay between the onset of a stimulus and an individual's response. For example, in a 100-metre dash a race official fires a pistol to signal the beginning of the race. The sound of the pistol firing is the stimulus and the time it takes for a runner to react is their reaction time.

   Write code to implement a simple reaction timer that works as follows and uses **interrupts**.

   i) Set the switch `SW1` in Off position. The program starts with a message on LCD as "Toggle SW1" on the first line and "if LED glows" on the second line. After displaying this message for 2 seconds, turn on the LED `P1.4`.

   ii) The user toggles `SW1` because of the displayed message. The LED should be turned Off as soon as the switch is toggled.

   iii) The program is expected to measure time between the instant the LED starts glowing and the instant the switch toggle is identified.

   iv) Next, the display on the LCD shows "Reaction Time" on the first line and the "Count is XX XXXX" on the second line. Assuming `T0` is used, display of count should be of 6 digits.
      - First 2 digits should show the number of times the timer has overflown
      - The next 2 digits should show the `TH0` value
      - The next 2 digits should show `TL0` value
      - All values need to be shown in hexadecimal format.
      - This message should be displayed for 5 seconds.

      Bonus 5 points: Optionally, you can display the reaction time in milliseconds. That is, instead of displaying "Count is XX XXXX", you can display "TTTT milliseconds". The TTTT should be in decimal format.

   v) Go back to Step (i) and repeat the process continuously.

2. [10 points] Write code to play some music corresponding to the note frequencies mentioned in Table 1 on the left hand side. The order and duration of the notes are given on the right hand side. The musical notes can be generated as square waveforms of specific frequencies. To generate silence, write the zero bit to the port pin `P0.7`. These can be generated and output using the speaker with appropriate interfacing.

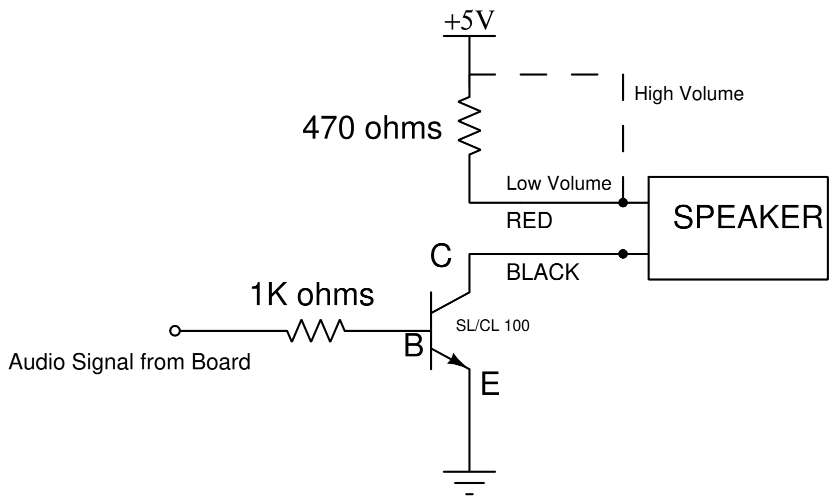
Table 1: Note frequencies, sequence and durations.

		S. No.	Note	Frequency (in Hertz)
	EE3F	1	N1	220 4545us
	F030	2	N2	247 4048us
	F2B7	3	N3	294 3401us
5405	F572	4	N4	370 2702us
6060	F42A	5	N5	330 3030us

Order	Note	Duration (ms)
1	N1	750
2	N2	750
3	N3	750
4	N2	750
5	N4	1000
6	Silence	500
7	N4	1000
8	N5	1000

1. **Use timer `T0` to generate the note (waveform) of appropriate frequency. Use timer `T1` to control the duration of the note**. The output has to be written to port pin `P0.7`. Generate the notes shown on the right hand side in Table 1 using the durations mentioned. Do this continuously and verify the output using the debugger.
2. Connect the speaker to the `Pt-51` kit using the SL100 transistor and interfacing circuit shown in Figure 1. You should be able to hear the notes being played out on the speaker. During this period, the text "ROLLING TIME" must be displayed on the LCD.
3. **Shoot a 30 second video of your kit playing the notes. Make sure your face appears at the beginning of the video.** The video needs to be uploaded using the **Video upload link for lab 7** assignment in the course team on MS Teams (it has an option to attach files). The file size cannot exceed 200 MB.

(a) SL100 pinout



(b) Speaker driver circuit.

Figure 1: Circuit for interfacing Pt-51 with speaker.

TA Checkpoints

1. For question 1, ask the students to show the working of the reaction timer. Do this with short and long reaction times. Keep the long reaction time to below 30 seconds to avoid wrap around.

2. For question 2, check the appropriate waveforms on the logic analyzer (Keil). Also ask them to play out the notes on the speaker.

1 cycle: 0.5us
2000cycle: 1ms