EE/CprE/SE 491 WEEKLY REPORT 1

01/20/26 – 2/13/26

Group number: SD26-10

Project title: Open-sourced Radio Microcontroller for Fabrication

Client &/Advisor: Henry Duwe duwe@iastate.edu

Client/Company/Organization: ISU; ECPE

Team Members/Role: Zane Salti, Jackson Doyle, Daniel Pytel, Tyler Bibus, Aiden Han-Lindemyer, Parker Pederson, Akash Gojuru, Seth Klaassen

Weekly Summary:

For this week, the team has merged with the Digital ASIC group. Meeting times have been developed (for the most part) as well as setting up our team communications and infrastructure (Teams, shared google doc folders, git repository structure, etc.). Members engaged with ChipForge over this week as well for learning the relevant tooling and completed access setup. Progress has also been made in scoping the project by a considerable amount.

Past week accomplishments

- Zane Salti: Went through the *Bluetooth Low Energy Primer* (shy of two chapters) and identified key aspects that should be helpful for scoping the project (this isn't to explain how BLE works or how the Link Layer states work but instead to mention aspects that would be helpful for scoping):

Bluetooth Low Energy came into the Bluetooth Core Specification with version 4.0. There is also backwards compatibility. Later versions may be scoped out.

There are 40 channels, each being spaced 2 MHz apart, ranging from 2400 MHz to 2483.5 MHz. A 1 symbol is encoded by an at least +185 kHz shift from the channel center and a 0 symbol is encoded by an at least -185 kHz shift from the center (for the LE 1M mode).

There are 3 "modulation schemes":

	LE 1M	LE 2M	LE Coded
Symbol Rate	1 Ms/s	2 Ms/s	1 Ms/s

Error Detection	Cyclic Redundancy Check (CRC)	Cyclic Redundancy Check (CRC)	Cyclic Redundancy Check (CRC)
Error Correction	None	None	Forward Error Checking (FEC)
Requirement	Mandatory	Optional	Optional

We can scope the project down to just using LE 1M.

There are two types of “advertising”: Legacy Advertising and Extended Advertising.

Legacy Advertising advertises on the three primary channels (37, 38, and 39). Scheduling is intentionally irregular (0-10 ms random value added to the advertising interval) to avoid consistent collisions.

Extended Advertising uses the three primary channels as well as the general purpose channels. It was introduced in version 5.0 and lots of the newer features seem dependent on it. For example, “periodic advertising” (PADVB and PAwR) are dependent on it. Periodic advertising tries to use a deterministic schedule which gives a way for other devices to synchronize their scanning with our advertising. Isochronous communication as well (and so LE Audio) are also dependent on Extended Advertising.

With **sticking to version 4.0 or version 4.x** for now, we can scope out the need for the “Synchronization” and “Isochronous Broadcasting” states in the Link Layer.

We can have a reduced Link Layer state machine as such:

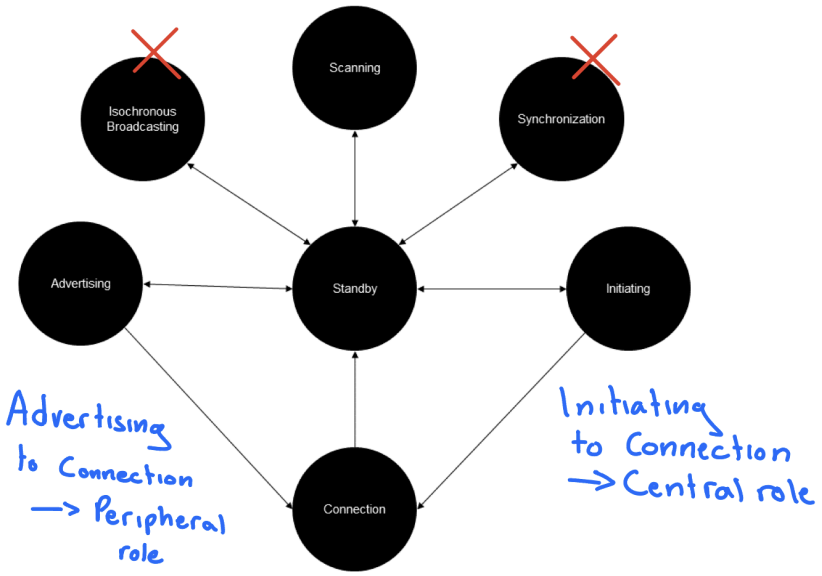
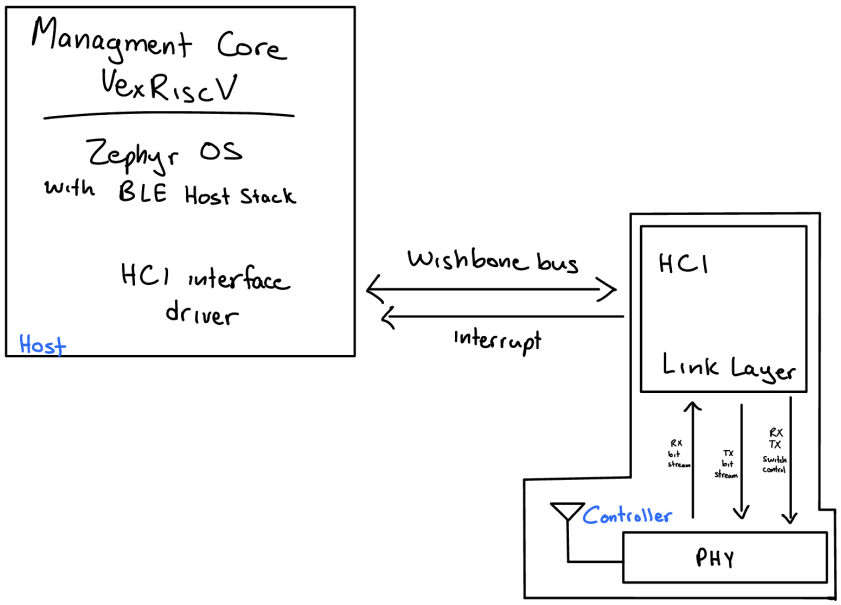


Diagram taken from *The Bluetooth Low Energy Primer* and then annotated

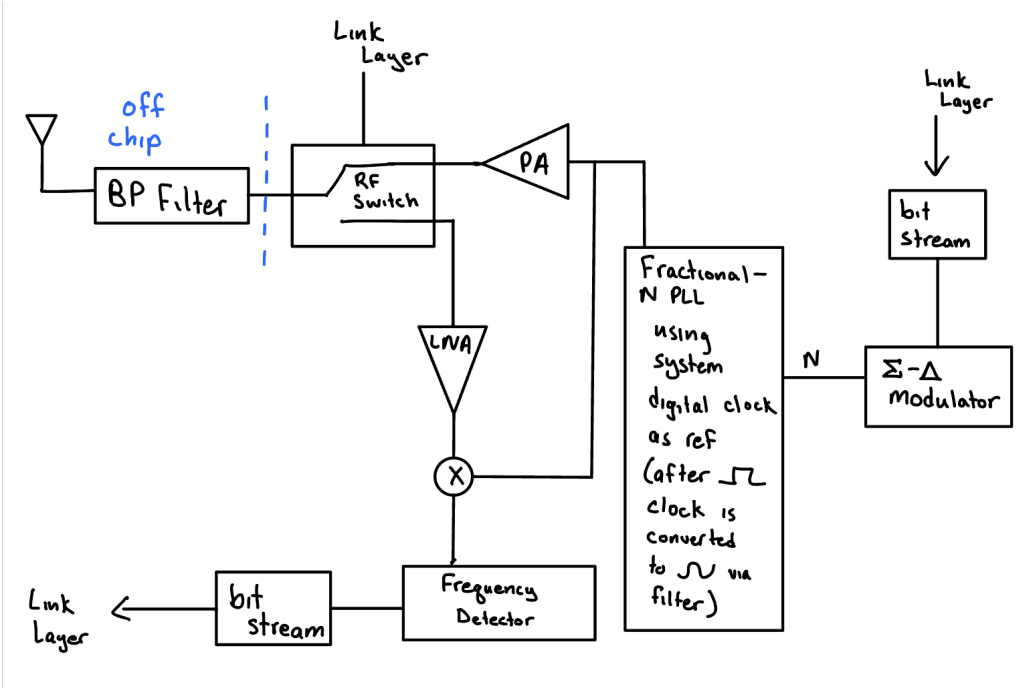
The Host Controller Interface is the standardized interface where the host and controller can communicate. The physical link for communication between the host and controller can be UART, USB, Secure Digital (SD), Three-wire UART, or custom. Due to the nature of this project —the host and controller being on the same chip— the physical link for communication would naturally be a memory bus (like AXIS, but in our case, “wishbone”). The parts of the BLE stack that go on the host controller (L2CAP, ATT, GAT, etc.) can be

implemented with Zephyr OS with its BLE stack. Zephyr OS BLE stack also has a LE link layer implementation for the controller side which may be an option instead of designing dedicated LE Link Layer digital hardware. Though as of now, the intent is to design dedicated hardware for the Link Layer. Here is a sample sketch block diagram of the overall system:



The Bluetooth Low Energy Primer mentions that, “The output power level at the maximum power setting shall be between 0.01 mW (-20 dBm) and 100 mW (+20 dBm).” Considering that this range is very generous (going all the way down to -20 dBm), our power amplifier design gain requirement can be flexible.

While a RF frontend architecture hasn’t been settled on yet, here is an idea for the PHY:



As BLE is half-duplex, we have a RF switch controlled by the Link Layer to switch between the TX and RX chain. A fractional-N PLL is convenient for BFSK as you can do fine fractional adjustments to the N divider value (by averaging integer N values) to adjust the frequency (which is helpful for finely shifting left and right from the channel center by at least +/- 185 kHz). The bandpass filter and antenna can be off chip.

Discussed the project with a graduate student with RFIC industry experience. One of the tips given was that relevant resources be found on IEEE Xplore for sky130 (by searching for open source 130 nm).

- Jackson Doyle: Joined chip forge and started the setup process. Made it to SSH and VDI workflow. Contributed to presentation.
- Daniel Pytel: Attended Chipforge meeting and began integrating their toolflow, albeit with some issues. Also read through parts of the previous senior design group's document, as well as the BLE primer to get acquainted with the terminology and the starting point.
- Tyler Bibus: Read through large portions of the primer, focusing on Link Layer related information. Generally getting a conceptual understanding around the scope, objectives, and a general action plan. Not too much done this week beyond conceptual work unfortunately, but a project as detailed and difficult as this requires strong knowledge.
- Aiden Han-Lindemyer: Merged with Radio team; Joined and completed ChipForge onboarding with setup (SSH and VDI workflow), but struggled in tutorial completion. Compiled report structure and finalized time structure.
- Parker Pederson: Merged with the Radio team. Joined chipforge and setup the toolflow for the vdi and comparch. Started tutorials for chipforge but got stopped in the wishbone adder tutorial.
- Akash Gojuru: Joined the chipforge and setup the toolflow for vdi. Started working on the tutorials.
- Seth Klaassen: joined chipforge and setup keys and toolchain, currently working through tutorials to learn the tools we are working with.

Pending issues

- Zane Salti: An RF frontend architecture has not been settled on yet.
- Jackson Doyle: Need to complete chip forge onboarding of toolflow. Continue with analysis of PLL and phy layer specifications of the previous project group. Begin with EE 4350 labs.
- Daniel Pytel: Still encountering some ssh errors with the chipforge toolflow.
- Tyler Bibus: Gather additional information regarding Link Layer and compile into personal notes. Figure out a good starting point from previous team implementation to carve out some tasks/issues.
- Aiden Han-Lindemyer: Struggled in ChipForge tutorial, primarily a statfs error in locating the directory. Will be regularly attending ChipForge meetings to get more familiar with the system and meet with previous semester Duwe project members as well as current LL team.

- Parker Pederson: Having statfs error during design hardening in digital design tutorial.
- Akash Gojuru: Need to complete the chipforge tutorials and still have the stafts error in the digital design tutorial.
- Seth Klaassen: need to complete the tutorials and gain familiarity with these programs as well as continue reading through the bluetooth link layer documentation

Time Tracking

NAME	Hours this week	HOURS cumulative
Zane Salti	~ 4	~ 5
Jackson Doyle	4	4
Daniel Pytel	2	2
Tyler Bibus	~3	3
Aiden Han-Lindemyer	4	4
Parker Pederson	4	4
Akash Gojuru	~4	4
Seth Klaassen	~3	3

Comments and extended discussion

The link layer specification represents more complex sections of the Bluetooth Core Specification. By limiting the scope to using the LE 1M modulation scheme as well as not adopting Extended Advertising, and so essentially focusing on Bluetooth LE 4.x support, implementation complexity can be reduced while still achieving wanted functionality. The decision to use an on chip wishbone for HCI communication is appropriate given the single chip design but will require driver custom development. While architecture decisions are being finalized, identifying common RF components allows for more simultaneous development progress.

Plans for the upcoming week

- Zane Salti: Meet with a former senior design member involved in sky130 RF analog architecture, explore and scope the security requirements of the Link Layer, and have the RF frontend architecture (from the idea PHY block diagram) get reviewed and commented on.

- Jackson Doyle: Attend chip forge meetings to finish onboarding and setup. Continue on to tutorials and sample chip forge projects. Develop baseline knowledge in regards to VLSI by completing EE 4350 labs and starting some work on xschem and magic. Develop a better understanding of prior phy layer work, understand and critique analysis.
- Daniel Pytel: Obtain more familiarity with comparch and the toolflow by completing the whole tutorial. Many others seem to have issues with it, so this may need to be revisited. Continue to form and place teams based on interests and skillsets.
- Tyler Bibus: Have completed the ChipForge tutorials before, will need to refresh compArch access and potentially redo ssh keygen and toolchain (likely on a lab computer for consistency rather than my personal laptop. Will also develop concrete plans for how link layer tasks will be divided, likely through an issue board on teams.
- Aiden Han-Lindemyer: Attend ChipForge meetings to get more familiar with the system. Goals are to get tutorials running in caravel, write samples that interact with C through the wishbone, and work with the LL team and Zane to find required sections for BLE. Finalize administrative documents and system (weekly presentation slides/reports, timetracking, Git repo workflow).
- Parker Pederson: Complete the chipforge tutorial set then write a sample that can interact with C code running on the CPU. Read the BLE primer to get update with link layer requirements. Begin determining which sections of BLE are required for minimal compliance.
- Akash Gojuru: Complete the ChipForge tutorial set and attend ChipForge meetings to get familiar with the tools. Read the BLE section and finalize the requirements. Work with the digital team and Zane to understand these topics better and get advice.
- Seth Klaassen: In this next week continue familiarizing with the Comparch system and the toolchain. Also continue reading through the bluetooth link layer datasheet.

Summary of weekly advisor meeting

The meeting was focused on scoping out the project. Discussed BLE legacy advertising and how it would need to be implemented in the digital logic of the physical layer, Duwe said it sounds reasonable. Discussed on chip host on controller vs having controller be a controller and how the physical bus would need to be created (wishbone). Emphasized focus on interface and controller. On the physical side the project could just not need a power amplifier, or have it be at a smaller scale because of the 0.01 mW (-20 dBm) and 100 mW (+20 dBm) range, all to narrow down scope on the phys side.Duwe emphasized the ability to look through components that are independent of the final architecture. Overall the team is still scoping out the project with Duwe. Duwe said that it is important to show that we did design work of a component, even if it is easy/on a smaller scale (such as an RF switch) so the team can still show a core component was completed from the ground up; show design and design analysis skill. LL layer team will regularly be attending ChipForge to grow skills with a primary goal to determine final architecture. Duwe recommended having a thesis project description (requirements etc) to keep the team on track and definable. Duwe has no concerns with a SOC design.