ECEN-5833

Low Power Embedded Design Techniques

Project Updates

Fitness Performance Tracking Vest

Team A.V.D

Aneesh Deshpande, Vaishnavi Patekar, Devang Boradhara

University of Colorado, Boulder

Vaishnavi Patekar Devang Boradhara Aneesh Deshpande

Index

Vaishnavi Patekar Devang Boradhara Aneesh Deshpande

Performance Tracker

We'll be designing and developing a GPS performance tracker which will measure important performance stats for most sports activities. From a demanding scenic trek to any high intense sports game and provide you an insight of your performance.

It will include the following statistics:

- Current Heart Rate
- Maximum Heart Rate
- Blood Oxygen
- Total Distance
- Altitude, Acceleration
- Max Speed
- No. of Laps/Sprints
- Calories.

Why this?

Performance tracking devices provide a level of individual performance analysis that far surpasses anything coaches and athletes from previous generations were familiar with. The advantages to early adopters of this technology are enormous. Individuals can keep a track of their fitness and get an in-depth understanding of strengths and weaknesses. This knowledge, put together with the right training can do wonders.

Managing Workload

For emerging athletes, consistent and intelligent training is crucial for long-term success, particularly for those aspiring to college-level sports and beyond. Field sports like soccer, football, and baseball demand skill development through year-round, focused training. However, overtraining poses risks to athletes and teams. With performance tracking technology, coaches can now make data-driven decisions to balance intensity and rest, ensuring peak performance on game days.

Injury Reduction

Injuries are a significant concern for athletes and teams, especially when they result from preventable factors, such as risky training practices. Performance tracking offers valuable insights into determining suitable training thresholds, preventing overexertion, and managing fatigue and stress during games and practices.

Block Diagram

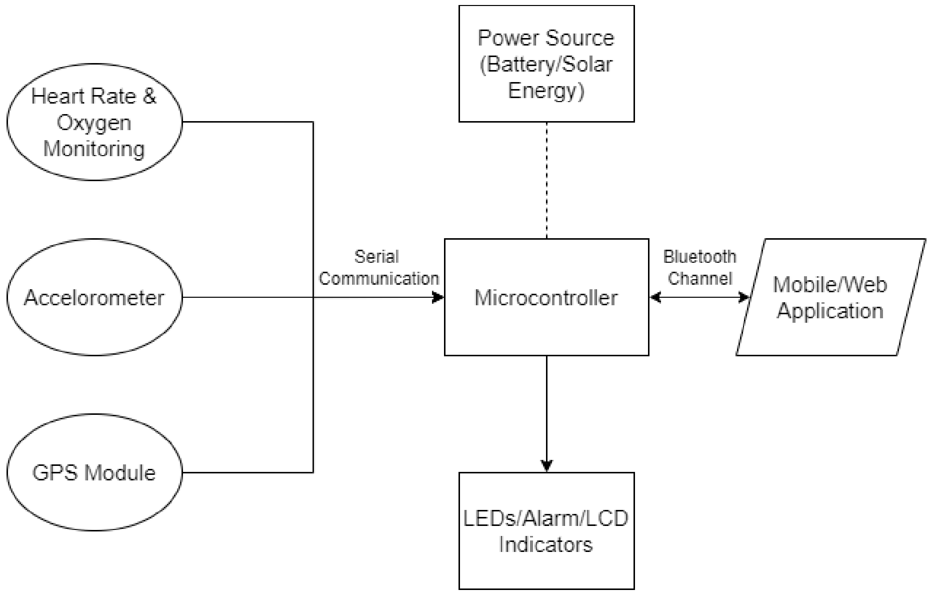


Figure 1: High Level Block Diagram

Data Flow Algorithm

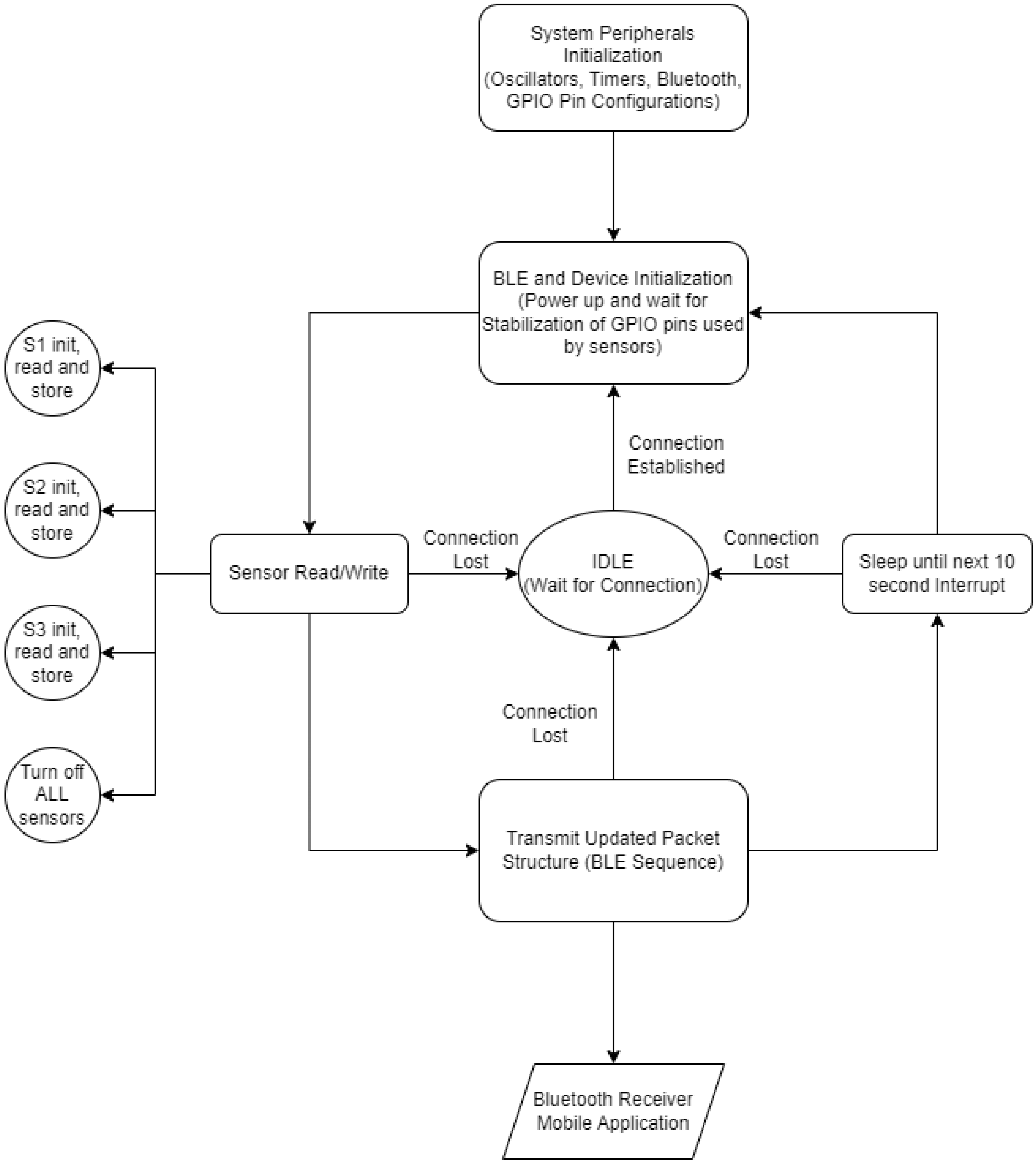


Figure 2 Data Flow Diagram

Microprocessor and Sensor Selection

EFR32 Blue Gecko

The EFR32 Blue Gecko is a ARM Cortex-M4 based family of wireless system-on-chip (SoC) devices used as an energy-efficient solution for wireless communication and IoT applications.

Operating Parameters	**Range**
Power Supply Voltage	1.8V – 3.8V
Wireless Connectivity	Bluetooth Low Energy (BLE), Zigbee, Thread
Peripherals	GPIO pins, UARTs, SPI, I2C, timers, and analog interfaces
Power Modes	active, sleep, deep sleep, and hibernate, (EM0 – EM3)
Operating Temperature Range	-40°C to +85°C

Heart Rate & Pulse Oximeter Sensor, *MAX30102*

The MAX30102 is used to measure both heart rate and oxygen saturation (SpO2). It works by emitting and detecting infrared light through the skin, measuring variations in light absorption due to blood flow, and using this information to calculate heart rate and SpO2 levels in wearable fitness devices. It plays a crucial role in monitoring the wearer's health and fitness parameters in real-time.

Operating Parameters	**Range**
Power Supply Voltage	3.3V – 5V
Communication Interface	I2C
Capability of heart rate measurement	30BPM – 200BPM
Oxygen Saturation Measurement	70% - 100%
Sampling Rate	50Hz – 1KHz
Operating Temperature Range	-40°C to +85°C

Accelerometer, LIS3DH Triple-Axis

The LIS3DH is a triple-axis accelerometer sensor used for motion sensing. Here are the working requirements and specifications of the LIS3DH Triple-Axis module:

Operating Parameters	Range
Power Supply Voltage	1.71V – 3.6V
Communication Interface	I2C/SPI
Measurement Axes	Three orthogonal axes: X, Y, and Z
Acceleration Range	70% - 100%
Operating Temperature Range	-40°C to +85°C

SAM-M8Q

- Initially, the NEO-6M GPS module was considered, but following thorough research, we opted for the SAM-M8Q GPS module due to its superior features and benefits compared to the former.

Parameters	SAM-M8Q	NE0-6M	MAX-M10S
Integrated Antenna	✓	×	×
Odometer	✓	×	✓
Tracking Channels	72	50	56
Dynamics	≤4 g	≤4 g	≤ 4 g
Altitude	50,000 m	50,000 m	80,000 m
Velocity	500 m/s	500 m/s	500 m/s
VCC Max	3.6	3.6	3.6

Product Features

- Free and easy to use Vest Mobile Application which enables the user to track the cardinal parameters related to human body (heartbeat rate and blood oxygen levels) and movement (speed, altitude, location) while performing any physical activity.
- Load power management would be implemented though software based on sensory data, such that, components consume as low power as possible.
- The device can run on rechargeable batteries, and it would also support charging over solar energy as a part of energy harvesting.
- If threshold values of heartbeat or oxygen levels are crossed above or below limits, the mobile app will support notifying the user with a warning.

Specs	Values
Dimensions	3.5in x 3.5in
Temperature	0 – 60 ˚C
Expected Wireless Range	10 meters (From EFRBG13)
Relative humidity	65 ± 20%
Warranty	~ 1.7 years

Table 1 Product Specifications.

Update 1: Week 3

Proposal Feedback Questions

1. I would like to know how often you plan on sampling your sensors and how often you are going to get GPS positioning?
    - We intend to sample each of the devices to ensure that new data becomes available every 10 seconds, aligning with our initial concept of transmitting data packets at this specific interval.

2. How are you going to implement load power management?
    - If the component under consideration supports low power or sleep mode with some condition/threshold, it will be incorporated for load power management, else the module's switching will be controlled by the software based on required conditions.

3. Under features, it's stated load power management will be implemented through software based on sensory data, but is there a low power mode on your GPS module?
    - Yes, SAM-M8Q has two low power modes out of which we will be using "Cyclic Tracking" mode which can sample data every 1 – 10 seconds.

4. Does your MAX30102 have a shutdown mode?
    - Yes, the MAX30102 sensor does have a shutdown mode. Its shutdown mode allows us to conserve power when the sensor is not actively needed, which is especially important in battery-operated devices.
    - In shutdown mode:
        - LEDs Turn Off: The MAX30102's LED drivers, which are used for emitting light into the skin to measure pulse and oxygen levels, are turned off. This significantly reduces power consumption as the LEDs are one of the most power-hungry components of the sensor.
        - Sensor Functions Pause: The sensor's data acquisition and processing functions are paused. It stops collecting and processing data, which further reduces power consumption.
        - Registers Retain Data: The sensor retains its configuration settings and previous data in its registers during shutdown. This means that when you exit shutdown mode and power it up again, you can resume data collection with the same settings.
        - To exit shutdown mode and bring the MAX30102 back into active operation, you typically need to write to the sensor's control registers to configure its mode of operation (e.g., heart rate or SpO2 mode) and start data acquisition.

5. How long do you expect your product to run (how long of a workout can it track before you need to charge it)?
    - As per initial estimations and design ideas, we plan to have battery that can keep the device up for at least 5 hours.

6. You mention an LCD in your indicators box, are you planning on implementing an LCD (think about your physical product specifications)?
    - The indication segment of the Block Diagram presented potential choices, but our current plan does not involve incorporating an LCD into the device. Instead, the device will feature status LED(s).

7. Also, please elaborate on your ideas for your mobile/web application (high level).
    - GUI Development Platform Considerations: Python/MIT App Inventor/Android Studio
    - App Features:

      Application will display following health, fitness & location parameters:

        - Heart rate
        - Blood Oxygen Level
        - Calories Burnt
        - Distance Covered
        - Location
        - Altitude
        - Motion Speed
        - Total activity time
        - Type of activity
    - User will have to set a profile during App initialization.
    - User Profile Parameters:
        - Profile name
        - Gender
        - Age
        - Height
        - Weight
    - App will be integrated with the fitness tracking vest through wireless communication protocol – Bluetooth.
    - Real-time data synchronization between the vest and the app

Past Week Progress

Sr. No.	Task	Date
1.	Sensor selection Verification	09/11/2023
2.	GPS sensor change based on specifications	09/12/2023
3.	Detailed study of sensor datasheets	09/14/2023
4.	Sensor working modes	09/14/2023

5.	Study and analysis of power consumption of each sensor and microcontroller	09/15/2023
6.	Load power management design	09/16/2023
7.	Storage element inclusion decision	09/16/2023

Plan for Next Week

Sr. No.	Task	Planned Date of completion
1.	Deciding basic process flow algorithm	09/18/2023
2.	Study: Basic Converters and Regulators	09/20/2023
3.	Study: PMIC and decide suitable one for our application	09/20/2023
4.	Components library creation in Altium	09/22/2023
5.	Study: App Development	09/23/2023
6.	Study: Health Parameters monitoring	09/24/2023

Gantt Chart

Please access the Gantt Chart here.

Support needed for following tasks/decisions

- Power Consumption Analysis
- Energy Harvesting Specifications
- Review of Process algorithm

Why these Sensors?

LIS3DH

- LIS3DH is a 3-axis accelerometer, measuring acceleration along the X, Y, and Z axes. The primary reasons to use this for our application are following:

- Known for low power consumption, suitable for battery-powered devices and applications where power efficiency is critical and can help in load power management of our device.
- Provides good resolution to capture a wide range of accelerations accurately.
- Supports both I2C and SPI communication interfaces with transaction cycles in the range of microseconds and nanoseconds respectively.

MAX30102

- The selection between the MAX30101 and MAX30102 sensors was deliberated. In cases where exclusive heart rate monitoring suffices, the MAX30101 could be deemed adequate and economically advantageous. Nevertheless, for our application precise SpO2 measurements are necessitated for a broad spectrum of scenarios, the MAX30102 emerges as the favored option owing to its dual-LED configuration and enhanced accuracy.

SAM-M8Q

- Initially, the NEO-6M GPS module was considered, but following thorough research, we opted for the SAM-M8Q GPS module due to its superior features and benefits compared to the former.

Parameters	SAM-M8Q	NE0-6M	MAX-M10S
Integrated Antenna	✓	×	×
Odometer	✓	×	✓
Tracking Channels	72	50	56
Dynamics	≤4 g	≤4 g	≤ 4 g
Altitude	50,000 m	50,000 m	80,000 m
Velocity	500 m/s	500 m/s	500 m/s
VCC Max	3.6	3.6	3.6

Table 2: GPS module selection based on key specifications

Use Case Model

Power Consumption

The following data regarding current consumption at specific voltage levels as per datasheet was used to determine an estimate of total power usage and which energy modes would the device operate at any instant. All these values are based on specific operating environments from datasheets, and thus the actual consumption based on time periods (power on, stabilizing time, data transfer, sleep/wake) would vary once the measurements are made using the actual sensor and the microcontroller.

		Current (uA)	Voltage	Power (uW)
Interfaces	MAX30102 (HR and SpO2)			
	HR + SpO2 Mode	1200	5	6000
	HR Mode	1200	2	2400

	Standby Mode	0.7	1.7	1.19
	LIS3DH (Accelerometer)			
	Normal Mode @50 HZ ODR	11	2.5	27.5
	Normal Mode @1 HZ ODR	2	2.5	5
	Low Power Mode @50 HZ ODR	6	2.5	15
	SAM-M8Q (GPS)			
	Continuous Mode	23000	3	69000
	Cyclic Tracking (@ 1Hz)	9500	3	28500
	Max Supply Current (@1Hz)	67000	3	201000

Table 3: Power Consumption for Interfacing Devices

For the microcontroller, since it has an onboard Bluetooth module, and the radio would not work beyond EM1 mode, the following current consumption data is available from the datasheet:

Energy Modes	Typical Current Consumption (uA)	Voltage	Power (uW)
EM0	128	3.3	422.4
EM1 (all peripherals disabled)	76	3.3	250.8
EM1 (with Radio)	9500	3.3	31350
EM2	2.2	3.3	7.26
EM3	1.5	3.3	4.95
EM4	0.4	3.3	1.32
EM4 Sleep	0.08	3.3	0.264

Table 4: Power Consumption for Microcontroller

Energy Mode Analysis

According to the initial proposal and brainstorming, we plan to transmit the data packets every 10 seconds to the mobile application over Bluetooth. As per the reference manual of EFR32BG13, the device supports active radio transmission only until EM1 mode. Therefore, based on power consumptions, reference manual data and initial brainstormed sampling rate for data transfers, the device would always be in either **Active Mode, EM0 or EM1 energy modes**. But, with load power management, the current consumption at any instant can be lowered for the time when no new reading of sensory data is required or using sensor's internal low power feature.

Update 2: Week 4

Feedback Questions from Update 1

1. In your Gantt chart, I would recommend splitting your tasks into subsections for easier navigation. The Gantt chart should fill in the right side with colored cells where tasks are completed/planned. Please keep adding your updates to a growing document.

- Refer here.

2. For this week, you should have your proposal and then your update 1 as an additional section.

- Updated

3. Don't forget to add if you are on track and if not why.

- Added here.

4. Please pick your desired energy harvesting option. If you need help with this, reach out to me/Randy and the TA's.

- We plan to use solar panels/cells as our energy harvesting source as our device's primary application is sufficed outdoors. As the energy harvesting source would contribute to 10% of battery capacity, we would finalize the size/number of panels/cells once the energy source device is finalized.

5. In your energy mode breakdown, which sensors are running in which energy mode and for how long? There is an example of this energy chart given in the slide deck.

- In summary, all sensors need to be at least in EM1 energy mode to receive peripheral interrupts. The detailed explanation for energy mode using average & maximum current values and time period of one initialization, read and write cycle is provided in Energy Mode Breakdown and Use Case and Energy Storage Element Calculation sections.

6. Additionally, you need to tell me more about your design. How are you going to connect your sensors? Are they all on the same I2C bus? Are they split? If a sensor supports different serial protocols, which one are you picking and why?

- Refer here

Energy Mode Breakdown

To calculate the maximum power being drawn by the device and which energy state it falls under, following parameters were considered:

Device	Current (mA)	Voltage (V)	Power (mW)	Power(uW)
Deep Sleep	0.0013	3.3	0.00429	4.29
Microcontroller (Tx)	8.5	3.3	28.05	28050
GPS	67	3.3	221.1	221100
HR and SPO2	1.2	3.3	3.96	3960
Accelerometer	0.01	3.3	0.033	33
Total:	76.71			

Table 5: Voltage and Current Consumption Summary

Device States	Power (mW)
GPIO Init/ Startup	4.95
Sensor Init + Read	225.093
Transmit	80
Deep Sleep	0.00429

Referred from datasheet, 19dBm is 0.0794 Watts

Table 6: Power Consumption Summary for each State

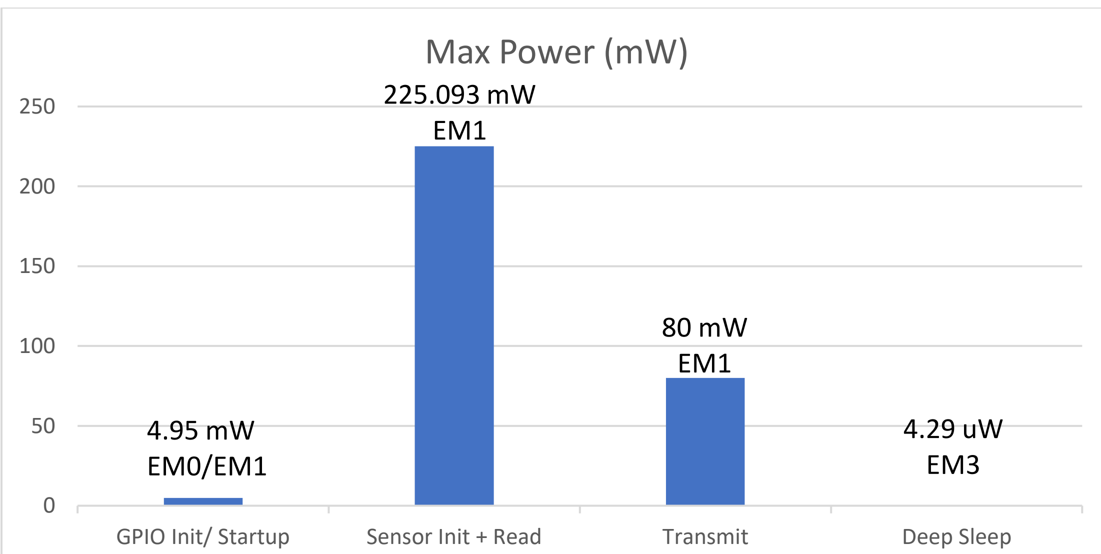


Figure 3 Energy Chart

The high-level design elaborating the timing, current and power consumption of each state is shown in the Use Case and Energy Storage Element Calculation section.

Past Week Progress

Sr. No.	Task	Planned Date of completion	Actual Date of Completion
1.	Deciding basic process flow algorithm	09/18/2023	Done (Refer here)
2.	Study: Basic Converters and Regulators	09/20/2023	Done
3.	Study: PMIC and decide suitable one for our application	09/20/2023	Done (Refer here)
4.	Components library creation in Altium	09/22/2023	Not started
5.	Study: App Development	09/23/2023	Not started
6.	Study: Health Parameters monitoring	09/24/2023	In progress

Plan for Next Week

Sr. No.	Task	Planned Date of completion
1.	Finish library creation of components to be used in the system.	09/29/2023
2.	Perform interfacing of all components using the development kits and verify the working.	09/29/2023

Gantt Chart

Please access the Gantt Chart here.

Progress Status

At this point, we've completed essential tasks, including the examination of fundamental regulators and the selection of a PMIC (Power Management Integrated Circuit). We have also determined the basic dataflow for the design. However, we realized that we underestimated our time for the completion of necessary tasks and planned other tasks such as studying app development and creating a component library. Unfortunately, due to other high-priority tasks such as energy management calculations and PMIC selection, we were unable to dedicate time to these tasks. Taking this phase of development into consideration, we are on schedule, except from verifying recharge time requirements for use cases and sizing storage based on the number of charge cycles required.

Support needed for following tasks/decisions

- Need help to precisely understand following two points and to verify if our analysis is right.
  1. Use-case recharge time requirements
  2. sizing storage based on use case number of charge cycles
- To figure out, calculation of total energy required between charging.
- Selection of energy harvesting component

Sensor Interfacing

Sr. No.	Sensor Name	Available communication Protocols	Selected communication protocol	Reason for selecting specific protocol
1.	MAX30102	I2C	I2C	Sensor supports only I2C
2.	LIS3DH	I2C, SPI	SPI	SPI is efficient in terms of power and time consumption
3.	SAM-M8Q	UART, DDC (I2C compliant)	UART	Less power consumption as compared to I2C

Though there are multiple pins available on EFR32BG13 to establish connections between the sensor protocol pins (I2C/SPI/UART) and the EFR32BG13 pins, the ultimate pin configurations will be determined during the schematic and layout design phase, taking into consideration design convenience and ease of implementation.

Use case and Energy Storage Element selection math

To calculate the amount of energy/charge for the battery following design has been taken into consideration:

State	State Name	Time Duration (sec)	Energy Mode
1	GPIO INIT	1	EM0/EM1
2	Sensor Setup and Read	4	EM1
3	Data Transmission	1	EM1
4	Idle	4	EM3/Deep Sleep

Table 7: Energy Mode and Duration for each state

The current consumed during the initialization of GPIO pins is calculated using the AEM current profiler in Simplicity Studio by enabling the GPIO pin for onboard temperature sensor, and for this use case of 3 sensors, it was found to be around 1.5mA. The states shown above would run periodically every 10 seconds. Apart from this, the maximum current, maximum average current and minimum average current values were calculated using following data from datasheets:

Device	Current (mA)	Voltage (V)	Power (mW)
Microcontroller Deep Sleep	0.0013	3.3	0.00429
Microcontroller (Tx)	8.5	3.3	28.05
GPS	67	3.3	221.1
HR and SPO2	1.2	3.3	3.96
Accelerometer	0.01	3.3	0.033

Table 8: Power Consumption Summary for each device

While using the above design as a reference, following average current and power consumption values were obtained:

Parameter	Min.	Max
Average Current (mA)	1.07091	14.52152
Average Power (mW)	3.304003	47.921016

Table 9 : Average Current and Power Consumption of System.

The average current and power values were calculated using the following use case assumptions from device datasheets:

Operation	Worst Case Time (sec)
GPIO Init	1
GPS	2
Accelerometer	1
HR and SPO2	1
Transmit	1
Sleep	4
Total	10

Table 10: Worst Case performance time summary

We have defined 3 battery selection options and they are as follows:

Case I: Assuming all sensors run continuously for 5 hours (Max Current Consumption)

Parameter	Value	Units
Voltage (V)	3.3	V
Current (I)	0.076	A
Duration (t)	5	h

Table 11: Calculated values for Max Current Consumption

From these input values, we get the following:

Resistance	43.42105263	Ohms	=V/I
Power:	0.2508	Watt	=V*I
Charge	0.38	Ah	=I*t
Energy (WattHour)	1.254	Wh	=P*t
Energy (Joule)	4514	J	=Wh*3600
Energy (Calorie)	1078.97	cal	=Wh*860.42065

Table 12: Electrical Parameters for Max Current Consumption

Thus, from the number of Joules, we can calculate the required mAH:

Energy (Joule)	4500	J
Voltage	4.2	V
Charge (= E/V/3.6)	297.62	mAh

Table 13: Required Charge Calculation for Max Current Consumption

Case II: Maximum Average Current Consumption

Parameter	Value	Units
Voltage (V)	3.3	V
Current (I)	0.014	A
Duration (t)	5	h

Table 14: Calculated values for Max Average Current Consumption

Resistance	235.714	Ohms	=V/I
Power:	0.0462	Watt	=V*I
Charge	0.07	Ah	=I*t
Energy (WattHour)	0.231	Wh	=P*t
Energy (Joule)	832	J	=Wh*3600
Energy (Calorie)	198.76	cal	=Wh*860.42065

Table 15: Electrical Parameters for Max Average Current Consumption

Energy (Joule)	1000	J
Voltage	4.2	V
Charge (= E/V/3.6)	66.14	mAh

Table 16: Required Charge Calculation for Max Average Current Consumption

Case III: Minimum Average Current Consumption

Parameter	**Value**	**Units**
Voltage (V)	3.3	V
Current (I)	0.001	A
Duration (t)	5	h

Table 17: Calculated values for Minimum Average Current Consumption

Resistance	3300	Ohms	=V/I
Power:	0.0033	Watt	=V*I
Charge	0.005	Ah	=I*t
Energy (WattHour)	0.0165	Wh	=P*t
Energy (Joule)	59	J	=Wh*3600
Energy (Calorie)	14.20	cal	=Wh*860.42065

Table 18: Electrical Parameters for Minimum Average Current Consumption

Energy (Joule)	100	J
Voltage	4.2	V
Charge (= E/V/3.6)	6.61	mAh

Table 19: Required Charge Calculation for Minimum Average Current Consumption

From the above 3 cases, we have an estimate of the minimum and maximum requirements of the system at any given instant and the data required to look for an energy storage element (Battery) that will suffice the energy requirements.

Energy Storage Element and PMU(s)

Energy Storage element: Battery (3898) Digi-Key Part No: 1528-2731-ND

Upon researching supercapacitors and batteries, it becomes evident that supercapacitors are used in situations requiring high or instantaneous power output in a short timeframe, prioritizing power density. Conversely, batteries are chosen when energy density is more important, typically when sustained energy supply is needed for extended periods, with less emphasis on immediate power demands. For our specific application, the primary requirement is an energy storage component with higher power density rather than energy density. Therefore, the optimal choice is a "**battery**."

Use-case recharge time requirements

Assuming maximum average current consumption of the system mentioned above the recharge time requirements can be calculated as follows:

Battery Capacity = 400 mAH

Max. avg system current (power on + initialization + data acquisition + transmission) = 14.5 mA (Discharge rate)

$$\text{Next Battery Recharge Time (hours)} = \frac{\text{Battery Capacity (mAh)}}{\text{Max. Avg System Current (mA)}}$$

Plugging in the values:

$$\text{Next Battery Recharge Time (hours)} = \frac{400\,\text{mAh}}{14.5\,\text{mA}} = 27.5\,\text{hours} \approx 27\,\text{hours}$$

Table 20 Battery Recharge Duration

Sizing storage based on use case number of charge cycles (Warranty)

For the selected Battery charge cycles from the datasheet are as follows:

Cycle Life @25℃ - Cycle **life ≥ 500**

Discharge to 3.0V @0.2C, then 0.5c CCCV 0.01C charge to 4.2V, rest for 10 min. discharge @ 0.2C to 3.0V and rest for 10 min. Continue the charge/discharge cycles until discharge capacity lower than 70% of rated capacity.

Based on our calculations the battery will discharge at a rate of 0.03C which is nearly 15% less than the discharge rate considered in the datasheet. So, the warranty of product can be calculated as follows:

Recharge Time = 27 Hrs.

Number of hours to complete 500 cycles = 27 x 500 = 13500 Hrs.

Since the discharge current is 15% less than discharge current in datasheet adding 15% to Number of hours to complete 500 cycles = 13500 + 2025 = 15525 Hrs.

500 cycles in days = 15545/24 = 647 days

500 cycles in years ~= 1.7 yrs.

Note: These values are true considering the condition that there is no charging until and unless the battery discharges to discharge cut-off voltage of 3.0 V. But if the battery is charged and discharged at the same time using the energy harvesting element then the product warranty can be further improved based on the difference between charging and discharging rate.

Use-case PMIC selection

We have chosen the BQ25570, an Ultra-Low Power Harvester Power Management IC with a boost charger, and a Nano power Buck Converter, to serve as the Power Management IC (PMIC) for our product.

We have opted not to use an unregulated power supply since all sensors and the microcontroller in our system have a maximum voltage limit of 3.6 V, whereas the energy storage element supplies 3.7 V. As a result, we will require a regulator to step down and maintain a constant supply voltage of 3.3 V from the battery.

Considering the selected microcontroller and all attached sensors, it is clear that a 3.3 V voltage rail will be necessary for their operation, and this can be achieved using the chosen PMIC. Additionally, since the energy harvesting element is a solar cell/panel, we will need a boost converter to convert the solar output into power levels that are sufficient for our needs.

The component we have selected fulfills all these requirements as it provides both Buck and Boost converters, along with energy harvesting functionality.

Update 3: Week 5

Feedback from Update 2

1. Gantt Chart Modifications
- Refer here.
2. Find your solar panel's peak wattage and estimate how much sunlight you would get per day. This will give you the energy put into your battery (minus conversion losses).
- Based on the power consumption, PMIC's input voltage range and the maximum input current, we are exploring the solar panel options and would like to discuss with SA's on it.
3. With your GPS, try to verify functionality with UART. If you don't get that far before your boards are sent to fab, you can add both UART and I2C lines to your board.
- We will try interfacing and reading data from the GPS Module using UART and if there are any issues or for backup solution, we will also add I2C lines on our final design.
4. In your energy storage element selection math, how did you get to your time duration for each state?
- For calculating the time duration in each state, we took into consideration the maximum time taken by the sensor to setup and give out sensed data from datasheet. Along with the current and power values used to determine the energy state, we considered worst case time (in resolution of seconds for all sensors) in addition to the known parameters such as GPIO stabilization time from previous experience with EFR32BG13. Here is the summary of data we took into consideration to conclude a preliminary period based on Datasheet values (resolution: seconds) for each state:

Operation	Worst Case Time (sec)
GPIO Init	1
GPS	2
Accelerometer	1
HR and SPO2	1
Transmit	1
Sleep	4
Total	10

Upon combining these tasks into common states, here's the summarized state information:

State	State Name	Time Duration (sec)	Energy Mode
1	GPIO INIT	1	EM0/EM1
2	Sensor Setup and Read	4	EM1
3	Data Transmission	1	EM1
4	Idle	4	EM3/Deep Sleep

- The precise calculations for timing of each sensor with respect to the communication bus is elaborated here.

5. Please include your system's specs (dimensions, temp, expected wireless range, humidity range, warranty time).

Specs	Values
Dimensions	3.5in x 3.5in
Temperature	0 – 60 ˚C
Expected Wireless Range	10 meters (From EFRBG13)
Relative humidity	65 ± 20%
Warranty	~ 1.7 years

Sizing Storage based on recharge requirements

Based on current and recharge requirements of our system the storage sizing was calculated which is linked here. For deciding the storage element size three different cases were considered for different current consumptions of the system which can be found under Use case and Energy Storage Element selection math.

Finalization of storage size was done based on different supply voltages for all sensors and on supply voltage for the controller. Apart from this tolerance of battery was considered to make sure that it has sufficient power available to supply to the system.

C-Rate and Battery Solution

As per the datasheet for the selected battery solution Maximum Constant Charging Current is 400mA that is 1C and Maximum Continuous Discharging Current is 600mA (1.5C).

Based on Use case model and Energy Storage Element selection math for three different cases the discharging current in C is calculated as follows:

Assuming all sensors run continuously for 5 hours (Max Current Consumption)

Parameter	Value	Units
Current (I)	76	mA
Battery Size	400	mAH
Required Duration (t)	5	h
C Rate	**0.19**	**C**

Table 21 Worst case C rate.

Maximum Average Current Consumption

Parameter	Value	Units
Current (I)	14	mA
Battery Size	400	mAH
Required Duration (t)	5	h
C Rate	**0.0350**	**C**

Table 22 Normal case C rate.

Minimum Average Current Consumption

Parameter	Value	Units
Current (I)	1	mA
Battery Size	400	mAH
Required Duration (t)	5	h
C Rate	0.0025	C

Table 23 Best case C rate.

Hence it can be considered that even with the worst-case discharge rate of ~0.2C the system will work easily for 5hrs before needing a recharge.

Parameters	Value	Units
Nominal Voltage	3.7	V
Charging Cut-off voltage	4.2	V
Discharge Cut-off Voltage	3	V
PMU support a low battery discharge cut-off voltage	1.91	V

Table 24 Cut-off voltages of Battery.

As our system requires only two power supplies (3.3V and 1.8V) which are less that the nominal voltage of 3.7V, we would only require Buck-convertor.

PMU Solution

Features
Cold-Start Voltage: VIN ≥ 600 mV
Continuous Energy Harvesting From VIN as low as 100 mV
Battery Charging and Protection
Internally Set Undervoltage Level
User Programmable Overvoltage Levels
Programmable Step-Down Regulated Output
High Efficiency up to 93%
Supports Peak Output Current up to 110 mA (typical)

Additional features considered during selection of PMIC other than Use-case PMIC selection can be found here.

Apart from the PMIC, we will also use an LDO to limit the buck output to 1.8V as it is required by the MAX30102 sensor.

Past Week Progress

Sr. No.	Task	Planned Date of completion
1.	Component Library Creation	In-progress
2.	Worst Case timing analysis for sensor communication	09/30/2023
3.	Verification of Battery math	09/29/2023

Plan for next week

Sr. No.	Task	Date
1.	Component Library Creation + Schematic	10/07/2023
2.	Check Solar panel specifications	10/05/2023
3.	Bulk cap simulation	10/06/2023
4.	Spice analysis of power supplies	10/06/2023

Unplanned Tasks accomplished during past week

Sr. No.	Task	Date
1.	Studied Mini Debug Connector (AN958)	09/30/2023

Progress Status

We are on track as per the plan.

Support needed for following tasks

We found it difficult to gather enough information regarding the worst-case communication bus time for SAM-M8Q and only worst-case device response time was available for our analysis. Any leads on same would help us reach a concrete conclusion regarding timing parameters.

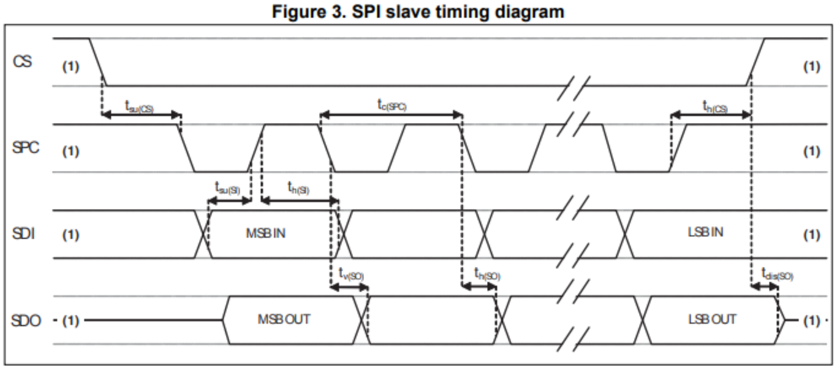
Worst case timing for communication bus

LIS3DH (SPI)

- Minimum Clock Cycle: 100ns
- Wait at least 5ms after powering up the sensor before reading/writing to the sensor.
- CS line takes minimum 5ns to change its state, should be held for at least 20ns.
- SDI line takes minimum of 5ns to change its state, should be held for at least 15ns.
- Valid output can be found on SDO line after a maximum of 50ns, can be held for at least 5ns before next clock pulse.
- The LSB of SDO will be available no later than 50ns after CS is pulled high.

Timing Diagram:

Figure 3. SPI slave timing diagram



1. When no communication is ongoing, data on SDO is driven by internal pull-up resistors.

Worst Case timing calculation based on datasheet:

LIS3DH		
Operation	**Cycles**	**Unit**
Read Operation	120	CLK cycles (16 CLK cycles per transaction x 6 read operations for 1 period)
Write Operation	96	CLK cycles (40 CLK cycles per transaction x 3 write operations for 1 period)
Total	216	CLK cycles

Timer Period for 1 CLK cycle	100	Nanoseconds

Worst Case time	21.6	Milliseconds (total cycles x 1 CLK cycle period)

MAX30102 (I2C)

- Frequency: 0 to 400 KHz.
- There has to be 1.3us time between start and stop conditions over I2C channel.
- The setup time for Data line is 100ns.
- SDA/SCL Rise/Fall time: 300ns.

Timing Diagram:

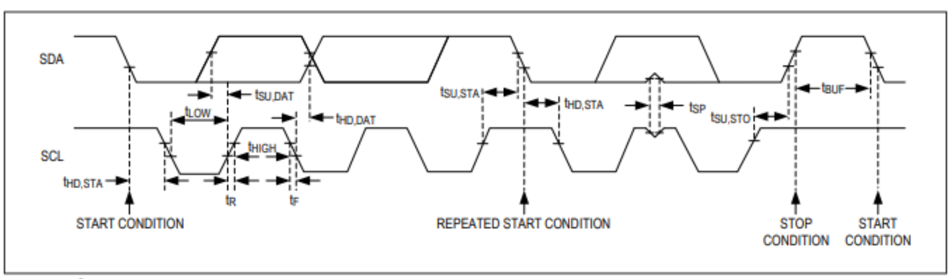


Figure 1. I²C-Compatible Interface Timing Diagram

Worst Case timing calculation based on datasheet:

MAX30102		
Operation	**Cycles**	**Unit**
Read Operation	102	CLK cycles (17 CLK cycles per transaction x 6 read operations for 1 period)
Write Operation	85	CLK cycles (17 CLK cycles per transaction x 5 write operations for 1 period)
Total	187	CLK cycles

Frequency	400	KHz
Timer Period for 1 CLK cycle	2500	Nanoseconds

Worst Case time	467.5	Milliseconds (total cycles x 1 CLK cycle period)

SAM-M8Q (UART/I2C)

The documentation (datasheet and supporting document) available for SAM-M8Q enlists initialization steps and information relating the commands to be sent over UART. It does not have any timing related worst-case information and just provides supporting baud rates.

Possible UART Interface Configurations

Baud Rate	*Data Bits*	*Parity*	*Stop Bits*
4800	8	none	1
9600	8	none	1
19200	8	none	1
38400	8	none	1
57600	8	none	1
115200	8	none	1
230400	8	none	1
460800	8	none	1

High-risk development items and mitigation plans

- GPS Accuracy
  Ensuring accurate GPS data can be a challenge, especially in areas with poor satellite visibility (e.g., urban canyons or dense forests).
  To mitigate this risk, we are storing the most recent GPS data and the user's current direction of movement.

- Communication Failure
  Failure of sensor communication over the physical bus can be risky.
  To address this, we are incorporating test points for the communication lines of each sensor as a precautionary measure.

- Signal Integrity
  Signal noise has the potential to impact the reliability of sensor readings, potentially resulting in errors in the collected sensor data and consequently yielding inaccurate results. To address this concern, we are taking measures to minimize signal noise by placing decoupling capacitors in close proximity to the power pins.

References

[1] U-blox-Receiver-Description, https://content.u-blox.com/sites/default/files/products/documents/u-blox8-M8_ReceiverDescrProtSpec_UBX-13003221.pdf (accessed Sep. 16, 2023).

[2] SAM-M8Q Hardware Integration Manual, https://content.u-blox.com/sites/default/files/SAM-M8Q_HardwareIntegrationManual_%28UBX-16018358%29.pdf (accessed Sep. 16, 2023).

[3] SAM-M8Q Power Management Application Note, https://content.u-blox.com/sites/default/files/products/documents/PowerManagement_AppNote_%28UBX-13005162%29.pdf?hash=undefined (accessed Sep. 16, 2023).

[4] Accelerometer LIS3DH Datasheet, https://cdn-learn.adafruit.com/assets/assets/000/085/846/original/lis3dh.pdf?1576396666 (accessed Sep. 16, 2023).

[5] MAX30102 high-sensitivity pulse oximeter an ... - analog devices, https://www.analog.com/media/en/technical-documentation/data-sheets/MAX30102.pdf (accessed Sep. 16, 2023).

[6] EFR32BG1 data sheet: Blue Gecko Bluetooth® low Energy Soc Family, https://www.silabs.com/documents/public/data-sheets/efr32bg1-datasheet.pdf (accessed Sep. 16, 2023).

[7] Recommended configurations and operating profiles for MAX30101/MAX30102 ..., https://www.analog.com/media/en/technical-documentation/user-guides/max3010x-ev-kits-recommended-configurations-and-operating-profiles.pdf (accessed Sep. 16, 2023).

[8] S. Biswas, "How do smartwatches measure calories: Tech," knowledge, https://www.vertexknowledge.com/post/how-do-smartwatches-measure-calories-tech-knowledge#:~:text=Smartwatches%20use%20their%20inbuilt%20accelerometer,calories%20to%20fuel%20their%20contractions (accessed Sep. 16, 2023).

[9] BQ25570. BQ25570 data sheet, product information and support | TI.com. (n.d.). https://www.ti.com/product/BQ25570

[10] Li-polymerbatterytechnologyspecification - Adafruit Industries. (n.d.-a). https://cdn-shop.adafruit.com/product-files/3898/3898_specsheet_LP801735_400mAh_3.7V_20161129.pdf

[11] The major differences between supercapacitors and batteries - Eaton. (n.d.-b). https://www.eaton.com/content/dam/eaton/products/electronic-components/resources/brochure/eaton-supercapacitor-vs-batteries-white-paper-elx1150-en.pdf

[12] AN958: Debugging and Programming Interfaces for Custom Designs | Silicon Labs https://www.silabs.com/documents/public/application-notes/an958-mcu-stk-wstk-guide.pdf