MLPerf™ Tiny Deep Learning Benchmarks for Embedded Devices

Gap 9
Hardware System Setup

This document explains how to set up the hardware part of the TinyML Perf benchmark. The hardware installation includes:

- GAP9_EVK
- Energy Monitor (STMicro. LPM01A (v1.0.6)
- IO Manager (Arduino UNO)
- Level shifter (BSS138)

Performance and Accuracy

To measure accuracy on GAP9, the setup is the following:

GAP9_EVK is directly connected to the host runner through an USB <-> UART key, the key must be transmitting at 1.8 V or level shifters must be used. Gap9 EVK Pin Connection:

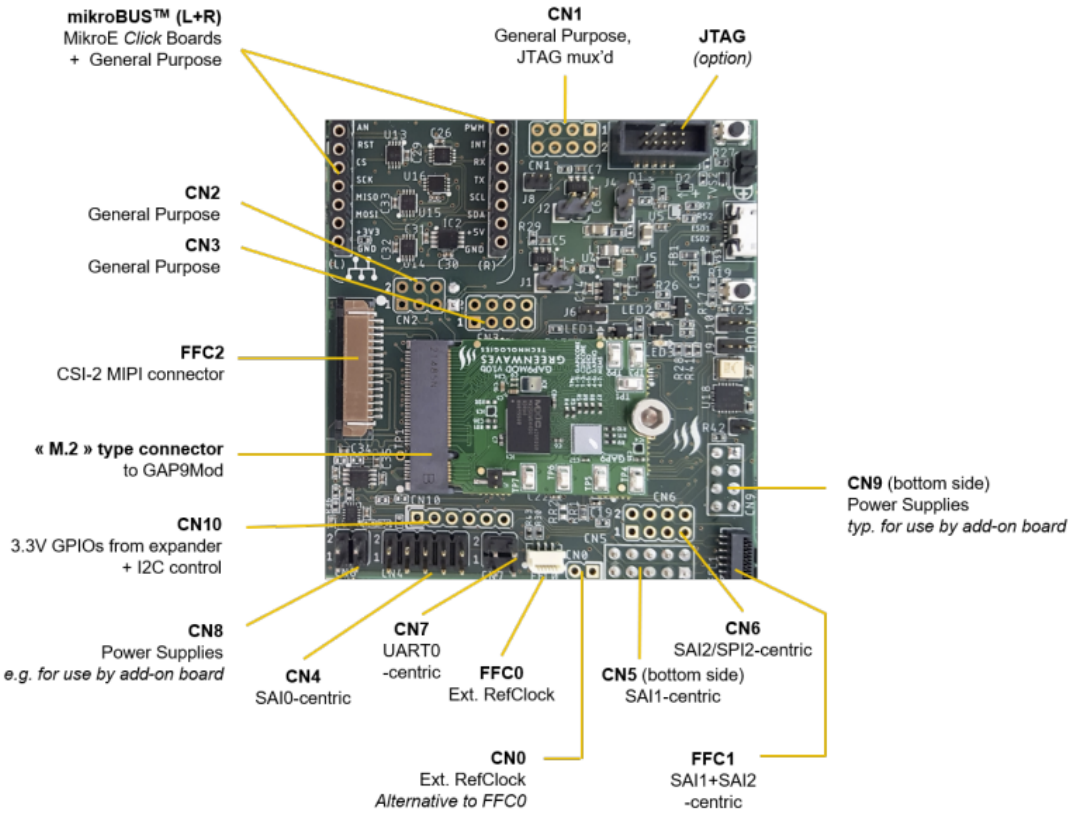


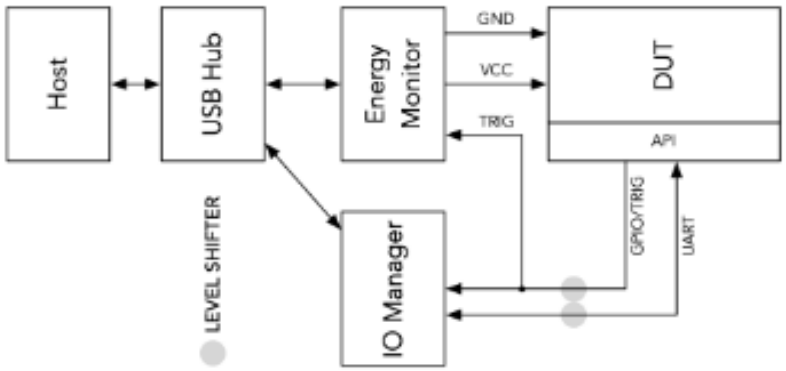
Fig. 8 – Expansion Connectors

CN8 pin3 - GND (or elsewhere on the board)
CN7 pin4 - GAP UART TX
CN7 pin1 - GAP UART RX

NOTE: The code running on GAP can be flashed using the onboard FTDI Chip and there is no need to boot from flash.

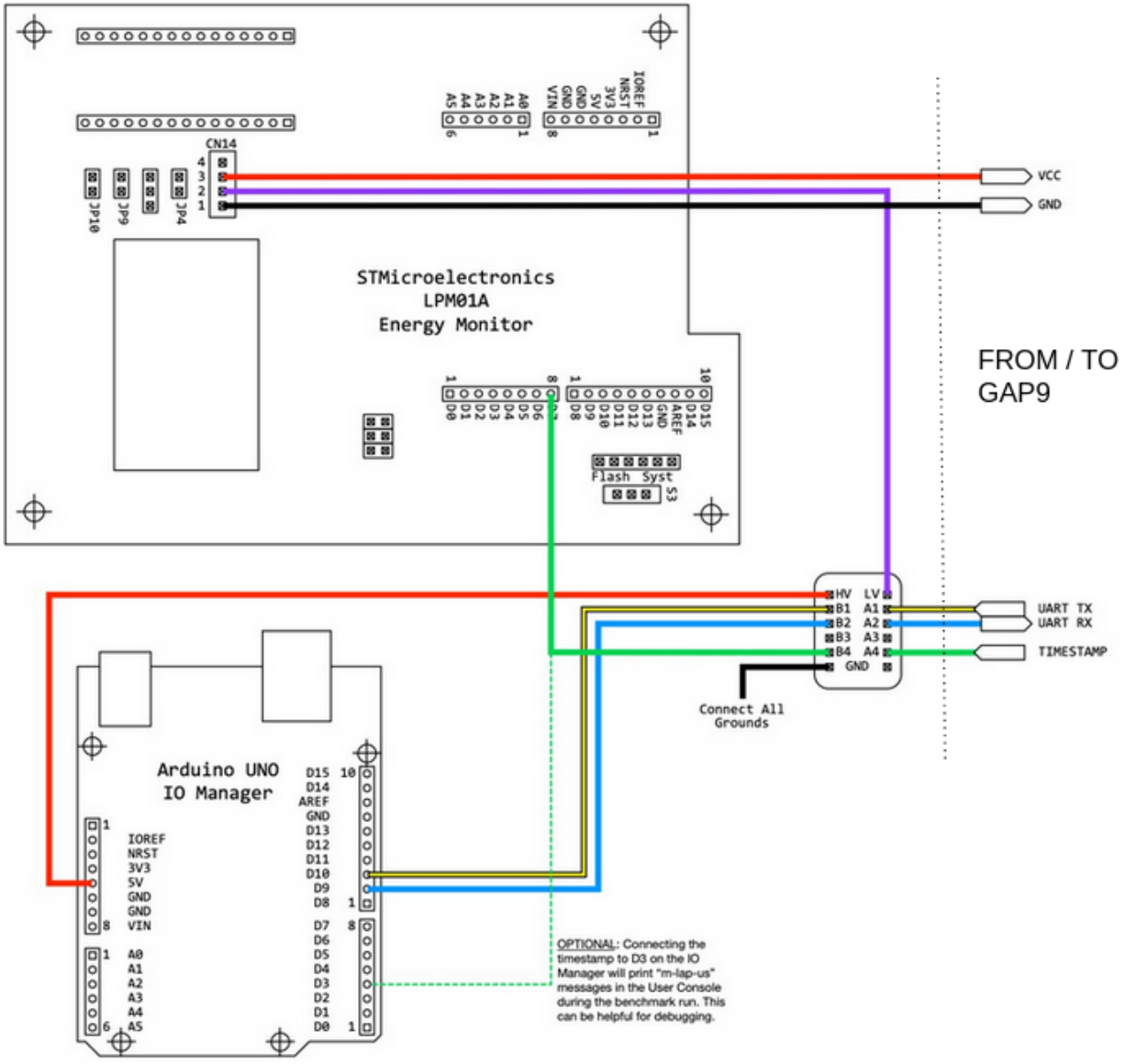
Latency and Energy

In order to measure latency and energy on GAP9, the setup is the following:



The Energy Monitor and the IO Manager are directly connected to the host runner through an USB cable. Then, GAP9_EVK UART and GPIO trigger (timestamp) are connected to the IO Manager through a level shifter 5V - 1V8. The Energy Monitor, in turn, powers the GAP9_EVK with 1.8V.

The following schematic representation shows the connections between the 4 modules.



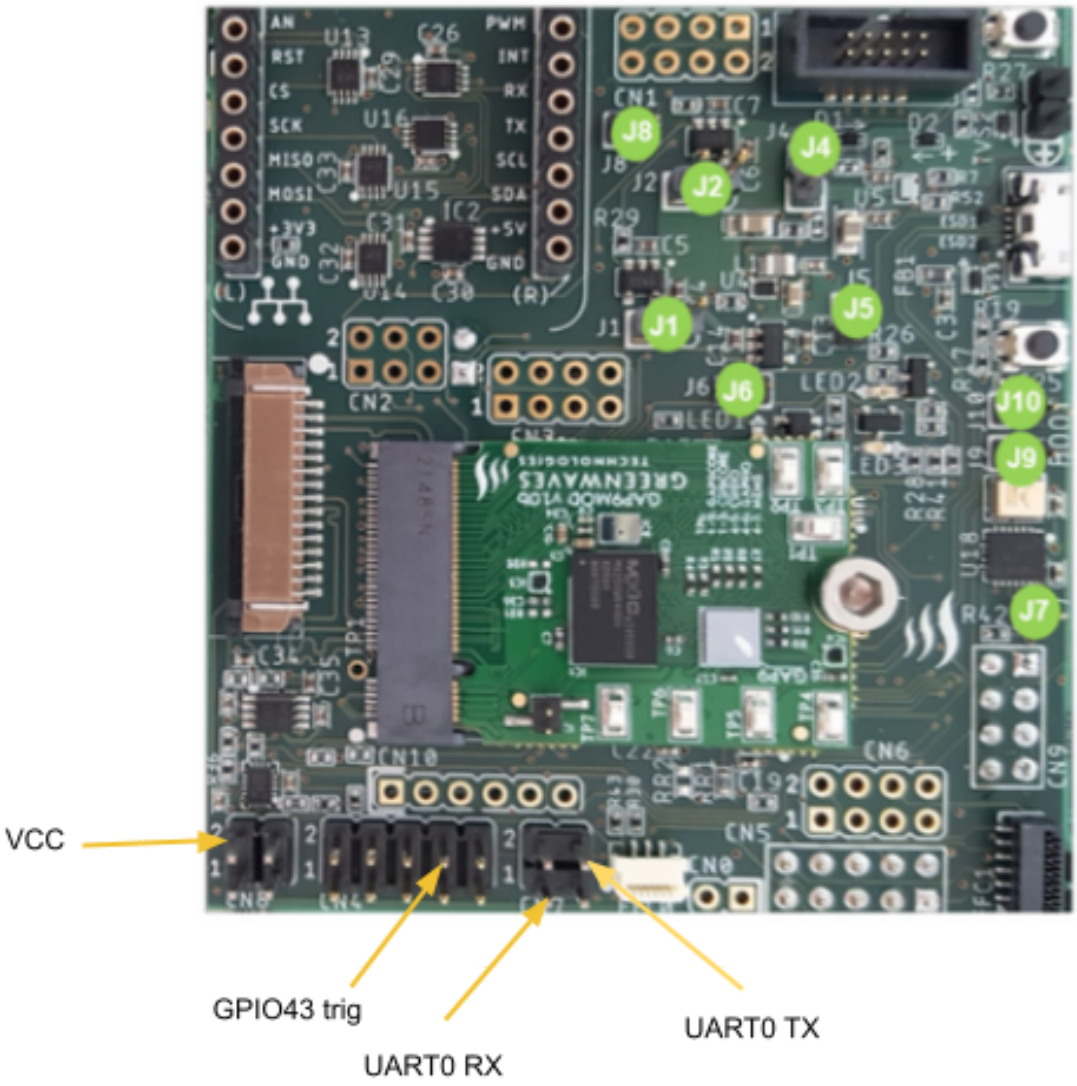
GAP9_EVK connection

In order to connect GAP9 to the benchmark, we are using the GAP9_EVK evaluation board.

These connections are:

- UART0 RX/TX used for the communication between host and DUT
- GPIO_A43 used for trigger (timestamp)
- VBAT used for power GAP9 during Energy and Latency measurements
- GND

The following picture describes where these connections are on the GAP9_EVK board.



CN8 pin2 - VCC 1.8 V
CN8 pin3 - GND (or elsewhere on the board)
CN4 pin8 - Timestamp, GPIO Trigger.
CN7 pin1 - GAP UART RX
CN7 pin4 - GAP UART TX

For the Latency and Energy mode, you will need to configure jumpers J5,J6 and J8 to power GAP9_EVK from the Energy Monitor:

- J5 - open circuit (remove jumper)
- J6 - closed on the “right”
- J8 - open circuit (remove jumper)

NOTE
In order to properly measure the energy you need to configure the boot from Flash (options provided in the makefile for the code) and then set BOOT pins accordingly, with jumpers JP9 and JP10:

BOOT[1:0] =

- 00 (binary) → eFuse-defined boot source
- 01 → boot from JTAG
- 10 → boot from HyperFlash
- 11 → boot from eMRAM

You will also need to disconnect the resistor RR1 to disconnect the FTDI chip from GAP9 (in case you have an EVK with Embedded FTDI) and shortcut the resistor RR2. An external JTAG probe is then required to flash the image (i.e. Jlink, Olimex, …) or use another EVK to flash it.