SAN FRANCISCO STATE UNIVERSITY
SCHOOL OF ENGINEERING

Engineering Design I Project Proposal

Fall 2022

Oct 15, 2022

Benediction Bora

Email: bbora@mail.sfsu.edu

ID#: 922175810

Project Proposal
Private Machine Learning with On-Device Training

Author: Benediction Bora
Email: bbora@mail.sfsu.edu

Problem

Machine Learning (ML) has achieved great success in the past decade for various intelligent applications such as computer vision [x], neural language processing [x], and speech recognition [x]. However, the process of training ML models is usually done on the cloud with powerful computing infrastructure (e.g., GPUs) due to the heavy computation cost of the ML models, and then deployed on the local devices for real-time inference. This process is computationally laborious, time-consuming, and expensive for cases where we want to implement ML on the edge/endpoints. **Training the ML model on the device has many advantages, including: 1) Privacy**: the data processing can happen on the user's device, which can apply learning algorithms to sensitive data such as photos and voice recordings. **2) Adaptation**: the ML model can continuously adapt to new data collected from the sensors, adjusting to personal preferences that are dynamic rather than static. For example, on our wrist, to analyze real-time health data, in an elevator or mechanical system to correct/analyze possible malfunctions, in a system with private data where information is critical to be uploaded to the cloud for analysis. And many more applications.

Goal

This project seeks to implement machine learning model training on a low-power hardware devices (e.g., microcontrollers) that lack rigorous computational power and cloud data direct and/or quick access. This approach of loading ML models on low-power devices is commonly known as TinyML. Specially, I will optimize and deploy one of the most representee machine learning or deep learning model called convolutional neural network (CNN) for the image classification task on the Sony spresense microcontroller. To achieve the on-device training, the neural network architecture will be optimized for efficient resource access by neural network compression [x].

Objective

- System can collect training data through the camera and/or on-board sensors
- System can use the removable flash memory to store data
- System can train CNN models to optimal accuracy
- System can update CNN model parameters based on real-time input (On-device Training)

Possible Implementation

- The system will be deployed on the Sony Spresense Board
- A Sony Spresense *camera* board will be used to capture the image in real time for on-device training
- TensorFlow lite will be used to load sample models for testing successful deployment
- Visual studio code shall be the preferred IDE, although the Arduino IDE could possibly be used in tandem
- User interaction and progress of the model training shall be observed via an LCD display
- For portability, the entire system will be operated by a battery

*Most of the work will be done on the Sony Spresense Microcontroller Board, therefore, the specifications will be laid out in detail below

Board Specifications (MAIN)

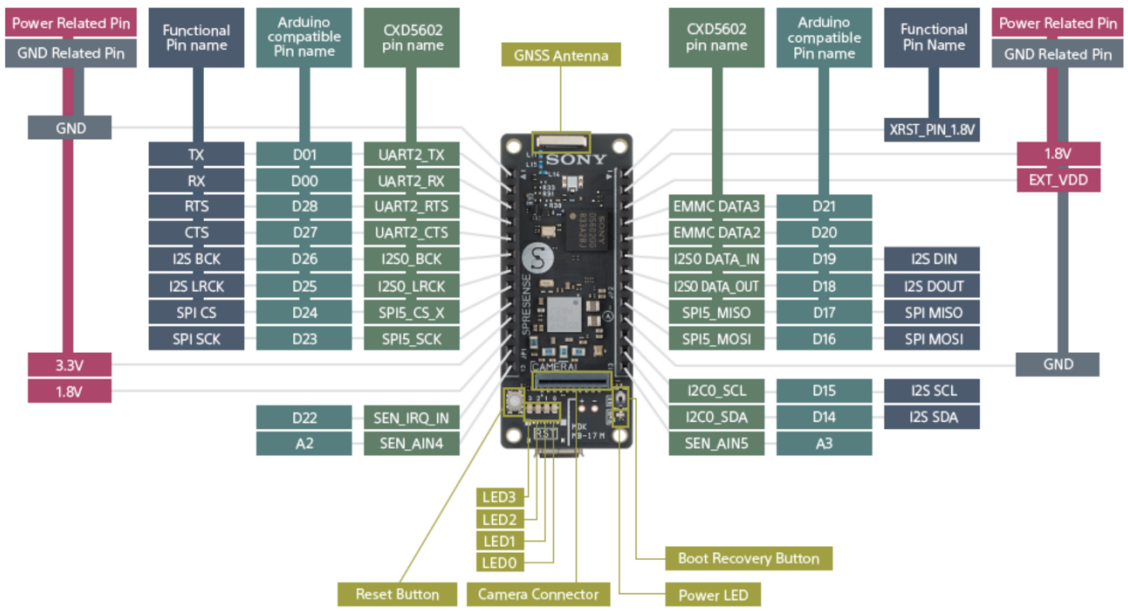
Model name	CXD5602PWBMAIN1
Size	**50.0mm x 20.6mm**
CPU	ARM Cortex-M4F x 6 cores
Maximum clock frequency	156 MHz
SRAM	1.5 MB
Flash Memory	8 MB
Digital input / output	GPIO, SPI, I2C, UART, I2S
Analog input	2 ch (0.7 V range)

GNSS	GPS (L1 C/A), GLONASS(L1OF), BeiDou(B1), Galileo (E1 CBOC), QZSS(L1 C/A, L1 S), SBAS(L1 C/A
Camera input	Dedicated parallel interface

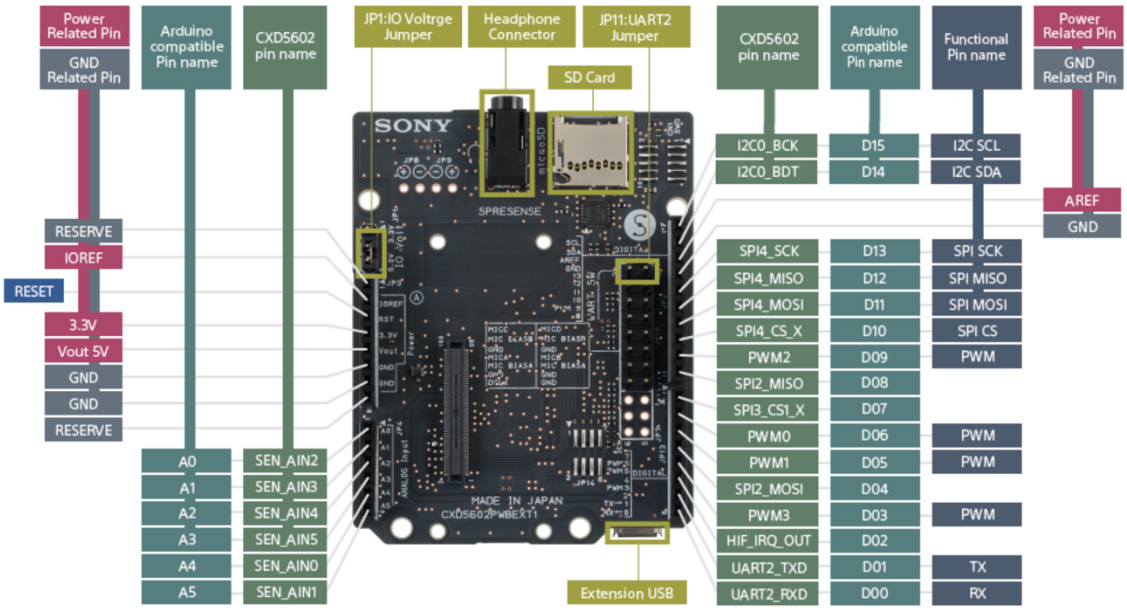
Extensions Board Specifications

Model name	CXD5602PWBEXT1
Size	68.6 mm x 53.3 mm
CPU	ARM Cortex-M4F x 6 cores
Audio input / output	4 Ch analog microphone input or 8 Ch digital microphone input, headphone output
Digital input / output	3.3 V or 5 V digital I/O
Analog input	6 Ch (5.0 V range)
External memory interface	microSD card slot

Pin Assignments (Main Board)



Pin Assignments (Extension Board)



Budget

Project Hardware Requirements		
Parts	Description	Price
CXD5602PWBMAIN1	Sony Spresense Main Board (price includes extension board and Camera)	$130.00
QVGA TFT SPI LCD Display (ILI9341)	LCD Display	$15.00
Software Requirements		
Visual Studio Code	Shall be used for loading software to a microcontroller board	open
TensorFlow Lite	Shall be used to train sample models and for converting models into C header files	open
Google Collab	May be used as secondary to TFL	open

References

Nil Llisterri, et al, 14 February, 2022. " On-Device Training of Machine Learning Models on Microcontrollers with Federated Learning".

Sony. https://developer.sony.com/develop/spresense/specifications/

Taro Yoshino, 4th June, 2022. Realtime person detection by Sony Spresense https://www.hackster.io/taroyoshino007/realtime-person-detection-by-sony-spresense-fde9c7