

# GAPMod 3.x

GAP8 Centric Core Module with QSPI memories

## HARDWARE OVERVIEW

Rel. 1.2  
11-Aug-21

---

## CONTENTS

Disclaimer.....	2
Preamble - GAPMod 3.x vs. GAPMod 2.x vs. GAPMod1.2.....	3
1. Introduction.....	3
2. Board Description.....	3
3. GAP8.....	6
4. On-board power management.....	6
5. Flash and RAM Memories.....	8
6. Low-cost 8-bit (STM8) MCU.....	8
7. Programming / Debugging.....	8
8. GAPMod I/O Signal Assignments.....	8
REFERENCES.....	9
DOCUMENT HISTORY.....	10

## **Disclaimer**

This information is subject to change without notice.

Information on this document is provided “as is” without any warranty of any kind, either express or implied, including but not limited to, the implied warranties of merchantability, suitability for a particular purpose, or non-infringement. The information provided in this document is intended for informational purposes only. Information may be changed or updated without notice.

The GAPMod design files and associated documentation are provided under the Solderpad Hardware License Version 2.0, a copy of which can be obtained at:

<http://solderpad.org/licenses/SHL-2.0/>

## Preamble - GAPMod 3.x vs. GAPMod 2.x vs. GAPMod1.2

GAPMod comes in 3 variants:

- **GAPMod1.x** : bears a HyperBus Flash+RAM MCM

- **GAPMod2.x** : bears a Quad-SPI Flash and Quad-SPI RAM. In addition, some non-essential hardware that had been provisioned on GAPMod 1.x has been removed.

*HyperBus memory provides more bandwidth than QSPI but is more expensive and possibly less easy to source than QSPI memories (AN003, "GAP8 Hardware Integration Guide", elaborates a bit on external memory selection).*

- **GAPMod3.x** : mostly identical to GAPMod2.x, plus on-board 8-bit, low-cost MCU (STM8) usable as system WatchDog/RTC/DeepSleep controller. Also includes the ability to keep GAP8 crystal oscillator always on while all other supplies are switched off, which was not possible on GAPMod2.x.

Pin numbering between GAPMod1.2 and GAPMod2.x/3.x has been rationalized but pin compatibility is maintained to a large extent.

**\*\* Beware \*\*** - Pin A15 of GAP8 is used to program the STM8 and features a 220ohm pull-up to 1.8V. Make sure it is not driven low when not used else 8mA current consumption will follow.

### **Note for 2021 produced GAPMod 3.0, valid until further notice :**

*With the worldwide shortage of semi-conductor components, the low-cost 8-bit MCU (STM8) included in the original GAPMod 3.0 design has become unavailable from any distributor. As a result, this part is not assembled on the latest GAPMod 3.0 modules. This has only a minor impact on GAPMod functionalities – specifically, this removes the ability to switch off the entire module under control of the on-board STM8 used as system RTC/WatchDog.*

## 1. Introduction

GAPMod is a core module architected around GreenWaves Technologies' GAP8 chip, an ultra-low power Application Processor for the IoT (*Ref[1]*).

GAPMod is intended to facilitate and speed up development of applications around GAP8.

It packs on a small board (26mmx36mm) the "invariant" part of any application based on GAP8 (GAP8 chip, external Flash+RAM memory, associated decoupling caps, crystal oscillator, passives for internal DC-DC, etc.). This core hardware happens to also be, in many use cases, the most demanding part of the overall design in terms of precautions to take and in terms of required PCB technology.

With GAPMod, an application PCB designer can simply drop the module in his/her design and rely on comparatively relaxed Design Rules and reasonably priced PCB technology for his/her own application-specific motherboard.

## 2. Board Description

GAPMod comes in the form of a 4-layer PCB to be assembled using SMT (Surface Mount Technology) on a target board. All components therefore sit on one side while the other side bears the SMT pads. As it implements a “castellated” pinout with mm-range pitch, it can also be hand-soldered.

Almost all GAP8 I/Os that are not dedicated to interconnecting chips present on GAPMod are made available to the target application board as GAPMod I/Os. For maximum flexibility, all core and I/O power supplies are kept independent and to be provided by the application motherboard, with the exception of that dedicated to the on-chip crystal oscillator.

Figure 1 highlights the major parts that constitute GAPMod, while Figure 2 details physical dimensions of the board and location plus size of its I/O pads.

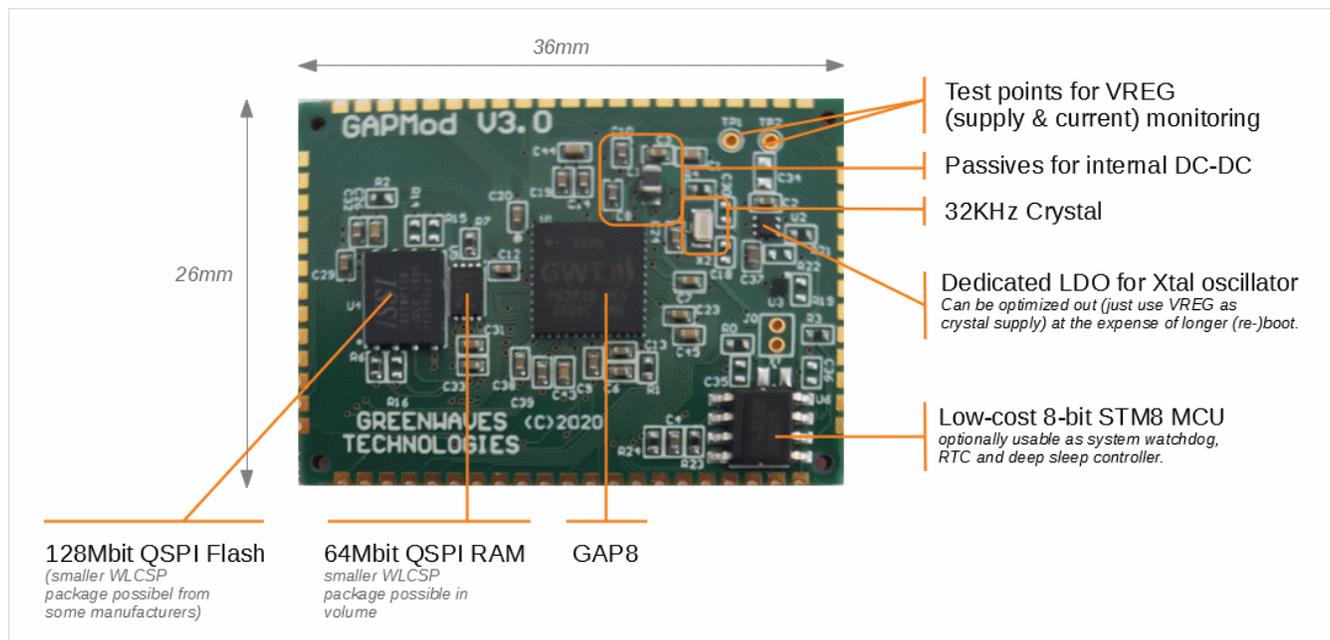


Fig. 1: Anatomy of GAPMod 3.0

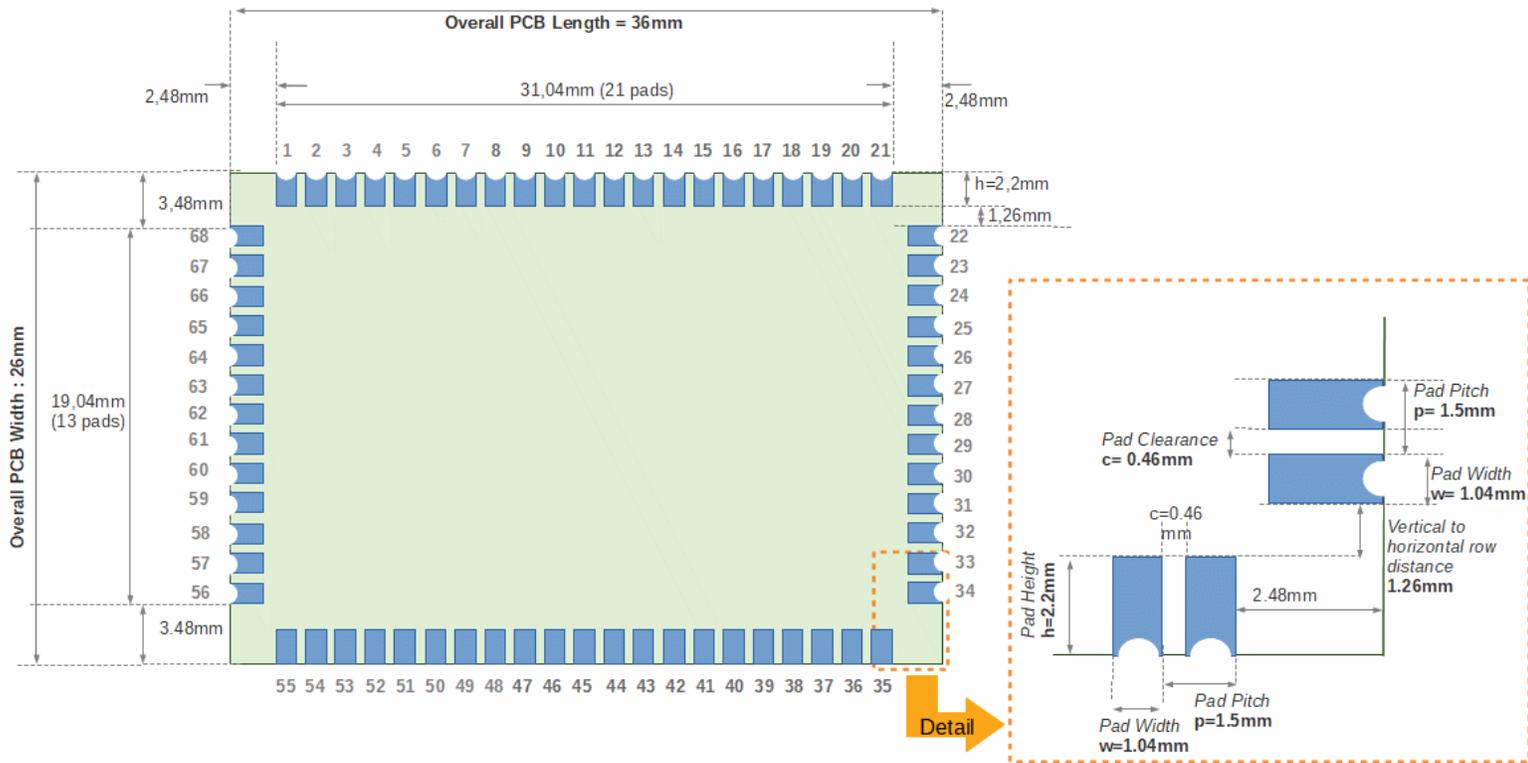


Fig. 2: GAPMod 3.0 Physical Information, TOP (see-through) View

### 3. GAP8

Central to the board is the GAP8 chip, an ultra-low power Application Processor for the IoT, bringing intelligence at the very edge of networks – that is, just next to sensors. GAP8 brings the ability to analyze and classify data from sensors (images, sounds, vibrations etc.) at very low cost and very low power, by extracting from sensor data: condensed, meaningful information which can then be sent efficiently over a network.

GAP8 consists on an MCU-like part (a core plus standard peripherals), a processing intensive part (8 identical parallel cores plus a hardware accelerator specialized in Convolutional Neural Network operations), with on-chip L1 and L2 RAM and a specialized DMA. The cores are 32-bit RISC-V enhanced with specialized instructions.

Refer to the GAP8 Product Brief and Datasheet (*Ref.1*) for details.

### 4. On-board power management

Refer to Fig. 3 for a graphical illustration of power management on GAPMod.

As mentioned earlier, most power supplies are to be provided by the application board in the sake of flexibility.

However, the crystal oscillator is powered on-board. By default it is powered off VREG, the 0.8V-1.2V voltage generated by the internal DC-DC converter – this is the lowest cost approach. A dedicated 1.2V LDO is also provisioned on-board, to provide a potentially cleaner and more stable voltage; also in case operating the crystal oscillator at 0.8V (limit of specification) in sleep mode does not work as reliably as expected (not fully characterized at the time of writing). Selection is through an assembly option (populating or not some parts).

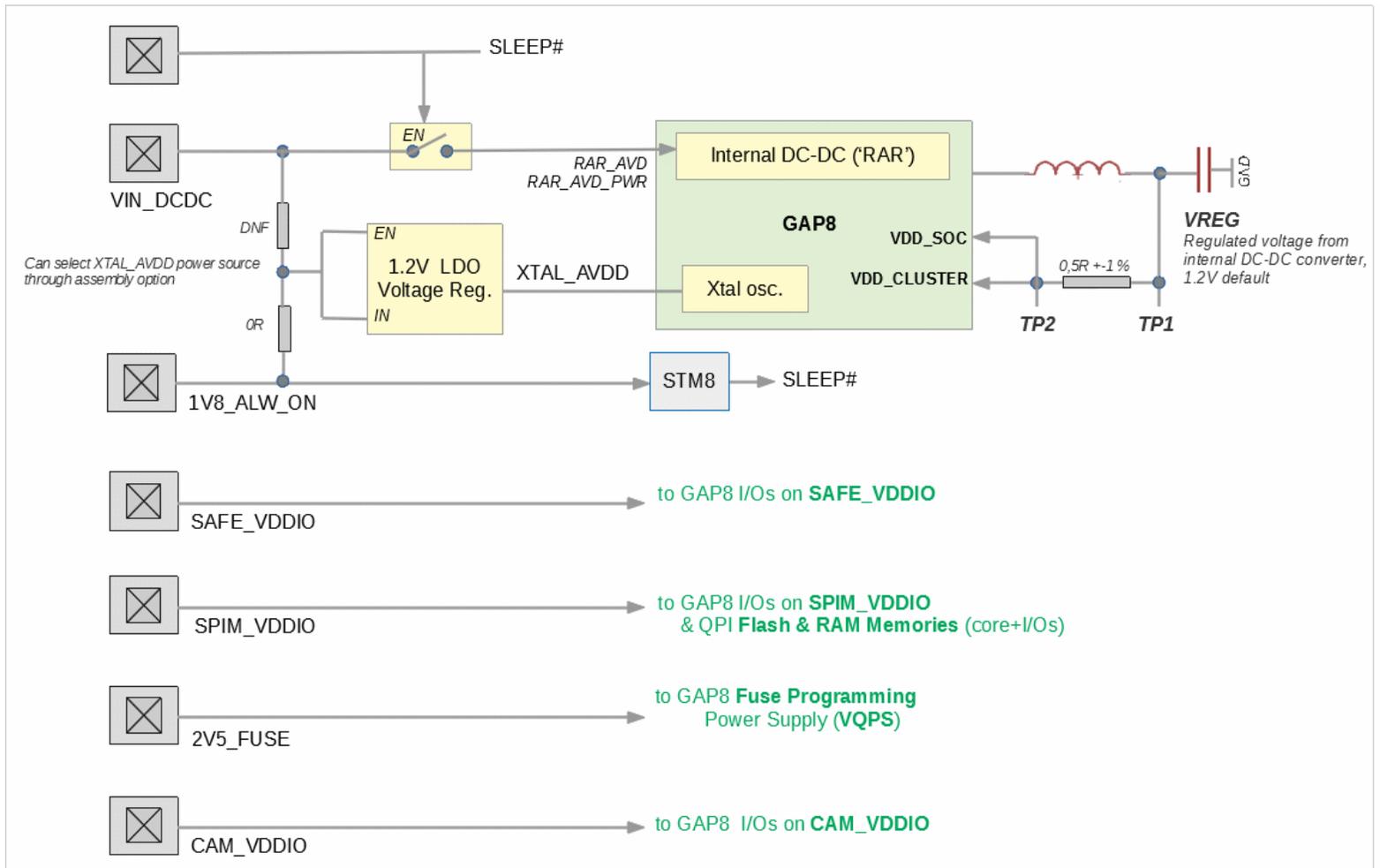


Fig. 3 – On-board power supply generation and control (notional)

### – Test Points / Power Supply Monitoring –

Two testpoints, TP1 and TP2, are provided to monitor the regulated voltage at output of internal DC-DC converter and the current drawn by the “SoC” part (a.k.a FC, Fabric Controller) and the “Cluster” part (i.e. full digital core, excluding I/Os and oscillator) of the GAP8 chip.

The current flows through a 0.5 ohm (+-1%) sense resistor placed on the current path between TP1 and TP2. Therefore the current consumption (in mA) is twice (2x) the value of the measured voltage drop (in mV) since  $I=U/R$ . The sense resistance is deliberately small to avoid severe supply voltage fluctuations when there are significant instantaneous current changes.

The current consumption is measured at output of the DC-DC converter. The efficiency of DC-DC conversion will determine how this translate as actual power drawn by GAP8 (SOC aka FC +Cluster) at system level.

In normal run mode, the regulated voltage is 1.2V by default and can be set to any value between 1.0V and 1.2V in increments of 50mV by adequate programming. When GAP8 enters sleep mode, the regulated voltage drops to 0.8V.

## 5. Flash and RAM Memories

GAPMod 3.x implements a 128Mbit Flash Memory (ISSI IS25WP128\_JKLE on initial batch, possibly equivalent chip on next ones) and a 64Mbit pseudo-static RAM (APMemory APS6404L-SQRH-ZR), both on the QPI (Quad-SPI) bus. They require 1.8V nominal power supply (1,7V min, 1,95V max).

The QPI memory bus is on the SPIM0 interface of GAP8 (see datasheet [Ref.1]). The Flash is addressed by asserting SPIM0\_CS0 (pin B8) while the RAM is addressed by asserting SPIM0\_CS1 (pin A8).

## 6. Low-cost 8-bit (STM8) MCU

A low-cost 8-bit MCU (STM8L050J3) is provisioned on the GAPMod 3.x core module to optionally be used, with appropriate firmware, to act as system-wide WatchDog and/or RTC and/or to generate a SLEEP# signal intended to enable/disable power supplies in the system. On GAPMod3.0 itself, when this signal is asserted (low), GAP8's internal DC-DC converter is no longer powered and therefore all digital logic inside GAP8 is off. In addition, that SLEEP# signal is provided on a pin of GAPMod3.0 so that other power supplies generated on the application motherboard can also be switched off under its control.

To achieve ultra-low power consumption, the firmware running on STM8 should enter active run mode only occasionally and stay in “halt” or “active halt” (with RTC on) most of the time (*active halt* mode consumes about 1uA).

If the STM8 is not programmed, it will execute an endless idle loop and consume about 1mA or so. This should be taken into account if performing system level power consumption measurements. To minimize power consumption when STM8 is unused, it should be programmed to enter a permanent *halt* state.

The STM8 can be programmed from GAP8 emulating the SWIM interface used by this small MCU. This protocol relies on a single data line in open-drain mode and the required pull-up resistance is rather low. In this implementation, GAP8 pin A15 is used as SWIM data line with a 220 ohm pull-up to 1.8V. Outside STM8 programming, pin A15 should be kept at high impedance (input mode with no pull), else significant current consumption (8mA) will follow.

## 7. Programming / Debugging

Code can be downloaded into GAP8 internal RAM or into the Flash Memory present on GAPMod through its JTAG interface. This is done using the **standard GAP8 SDK**, please refer to its documentation [Ref.2].

## 8. GAPMod I/O Signal Assignments

Please refer to Table “GAPMod I/Os” (file *GAPMod\_v3.x\_IO\_Definition.xls*, provided as a separate spreadsheet ([Ref.3]).

Pin compatibility with GAPMod1.2 and GAPMod2.x is mostly (but not fully) preserved, so that only limited updates would be required on a motherboard to switch from one variant to another.

## REFERENCES

*Ref[1] -*

Product Brief “GAP8 IoT Application Processor” – available from [greenwaves-technologies.com](https://greenwaves-technologies.com)  
DataSheet “GAP8 Hardware Reference Manual” – available from [greenwaves-technologies.com](https://greenwaves-technologies.com)  
through: <https://greenwaves-technologies.com/sdk-manuals/> > Hardware Documentation > GAP8  
Application Processor Chip

*Ref[2] -*

GreenWaves GAP8 SDK & Manuals – available from [greenwaves-technologies.com](https://greenwaves-technologies.com)  
through: <https://greenwaves-technologies.com/sdk-manuals/> > SDK Documentation

*Ref[3] -*

Functional Definition of GAPMod 2.x I/Os :  
Spreadsheet *GAPMod\_v3.0\_IO\_Definition.xlsx* available from GreenWaves Technologies

## **DOCUMENT HISTORY**

Rel.1.0 – August, 2020 (XC)

Initial Release

Rel.1.1 – April, 2021 (XC)

Highlight presence of pull up to 1.8V on pin A15 – hence significant current consumption if driven low

Rel.1.2 – Aug, 2021 (XC)

Added note in preamble about absence of auxiliary STM8 on latest GAPMod3.0