Requirements, Constraints, and Engineering Standards

Arinc 429 Portable Receiver and Flutter Application

Problem Statement

Currently in order to read Arinc 429 data there is a need for a bulky and expensive data receiver and transmitter. We will be altering a size and cost-effective bus reader to meet a variety of requirements so it can be used in the avionics industry and replace the bulky and expensive one.

Constraints

1	The project must be completed by the end of the Fall 2023 semester
2	The project must use the ESP23S3 Microprocessor
3	The mobile app needs to use the Flutter framework
4	The microcontroller should communicate with the mobile app over Bluetooth Low Energy

Firmware Functional Requirements

Functional Requirements	
1	ESP32S3 to read in data from Holt 429 receiver
2	ESP32S3 to send data over Bluetooth Low Energy
3	ESP32S3 to connect to Smartphone
4	ESP32S3 to receive labels over Bluetooth Low Energy
5	ESP32S3 rejects incorrectly formatted words (incorrect parity, etc.)
6	Send multiple labels over BLE

Firmware Non-Functional Requirements

Non-Functional Requirements	
1	Ability to quickly and easily update firmware
2	Easy to understand and use
3	Send labels quickly (numbers tbd)
4	Receive labels quickly (numbers tbd)
5	Decode labels quickly (numbers tbd)
6	Flutter App is able to decode and receive multiple incoming labels at a time

Flutter App Functional Requirements

Functional Requirements	
1	Communicate with the chip over Bluetooth Low energy
2	Read Arinc labels from the BLE
3	Decode the Arinc labels
4	Send Arinc labels over BLE
5	Receive multiple incoming labels at a time
6	Handle errors with the data(incorrect parity, etc.)
7	Display rate of information received
8	Define and store new labels
9	Display SDI(Source Direction Indicator)
10	Display SSM(Sign/Status Matrix)
11	Listen for multiple labels being sent over BLE
12	Display the data bits according to their meaning(defined by the labels)

Flutter App Functional Requirements

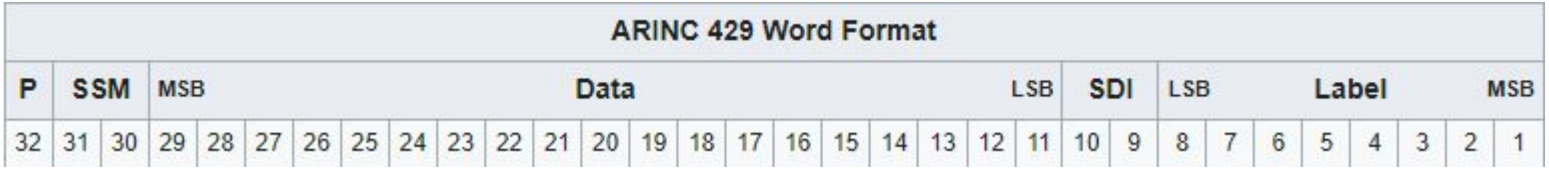
Functional Requirements	
1	Communicate with the chip over Bluetooth Low energy
2	Read Arinc labels from the BLE
3	Decode the Arinc labels
4	Send Arinc labels over BLE
5	Receive multiple incoming labels at a time
6	Handle errors with the data(incorrect parity, etc.)
7	Display rate of information received
8	Define and store new labels
9	Display SDI(Source Direction Indicator)
10	Display SSM(Sign/Status Matrix)
11	Listen for multiple labels being sent over BLE
12	Display the data bits according to their meaning(defined by the labels)

Standards

We will be discussing the Arinc 429 standard, Standards for Embedded Systems, and Standards for Bluetooth. The Arinc 429 standards aren’t from IEEE, however they are standard for the industry they are used in.

Arinc 429 Serial Bus

Arinc 429 is a serial bus commonly used in the avionics industry. The standard communicates 32 bit words containing a parity bit, 2 bits of sign/status matrix, 19 data bits, 2 bits describing where the data is coming from, and an 8 bit label.



ARINC 429 Word Format

P	SSM		MSB									Data									LSB	SDI		LSB			Label				MSB
32	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1

The SSM bits are the Sign/Status Matrix
The SDI is the Source/Destination Identifier which means different things depending on the label
The label is 8 bits which is grouped into a 2 bit octal digit and two 3 bit octal digits. The label describes how to interpret the other data in the word such as the actual data bits, and the SDI

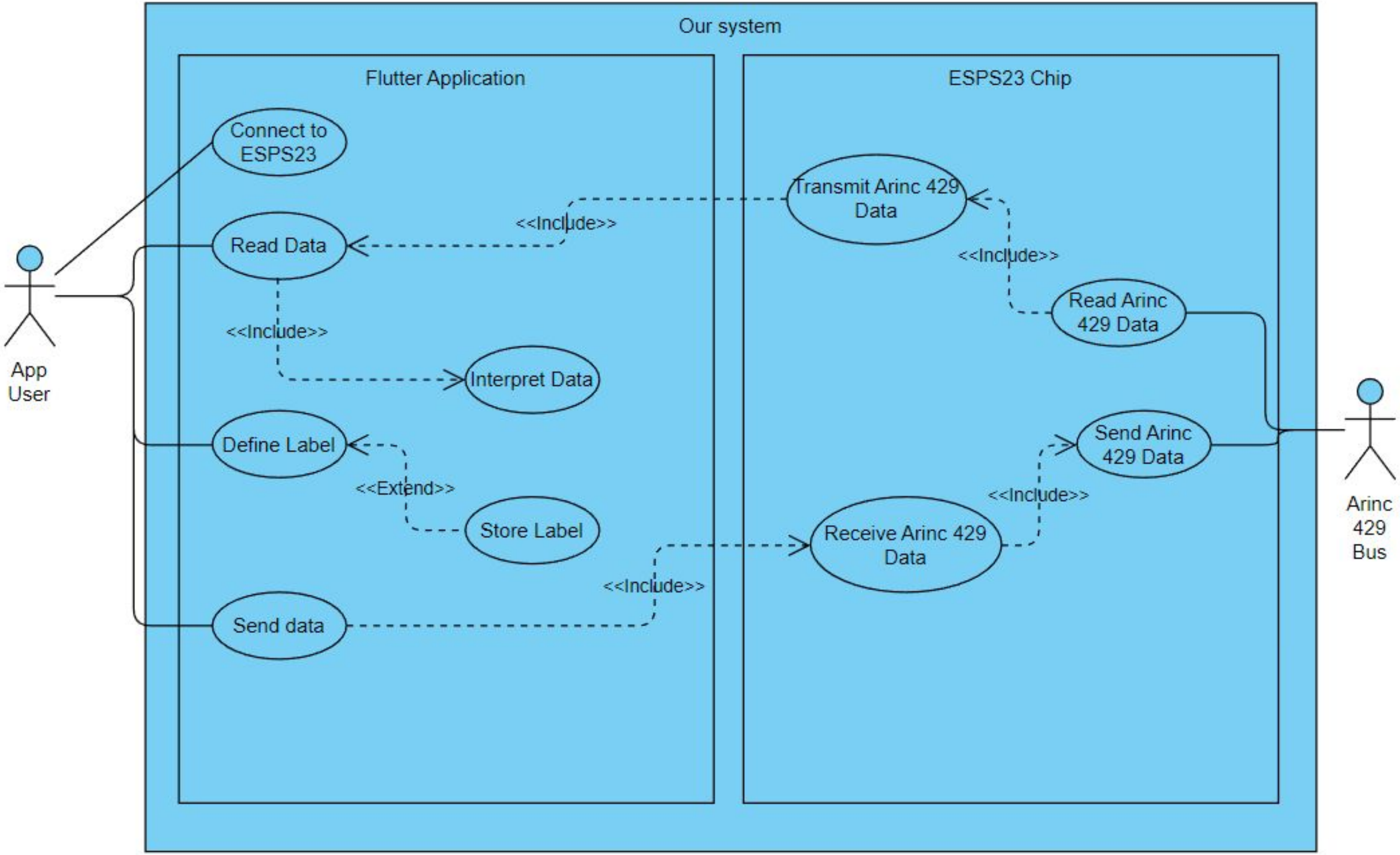
Bluetooth

The project requires that we transmit the data over Bluetooth Low Energy in order to transmit the data to the mobile application

Embedded Systems

The project requires us to write firmware for an embedded system to read the data from the Arinc 429 bus and transmit it over bluetooth

Users and Use Cases


Our system
Flutter Application
ESPS23 Chip
App User
Connect to ESPS23
Read Data
Interpret Data
Define Label
Store Label
Send data
<<Include>>
<<Include>>
<<Extend>>
<<Include>>
Transmit Arinc 429 Data
<<Include>>
Read Arinc 429 Data
Send Arinc 429 Data
<<Include>>
Receive Arinc 429 Data
Arinc 429 Bus