Question 1:

Phidgets: Basics and Parking Assistant **[25 Marks]**

a) An analog Phidget input sensor, e.g. touch sensor, is connected via a USB port to a PC. An application running on the PC creates a COM object for the Phidget sensor and opens the communication between the application and the device (sensor). Also, the COM object has a listener interface that handles events coming from the sensor. Sketch an abstract diagram that explains how in the software levels

- *Application*
- *Phidget Object*
- *Device*
- *Listener*

communicate with each other.

[7 marks]

b) Write a pseudo code in any programming language to connect a touch sensor to the application.

[8 marks]

c) Design a *Parking Assistant* that acts as a stop light and hangs on the front wall of a garage. It senses a vehicle coming into the garage and displays a green light. As the car gets closer to the wall, the yellow light comes on, warning the driver to slow down. Finally, the red light signals that the driver is close enough to the wall and should stop.

Describe a suitable control interface for the parking assistant that could be piloted using a set of Phidgets. Assume there is no limit to the number of any type of phidgets. Explain and justify your choice of individual phidget types, and how they would assist the overall interface design.

[10 marks]

Question 2:
Phidegts+SHAKE: Music Player **[25 Marks]**

Scenario: Design a music player, which will have a small display. The target user **will not be fully** relying on sight for the control of the device and it should support the full range of access on an advanced personal music player including:

- *Play lists*: by Artist and/or Album.
- *Feedback*: Information about current music track is displayed.
- *Basic controls*: Play, pause, next/last track, volume control

The device should be controllable by SHAKE and Phidgets, however, buttons must not be used.

a) Describe a suitable control interface for the *basic controls* that could be piloted using a set of Phidgets and SHAKE sensors. Assume there is no limit to the number of any type of Phidgets, just no buttons. Explain and justify your choice of individual Phidget types and SHAKE sensors, and how they would assist the overall interface design.

[10 marks]

b) Further describe which Phidget types and SHAKE sensors you would add for *feedback* and *play list* controls, again justifying your choices. Also describe how the feedback would be provided (e.g. by haptic and/or audio and/or visual cues) and how this would interact with the basic functions of the device.

[8 marks]

c) Outline a method for providing interaction between different Phidgets and SHAKE sensors and distinguishing their role (which sensor does what) in the interface, and the appearance of the device. Use sketch drawings and/or diagrams to clarify your overall design.

[7 marks]

Question 3:

SHAKE: Basics **[25 Marks]**

a) Describe what has made SHAKE a 6 Degrees of Freedom (DOF) inertial sensing tool and give a definition for each sensor incorporated in inertial sensing.

[4 marks]

b) SHAKE has an onboard vibration motor with variable speed control and active braking. A vibration profile can be programmed and triggered by the host, such as the HyperTrm application. To upload a vibration profile to the SHAKE this packet format has been sent via HyperTrm

$VIB,res1,res2,speed1,end1,speed2,end2,speed3,end3

Which data field (s) is (are) missing in this packet?

[3 marks]

c) Improve the packet format in (b) and upload a profile to address space number 20 that drives the motor full speed for the first 200ms, at half speed for the next 200ms and then applies the brake.

[8 marks]

d) Write a pseudo code in any high-level programming language (C, C++, Java, C#, or Python) that uploads a vibration profile to address space number 200 that drives the motor full speed for the first 400ms, at half speed for the next 300ms and then applies the brake for 100 ms. Also, write a pseudo code that plays the uploaded pattern in the main vibration channel.

[10 marks]