# 3. SYSTEM REQUIREMENTS SPECIFICATIONS (SRS)

## 3.1 User Requirement Specification

The system consists of an armband and a window application.

When user wears the armband, it must record hand gestures of user in the form of EMG data. Then the EMG data must be sent to window application via Bluetooth connection. The armband has five main functions:

* Get analog signal from hand gestures
* Process analog signal: this function requires following steps:
* Pre-amplify: Because of the amplitude of signal is too minimal for rectify and amplify, so this step will make signal amplitude increased.
* Rectify: In order to keep the value of signal to be positive for microcontroller, this step will simply convert all negative value to positive value.
* Amplify: This step will again widen the amplitude of signal.
* Filter: This step is to eliminate the noises of the signal.
* Convert analog signal to digital signal: Convert analog signal to 12-bit digital signal.
* Package digital signal (called EMG data): Data will be packaged in form of services that available in Low Energy Bluetooth.
* Send EMG data from the armband to window application through Low Energy Bluetooth protocol.

The purpose of the window application is to make communication between armband and PC. When user starts the application, user can:

* **Connect with armband:** User plugins Bluetooth Adapter, and launches the application. The application must recognize the armband.
* **Perform Sync Gesture:** User performs a hand gesture that is shown on the screen, then the application must sync armband with PC. This function makes the armband ready to use with window application.
* **Test armband with hand gestures**: User performs each hand gesture shown on the screen in turn. This function makes sure that the armband work correctly.
* **Manage the armband:**

***Turn on/off presentation control*:** User can turn on or turn off Microsoft PowerPoint presentation control.

***Disconnect armband*:** User can disconnect with armband. This function helps user to refresh and reconnect armband in case of the armband doesn’t work correctly.

***View armband battery level*:** User can view the battery level of the armband.

* **Control Microsoft PowerPoint presentation with hand gestures:** User can perform some simple hand gestures to control PowerPoint presentation:

**Spread fingers:** Zoom Out of the Slide

**Make a fist:** Zoom In to the Slide

**Wave Right:** Move to next Slide

**Wave Left:** Move to previous Slide

**Double tab:** Exit Slide Show

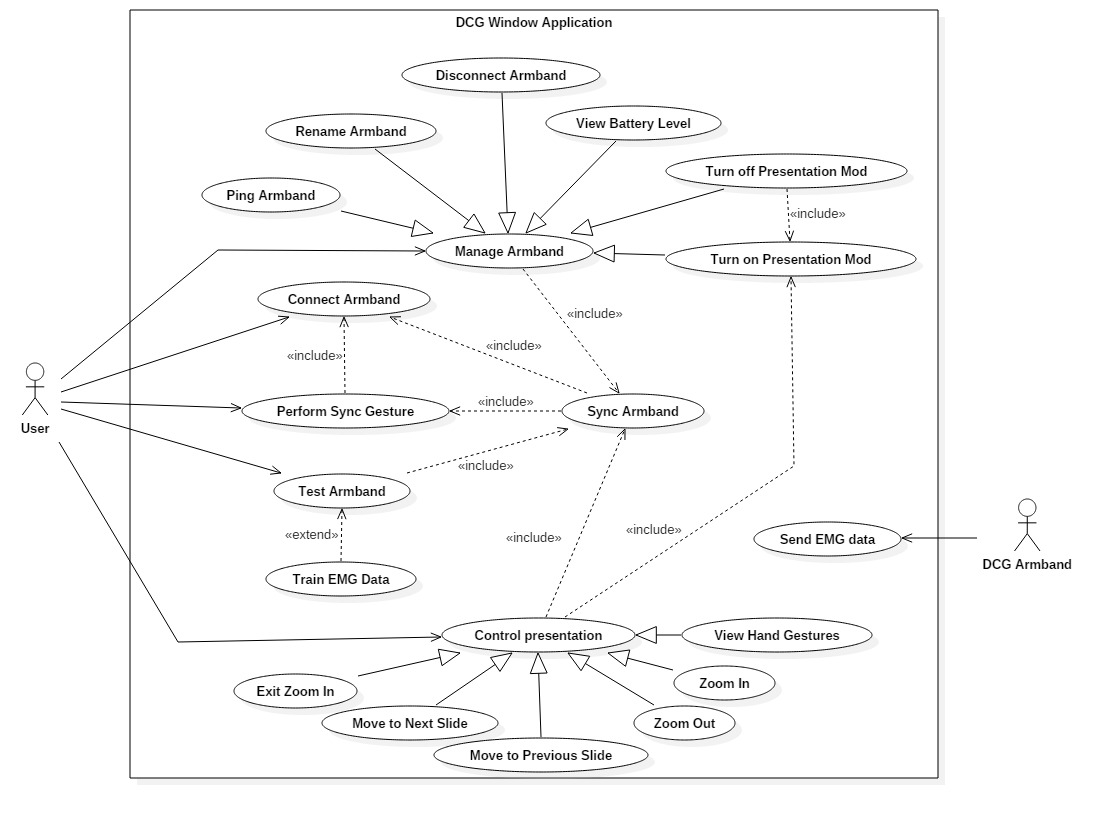
Window Application must do background tasks:

* Automatically connect armband with PC when user press connect button.
* Sync armband to PC when user perform Sync Gesture.
* Train EMG data of sent from armband, when user tests their hand gestures.
* Get and analyze EMG data from armband, then map those data with correct expected function.

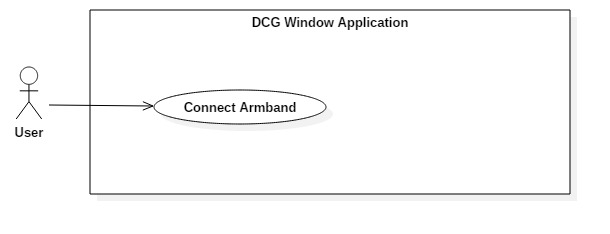
## 3.2 System Requirement Specification

### 3.2.1 Functional Requirement

#### 3.2.1.1 Software Requirement

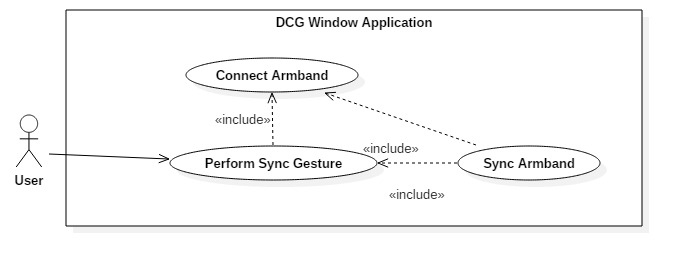


##### **Connect Armband**



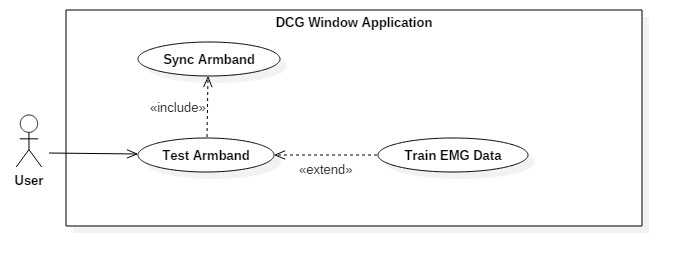
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| USE CASE - UC001 – Connect Armband | | | | |
| Use-case No. | UC001 | **Use-case Version** | | 1.0 |
| Use-case Name | Connect Armband | | | |
| Author | Đinh Bảo Trân | | | |
| Date | 2016/05/30 | **Priority** | Normal | |
| Actor:  User  Summary:   * + This use case allow user to connect armband with window application.   Goal:   * + The window application successfully connect armband via Bluetooth.   Triggers:   * + User launches the application.   Preconditions:   * + Armband is ON   + Armband is not connected to window application.   Post Conditions:   * + Armband is connected to window application   + Windows application changes connection status from *Disconnected* to *Connected* in Manager Screen   + Windows application changes “Connect” button in TATUS Tab of Manager Screen to “Disconnect”   Main Success Scenario:   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Launch the application | Application appears. The Welcome Screen is shown. | | 2 | Click on “**Continue**” button | Plugin Screen is shown. | | 3 | Plugin Bluetooth Adapter  Plugin USB Cable | 1. The first Green Checked Icon appears  2. The second Green Checked Icon appears  3. “Continue” button changes from Disable to Enable. |   Alternative Scenario:   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1a | Move to **Status Tab** in **Manager Screen**  Click on “**Connect**” button | Application appears. The **Welcome Screen** is shown. |   Exceptions:  N/A  Relationship:  Sync Armband | | | | |

##### **Perform Sync Gesture**



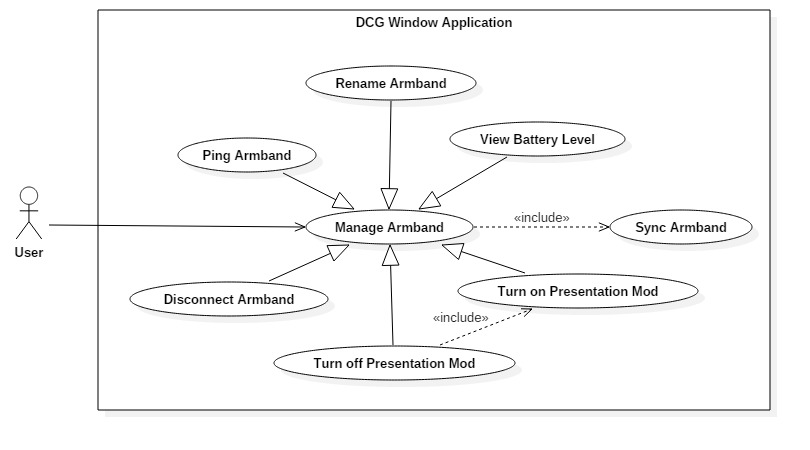
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| USE CASE - UC002 – Perform Sync Gesture | | | | |
| Use-case No. | UC002 | **Use-case Version** | | 1.0 |
| Use-case Name | Perform Sync Gesture | | | |
| Author | Đinh Bảo Trân | | | |
| Date | 2016/05/30 | **Priority** | Normal | |
| Actor:  User  Summary:   * + This use case allow user to sync armband with window application, which means application start to get data from armband.   + After sync function is finished, application can now start to communicate with armband.   Goal:   * + The window application successfully sync with armband.   Triggers:   * + User clicks “Continue” button in Plugin Screen.   Preconditions:   * + User has moved to Plugin Screen.   + Armband is ON   + Armband is connected to window application.   Post Conditions:   * + Armband is synced to window application   Main Success Scenario:   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click “**Continue**” button in **Plugin Screen** | The Syncing Screen is shown. | | 2 | Perform Sync Gesture | 1. Sync Icon appears while syncing  2. Green Checked Icon appears when sync successfully  3. “Continue” button in changes from Disable to Enable |   Alternative Scenario:  N/A  Exceptions:  N/A  Relationship:  Test Armband, Manage Armband, Control Presentation | | | | |

##### **Test Armband**



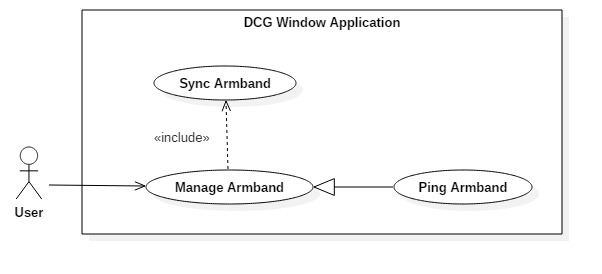
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| USE CASE - UC003 – Test Armband | | | | |
| Use-case No. | UC003 | **Use-case Version** | | 1.0 |
| Use-case Name | Test Armband | | | |
| Author | Đinh Bảo Trân | | | |
| Date | 2016/05/30 | **Priority** | Normal | |
| Actor:  User  Summary:   * + This use case allow user to test armband with 5 hand gestures.   Goal:   * + The window application successfully sync armband with application.   Triggers:   * + User clicks “Continue” button in Syncing Screen.   Preconditions:   * + User has moved to Syncing Screen.   + Armband is ON   + Armband is synced to window application.   Post Conditions:   * + Armband is ready for use.   + “Continue” button in Test Screen changes from Disable to Enable   Main Success Scenario:   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click “**Continue**” button in **Syncing Screen** | The Test Screen is shown. | | 2 | Perform 5 hand gestures one by one | The symbol of corresponding gesture changes color from Grey to Blue. |   Alternative Scenario:  N/A  Exceptions:  N/A  Relationship:  Sync Armband | | | | |

##### **Manage Armband**



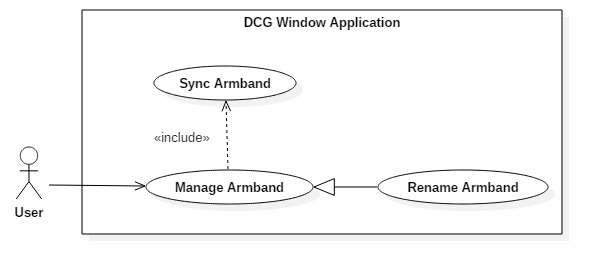
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| USE CASE - UC004 – Test Armband | | | | |
| Use-case No. | UC004 | **Use-case Version** | | 1.0 |
| Use-case Name | Manage Armband | | | |
| Author | Đinh Bảo Trân | | | |
| Date | 2016/05/30 | **Priority** | Normal | |
| Actor:  User  Summary:   * + This use case allow user to start manage armband.   Goal:   * + The window application successfully shows Manager Screen.   Triggers:   * + User clicks “Continue” button in Test Screen.   Preconditions:   * + User has moved to Test Screen.   + Armband is ON   + Armband is synced to window application.   Post Conditions:  N/A  Main Success Scenario:   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click **“Continue”** buttonin **Test Screen** | The Ready Screen is shown. | | 2 | Click “Start” button in Ready Screen | The Manager Screen is shown |   Alternative Scenario:  N/A  Exceptions:  N/A  Relationship:  Sync Armband | | | | |

###### Ping Armband



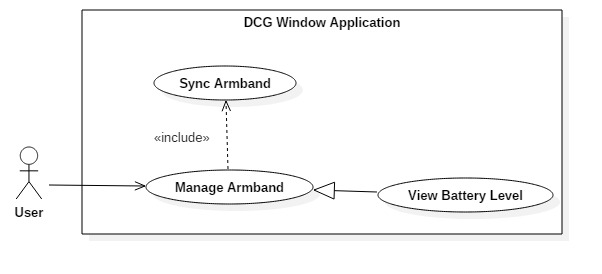
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| USE CASE - UC004.1 – Ping Armband | | | | |
| Use-case No. | UC004.1 | **Use-case Version** | | 1.0 |
| Use-case Name | Test Armband | | | |
| Author | Đinh Bảo Trân | | | |
| Date | 2016/05/30 | **Priority** | Normal | |
| Actor:  User  Summary:   * + This use case allow user to make armband vibrate.   Goal:   * + The window application successfully send signal to vibrate armband.   Triggers:   * + User clicks on “STATUS” tab.   Preconditions:   * + User has moved to Manager Screen.   + Armband is ON   + Armband is synced to window application.   Post Conditions:  N/A  Main Success Scenario:   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click on “**STATUS**” tab | The STATUS Tab is shown. | | 2 | Click “Ping” button |  |   Alternative Scenario:  N/A  Exceptions:  N/A  Relationship:  Sync Armband | | | | |

###### Rename Armband



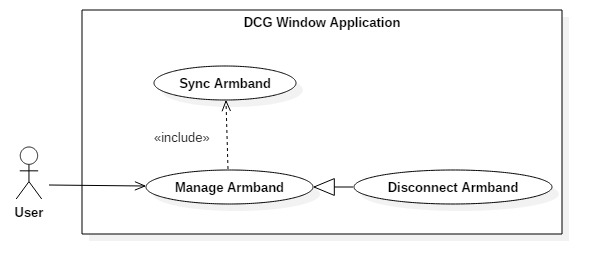
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| USE CASE - UC004.2 – Rename Armband | | | | |
| Use-case No. | UC004.2 | **Use-case Version** | | 1.0 |
| Use-case Name | Rename Armband | | | |
| Author | Đinh Bảo Trân | | | |
| Date | 2016/05/30 | **Priority** | Normal | |
| Actor:  User  Summary:   * + This use case allow user to rename armband on application.   Goal:   * + The window application successfully rename armband.   Triggers:   * + User clicks on “DETAIL” tab.   Preconditions:   * + User has moved to Manager Screen.   + Armband is ON   + Armband is synced to window application.   Post Conditions:  N/A  Main Success Scenario:   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click on “**DETAIL**” tab | The DETAIL Tab is shown. | | 2 | Enter new name in text field  Click “Rename” button | Change the name of armband. |   Alternative Scenario:  N/A  Exceptions:  N/A  Relationship:  Sync Armband | | | | |

###### View Battery Level



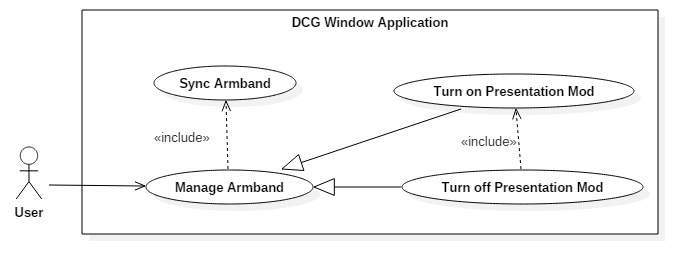
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| USE CASE - UC004.3 – View Battery Level | | | | |
| Use-case No. | UC004.3 | **Use-case Version** | | 1.0 |
| Use-case Name | View Battery Level | | | |
| Author | Đinh Bảo Trân | | | |
| Date | 2016/05/30 | **Priority** | Normal | |
| Actor:  User  Summary:   * + This use case allow user to view armband battery level.   Goal:   * + The window application successfully gets battery level from armband.   Triggers:  Preconditions:   * + User has moved to Manager Screen.   + Armband is ON   + Armband is synced to window application.   Post Conditions:  N/A  Main Success Scenario:   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Look at the top-left side of the Manager Screen | Battery Icon changes its level according to battery level of armband. |   Alternative Scenario:  N/A  Exceptions:  N/A  Relationship:  Sync Armband | | | | |

###### Disconnect Armband



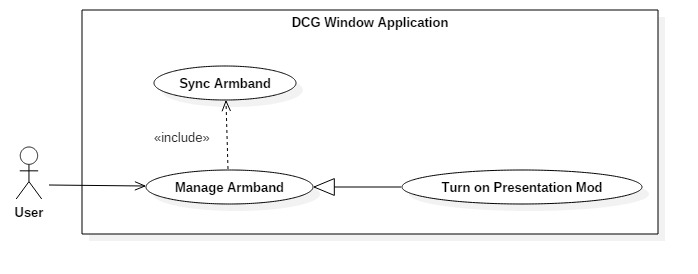
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| USE CASE - UC004.4 – Disconnect Armband | | | | |
| Use-case No. | UC004.4 | **Use-case Version** | | 1.0 |
| Use-case Name | Disconnect Armband | | | |
| Author | Đinh Bảo Trân | | | |
| Date | 2016/05/30 | **Priority** | Normal | |
| Actor:  User  Summary:   * + This use case allow user to disconnect with armband.   Goal:   * + The window application successfully disconnect with armband.   Triggers:   * + User clicks “STATUS” tab.   Preconditions:   * + User has moved to Manager Screen.   + Armband is ON   + Armband is synced to window application.   Post Conditions:   * + If user is controlling presentation, application will execute function “Turn off Presentation Mod” automatically.   Main Success Scenario:   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click “**STATUS**” tab | The STATUS Tab is shown | | 2 | Click on “Disconnect” button | 1. Change Connection Status from *Connected* to *Disconnected*  2. Change Checked Icon color from Green to Grey |   Alternative Scenario:  N/A  Exceptions:  N/A  Relationship:  Sync Armband | | | | |

###### Turn off Presentation Mod



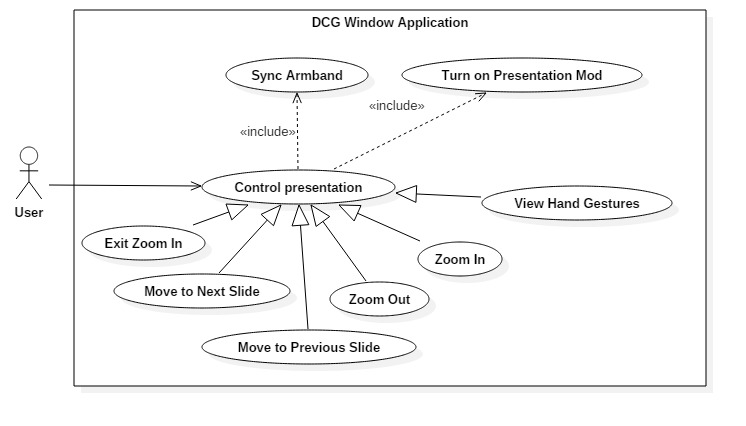
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| USE CASE - UC004.5 – Turn off Presentation Mod | | | | |
| Use-case No. | UC004.5 | **Use-case Version** | | 1.0 |
| Use-case Name | Turn off Presentation Mod | | | |
| Author | Đinh Bảo Trân | | | |
| Date | 2016/05/30 | **Priority** | Normal | |
| Actor:  User  Summary:   * + This use case allow user to turn off presentation mod.   Goal:   * + The window application successfully turns off presentation control.   Triggers:   * + Click “DETAIL” tab.   Preconditions:   * + User has moved to Manager Screen.   + Armband is ON   + Armband is synced to window application.   + Presentation Mod is ON   Post Conditions:  N/A  Main Success Scenario:   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click “**DETAIL**” tab | The DETAIL Tab is shown | | 2 | Click on “Deactivate” button | 1. The Presentation Screen disappears  2. Change “Deactivate” button to “Activate” |   Alternative Scenario:  N/A  Exceptions:  N/A  Relationship:  Sync Armband, Turn on Presentation Mod | | | | |

###### Turn on Presentation Mod



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| USE CASE - UC004.6 – Turn on Presentation Mod | | | | |
| Use-case No. | UC004.6 | **Use-case Version** | | 1.0 |
| Use-case Name | Turn on Presentation Mod | | | |
| Author | Đinh Bảo Trân | | | |
| Date | 2016/05/30 | **Priority** | Normal | |
| Actor:  User  Summary:   * + This use case allow user to turn on presentation control mod.   Goal:   * + The window application successfully turns on presentation control.   Triggers:   * + User clicks “DETAIL” tab.   Preconditions:   * + User has moved to Manager Screen.   + Armband is ON   + Armband is synced to window application.   + Microsoft PowerPoint is opened.   Post Conditions:  N/A  Main Success Scenario:   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click “**DETAIL**” tab | The DETAIL Tab is shown | | 2 | Click on “Activate” button | 1. The Presentation Screen is shown  2. Change “Activate” button to “Deactivate” |   Alternative Scenario:  N/A  Exceptions:   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 2 | MicrosoftPowerPoint is not opened | Show message *“Please Run PowerPoint Firstly”* under “**Activate**” button |   Relationship:  Sync Armband | | | | |

##### **Control Presentation**



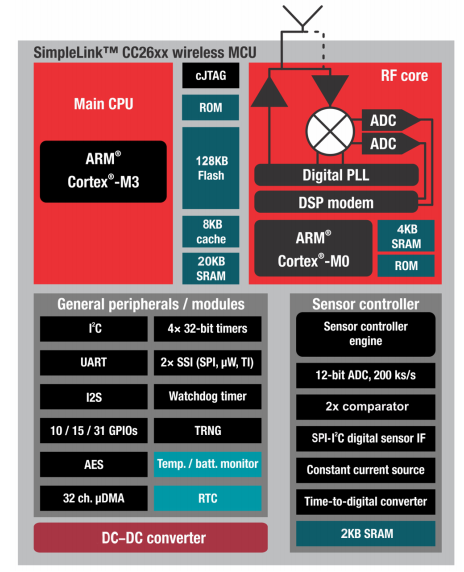
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| USE CASE - UC005 – Control Presentation | | | | |
| Use-case No. | UC005 | **Use-case Version** | | 1.0 |
| Use-case Name | Control Presentation | | | |
| Author | Đinh Bảo Trân | | | |
| Date | 2016/05/30 | **Priority** | Normal | |
| Actor:  User  Summary:   * + This use case allow user to control PowerPoint presentation by hand gestures.   Goal:   * + The window application successfully gets data from armband, analyses data, and maps each gesture’s data with correct function.   Triggers:  Preconditions:   * + User has turned on presentation mod   + Armband is ON   + Armband is synced to window application.   + Presentation Screen has been shown   Post Conditions:  N/A  Main Success Scenario:   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Perform hand gestures to control PowerPoint presentation | 1. Presentation Screen shows the image of the gesture that map with user’s gesture.  2. PowerPoint presentation will “Move to Next Slide”, “Move to Previous Slide”, “Zoom In”, “Zoom Out”, and “Exit Zoom In” according user gestures |   Alternative Scenario:  N/A  Exceptions:  N/A  Relationship:  Sync Armband, Turn on Presentation Mod | | | | |

#### 3.2.1.2 Hardware Requirements

##### Microcontroller



The **CC2650** device contains a 32-bit ARM Cortex-M3 processor that runs at 48 MHz as the main processor and a rich peripheral feature set that includes a unique ultralow power sensor controller. This sensor controller is ideal for interfacing external sensors and for collecting analog and digital data autonomously while the rest of the system is in sleep mode. Thus, the **CC2650** device is ideal for applications within a whole range of products including industrial, consumer electronics, and medical.



The **Bluetooth Low Energy controller** and the IEEE 802.15.4 MAC are embedded into ROM and are partly running on a separate ARM Cortex-M0 processor. This architecture improves overall system performance and power consumption and frees up flash memory for the application.

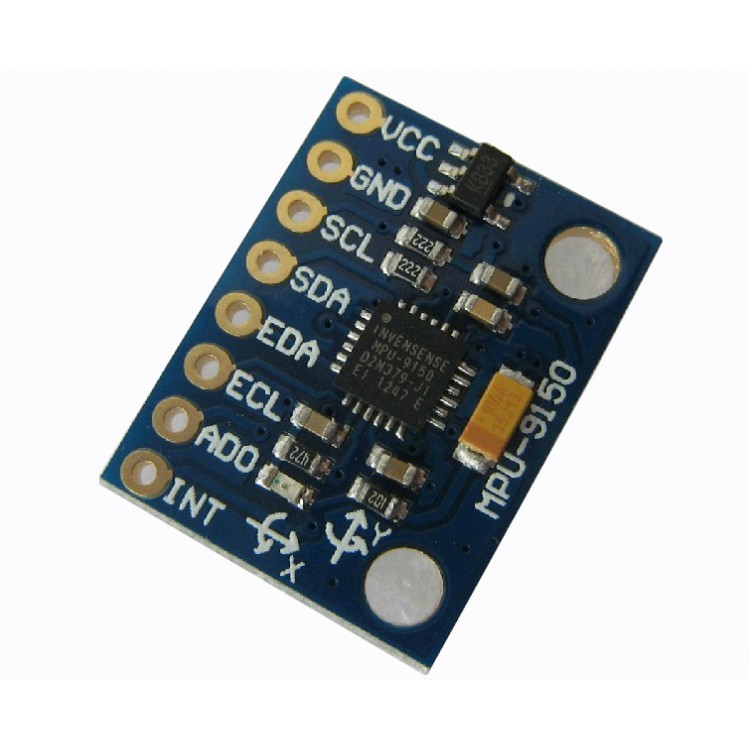
Microcontroller:

* Powerful ARM® Cortex®-M3
* EEMBC CoreMark® Score: 142
* Up to 48-MHz Clock Speed
* 128KB of In-System Programmable Flash
* 8KB of SRAM for Cache
* 20KB of Ultralow-Leakage SRAM
* 2-Pin cJTAG and JTAG Debugging
* Supports Over-The-Air Upgrade (OTA)

Ultralow-Power Sensor Controller:

* Can Run Autonomous From the Rest of the System.
* 16-Bit Architecture.
* 2KB of Ultralow-Leakage SRAM for Code and Data.

##### MPU9150 module



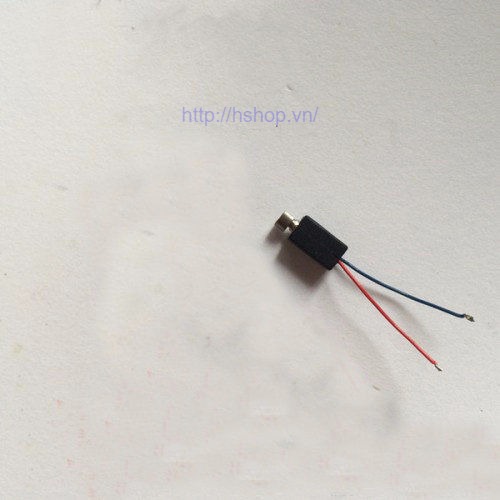
**MPU-9150** is the world’s first nine-axis Motion Tracking components designed for low-power, low-cost, high-performance consumer electronics products as smart phones, tablet, wearable sensors and other devices.

**GY-9150** Module integrates **MPU6050** (three-axis gyroscope + tri-axial accelerometer) and **AK8975** three-axis digital electronic compass on one chip, simplifying the hardware circuit, reducing the PCB size. In addition, it contains a DMP (Digital Motion Processor) that can process 9-axis sensor component fusion algorithms and a run-time calibration firmware. GY-9150 Module can help customers eliminate troubles of signals-regulating, combining fusion algorithms technology and factory calibration when choosing different brands of Motion Tracking components.

Feature:

* Operating Voltage: 2.375V - 3.46V
* Internal 16bit ADC 16-Bit Data Output
* Gyro Range: ±250 ±500 ±1000 ±2000°/s
* Acceleration Range: ±2g ±4g ±8g ±16g
* Communication: I2C Protocol Standard
* Digital Motion Processing™ (DMP™) Engine
* Embedded Calibration Algorithms
* Digital Output Temperature Sensor
* 3-Axis Digital Compass AK8975
* Internal 13bit ADC for the compass

##### Vibration Motor



The small, convenient motor, can be found in mobile phone.

* Operating voltage: 1.8 – 6V (best at 3V).
* Speed: 6000 rpm at 3V.
* Diameter of the pendulum: 5mm
* Engine size: 4.7 \* 9mm
* Total size: 12mm

##### Electrode and Cable

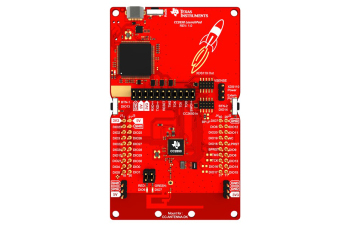


These cables feature 3.5mm Audio plug connectors with snap style receptacles. 24 inch length leads. These are intended to replace the cables that came with your sensor.

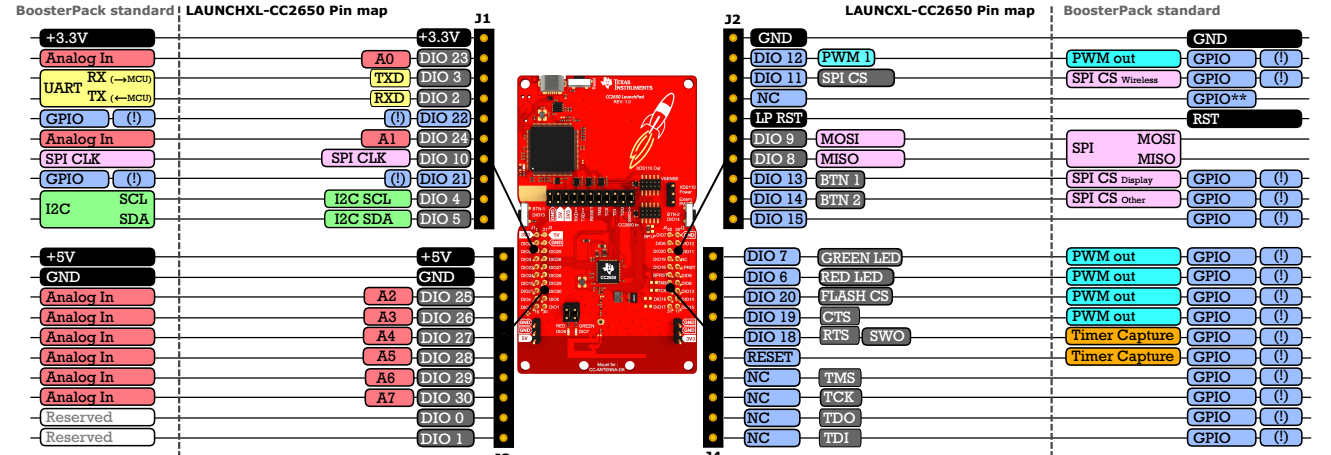
Features:

* 24 inch cable leads
* 3.5mm Audio plug style connection

##### CC2650 Launchpad XL



The CC2650 LaunchPad kit supports programming and debugging from Code Composer Studio™ and IAR Embedded Workbench® integrated development environments (IDEs). Save time and production cost by quickly prototyping your system with the LaunchPad kit I/O connectors that allow for quick interfacing to a variety of evaluation modules (EVMs) and BoosterPack™ plug-in modules.



Feature:

* Connect your LaunchPad kit to the cloud with Bluetooth Smart on your smartphone.
* Access all I/O signals with the BoosterPack plug-in modules I/O interface.
* On-board emulator with a USB serial port gets you started with instant code development.
* Upgrade the LaunchPad kit firmware over-the-air from the SimpleLink Starter app.
* Code Composer Studio IDE full license included.
* Includes 1 MB external Flash for support of multiple protocols or storing user data.
* Up to 5 dBm output power.
* Optional Sharp® Memory LCD BoosterPack plug-in module support.

#### 3.2.1.3 Communication Protocol

* Use Bluetooth 4.0 low energy protocol for communication between the armbands with the window application.
* Use I2C protocol for communication between CC2650 and MPU9150.

### 3.2.2 Non-functional Requirement

#### Usability

##### Training and Learning

* The system is intuitive and easy to use. The basic skills needed to surf and basic interact with the window application, basically is clicking to choose.
* System also provides user GUI with instruction step by step follow with picture and videos.

##### Graphic User Interfaces

* All the texts, labels, alerts and messages will be written in English.

#### Reliability

##### Availability

The system is available 24 \* 7, for 365 day year, less the downtime due to maintain of the system.

##### Mean time between failures

10000 hour.

##### Mean time to repair

Support team will provide solution handle the error and solve it in short time.

##### Accuracy

In desire, 90% of user gesture can be recognize precisely.

#### Performance

##### Response time

* The maximum response time for system to realize user gesture is 1 minute.

##### Throughput

* The arm band can send 10 EMG data per second.

##### Capacity

* System can take 4 gesture as the moment.

##### Supportability

* Phone call technical support open at 6am – 5pm from Monday to Sunday.

##### Security

N/A.

##### Maintainability

* System is divided into modules.
* When a module of a function is down, it is easy to take it down to fix without impact other functions or component.

##### Portability

User can stay away from their laptop over 10m while performing their presentation. Bluetooth Low Energy also provide power consumption techniques 0.01 to 0.5 W (depending on use case).

## 3.3 Conceptual Diagram

