Module Interface Specification for SE 4G06, TRON 4TB6

Team 26, STRONE
Jordan Bierbrier
Azriel Gingoyon
Taranjit Lotey
Udeep Shah
Abraham Taha

April 5, 2023

Revision History

Date	Version	Notes
18-01-23	1.0	Added Content

Symbols, Abbreviations and Acronyms

See SRS Documentation at *SRS.pdf Document Link*.

Contents

1 Introduction **1**

2 Notation **1**

3 Module Decomposition **1**

4 MIS of Battery Status **3**
- 4.1 Module . . . 3
- 4.2 Uses . . . 3
- 4.3 Syntax . . . 3
  - 4.3.1 Exported Constants . . . 3
  - 4.3.2 Exported Access Programs . . . 3
- 4.4 Semantics . . . 3
  - 4.4.1 State Variables . . . 3
  - 4.4.2 Environment Variables . . . 3
  - 4.4.3 Assumptions . . . 3
  - 4.4.4 Access Routine Semantics . . . 3
  - 4.4.5 Local Functions . . . 4

5 MIS of Battery Status **4**
- 5.1 Module . . . 4
- 5.2 Uses . . . 4
- 5.3 Syntax . . . 4
  - 5.3.1 Exported Constants . . . 4
  - 5.3.2 Exported Access Programs . . . 4
- 5.4 Semantics . . . 4
  - 5.4.1 State Variables . . . 4
  - 5.4.2 Environment Variables . . . 4
  - 5.4.3 Assumptions . . . 4
  - 5.4.4 Access Routine Semantics . . . 5
  - 5.4.5 Local Functions . . . 5

6 MIS of Login Module **5**
- 6.1 Module . . . 5
- 6.2 Uses . . . 5
- 6.3 Syntax . . . 5
  - 6.3.1 Exported Constants . . . 5
  - 6.3.2 Exported Access Programs . . . 5
- 6.4 Semantics . . . 5
  - 6.4.1 State Variables . . . 5
  - 6.4.2 Environment Variables . . . 6
  - 6.4.3 Assumptions . . . 6

6.4.4 Access Routine Semantics . . . 6

7 MIS of Keyword Selection Module **7**
7.1 Module . . . 7
7.2 Uses . . . 7
7.3 Syntax . . . 7
7.3.1 Exported Constants . . . 7
7.3.2 Exported Access Programs . . . 7
7.4 Semantics . . . 7
7.4.1 State Variables . . . 7
7.4.2 Environment Variables . . . 7
7.4.3 Assumptions . . . 7
7.4.4 Access Routine Semantics . . . 8

8 MIS of Bluetooth Communication Module **9**
8.1 Module . . . 9
8.2 Uses . . . 9
8.3 Syntax . . . 9
8.3.1 Exported Constants . . . 9
8.3.2 Exported Access Programs . . . 9
8.4 Semantics . . . 9
8.4.1 State Variables . . . 9
8.4.2 Environment Variables . . . 9
8.4.3 Assumptions . . . 9
8.4.4 Access Routine Semantics . . . 9

9 MIS of Microphone Module **11**
9.1 Module . . . 11
9.2 Uses . . . 11
9.3 Syntax . . . 11
9.3.1 Exported Constants . . . 11
9.3.2 Exported Access Programs . . . 11
9.4 Semantics . . . 11
9.4.1 State Variables . . . 11
9.4.2 Environment Variables . . . 11
9.4.3 Assumptions . . . 11
9.4.4 Access Routine Semantics . . . 11

10 MIS of Output Signal Module **12**
10.1 Module . . . 12
10.2 Uses . . . 12
10.3 Syntax . . . 12
10.3.1 Exported Constants . . . 12

10.3.2 Exported Access Programs . . . 12
10.4 Semantics . . . 12
10.4.1 State Variables . . . 12
10.4.2 Environment Variables . . . 12
10.4.3 Assumptions . . . 12
10.4.4 Access Routine Semantics . . . 12

11 MIS of Bluetooth Connection Module **13**
11.1 Module . . . 13
11.2 Uses . . . 13
11.3 Syntax . . . 13
11.3.1 Exported Constants . . . 13
11.3.2 Exported Access Programs . . . 13
11.4 Semantics . . . 13
11.4.1 State Variables . . . 13
11.4.2 Environment Variables . . . 13
11.4.3 Assumptions . . . 13
11.4.4 Access Routine Semantics . . . 13

12 MIS of Profile Module **15**
12.1 Module . . . 15
12.2 Uses . . . 15
12.3 Syntax . . . 15
12.3.1 Exported Constants . . . 15
12.3.2 Exported Access Programs . . . 15
12.4 Semantics . . . 15
12.4.1 State Variables . . . 15
12.4.2 Environment Variables . . . 15
12.4.3 Assumptions . . . 15
12.4.4 Access Routine Semantics . . . 16

13 Appendix **17**

1 Introduction

The following document details the Module Interface Specifications for Synesthesia Wear, a wearable product that assists users by using signal processing on gathered sounds to provide appropriate feedback (via vibrations) to the user according to inputted sound configuration settings. As a result, this gives the users peace of mind knowing that if their attention is needed (doorbell, ring, name call, etc.), Synesthesia Wear will be able to alert them.

Complementary documents include the System Requirement Specifications and Module Guide. The full documentation and implementation can be found at *Team 26 Capstone GitHub Repository*.

2 Notation

The structure of the MIS for modules comes from **?**, with the addition that template modules have been adapted from **?**. The mathematical notation comes from Chapter 3 of **?**. For instance, the symbol := is used for a multiple assignment statement and conditional rules follow the form $(c_1 \Rightarrow r_1|c_2 \Rightarrow r_2|...|c_n \Rightarrow r_n)$.

The following table summarizes the primitive data types used by SE 4G06, TRON 4TB6.

Data Type	Notation	Description
character	char	a single symbol or digit
integer	$\mathbb{Z}$	a number without a fractional component in (-∞, ∞)
natural number	$\mathbb{N}$	a number without a fractional component in [1, ∞)
real	$\mathbb{R}$	any number in (-∞, ∞)

The specification of SE 4G06, TRON 4TB6 uses some derived data types: sequences, strings, and tuples. Sequences are lists filled with elements of the same data type. Strings are sequences of characters. Tuples contain a list of values, potentially of different types. In addition, SE 4G06, TRON 4TB6 uses functions, which are defined by the data types of their inputs and outputs. Local functions are described by giving their type signature followed by their specification.

3 Module Decomposition

The following table is taken directly from the *Module Guide Document* for this project.

Level 1	**Level 2**
Hardware-Hiding Module	
Behaviour-Hiding Module	~~Login Module~~ Bluetooth connection ModuleM2 Keyword Selection ModuleM3 Output Signal ModuleM4 Profile ModuleM5 Battery Status ModuleM6
Software Decision Module	Sound Classification ModuleM7 Bluetooth Communication ModuleM8 Microphone ModuleM9

Table 1: Module Hierarchy

4 MIS of Battery Status

4.1 Module

~~Battery Status~~

4.2 Uses

Indicator

4.3 Syntax

4.3.1 Exported Constants

N/A

4.3.2 Exported Access Programs

Name	In	Out	Exceptions
Low Battery	N/A	Low Status	Low Battery
Battery Level	N/A	Percentage	Battery Status

4.4 Semantics

4.4.1 State Variables

N/A

4.4.2 Environment Variables

keyOutput : key.Low Battery, key.Battery Status

4.4.3 Assumptions

Battery will always have some charge on it.

4.4.4 Access Routine Semantics

- transition: N/A
- output: Battery level
- exception: N/A

4.4.5 Local Functions

- get level(int level)
- battery condition(int cond)

5 MIS of Battery Status

5.1 Module

Sound Classification M7
Dependent on Bluetooth communication M8

5.2 Uses

Categorize detected sounds

5.3 Syntax

5.3.1 Exported Constants

N/A

5.3.2 Exported Access Programs

Name	In	Out	Exceptions
Sound level	Microphone	decibels	High/Low Level

5.4 Semantics

5.4.1 State Variables

N/A

5.4.2 Environment Variables

keyOutput : key.microphone $\mathbb{N}$
keyOutput : key.decibels $\mathbb{N}$

5.4.3 Assumptions

Surrounding sound levels are null compared to required input level.

5.4.4 Access Routine Semantics

- transition: N/A
- output: decibels
- exception: N/A

5.4.5 Local Functions

- sound level(float volume)
- sorting volume()

6 MIS of Login Module

6.1 Module

~~Login Type~~

6.2 Uses

N/A

6.3 Syntax

6.3.1 Exported Constants

N/A

6.3.2 Exported Access Programs

Name	In	Out	Exceptions
authorize	keyInput	Authorized	
login	keyInput		Not_A_Character
switchCue	keyInput		Not_Cueable

6.4 Semantics

6.4.1 State Variables

Authorized: Authorized is a boolean that is true when the correct credentials are entered and false otherwise.

6.4.2 Environment Variables

keyInput: {key.Enter, key.AlphabetCharacters, key.LeftClick}

6.4.3 Assumptions

The Synesthesia Wear application is successfully installed on the user's device and the login page has loaded onto the screen.

6.4.4 Access Routine Semantics

switchCue(key.LeftClick):

- transition: mouseLocation.navigate()
- output: None
- exception: Not_Cueable

login(key.AlphabetCharacters):

- transition: username.addCharacter() or password.addCharacter()
- output: None
- exception: Not_A_Character

authorize(key.Enter):

- transition: login.submit()
- output: Authorized
- exception: None

7 MIS of Keyword Selection Module

7.1 Module

Keyword Selection Type

7.2 Uses

- Profile Module M5
- Bluetooth Communication Module M8

7.3 Syntax

7.3.1 Exported Constants

N/A

7.3.2 Exported Access Programs

Name	In	Out	Exceptions
save	keyInput	Saved	Mouse_Not_On_Save_Button
keyword	keyInput		Not_A_Character
switchCue	keyInput		Not_Cueable

7.4 Semantics

7.4.1 State Variables

Saved: Saved is a boolean that is true when the "Save" button has been pressed and false otherwise.

7.4.2 Environment Variables

keyInput: {key.AlphabetCharacters, key.LeftClick}

7.4.3 Assumptions

The Synesthesia Wear application is successfully installed on the user's device, the user was able to log into the app, and the sound configuration settings page is loaded onto the screen.

7.4.4 Access Routine Semantics

switchCue(key.LeftClick):

- transition: mouseLocation.navigate()
- output: None
- exception: Not_Cueable

keyword(key.AlphabetCharacters)char:

- transition: keyword.addCharacter()
- output: None
- exception: Not_A_Character

save(key.LeftClick):

- transition: keyword.save()
- output: Saved
- exception: Mouse_Not_On_Save_Button

8 MIS of Bluetooth Communication Module

8.1 Module

BTComuModule

8.2 Uses

None

8.3 Syntax

8.3.1 Exported Constants

DataTx:{ BT.send()}

8.3.2 Exported Access Programs

Name	In	Out	Exceptions
RecvNewClass	DataRx	-	Class_Full
RmClass	DataRx	-	Class_Empty
UpdatePref	DataRx	-	-
SendBattV	BattV	DataTx	-

8.4 Semantics

8.4.1 State Variables

None

8.4.2 Environment Variables

~~BattV: {AnalogRead(BatteryVolt)}~~
DataRx: {BT.recieve(T1),BT.recieve(T2),BT.recieve(T3)}ℕ

8.4.3 Assumptions

Connection with the application is already establised. Bluetooth tries to automatically reconnect if the application is disconnected.

8.4.4 Access Routine Semantics

RecvNewClass():

- transition: if(DataRx == T1) then addclass(dataRx)

- output: None
- exception: Class_Full

RmClass():

- transition: if(DataRx == T2) then rmclass(dataRx)
- output: None
- exception: Class_Empty

UpdatePref():

- transition: if(DataRx == T3) then prefchange(DataRx.class,DataRx.pref)
- output: None
- exception: None

~~SendBattV(BattV)~~:

- transition: None
- output: DataTx == BattV
- exception:

9 MIS of Microphone Module

9.1 Module

MicroMod

9.2 Uses

None

9.3 Syntax

9.3.1 Exported Constants

None

9.3.2 Exported Access Programs

Name	In	Out	Exceptions
updateData	SoundIN,CLKINT	SoundData	-

9.4 Semantics

9.4.1 State Variables

SoundData - Array of sound inputs recorded at every sample interval

9.4.2 Environment Variables

SoundIN: { Digital.Read(Digital_Microphone) }
CLKINT: {clk.interupt(sample time)}

9.4.3 Assumptions

Clock interrupt is already set up to issue interrupts every 1/(sample frequency)

9.4.4 Access Routine Semantics

- transition: N/A
- output: decibels
- exception: N/A

10 MIS of Output Signal Module

10.1 Module

Output Signal Module

10.2 Uses

- Sound Classification Module

10.3 Syntax

10.3.1 Exported Constants

N/A

10.3.2 Exported Access Programs

Name	In	Out	Exceptions
onVolt	(Boolean, Class_Id)		Null_class

10.4 Semantics

10.4.1 State Variables

current_Volt: Integer of the current voltage being supplied to the motor

10.4.2 Environment Variables

N/A

10.4.3 Assumptions

We are assuming that when this module is called that the battery has sufficient charge to power the motor for the called upon number of requests

10.4.4 Access Routine Semantics

onVolt(Boolean onOFF, Class_id Current):

- transition: Count = Current | While count ! = 0 $\implies$ (current_volt $\equiv$ 0 $\wedge$ onOFF) $\implies$ current_Volt = 5 | count $-=$ count | current_Volt = 0
- output: None
- exception: Class_id $\equiv$ Null

11 MIS of Bluetooth Connection Module

11.1 Module

Bluetooth Connection Module

11.2 Uses

N/A

11.3 Syntax

11.3.1 Exported Constants

N/A

11.3.2 Exported Access Programs

Name	In	Out	Exceptions
Find_Device Connect_Device	Boolean onClick (Boolean, onClick Device_ID)	Device_ID Array	device_NotFound

11.4 Semantics

11.4.1 State Variables

- known_Device: Device_ID
- Paired_Device

11.4.2 Environment Variables

N/A

11.4.3 Assumptions

We assume that the mobile device that the user has the application on has a bluetooth module.

11.4.4 Access Routine Semantics

Find_Device(Boolean onClick):

- transition: (Onclick $\equiv$ True) $\implies$ Found_Devices = $\implies$ Found_Devices.Add(known_Devices) $\implies$ Found_Devices.Add(device_id1)

- output: Found_Devices(known_Device, device_id1, device_id2 ... device_idN)
- exception: Device_NotFound

Connect_Device(Boolean onClick):

- transition: (Onclick $\equiv$ True) $\implies$ Paired_Device = device_id2
- output: None
- exception: None

12 MIS of Profile Module

12.1 Module

Profile Module

12.2 Uses

Login Module

12.3 Syntax

12.3.1 Exported Constants

N/A

12.3.2 Exported Access Programs

Name	In	Out	Exceptions
Set_First_Name	String		Type_Error
Set_Last_Name	String		Type_Error
Get_First_Name		String	Null_Exception
Get_Last_Name		String	Null_Exception
change_Password	String		Type_Error
Change_Username	String		Type_Error
Get_Username		String	Null_Exception

12.4 Semantics

12.4.1 State Variables

- first_name
- last_name
- password
- username

12.4.2 Environment Variables

key_input(keyboard)

12.4.3 Assumptions

We are assuming that the user has either a touchscreen keyboard or an external keyboard.

12.4.4 Access Routine Semantics

Set_first_name(String Name):

- transition: first_name = Name
- output: None
- exception: Type_Error

Set_last_name(String Name):

- transition: last_name = Name
- output: None
- exception: Type_Error

Get_first_name():

- transition: None
- output: First_name
- exception: Null_Exception

Get_last_name():

- transition: None
- output: last_name
- exception: Null_Exception

Change_password(String Name):

- transition: password = Name
- output: None
- exception: Type_Error

Change_username(String Name):

- transition: username = Name
- output: None
- exception: Type_Error

Get_username():

- transition: None
- output: username
- exception: Null_Exception

13 Appendix