UVDose

Version: 23/09/19

Notes:

- For the purposes of this document, we consider "code" or "software" to refer to programs which runs on a PC and/or on an embedded system.
- While we try our best to allow students to have the freedom to make their own design choices, it is not possible for the teaching staff to support every possible development system.
- **This specification is a *living* document!** Changes will occur and will be released throughout the semester.
- You are permitted to bundle outside orders with other teams in order to reduce the overall cost of shipping, as long as you take into account the UQ Academic Integrity and Student Conduct policy.
- **You must follow all other requirements outlined in the *TP-STD* standards.**
- Any text which applies to ENGG2800 only will be highlighted in green.
- Any text which applies to ENGG3800 only will be highlighted in blue.

Introduction and background

Exposure to ultraviolet (UV) light from the sun is believed to cause the majority of skin cancers in Australia. For this project, you will be developing a UV dosimeter, which would allow people to become more aware of their UV exposure over time (also known as dosage) in order to help them to better understand their risk levels.

In the context of exposure limits, the standard ISO/CIE 17166:2019 (*Erythema reference action spectrum and standard erythema dose*) splits UV light into three bands: UVA, UVB and UVC. The UVA band encompasses light with a wavelength in the range of 400nm to 315nm. Similarly, the UVB band covers the wavelengths of 315nm and 280nm and the UVC band between 280nm to 100nm.

Although UVC is considered to be the most dangerous of the three as it can be directly ionising (much like alpha, beta, gamma radiation), it is absorbed by the ozone layer and atmosphere to the point where people on the surface of the earth receive negligible UVC exposure from the sun (although other sources of UVC exist, such as the electrical arc when welding). Light with a wavelength below 100nm is also considered to be UV light, but it is practically non-existent from the sun as it is readily absorbed by atmospheric oxygen. UVA and UVB are the bands which are associated with sun tan, sunburn and skin cancer. The exact mechanism of the damage with respect to skin cancer is beyond the scope of this project.

The most common measure of UV intensity in the context of human exposure is the UV Index, which is often present in weather forecasts and in educational materials. The World Health Organisation publishes the following figure showing risk levels:

UV exposure dosage calculation

Calculation of dose is to be performed via the UV index measurement. 1 Standard Erythemal Dose (SED) is defined as 100 $J.m^{-2}$. A UV index of 1 is defined as UV light exposure of 25 $mW.m^{-2}$. The UV index increases linearly with exposure rate, so a UV index of 2 is equivalent to 50 $mW.m^{-2}$, and so on. Given these rules, and given that that 1 W is defined as 1 $J.s^{-1}$, the following formula can then be derived to compute the total accumulated dose, *S* (in SED) given constant exposure at UV index, *i* (unitless) for the time period *t* (in sec):

$$S = \frac{t \times 0.025i}{100}$$

So for a person who is exposed to radiant sunlight with a UV index of 4 for 180 seconds, the dose in SED would be calculated as follows:

$$S = \frac{180 \times 0.025 \times 4}{100} = 0.18 \text{ SED}$$

Your device should measure UV index to at least 1 decimal place. SED must be computed and saved to at least 3 decimal places.

For the purposes of testing on demo day, we will be requiring you to increase the dosage rate of your device using the following formula instead:

$$S = t \times 0.25i$$

Device behaviour

Your dosimeter device is required to be in the form of a lanyard attachment/badge. At a fixed interval, the device will take a UV index reading, update the internal statistics and display the result to the user; this cycle is hereafter referred to as a *measurement*. After the measurement is complete, the device must enter a low power sleep mode in order to conserve battery. Your device must perform all of these tasks in less than 10 seconds. To calculate the dosage for the previous sleep period, your device must perform a measurement, and then use the formula above, assuming that the UV index was constant and the same as your measurement during the sleep period.

For ENGG2800, after the measurement is complete the device must sleep for 60 seconds. Your device must store the last UV index reading and the total SED to non-volatile memory, so that it is maintained even if the battery is removed from the device.

For ENGG3800, after the measurement is complete the device must sleep until 60 seconds after the last wakeup time. For example, if the device last woke at 11:53:21 to take a measurement, the next measurement taken must be at 11:54:21. An error of +/- 1 second is acceptable. Your device must store the measurements on a microSD card, hereafter referred to as "non-volatile memory". Your device must store files in the following directory structure:

uv/<YYYY>-<MM>-<DD>/<hh>-<mm>/<ss>.csv where <YYYY> is the current year, <MM> is the current month, <DD> is the current day of the month, <hh> is the current hour in 24 hour time, <mm> is the current minute of the hour and <ss> is the current seconds of the minute. These paths are not case sensitive.

The file must be an ASCII formatted CSV file with a single line; this line will contain the measured UV index followed by a comma, and then the SED up to and including this time point.

Your device must also store the most recent UV index measurement as uv/latest-index.txt and the most recent SED calculation as uv/latest-sed.txt (both as ASCII formatted numbers).

You may store other files on the SD card as necessary.

Controls and user interface

Your device must use the following e-ink display:
https://www.waveshare.com/wiki/2.9inch_e-Paper_Module (ETSG: 28-11-03)

When first powered on, the device must display a busy indicator, showing the user that it is taking an initial measurement (for example, the text "Busy…" or "Starting…"). At all other times, including while the device is in sleep mode, the display must show:

- The most recent UV index measurement as an integer.
- A bar graph with a single element spanning the entire width of the display showing the most recent UV index reading from 0 to 11+
- The Fitzpatrick skin type of the user (I to VI), or an indication that this has not been specified yet.
- The specified MED of the user, and a clear indication if the user has exceeded 40% of their MED in accumulated dosage.
- A clearly visible alert indicating that sun protection is necessary if the UV index is 1 or above. (note that in real life, a UV index of 3 should be used here. This is just to make testing easier)
- The accumulated dosage for the user as a number (in SED). If the SD card is re-inserted into the device, the device must recompute the total SED using all of the data available on the SD card. It is acceptable if this takes more time than the allowed 10 seconds. Recomputation starts from the oldest data point.

- For ENGG2800, the total time that the device has been powered on, including any sleep time. Format is hh:mm:ss.
- The time and date of the most recent UV index measurement (in the format YYYY/MM/DD hh:mm, where hours are in 24 hour time)
- The total number of logged data points currently saved on the SD card, or an indication that the SD card was not inserted at the time of the last measurement.

If a previous measurement is present in non-volatile memory, the display must not show the busy indicator, but instead show the last collected data.

For ENGG3800, display changes must be performed using the partial update functionality of the display; a full refresh should only occur once every 5 measurements. This counter does not need to persist using non-volatile memory.

Your device must have a switch to change into a mode which causes the device to stop taking measurements, hereafter referred to as "pause mode". However, when pause mode is active, the device must still keep accurate time. If pause mode is activated while a measurement is underway, the measurement must complete and not be aborted. When this is turned back on, the device must immediately take a measurement and begin the normal measurement cycle again. When exiting pause mode, the device does not need to compute the UV dosage over the extended pause sleep period; if the time since the last measurement is greater than 1 minute, you should just compute the dosage over the last minute and ignore the remaining off time. This measurement must be immediately taken, as opposed to waiting for the minute schedule. While this switch is off, the device must not perform any operations on the SD card (except for the case where it is switched off during a measurement; all operations must cease after the measurement completes).

Although the normal acquisition cycle is 1 minute, your device must have a switch which forces the device to continuously acquire measurements (ie every 1 to 2 seconds or so), hereafter referred to as "force mode". This is purely for testing and demo purposes. The device must not enter sleep mode after each measurement when in force mode, even for a very short period. If the force mode and pause mode switches are activated, the pause mode shall take precedence and the device must not perform any measurements and instead stay in sleep mode. The device must still compute dosage correctly; if the time between measurements is only a few seconds, it must not assume that the dosage was incurred over the usual 1 minute sleep period, and instead use the actual time between measurements.

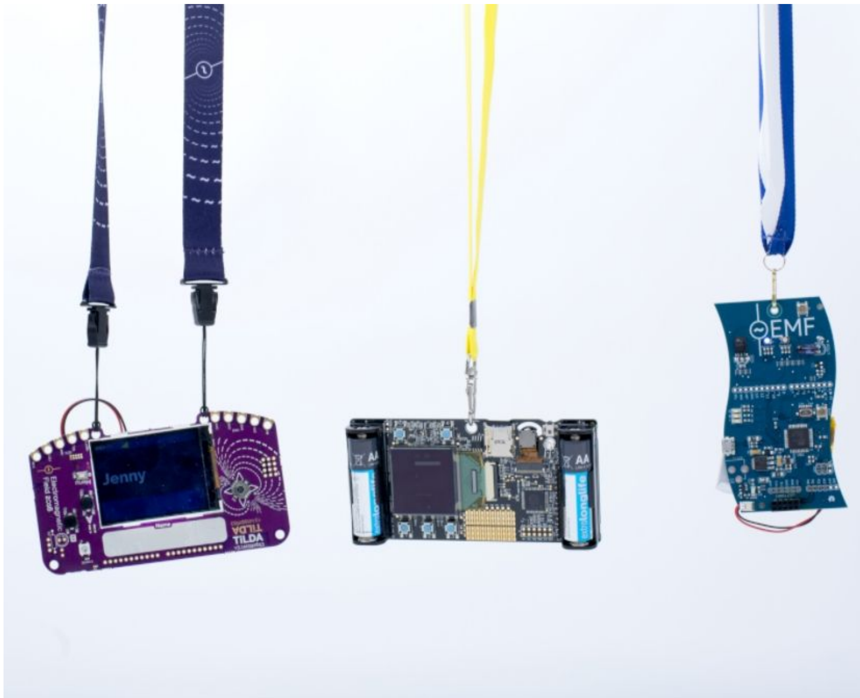
Instead of using two switches to implement the various modes, you may instead use a single SP3T switch.

Your device may wake up during the sleep period for very short amounts of time (eg 100ms or so every few seconds) if it needs to process some interrupts or perform some other management tasks. This must not affect the expected measurement cycle timing.

Your device must have a microUSB socket and associated hardware to communicate with the PC via a USB serial port (for example, using the UMFT234XF that you have been provided).

Your device must have a microSD card socket for logging purposes.

At least one 5mm hole must be on the PCB for the purposes of hanging the device from a lanyard. The hole should be centered along one side of the device; it must not be in a corner. The center can be defined as either the geometric center, or the centre of mass (so that when hanging the device on a lanyard, the bottom edge of the device remains horizontal under the influence of gravity like in the image below) .



Your device must have a length and width of less than 100mm.

A green and red LED must be present which quickly switch on and off at specified events. The red must blink immediately when the device wakes up from sleep, and the green must blink as the very last step before going to sleep. An additional blue LED must be present which is switched on while the device is accessing the SD card.

If the device wakes up for a reason that isn't a measurement, the red and green LEDs must not be activated.

You may use a multicolour LED for this.

All LEDs must not be obscured on your PCB, but they may be on the reverse side of the device so that they are not visible when the badge is worn normally.

All mounting must be done using the PCB only; no 3d printing is allowed.

The UV sensor must not be obstructed by any component, i.e. it must be visible to ambient light.

The UV sensor must also be positioned on the same side as the display as to obtain accurate UV index readings whilst also having any warnings / statistics clearly visible.

Power supply and consumption

For ENGG2800, you are required to power your device using 2-4 AA or AAA cells. Your device does not need to maintain an accurate time while the batteries are disconnected, but it does require accurate time to be kept during device sleep. During sleep mode, your system must consume less than an average of 1mA from the battery regulator. The batteries can either be on-board using PCB mount battery holders, or using an off-board battery holder with wires of reasonable length (~500mm) to allow the user to put the battery in their pocket.

For ENGG3800, you **must** use the following LiPo: https://core-electronics.com.au/polymer-lithium-ion-battery-1000mah-38458.html (ETSG: 28-11-05). Under no circumstances is any replacement part number acceptable. Your device must maintain accurate time while the LiPo battery is disconnected; as such, you are permitted to use a single coin cell battery no bigger than a CR2032 (the coin cell is not intended removed from your device during testing). Note that this coin cell must only be used to retain backup SRAM and/or allow a clock to continue running; it must not be used to power the MCU or any other circuits. All of the sleep current must be drawn from the LiPo battery. Your device must also charge the LiPo battery using a Lithium battery charger IC while a micro USB cable is connected to the device. During sleep mode, your system must consume less than an average of 200uA from the battery. The battery must be securely mounted to your PCB, and you must use the appropriate connector to electrically connect it to your PCB; soldering the bare wires is unacceptable, as is removing and replacing the existing connector on the battery.

PC software

You must develop PC software that can communicate with your device over a USB serial port. You must provide a mechanism for the user to select which serial port the device is connected to, and you must also provide connect/disconnect buttons, and some indicator of whether the device is currently connected or not. The software interface must not be unresponsive while the device is connecting. For ENGG2800, your device must connect within 1 minute of clicking the connect button. For ENGG3800, your device must connect within 10 seconds of clicking the connect button.

The device must not enter sleep mode while the device is connected to the PC software. Instead, the device must repeatedly perform one UV index reading per second and send the reading to the PC software. These fast measurements must not be stored in non-volatile memory, and they must not be used to update the dosage calculation. This UV index reading must be shown in the PC software interface.

If the device is plugged in to the PC but the software is not yet connected, or the device is plugged in to a USB charger, it must behave normally as if it was not plugged in (aside from the battery being able to charge).

Your software must contain an interface that allows the user to determine their Fitzpatrick skin type using the exact ARPANSA quiz wording: https://www.arpansa.gov.au/sites/default/files/legacy/pubs/RadiationProtection/FitzpatrickSkinType.pdf (there are a few spelling mistakes in this document, you are welcome to correct them)

The quiz interface must not require the user to directly enter the numbers as answers; instead it must use radio buttons, dropdown boxes, etc. However, if the user already knows their Fitzpatrick skin type, there should be an override which allows the user to enter the numeral directly (or select it from a list). A user must also be allowed to retake the quiz without restarting the GUI.

Using the Fitzpatrick skin type, the approximate MED should be calculated. By default, the MED of a particular skin type shall be defined as (see https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0190233):

Skin type	MED (in units of SED)
I	150
II	220
III	290
IV	370
V	440
VI	440

However, if the user is aware of their own MED, an override must be present which allows them to edit the number (as a positive integer). You should reject a number larger than 440.

Once the skin type of the user is known, the MED is known and the device is connected, there must be a button which sends these details to the device to be stored in non-volatile memory. This button must not be clickable if the previous conditions are not satisfied.

For ENGG2800, your software must have a mechanism to reset the accumulated SED stored in the non-volatile memory on the device to zero.

For ENGG3800, your software must have a mechanism to synchronise the current date and time on the device to the system clock on the PC.

Bill of Materials

You will have $150 AUD provided to you for development purposes at ETSG. Your product Bill of Materials (BOM) must be less than or equal to $100 AUD. You do not need to include the cost of a USB cable in your BOM.

For ENGG2800, you do not need to include the cost of batteries in your BOM.

For ENGG3800, you do not need to include the cost of the microSD card in the BOM, but you must provide one for testing purposes. You must include the cost of all batteries. For demos, you are also welcome (and encouraged) to provide a microSD card reader that you have tested with your card; it will be returned to you after your demo has been completed. This microSD card reader does not need to be added to your BOM.