



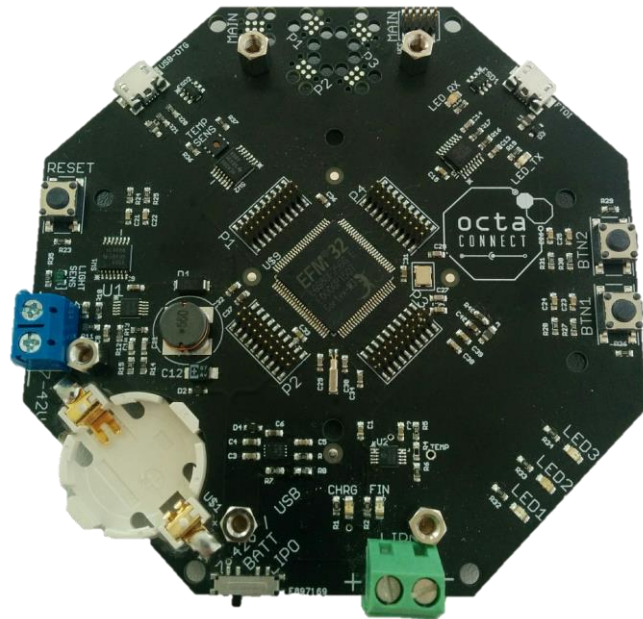
# mec

OCTA DOCUMENTATION

OCTA-GATEWAY

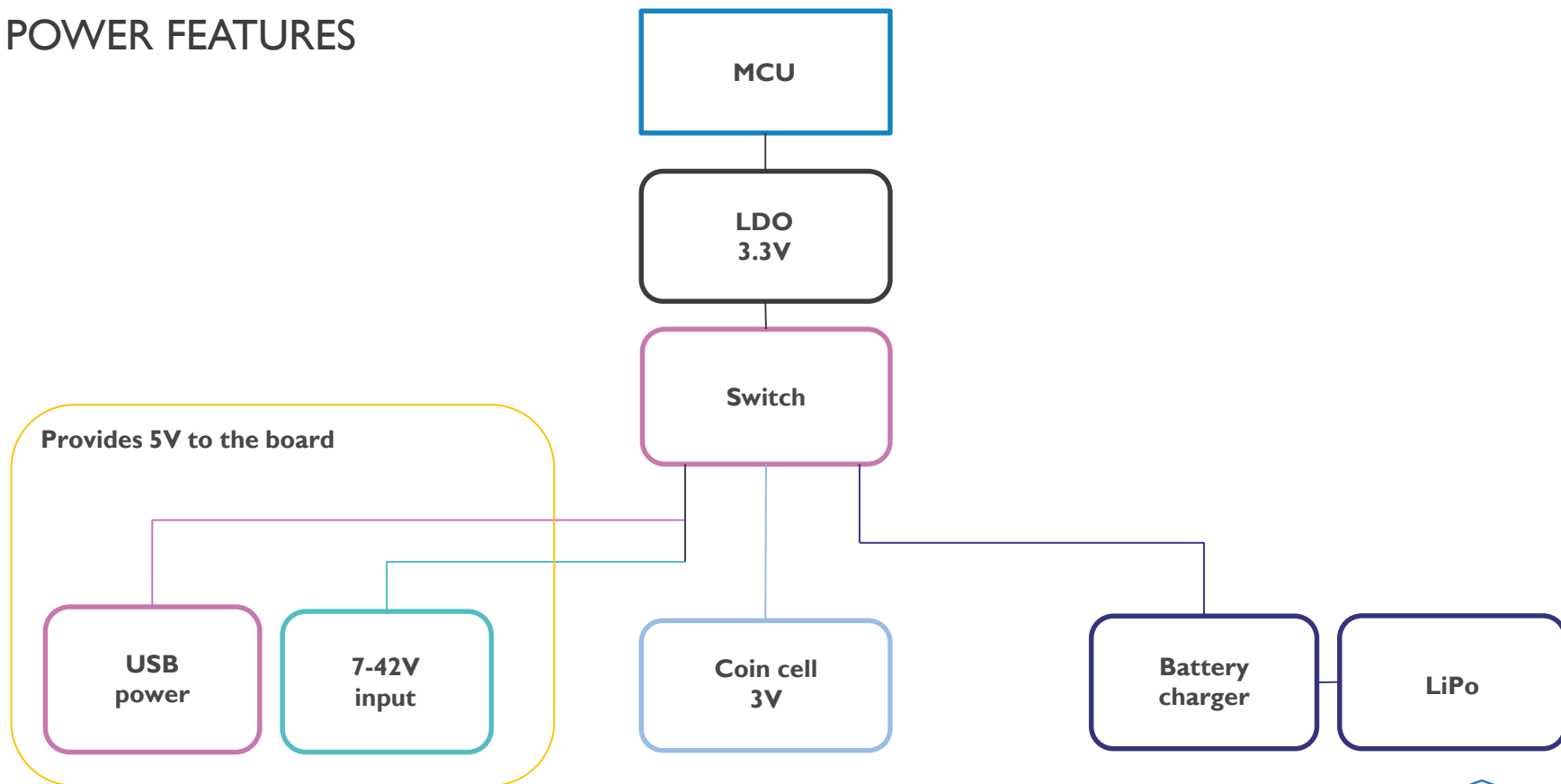
# OCTA-GATEWAY

## HW ARCHITECTURE



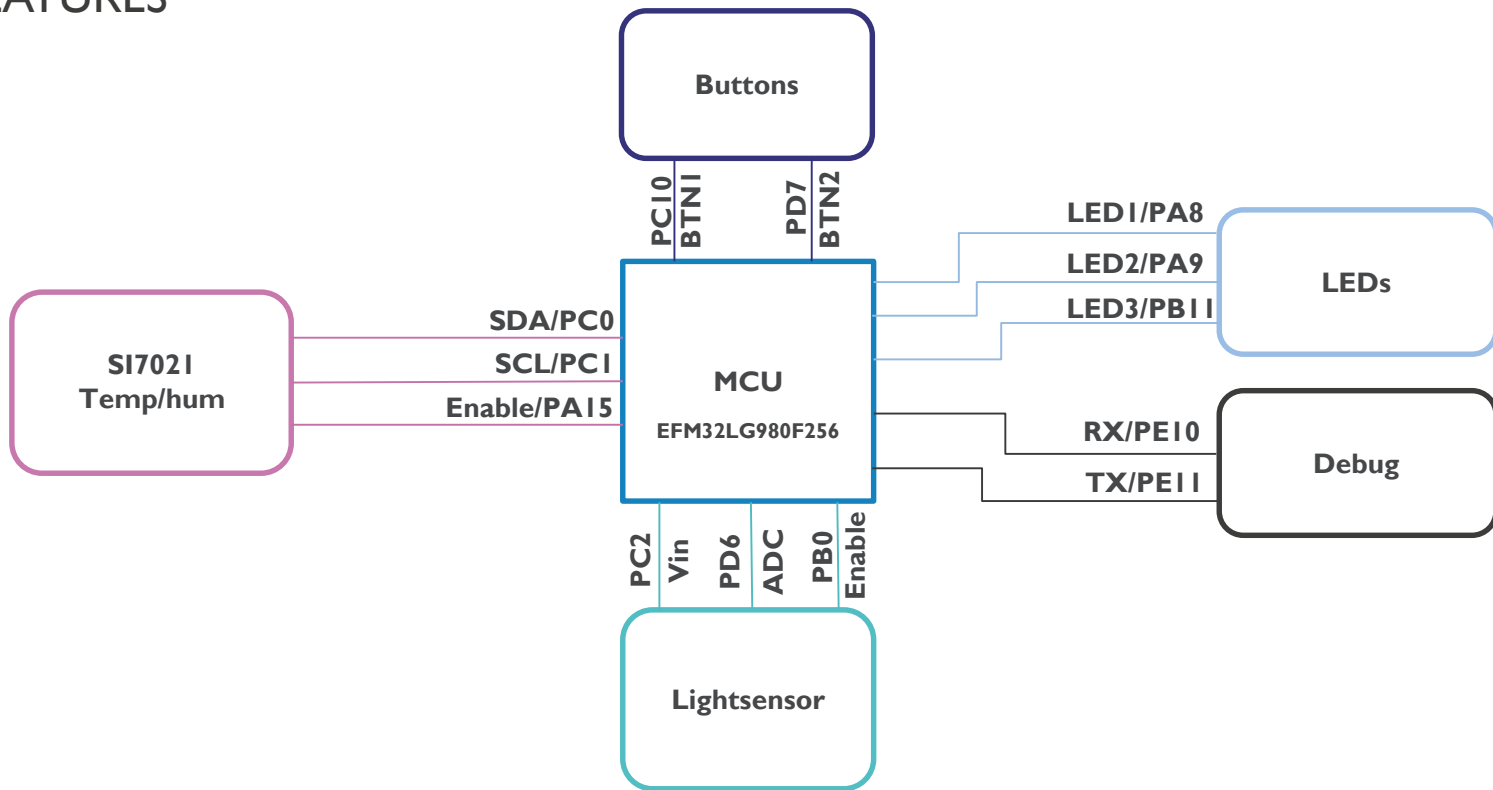
# OCTA-GATEWAY

## POWER FEATURES



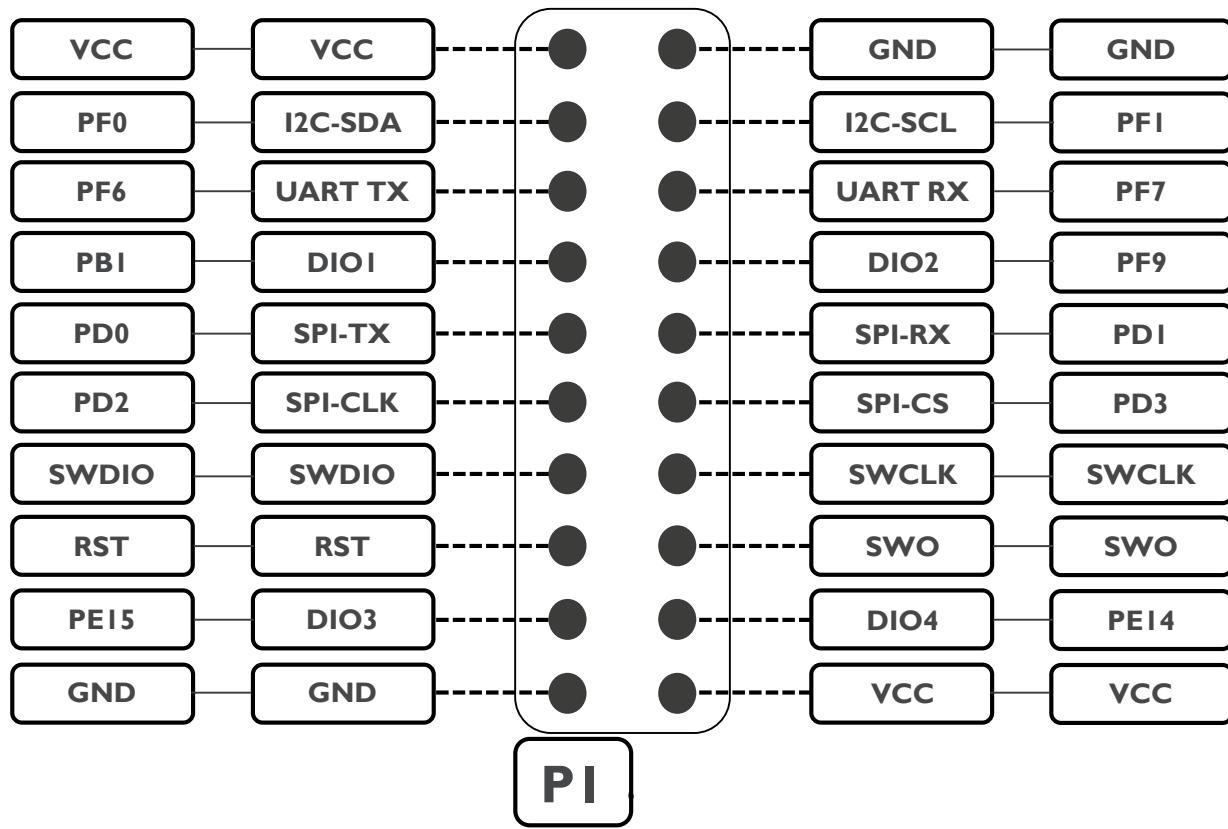
# OCTA-GATEWAY

## HW FEATURES



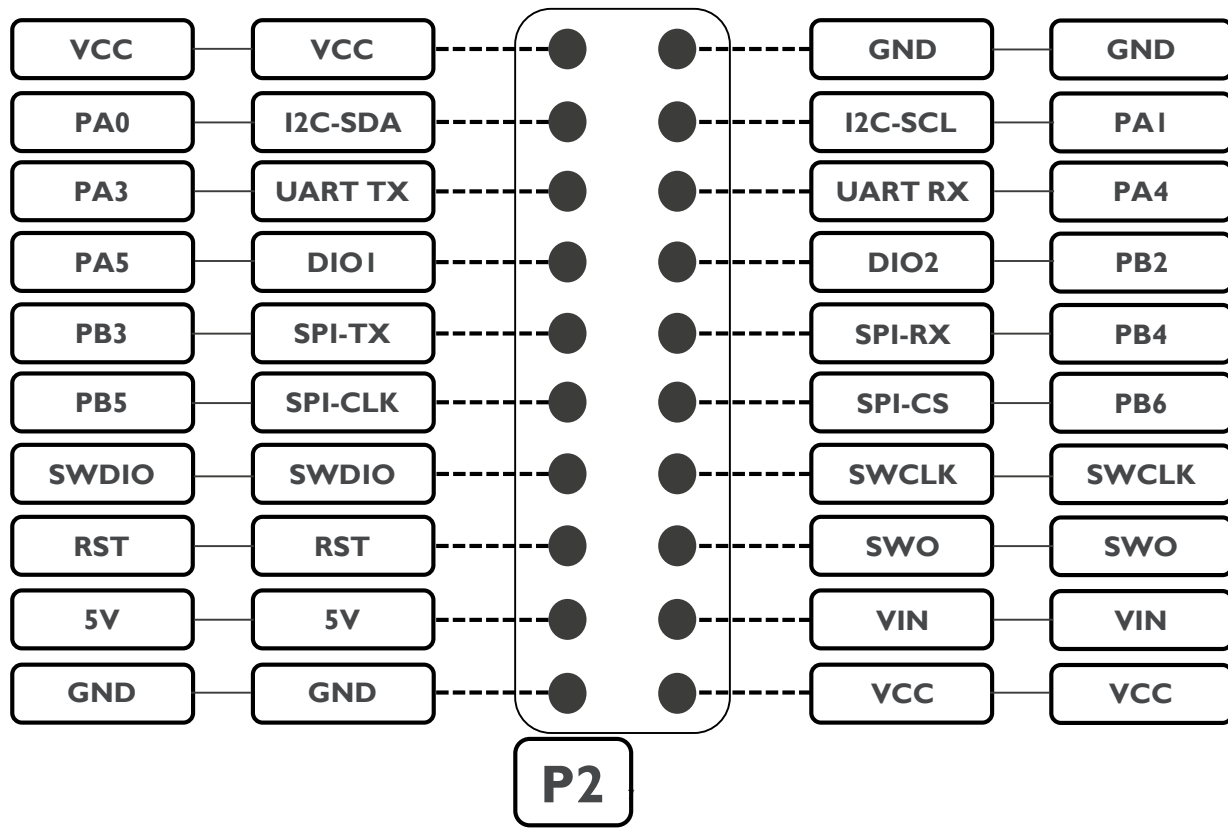
# OCTA-GATEWAY

## CONNECTOR PI



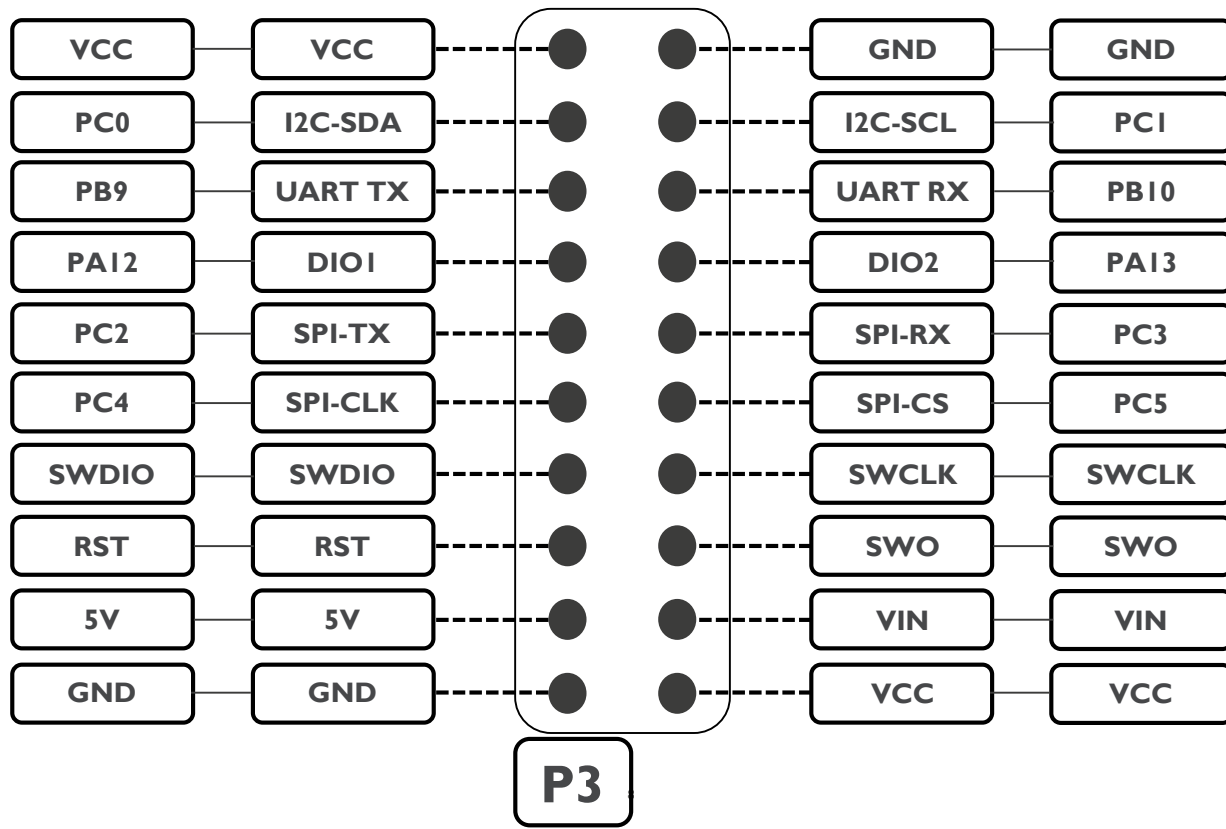
# OCTA-GATEWAY

## CONNECTOR P2



# OCTA-GATEWAY

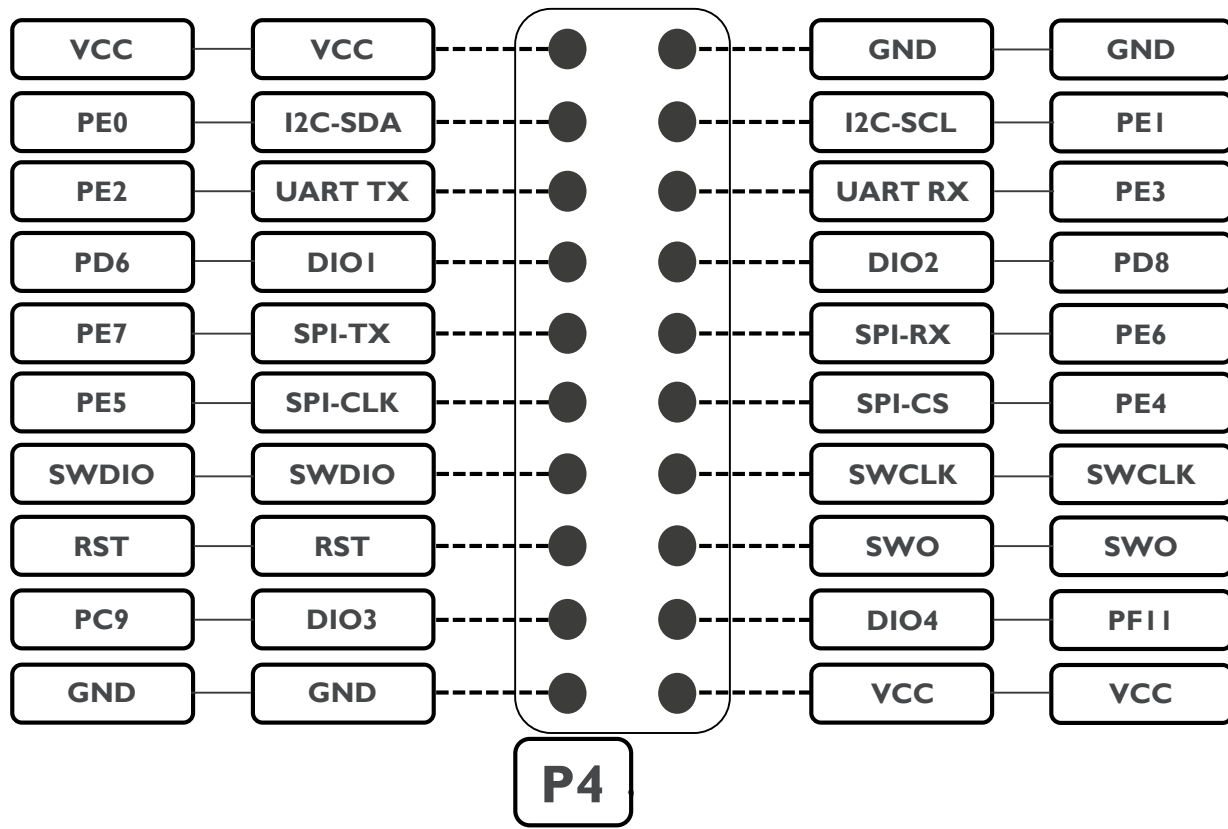
## CONNECTOR P3





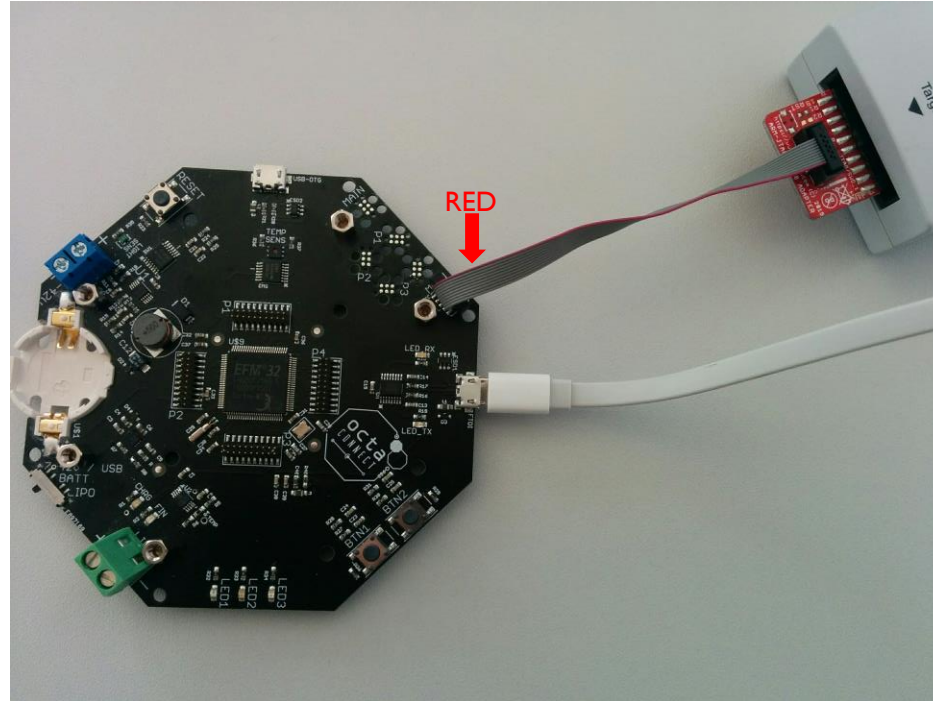
# OCTA-GATEWAY

## CONNECTOR P4



# OCTA-GATEWAY

## PROGRAMMER POSITION

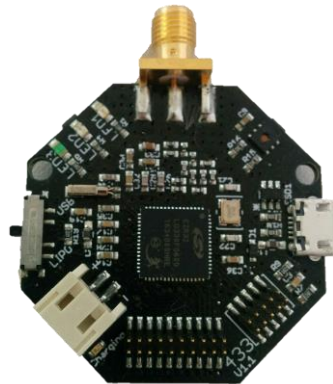


**\*needs to be powered externally**

OCTA-MINI

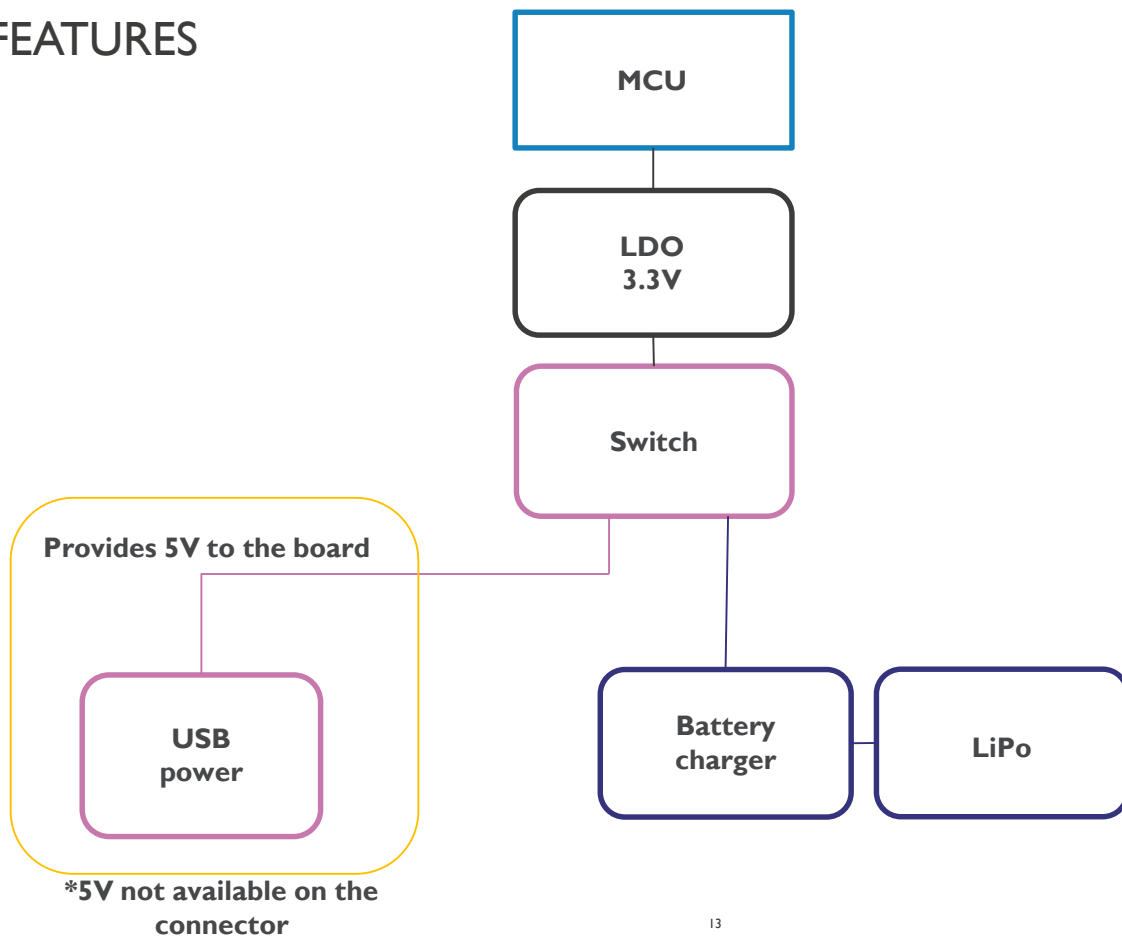
# OCTA-MINI

## HW ARCHITECTURE



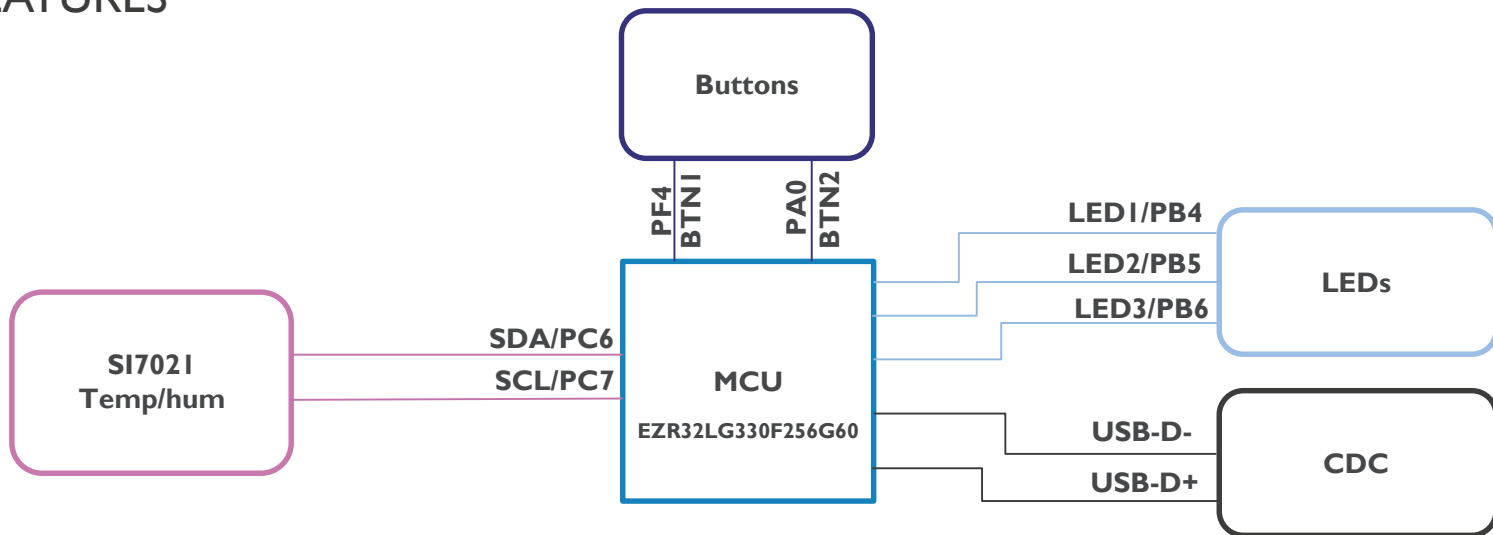
# OCTA-MINI

## POWER FEATURES

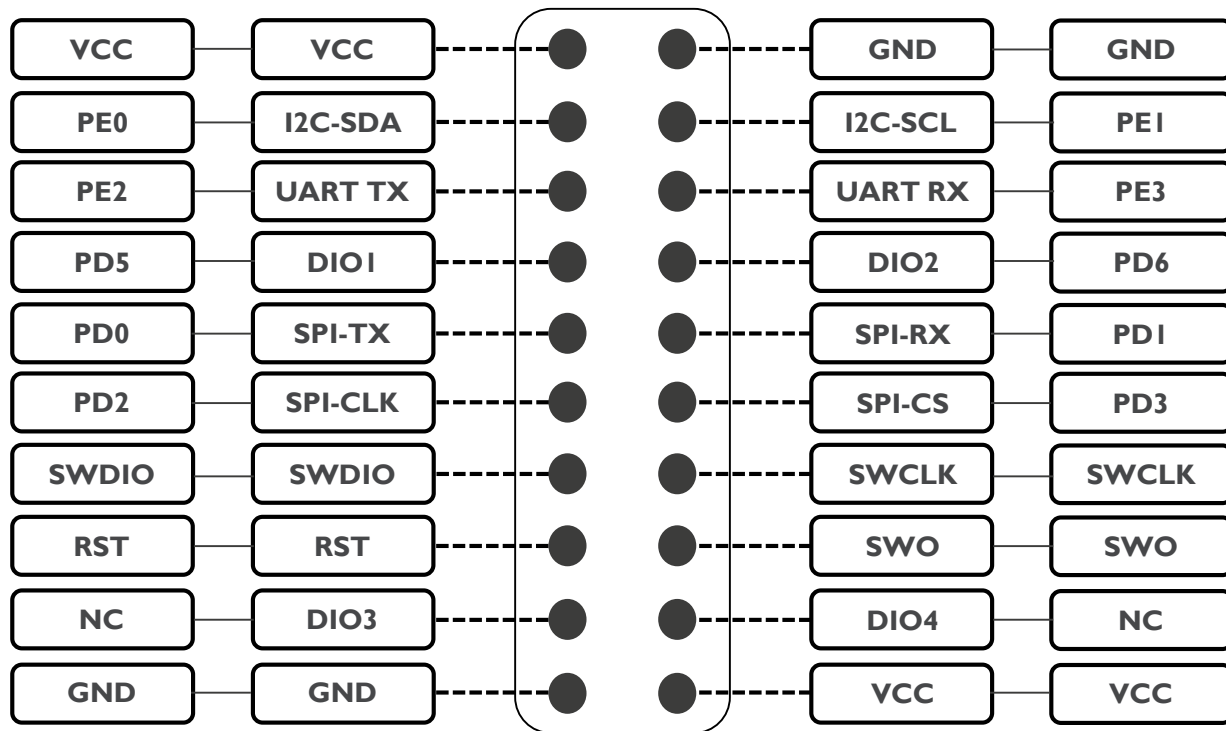


# OCTA-MINI

## HW FEATURES

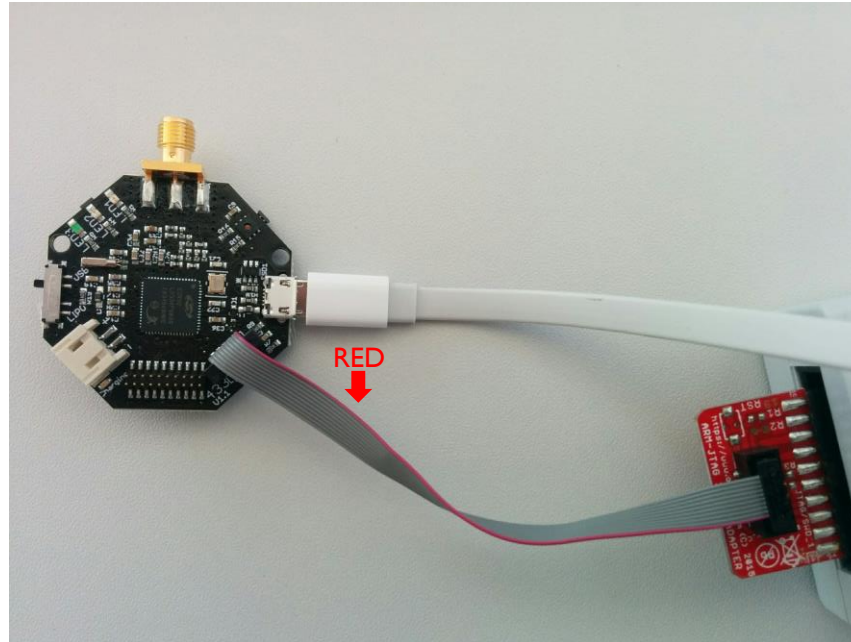


# OCTA-MINI CONNECTOR



# OCTA-MINI

## PROGRAMMER POSITION



\*needs to be powered externally



SOFTWARE

# SOFTWARE

## IDE ENVIRONMENT

- Simplicity studio from SiLabs is the most easiest way to start. Based on eclipse with a custom skin. GCC and GDB tools already installed.
- Other IDEs also possible (this requires more manually setup of extra tools):
  - Eclipse
  - IAR
  - ...

# SOFTWARE

## EXAMPLES

- Basic drivers and examples for the EFM32:  
<https://eewiki.net/display/microcontroller/Getting+Started+with+EFM32+Giant+Gecko+ARM+Cortex-M3>
- OSS7 DASH7 stack is already configured for the OCTA, custom Makefiles are already included in the project on Github:  
<https://github.com/MOSAIC-LoPoW/dash7-ap-open-source-stack>