University of Colorado Boulder

Faculty of Electrical, Computer & Energy Engineering

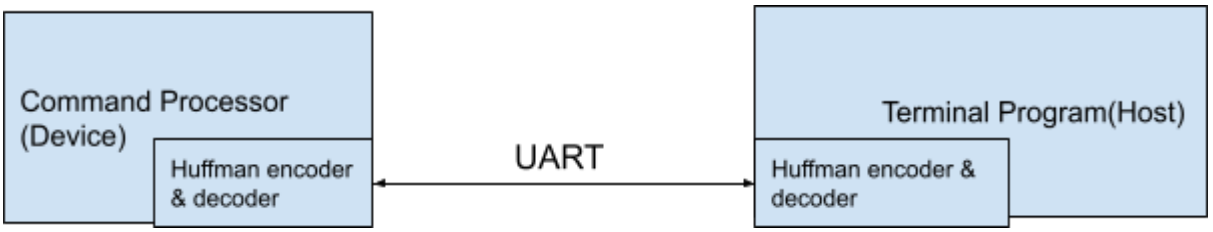
ECEN5813

Written Document: Final Project Proposal

Date	2021/11/30
Prepared by (Name)	Shuran Xu

Q1: What functionality will your project demonstrate ?

The system to be built consists of two parts: a command processor running on the target device and a command terminal application running on the host. The two parties are communicated via UART with data encoded using the Huffman encoding algorithm for better throughput.



Specifically, the terminal program on the host side sends the encoded command supplied by the user to the command processor, and then the command processor performs the corresponding action and sends back the encoded results back to the terminal program, which will display the decoded results on to the user.

As for the supported commands, the command processor primarily focuses on the tone frequency generation.Not only users can input a particular frequency but also can change the frequency at run time by sliding the TSI slider. Additionally, a speaker will be used to produce the sound associated with the current tone frequency.

The commands supported by the command processor are as follows:

Command	Description
Tone at particular frequency	Produce the tone of particular frequency sound via a speaker and display the statistical information regarding the samples collected
TSI-controlled Tone	Produce the tone at a known frequency, and users can change the tone frequency by sliding the TSI slider. The sound is delivered via a speaker and the information regarding the frequency alternation is displayed.
Memory Dump	Display the top 128 bytes of memory of designated memory address. However, this might introduce a program crash.
Help	Display the usage of all supported commands.

The speaker controlled via the PWM module is used to produce the sound associated with the designated tone frequency. It is connected via a resistor to a GPIO pin configured for use for a PWM module.

Q2:What technologies will you use? For those areas we have covered in class, how will you demonstrate deeper knowledge than what we covered in the biweekly homework assignments?

The final project covers a variety of technologies that can be listed as follows:

Technology	Purpose
UART	Communication means between the user and the device.
Circular buffer	Buffering mechanism for UART data transmission.
Command Processing	Main interface for the program.
DMA	Data transmission for DAC
DAC	Generation of waveforms associated with user commands.
GPIO Lines	Connection with an external speaker connected to a breadboard.
Interrupts	Means to initiate requests and respond to device behaviour.
Huffman encoding	Compress communication data via UART
PWM	Generation of output signals to an external speaker
TSI	Touchpad controlling the tone frequency
CUnit	Unit test framework for modular testing
Low power Design	System resource management for power consumption minimization purpose

In class, I learned the basic mechanism of each module and performed hands-on practice through each assignment. Such a learning process helps me gain better understanding of each module both theoretically and practically, but there is no practice yet to develop a comprehensive application system on the microcontroller that is highly optimized in terms of resource usage as well as power efficient. Therefore, in this project, I will not only manipulate a set of peripherals tailored to one comprehensive command processor on the device side, but also design a terminal program on the host for command interaction. Besides, the Huffman encoding & decoding engine will be employed for data compression to increase the UART throughput. Low power mode operations will be used for on-chip peripherals. Additionally, flexible power off and wake up of in-use peripherals will be implemented to minimize the dynamic power consumption. Upon the system design, the final deliverable will be a command processor system that responds to user commands from the terminal program in real time.

In terms of the implementation plan, interrupts of peripherals in use will be employed and polling will not be applied to save power consumption. Inside the interrupt handling, tasks such as data sampling and DMA request generation will be performed. Besides, command tables as well as custom-defined data structures will be used to construct an array of supported commands and the flow of the command processing. Table lookup will be employed on applications such as trigonometric calculations to save CPU cycles. Additionally, the spontaneous management of each peripheral will be exercised so that unused peripherals can be either disabled or operate under low power mode to minimize the overall system power consumption. Furthermore, in case multiple interrupts are arrived at the same time, critical section protection will be performed if any shared resources are present. Interrupt priority configuration will be exercised to ensure the lower-priority interrupts will not prevent the high-priority interrupts from being handled. Moreover, the Huffman encoder and decoder will be implemented so that compressed data is communicated between the two parties and later decoded for processing.

Q3: What do you anticipate needing to learn in order to develop your project? What sources (KL25Z Reference Manual, Internet sites, etc.) do you plan to use to figure out how to do whatever it is you are attempting?

I need to learn at least the following:

Materials to learn	Potential source
waveform signals for common music tones	Internet sites
Various peripheral configuration	the KL25Z reference manual
Huffman encoding implementation	Lecture slides
CUnit setup and practice	Internet sites
Good practices and tricks to optimize the program performance	Textbooks such as the White book

Q4: Does your project require any additional hardware? If so, what will you acquire, and what is your plan for assembly? (Again, my focus is on the software you develop. I am asking about your hardware plans just so I can ensure that whatever you are planning in this area is relatively straightforward.)

Yes, I do require hardware support. Specifically, I will have to connect certain GPIO pins to an external speaker. I will perform the setup in the following way: I will use a breadboard to place the speaker with a resistor. There will be a wire connecting one slot on the breadboard

with the designated GPIO pin, which is used to deliver signals generated from a PWM module.

Q5: Finally, what is your testing strategy for your project? Will you develop automated tests, will you use manual tests, or will you use a mixture of both?

I will perform both the unit testing and integration testing. For unit testing, I will implement a CUnit test program to test modular components, and implement a standalone test program for the Huffman encoding algorithm. And in terms of the integration testing , I will thoroughly test all possible use cases.