Massey University
School of Engineering and Advanced Technology
Computer Science and Information Technology

Discover Computing With the Raspberry Pi 3 and the Sense HAT

Activity 3: Composing Music Using Sonic Pi on the Raspberry Pi 3

Description and Aim of Activity 3

Sonic Pi is an open-source programming environment, designed for creating new sounds with code in a live coding environment. It was developed by Dr Sam Aaron at the University of Cambridge. He uses the software to perform live with his band. The aim of this activity is to help get you started with the basics of Sonic Pi programming so that you can code your own music.

Learning Objectives for Activity 3

At the end of this activity, you should be able to:

1. Make sounds by typing text into Sonic Pi
2. Loop a tune so that it repeats
3. Explain what MIDI note numbers are;
4. Convert MIDI notes to music notes
5. Play two tunes at the same time using threads
6. Write a Sonic Pi program that plays a tune for a song that you know

The Sonic Pi Graphical User Interface

Run Sonic Pi by going to **Menu → Programming → Sonic Pi**
You will then get the following graphical user interface.

This is the Sonic Pi interface; it has three main window areas. The largest one is for writing your code, and we call it the Programming Panel. There is also an output panel that displays information about your program as it runs. When you click on the help button at the top of the window, the third panel appears along the bottom displaying help documentation. This contains information about different code you can try and use, as well as different synth sounds, samples, and much more.

To complete the following tasks, you will need to connect speakers or headphones to the Raspberry Pi 3 and configure the audio output appropriately.

Main Challenge Task: Composing the Tune for the Nursery Song *"Mary had a little lamp"*

The main challenge for this task is to:

> **Write the Sonic Pi program,** `marry-had-a-little-lamp.rb`**, for the tune of the nursery song, "Mary had a little lamp". Your program must:**
>
> 1. **Play the 3 versions, Key of C, Key of F and Key of G, at the same time; and**
> 2. **Repeat the tune 3 times.**

You are given the tune in the Key of C in the form of the WAV file: `mary-had-a-little-lamp.wav`. This wave file was created in Sonic Pi by recording the play of the program that we wrote for this song.

As you search for the right notes, you can play the tune several times.

To play the tune do the following:

1. On the main menu bar, click **`Terminal`**;
2. Type: `cd rpi/discover-computing/03-sonic-pi-music/`
3. Press Enter
3. Type: `aplay mary-had-a-little-lamp.wav`

In order to do this task, you need to take a few minutes to quickly learn how to create the right sounds in Sonic Pi. Please, quickly do Practice Tasks 1 – 4, which follows. Once you complete the practice tasks, return to start working on this challenge task.

Practice Task 1: Basic Sounds With the Sonic Pi

1. Launch **Sonic Pi** from the desktop or applications menu.

2. Select `Buffer 1` and type:

```
play 60
```

3. Click on the play icon at the top of the screen. What happens?

4. What happens if you type `pley 60` and click on the play icon?

This is an example of a bug in your code. In later activities, if the error panel displays text you will know that you have a bug that you need to fix. It could be that you have misspelt a word like `play`.

5. Now type:

```
play 60
play 67
play 69
```

6. Click on the play icon at the top of the screen. What happens?

7. The computer is playing each note in sequence (one after the other), but it is happening so fast that to us they sound like they are playing at the same time.

We need to tell the computer to pause between each note. We can do this by typing the following after each `play`:

```
sleep 1
```

The value entered after the word `sleep` represents time in seconds. Using the value 1 represents one second. What would you type for half a second?

8. Now write a sequence of `play` and `sleep` to make a cool-sounding tune!

Practice Task 2: Looping or Repeating a Tune

Type and play the following in Sonic Pi. The code is for the first part of the tune, Frère Jacques.

```
play 60
sleep 0.5
play 62
sleep 0.5
play 64
sleep 0.5
play 60
sleep 0.5
```

In the tune, *this first section plays twice.*
How could you repeat it?
You could type the same section out again, or we could start to introduce loops to your code.

In the example below, illustrates code for looping twice. You can see that some lines of code are indented. This makes it easier to read your code, and check for any bugs if it does not work when you press the play button. You can press the space bar twice to indent a line of code or the tab key once.

Type and play the

```
2.times do
  play 60
  sleep 0.5
  play 62
  sleep 0.5
  play 64
  sleep 0.5
  play 60
  sleep 0.5
end
```

In the above code, you could replace 2 with any number

To loop for ever, you replace the first line with the following code:

```
loop do
```

Practice Task 3: MIDI and Music Notes

The values that you have been typing after the word `play` represent notes; in fact, they are MIDI note numbers. This means we can translate songs played on a piano into Sonic Pi using a table like so:

Note	Octave										
	-1	0	1	2	3	4	5	6	7	8	9
C	0	12	24	36	48	**60**	72	84	96	108	120
C#	1	13	25	37	49	**61**	73	85	97	109	121
D	2	14	26	38	50	**62**	74	86	98	110	122
D#	3	15	27	39	51	**63**	75	87	99	111	123
E	4	16	28	40	52	**64**	76	88	100	112	124
F	5	17	29	41	53	**65**	77	89	101	113	125
F#	6	18	30	42	54	**66**	78	90	102	114	126
G	7	19	31	43	55	**67**	79	91	103	115	127
G#	8	20	32	44	56	**68**	80	92	104	116	
A	9	21	33	45	57	**69**	81	93	105	117	
A#	10	22	34	46	58	**70**	82	94	106	118	
B	11	23	35	47	59	**71**	83	95	107	119	

For example `60 62 64 60` in MIDI notes are the piano notes: `C D E C`, respectively.

To test the first part of the *Frère Jacques tune* using piano notes, type and play the following code:

In a new buffer tab type:

```
play :c4
sleep 0.5
play :d4
sleep 0.5
play :e4
sleep 0.5
play :c4
sleep 0.5
```

Press play to hear your tune.

Does it sound the same as when you used MIDI notes?

Practice Task 4: Playing more than one tune at the same time

Music often has a repeating backing track, with a separate melody played over the top. So far in Sonic Pi you have played one tune. Let's try playing two tunes at the same time!

1. Click on a new buffer tab.
2. The code we use to play two tunes at the same time needs to be between `in_thread do` and `end`.
3. Underneath `in_thread do`, type your tune. Here we use a *sample* for the backing track:

```
in_thread do
  loop do
    sample :loop_amen
    sleep 1.753
  end
end
```

This first *'thread'* will act as the melody of your music. Underneath, you can type the code for your backing track or baseline.

4. Type:

```
in_thread do
  16.times do
    play 75
    sleep 1.753
    play 74
    sleep 0.25
  end
end
```

Now press play and you should hear both threads playing at the same time.
In Sonic Pi, you can also *modularise* by declaring a piece of music and then playing it by name. For example:

```
define :play_my_synth do
  use_synth :prophet
  play 50, attack: 0.2, release: 1.3
  sleep 0.5
end

loop do
  play_my_synth
end
```