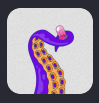
classpert



Making Embedded Systems

Final Project Evaluation

Red Jellies Cohort

Calor
by David Slik

Reviewers

Erin Kennedy

Elecia White

Contents

1 Overview

The objective of this document is to assess and give you a high-level feedback on your final project. Completing it and receiving a passing grade is a pre-requisite for the certificate of course conclusion to be issued. Your project was reviewed and graded by mentor **Erin Kennedy** and instructor **Elecia White**.

1.1 Project Details

Project Title

`Calor`

Student Name

`David Slik`

Enrollment ID

`MES1C23`

Deliverables	Links
Report	open
</> Code	open
▶ Video	open

2 Final Evaluation

For each criteria, a score was given according to the grading rubric (see appendix). The total achievable score was **24**, of which **18** are `common credits` and **6** are `bonus credits`.

Credits exceeding **18** are not considered for the final grade calculation.

Your final grade was calculated using the formula

$$grade = \frac{applicable\_score \times 100}{18}$$

2.1 Final Grade

Your final grade: `100` (`PASS`)

Common score: `17.8/18`
Bonus score: `6/6`
Applicable score: `18`

$$grade = \frac{applicable\_score \times 100}{18} = \frac{18 * 100}{18} = 100$$

Criteria	Score	Notes
Project meets minimum project goals	** breakdown3.2	
Completeness of deliverables	**	
Clear intentions and working code	**	
Reusing code	**	
Originality and scope of goals	**	Creative use of NFC to wake the device, and using coloured light to encode data
Self-assessment (mentor category only)	**	The estimation was very close but give yourself more credit in the future
Power analysis, firmware update, or system profiling	**	
Version control was used	**	Extra points for using Issues / Projects to keep track of the upcoming features
	Total: **	

*Grades have been hidden

2.2 Reviewers' Feedback

2.2.1 Mentor Comments

by Erin Kennedy

Starting the project with the requirements from the user perspective allowed for a creative solution to emerge. Precision sensing for long temporal resolutions is commonly burdened by deployment and retrieval — this concept addresses this in a way that's only possible with great embedded systems engineering. The documentation for the project was a joy to read.

While there may not be quick modifications to improve the prototype at this point, the next steps listed in the Github repository issues make sense to move this towards a functional minimal viable prototype (MVP).

A calibration target may be helpful in testing the number of colours that can be reliably detected in different lighting conditions:

Source

I am curious to know how 32 colour-limit was reached (as mentioned in the report), and what combination of brightnesses would achieve this resolution. If all 3 colours are combined as 100%, 50%, 0% brightness, this would be $3\hat{3}$ (27) colours, which is seen in this optical_output_table with 27 values.

Clever use of capturing at a higher frame rate to both reduce light and increase resolution for data being transferred. Will rolling shutter be a concern? Further work on validating the transmission rate would be interesting to read about.

The reduction in current consumption diagram is excellent for visualizing the changes that were made. It is likely that future cohorts could benefit from this example.

Before bringing this to an interview (assuming a pitch style interview for financing), it would be helpful to have the testing framework complete. This would address reducing technical risk by showing the MVP is functional, and by already being ready to deploy more sensors.

For a pilot deployment, a potential set of early adopter users may be hobby farmers as they tend to be more receptive to new ideas.

Design-wise, it would be worth looking into the products MG Chemicals offers for resin. They are responsive in their support email answering questions about which product they would recommend for a specific application. Be careful with the button component when potting.

Excited to see how far you will take this. Incredibly curious to hear if there are correlations between temperature and soil moisture. Congrats on the work that went into making this excellent project!

2.2.2 Instructor Comments

by Elecia White

This is fantastic. I am amused that the worst score you received was for self-assessment. You really went above and beyond the requirements (and scope) of the final project. Thank you for sharing Calor with us.

The GPS with the phone picture is an ingenious solution, saving the BOM cost of a GPS while having the user do something they'd probably already be doing. It is very much an obvious "why didn't I think of that?" solution that I've never seen done.

It does leave you open to an issue with time synchronization between devices. If a sensor is in a warm place, its clock will go a little faster than a cool sensor. For an hour, that doesn't matter. For a month, it may cause a significant discrepancy which will cause issues with your data. Happily, the solution is in your processor already: a real time clock (RTC). Your phone app will need to set the RTC's clock (or record its current value with the GPS which can then be used to calculate an absolute time. Ahh, I see that this is already on your Future Enhancements, excellent.

I appreciate how much of the system can be tested. Your command line is great. Though, I wish I'd talked you into a command table because it is more extensible.

I'm surprised the optical signaling is so slow. 120 baud is less than I'd expect of a white LED. Your additional colors should make it faster. I don't see any specifics about the Electric Imp BlinkUp speeds (it does some auto-baud as the firmware update runs at different speeds depending on the phone). Take a look into that to see if it gives you ideas for faster information transfer. If you can get to 500 baud, the initial setup will feel like magic. Though... how long will it take to get a month's worth of data from the device? Thank you for purchasing Calor temperature sensors.

I look forward to this being a reality. I've really enjoyed having you in the course. Your feedback has been wonderful and, as Erin said, your final project documentation is a joy to read.

3 Appendix

3.1 Grading Rubric

Criteria	Score		
	1 - Needs Improvement	2 - Meets Expectation	3 - Exceeds Expectation
Project meets minimum project goals	• Any project goals are not met	• All project goals are met. The state machine may be basic	• Additional sensors, actuators • Well documented and implented state machine • Comprehensive command line on serial port
Completeness of deliverables	• Lacks report, video or code • Report does not cover all sections • Code has obvious errors that would cause it not to compile	• Report covers all sections but some are answered incompletely leaving questions for the reader • Code is readable given the report as a description • Video shows code working.	• Code is readable on its own, without the report • Report addresses each point thoroughly, demonstrating understanding as it related to the course • Video demonstrates the project and is explanatory
Clear intentions and working code	• What the system is supposed to do (based on the report or code) doesn't seem to be what the system does in the video	• The system performs approximately as described in the report and code	• The system performs as described in the report in a manner that is professionally polished • The code shows how it works in a way that is easy for a maitainer to see
Reusing code	• No code was used from other sources or it is unclear what code was used from other sources	• Student code was identified	• Versioning of reused code was included along with a license document that describes the license for the student's code and the reused code as well as shipping implications • Reader is confident they could rebuild the student's system
Originality and scope of goals	• The student did the bare minimum to meet the goals • No originality	• Some areas of interest were noted in the report but they were minor extensions of the existing examples	• The student has gone far beyond the requirements to make something novel and awesome
Self-assessment (mentor category only)	• Self-assessment was significantly different from mentor assessment	• Self assessment was +/- 25% of mentor assessment	• Self-assessment was +/- 10% of mentor assessment
Power analysis, firmware update, or system profiling	• None	• Described	• Described, has graphs, and is accurate
Version control was used	• None or a single commit		• The log shows the project being built, though the messages may be terse but should be descriptive

3.2 Requirements

3.2.1 Project

Features	Delivered	Note
Video turned in	✔	
Link to code	✔	
Report turned in	✔	
Use a Cortex-M processor	✔	RP2040
Button with interrupt	✔	(line) and (line)
Has serial port output	✔	Yes and command handler
Implements a state machine	✔	Yes State machine started from command (line)
Algorithmic piece	✔	Conversion from temperature (uint8_t) to light (RGB LED) (line)
Peripheral 1	✔	WS2812C via PWM
Peripheral 2	✔	TMP117 or LM75 via I2C
Peripheral 3	✔	On board Flash via SPI
Other*	⚠	Unsure if the NFC power input feature was fully implemented, or if it was simulated in software only, or if it was simulated using a potentiometer (in the video)
Uses a HAL*	✔	RP2040
Analysis of Power**	✔	Excellent annotations
Firmware update**	✖	
System Profiling**	✖	
Version control with history	✔	~40 commits to /final-project starting on Jan. 14, 2022

✔ Delivered ⚠ Partially Delivered ✖ Not Delivered * Not Required ** Extra Credit

3.2.2 Report

Features	Delivered	Note
Application Description	✔	Great definition of requirements from user point of view
Hardware Description	✔	
Software Description	✔	
Identify written vs reused code	✔	Yes, mentioned in the code (line) and in the report (line) and license (line)
Architecture Diagrams	✔	Clarifying which rail Vca and Vcb are would be helpful
Build Instructions (HW)	✔	Breakdown of the components in each subsystem is great
Build Instructions (SW)	✔	Great
Debug Instructions	✔	Comprehensive information in other sections assists debugging
Future Plans	✔	
Self Assessment	✔	In the repository

✔ Delivered ⚠ Partially Delivered ✖ Not Delivered