SAMA7G5 Series Power Consumption and Thermal Considerations

AN4797

www.microchip.com Product Pages: SAMA7G54

Scope

This application note discusses both the power consumption and thermal considerations of the SAMA7G5 Series processor. It provides system designers with a reminder of the basic thermal theory, a set of data measured on the SAMA7G54 standalone device and the SAMA7G5 Series System-in-Package (SiP) devices, and some thermal design guidance on both hardware and software aspects.

Reference Documents

Type	Title	Literature No.	Available
Data sheet	SAMA7G5 Series	DS60001765	www.microchip.com
Data sheet	SAMA7G5 Series SiP	DS50003577	www.microchip.com
Application note	SAMA7G5/D6 Series Product Lifetime Estimation	DS00004532	www.microchip.com
Application note	SAMA7G5 Series Temperature Sensor Calibration	DS00004530	www.microchip.com
User's guide	SAMA7G54-EK	DS50003273	www.microchip.com
Data sheet	PAC1934	DS20005850	www.microchip.com
Web page	Linux4SAM home page	–	www.linux4sam.org

1. Introduction

The device data sheet specifies a maximum operating junction temperature (T_J) (105°C for industrial devices, 125°C for extended industrial and automotive devices). This limit must be respected to ensure both correct operation and long lifetime(1) for the device. For this purpose, a thermal study should be included in the design process to verify the device operating temperature in the application and, if required, to take corrective action if this temperature is deemed too high.

Note:

1. For detailed information, refer to the “SAMA7G5/D6 Series Product Lifetime Estimation” application note. See Reference Documents.

This document is organized in the following way:

- First, the basic thermal theory applying to semiconductor devices is reminded.
- Then, the reader is provided with thermal measurements carried out on the SAMA7G54 evaluation kit and on SAMA7G5 Series SiP devices mounted on a similar board.
- Finally, some software and hardware recommendations are provided to help system designers optimize applications from a thermal performance perspective.

In addition, Appendix A provides insights on Linux® operation with regards to these thermal aspects.

2. Basic Thermal Concepts

2.1 Junction Temperature

The junction temperature (T_J) of a semiconductor device is the temperature of the silicon die embedded in the device package. T_J depends on both the external environment (PCB design, ambient temperature, air flow speed, presence or absence of heatsink, etc.) and the internal power consumption. Silicon vendors always specify the T_J operating range on which the device data sheet parameters apply. As an example, for SAMA7G54 industrial grade devices, T_J ranges from -40°C to +105°C and for SAMA7G54 extended industrial or automotive grade devices, T_J ranges from -40 to +125°C.

Even though SAMA7G5 devices can be operated up to their upper temperature limit, it is good practice to operate them at the lowest possible temperature for the following reasons:

1. As described in the ”SAMA7G5/D6 Series Product Lifetime Estimation” application note, a lower junction temperature entails slower ageing mechanisms and hence a longer lifetime.
2. Such precaution reduces the overall power consumption of the device, which is appreciated in any power-constrained applications, and which avoids entering a vicious circle where more power means a higher junction temperature and therefore a higher leakage power.

SAMA7G5 devices feature an integrated temperature sensor to measure their junction temperature, with an accuracy of ±5°C. Microchip recommends to periodically measure T_J in any application. The frequency of these measurements is application-dependent, and should therefore be tuned to each specific case. For further details on the temperature sensor, refer to:

- Application note “SAMA7G5 Series Temperature Sensor Calibration” (see Reference Documents)
- Appendix A for Linux-specific information (temperature reading, etc.)

2.2 Thermal Equations

The generic equation to compute T_J is the following:

$$T_J = T_A + R_{J\text{-}A} \times P_{TOT}$$

where:

- T_J is the junction temperature.
- T_A is the ambient air temperature.
- $R_{J\text{-}A}$ is the junction-to-air thermal resistance (see Junction-to-Air Thermal Resistance for details).
- P_{TOT} is the total power consumed in the product.

The device data sheet limit on T_J (T_{J_Limit}) induces a limit on T_A, as shown in the following equation:

$$T_{A\_MAX} = T_{J\_Limit} - R_{J\text{-}A} \times P_{TOT}$$

The maximum ambient temperature (T_{A_MAX}) an application can support is a function of T_{J_Limit}, the total power dissipation and the junction-to-air thermal resistance.

2.3 Junction-to-Air Thermal Resistance

The thermal resistance parameter ($R_{J\text{-}A}$) characterizes the heat transfer (or heat flow) between the silicon die, where the heat is generated, and the external ambient air surrounding the IC. As the heat flows from the IC in all possible directions, one part of it is conducted (by physical contact) to the PCB and then dissipated in ambient air, and the other part is conducted directly to the device’s package edges where it is dissipated in ambient air as well.

For this reason, $R_{J\text{-}A}$ strongly depends on the following parameters:

- PCB layout (PCB size, copper routing area, number of layers, proximity of these layers, etc.)

- Presence or absence of a heat sink
- Presence or absence of air flow

The lower the $R_{J\text{-}A}$, the more efficient the heat transfer between the silicon die and ambient air. For example, $R_{J\text{-}A}$ = 30°C/W means that for 1W of dissipated power on the die, T_J settles 30°C above the ambient air temperature. With a 20°C/W value, T_J settles only 20°C above the ambient temperature for the same power dissipation.

In applications without heatsink attached to the top of the device case, and in still air conditions, it is commonly assumed that about 70% of the heat generated by the processor is conducted to the PCB. In these cases, the PCB design has a major impact on $R_{J\text{-}A}$.

The device data sheet provides one $R_{J\text{-}A}$ value for a very specific measurement condition specified by the JEDEC JESD51-2 standard (a 10x10 cm, four-layer PCB, with two signals and two power layers). This $R_{J\text{-}A}$ value only aims at making device comparisons for one given PCB condition. It cannot be used for a rigorous thermal study on a specific PCB.

Any system design should include a thermal study to minimize $R_{J\text{-}A}$ as much as possible. An efficient thermal design (with low $R_{J\text{-}A}$ value) brings the following benefits:

- More power can be dissipated.
- The application ambient temperature operating range can be extended.

On the contrary, a poor thermal design (high $R_{J\text{-}A}$ value) may raise the following issues:

- The device limit junction temperature is reached.
- The application ambient operating temperature range is reduced.

2.4 Total Power Dissipation

The total power dissipation of an IC is the sum of the power consumed by each power supply(1)(2) and it can be divided into two parts:

- The power consumed by the core logic of the circuit: in the SAMA7G54 device, this core logic power is drawn by VDDCORE and VDDCPU rails. It accounts for the dynamic power and leakage power of the device logic gates and embedded memories.
- The power consumed by analog circuits and digital I/Os: this is the power drawn by all the other power supply inputs.

Notes:

1. Refer to the table “Power Supply Inputs” in the Electrical Characteristics section of the device data sheet for a supply rail list.
2. When counting the power consumed in a power supply, care should be taken to subtract any power transmitted to an external component that does not contribute to the “on-die” dissipation.

2.4.1 VDDCORE and VDDCPU Dynamic Power

The dynamic power consumed by the logic circuits in the VDDCPU and VDDCORE domains depends on the application use case, i.e., the circuit activity (enabled circuits and their frequency) and the supply voltage at which they are operated. It is commonly known that the dynamic power consumption of a digital circuit has:

- a linear dependence to its operating frequency, and
- a quadratic dependence to its supply voltage.

For example, operating a CPU circuit at 400 MHz instead of 800 MHz divides by two its dynamic power consumption. Furthermore, if the reduced operating frequency allows a 5% VDD reduction, then an additional 10% power is saved. Adjusting at runtime the frequency and the voltage of a digital circuit to optimize its power consumption is a technique known as Dynamic Voltage and Frequency Scaling (DVFS).

SAMA7G5 devices are designed to support DVFS techniques on their VDDCPU supply:

- The device data sheet defines several voltage ranges for VDDCPUs that have corresponding CPU frequency ranges.
- The clock control of the CPU island, in the Power Management Controller (PMC), is designed to scale the processor frequency upon request at runtime.

For example, for the experiments covered in this application note, the following Power-Performance states (P states) are defined in the Microchip Linux Device Tree (DT) for the SAMA7G54 processor. These are the operating points the kernel can choose to operate on:

- f_{CPU} = 90 MHz / VDDCPU = 1.05V
- f_{CPU} = 250 MHz / VDDCPU = 1.05V
- f_{CPU} = 600 MHz / VDDCPU = 1.10V
- f_{CPU} = 800 MHz / VDDCPU = 1.15V
- f_{CPU} = 1 GHz / VDDCPU = 1.25V

The SAMA7G5 logic circuits powered by VDDCORE cannot have their frequency changed dynamically. To optimize the dynamic power consumption in this power domain, the following actions are possible:

- The generic and peripheral clocks feeding unused peripherals must be disabled. This can be done at runtime to save power when needed in the PMC.
- The frequency of the MCKx clocks can be set to 100 MHz instead of 200 MHz, for example. This limits the internal bandwidth of the interconnect but may be possible in some applications cases. Note that, unlike the CPU frequency, this is a static setting that cannot be changed dynamically at runtime.

2.4.2 VDDCORE and VDDCPU Leakage Power

Leakage power is the power drawn by a digital circuit when this circuit is powered but has no activity (no clock). This power occurs mainly on core logic supplies for circuits built on advanced CMOS technologies.

On SAMA7G5 devices, leakage power should be considered only on VDDCORE and VDDCPU. It can be neglected for any other supply inputs. The device leakage power has the following properties:

- The leakage power increases exponentially with temperature and its contribution to the overall power starts to be significant for junction temperatures greater than 60°C.
- The leakage power increases exponentially with the supply voltage, therefore the supply voltage should be minimized whenever possible.

2.4.3 Analog and I/O Power

Contrary to the core logic power described in the previous sections, the power consumed in the analog and I/O cells has a low temperature dependence. Typically, it drifts by about 10 to 20% over the whole temperature range of the device. This power depends on the application use case (number of PLLs used, number of USB ports used, ADC enabled or disabled, MIPI D-PHY enabled or disabled, number of serial ports in use and their speed, GMAC interface use and its mode, etc.).

Tip: This power can generally be reduced by operating the device power rails at lower and optimum operating voltages. For instance, interfacing a Gigabit Ethernet transceiver(1) at 1.8V consumes less power than at 3.3V. Similarly, avoiding high voltage conditions on 3.3V inputs, i.e., avoiding 3.6V, by choosing accurate voltage regulators(2) is a good practice to keep the rail away from unnecessary power dissipation.

Notes:

1. Microchip Gigabit Ethernet transceiver KSZ9131 or LAN8840 can be interfaced at 1.8V.
2. Microchip MCP16501/502 Power Management Integrated Circuits (PMIC), with better than ±3% accurate voltage regulators, are recommended power supply solutions.

2.4.4 Part-to-Part Variability of Total Power

Variability in the manufacturing process of the silicon die leads to variability in most electrical parameters of the processor. The total power consumed on the die follows this rule and some variations can be observed from part to part when they are placed in identical software and hardware conditions. This variation is both software and hardware dependent and it should therefore be characterized in real case conditions to estimate the expected maximum when in production. To do so, it is recommended to build at least 20x PCBs and measure their individual power and junction temperature during the application-specific use case. From this measurement set, classical statistical tools (average and standard deviation) can be used to assess the production maximum.

The following table provides the results of measurements carried out on 20x SAMA7G54-EK evaluation kits when running an iPerf(1) test at f_{CPU} = 1 GHz, with T_A = 70°C.

Note:

1. iPerf is a Linux network benchmark tool. See Use Case 2: iPerf Test.

Table 2-1. Measurement Results on 20x SAMA7G54-EK Boards

	Total Power	Junction Temperature
Average (μ)	738 mW	90°C
Standard deviation (σ)	34 mW	0.92°C
Estimated maximum (μ + 3 x σ)	840 mW	92.8°C

3. SAMA7G54 Evaluation Kit Measurements

The SAMA7G54-EK evaluation kit is intended for evaluating and prototyping with the SAMA7G54 microprocessor. Supported by mainline Linux distribution as well as bare metal software frameworks and RTOS, the kit allows easy use case implementation and performance measurements.

Refer to www.microchip.com/en-us/development-tool/EV21H18A.

SAMA7G54-EK is a great tool to make first power consumption and junction temperature evaluations when running a real application software:

- The power consumption is measured using the on-board Microchip Quad DC Power Monitor PAC1934.
- The junction temperature is measured with the SAMA7G5 internal die temperature sensor. See Appendix A for details on junction temperature reading on Linux.

Two use cases are considered in the following sections:

- Use Case 1: Linux idle, the O/S prompts for commands.
- Use Case 2: iPerf3 test on the Gigabit Ethernet interface generating a high CPU load along with large memory transfers.

Note:

1. The results presented hereafter strongly depend on the PCB on which the SAMA7G54 device is mounted, and more specifically on the effective thermal resistance ($R_{J\text{-}A}$) of the chip soldered on the PCB. For the SAMA7G54-EK board, the junction-to-air thermal resistance was first characterized with a set of measurements (see the method presented in In-Application $R_{J\text{-}A}$ Measurement) and its value is estimated to 30°C/W (±10%).
2. Use Case 1: Linux Idle and Use Case 2: iPerf Test show the measurements of two parts, called respectively “Typical” and “Maximum.” Typical corresponds to a part having a power consumption identical to the average of the measured batch. Maximum is an extrapolation of the maximum production part ($\mu + 3 \times \sigma$).
3. T_J measurement accuracy is ±5°C.
4. To ease comparisons between curves, the x-axis always represents the device junction temperature, and not the ambient temperature. A vertical line at T_J = 105°C indicates the maximum specified junction temperature for the mounted (industrial grade) SAMA7G54 devices.

3.1 Use Case 1: Linux Idle

In this use case, Linux is running and displaying the classical prompt, waiting for commands to be entered. This use case results in low power consumption and low junction temperature.

3.1.1 Experiment Description

The boards are placed in an oven to characterize their responses (T_J and P_{TOT}) to an ambient temperature (T_A) sweep.

⚠ CAUTION For characterization purposes, the oven temperature is increased up to T_J = 125°C. For industrial devices, that temperature exceeds the maximum specified range (105°C) and should never be applied during normal operation. For Extended Industrial or Automotive device values, operation at T_J = 125°C is valid.

- No cooling system and no air flow is provided.
- VDDCPU = 1.05V, VDDCORE = 1.15V

- CPU frequency = 90 MHz (the Linux kernel reduces the CPU clock when the activity is low. See Appendix A).
- All clocks are activated.
- Linux thermal control is disabled.

3.1.2 Measurement Results

Figure 3-1. Use Case 1: P_{TOT} versus T_J

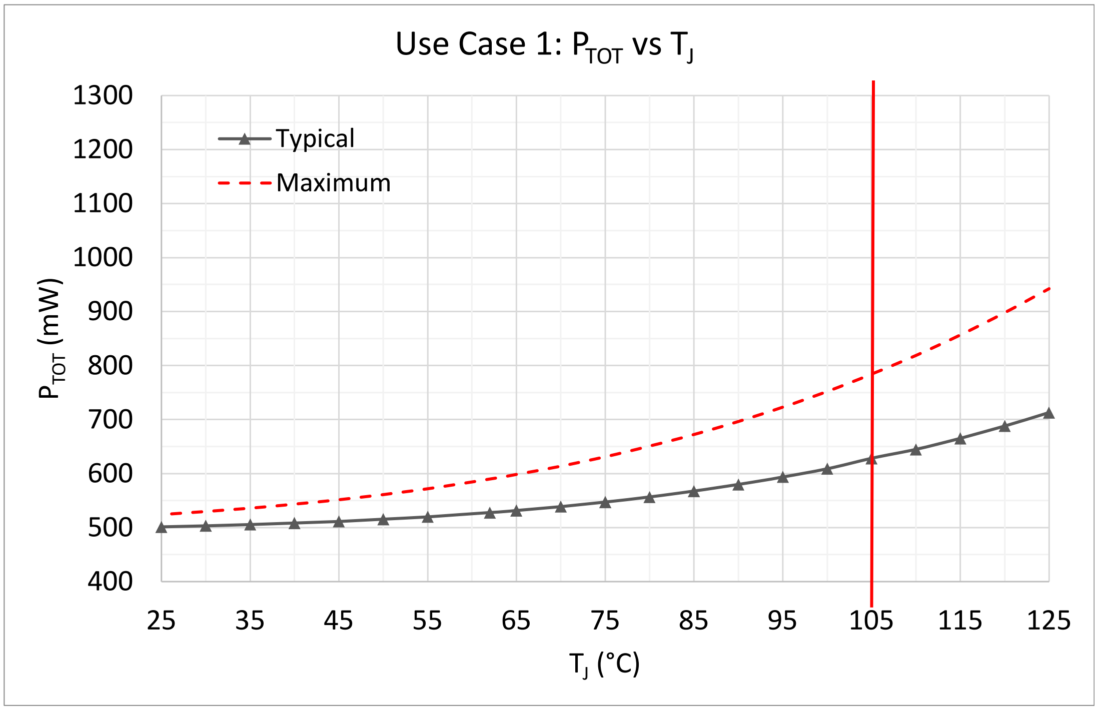
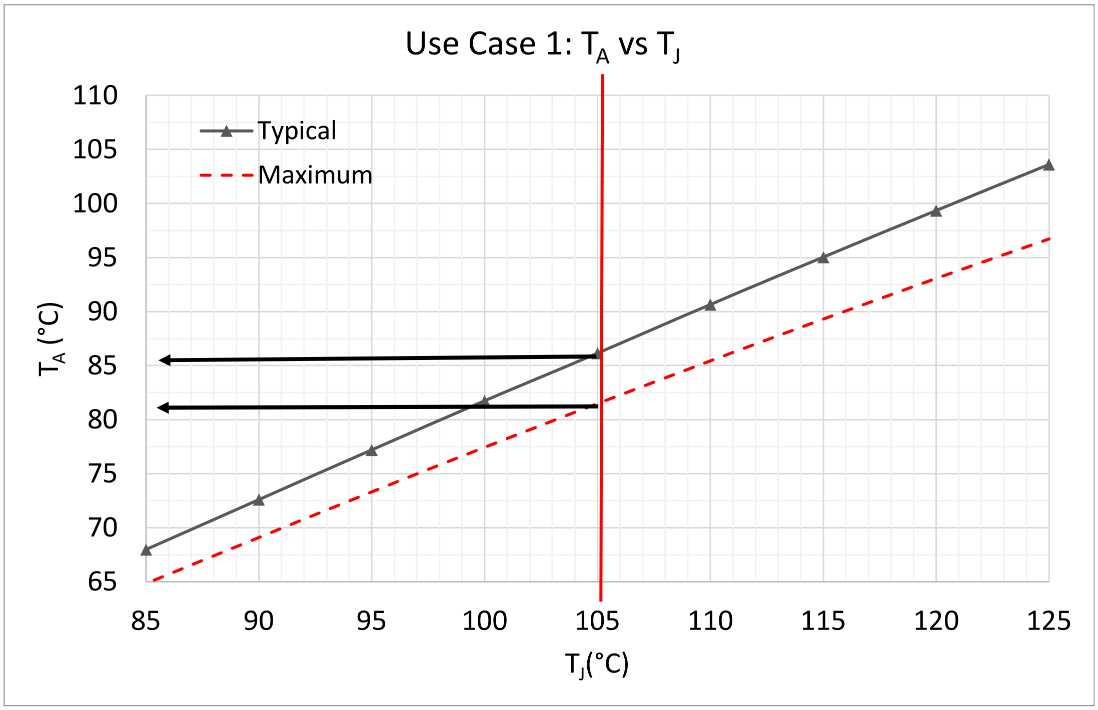


Figure 3-2. Use Case 1: T_A versus T_J



The following can be inferred from the above measurements:

- The total power consumed by this use case ranges from 630 to 780 mW at T_J = 105°C.
- The ambient temperature at which T_J reaches 105°C varies between 81°C and 86°C, depending on the device variability.

Note that running the same test on a different hardware (PCB) would lead to different results.

3.2 Use Case 2: iPerf Test

From the Use Case 1 "Linux Idle" condition, a network bandwidth test (iPerf3)(1) is started on the Gigabit Ethernet interface. SAMA7G54-EK serves as the client and a distant PC runs the server side. This use case results in maximum(2) power consumption values on the VDDCPU and VDDCORE.

Notes:

1. For more information on iPerf3 tests, refer to https://iperf.fr/.
2. Other tests were conducted on SAMA7G54, involving the camera sensor interface and/or additional DMA transfers, but none of them resulted in more power than the iPerf3 test. This test can therefore be considered as a "maximum power" test for SAMA7G54.

3.2.1 Experiment Description

As for Use Case 1, the boards are placed in an oven to study their response to the ambient temperature sweep.

⚠ CAUTION For characterization purposes, the oven temperature is increased up to T_J = 125°C. For industrial devices, that temperature exceeds the maximum specified range (105°C) and should never be applied during normal operation. For Extended Industrial or Automotive device values, operation at T_J = 125°C is valid.

- No cooling system and no air flow is provided.

- VDDCPU = 1.25V, VDDCORE = 1.15V
- CPU frequency = 1 GHz
- Linux thermal control is disabled.

3.2.2 Measurement Results

Figure 3-3. Use Case 2: P_{TOT} versus T_J

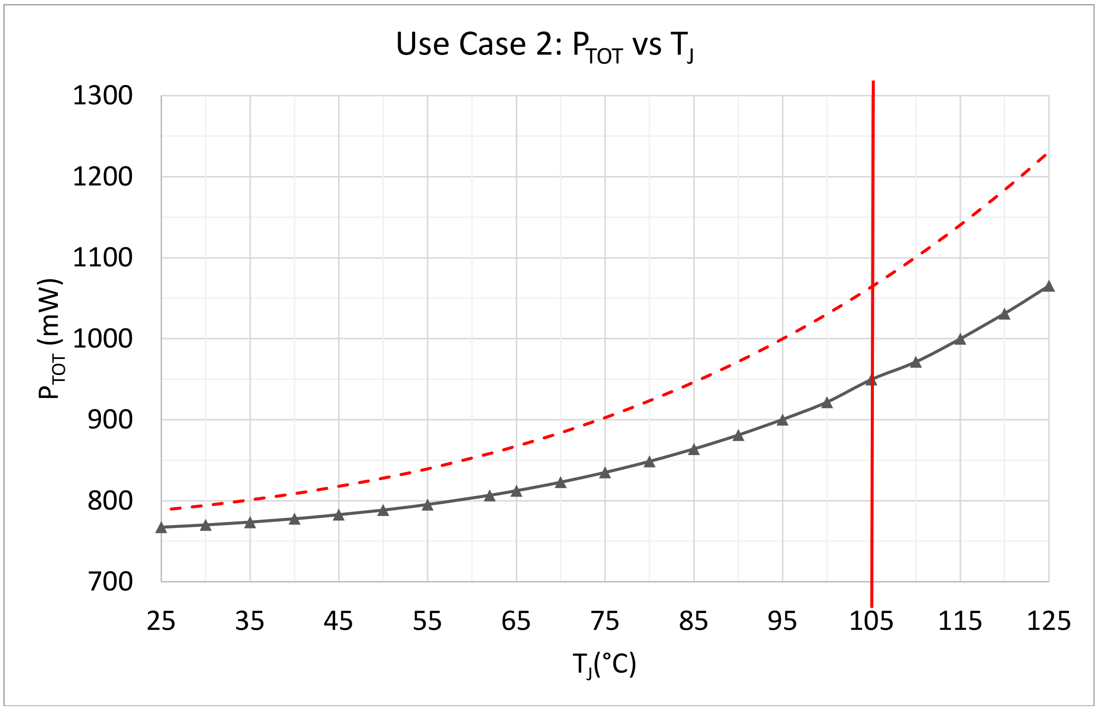
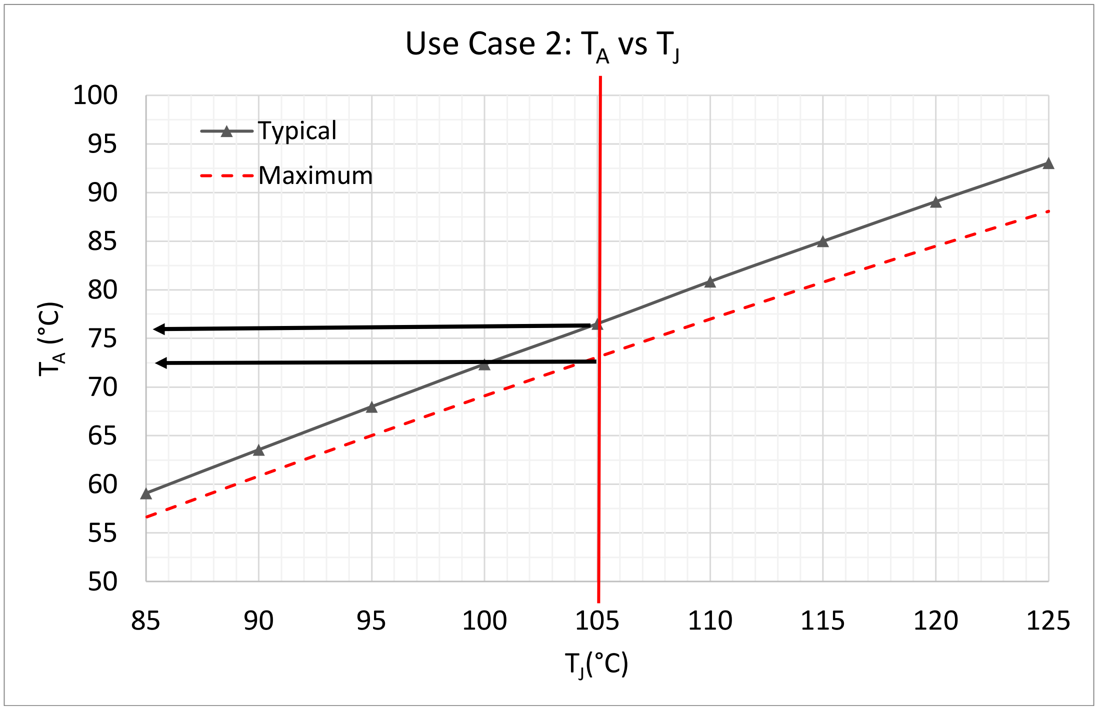


Figure 3-4. Use Case 2: T_A versus T_J



The following can be inferred from the above measurements:

- The total power consumed by this use case ranges from 950 to 1070 mW at T_J = 105°C.
- The ambient temperature at which T_J reaches 105°C varies between 73°C and 77°C, depending on the device variability.

Note that running the same test on a different hardware would lead to a different result.

4. Application-Specific Use Cases

One important question to address in many applications is the maximum ambient temperature (T_{A_MAX}) the application can run. As discussed earlier, the following equation applies:

$T_{A\_MAX} = T_{J\_Limit} - R_{J\text{-}A} \times P_{TOT}$

Where:

- T_{J_Limit} is provided in the device data sheet, for example 105°C for industrial parts and 125°C for extended industrial and automotive parts.
- $R_{J\text{-}A}$ depends on the application hardware: PCB design, presence or absence of heatsink, presence or absence of air flow, enclosure, etc.
- P_{TOT} represents the application activity and is largely influenced by the software run by the processor.

As a rule of thumb, for applications requiring T_{A_MAX} lower than 60°C, the MPU junction temperature remains below the 105°C limit. For T_{A_MAX} greater than 60°C, the operating junction temperature verification is required during the prototyping phase to ensure that the 105°C or 125°C limit is not exceeded. Measuring the junction temperature is also a way to estimate more accurately the device lifetime in real conditions. Refer to the “SAMA7G5/D6 Series Product Lifetime Estimation” application note (see Reference Documents).

In this section, simulated curves show the impact of various parameters ($R_{J\text{-}A}$, P_{TOT}, time) on the device junction temperature and power dissipation.

4.1 Board-Specific Junction-to-Air Thermal Resistance

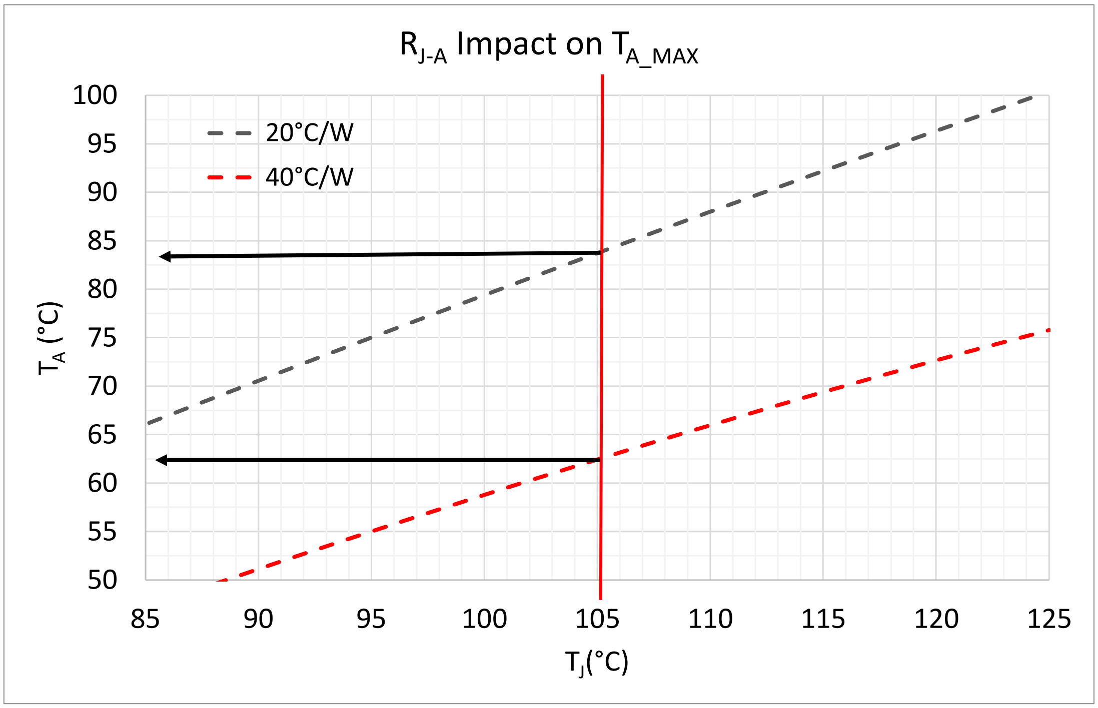
Junction-to-air thermal resistance is a critical parameter of any thermal system, and as such it needs to be evaluated in real conditions. The application board thermal resistance can be significantly different from the Microchip SAMA7G54-EK evaluation board resistance or from JEDEC-specified conditions. Depending on various parameters (PCB material and geometry, number of layers, etc.), that value may vary from 20°C/W to 40°C/W.

4.1.1 $R_{J\text{-}A}$ Impact on T_{A_MAX}

As an example, the figure below shows the simulation result $T_A = f(T_J)$ of two (theoretical) boards running the same software as in Use Case 2 (iPerf test on typical industrial grade part). The simulation condition is $R_{J\text{-}A}$ = 20°C/W for the first board, and $R_{J\text{-}A}$ = 40°C/W for the second board.

The ambient temperature (T_{A_MAX}) for which T_J = 105°C is extracted in both cases. T_{A_MAX} is 83°C for the first board and 62°C for the second board.

Figure 4-1. $R_{J\text{-}A}$ Impact on T_{A_MAX}



4.1.2 In-Application $R_{J\text{-}A}$ Measurement

The junction-to-air thermal resistance is a physical constant depending on the mechanical characteristics of the IC package mounted on its PCB. It does not vary with temperature or power. Thus, by measuring the junction temperature and the power consumed by the product in a controlled ambient temperature, the thermal resistance can be computed with the formula:

$R_{J\text{-}A} = (T_J - T_A)/P_{TOT}$

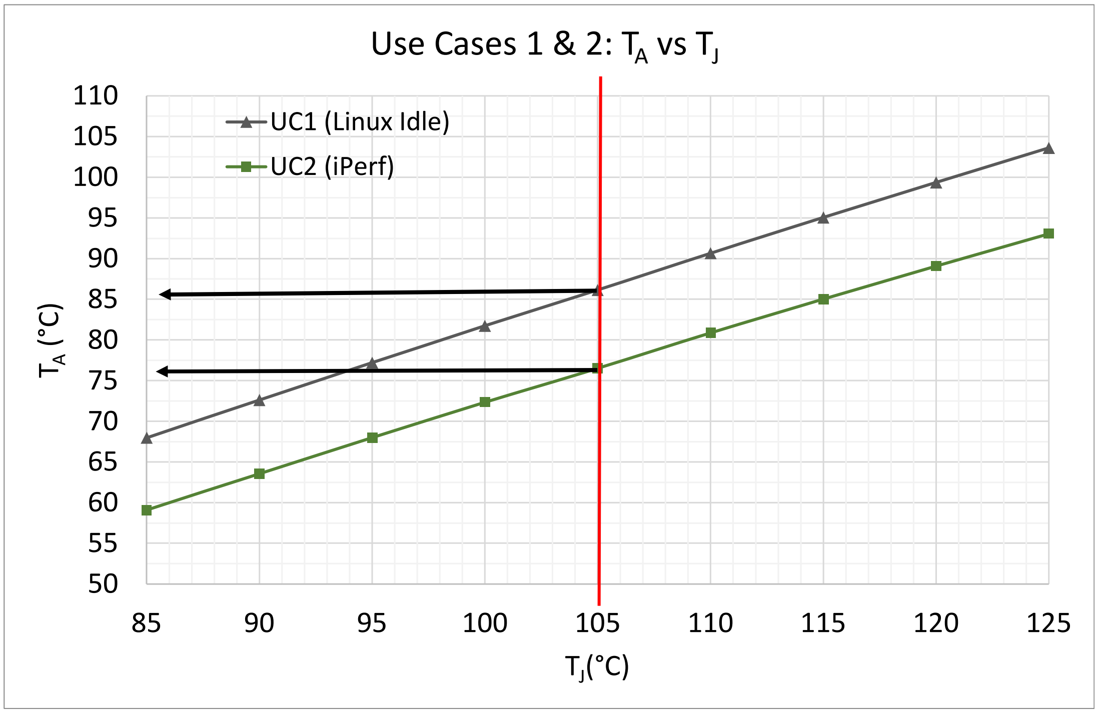
Notes:

- The SAMA7G54 device junction temperature can be read from the internal calibrated temperature sensor. The Linux kernel reports this junction temperature in thermal zone 0 (see Appendix A). For other implementations, refer to the “SAMA7G5 Series Temperature Sensor Calibration” application note (see Reference Documents).
- For power measurements, we recommend to make provision for one or more Microchip PAC1934 footprints on the application PCB as implemented on the SAMA7G54-EK board. For more information, refer to the PAC1934 data sheet (see Reference Documents).

4.2 Junction Temperature as a Function of the Use Case

To illustrate how different software complexities may lead to different possible operating temperatures, and for easy comparison, all SAMA7G54-EK measurement results presented in SAMA7G54 Evaluation Kit Measurements are shown in the same graph below.

Figure 4-2. Use Cases 1 & 2: T_A versus T_J



The above figure shows that for an identical hardware configuration, the T_A difference between Use Case 1 and Use Case 2 at T_J = 105°C is ≈9°C.

4.3 Junction Temperature Response Time

In many systems, the activity profile is not constant over time, and the processor alternates between high and low activity periods. Depending on the speed at which the load profile is modulated, the junction temperature response to this load profile change may differ. In particular, when a burst of activity has a very short duration compared to the system thermal time constant, T_J does not reach its steady state value.

In this section, the SAMA7G54 junction temperature response to a CPU load step is studied through measurements carried out on SAMA7G54-EK.

4.3.1 CPU Load Step Experiment Description

SAMA7G54-EK is placed in the oven at T_A = 70°C and the SAMA7G54 junction temperature evolution is measured when the system switches from a first state to a second one. The first state is Use Case 1 described above where Linux prompts for a command to be entered. The steady state junction temperature for state 1 is T_J = 86.5°C. State 2 is reached by launching a “dummy loop” that loads the CPU up to 100%. The temperature acquisition is made every two seconds, which is slow but still fast enough to capture the temperature variation in the device.

This load step scenario is measured twice with Linux thermal control mechanisms first disabled, then enabled (see Appendix A).

4.3.2 Linux Thermal Control Disabled

In this first experiment, the Linux thermal control is disabled:

```
# echo disabled > /sys/class/thermal/thermal_zone0/mode
```

During the first state of this experiment, the Linux CPU frequency scaling governor chooses the “powersafe” policy due to the low CPU activity.

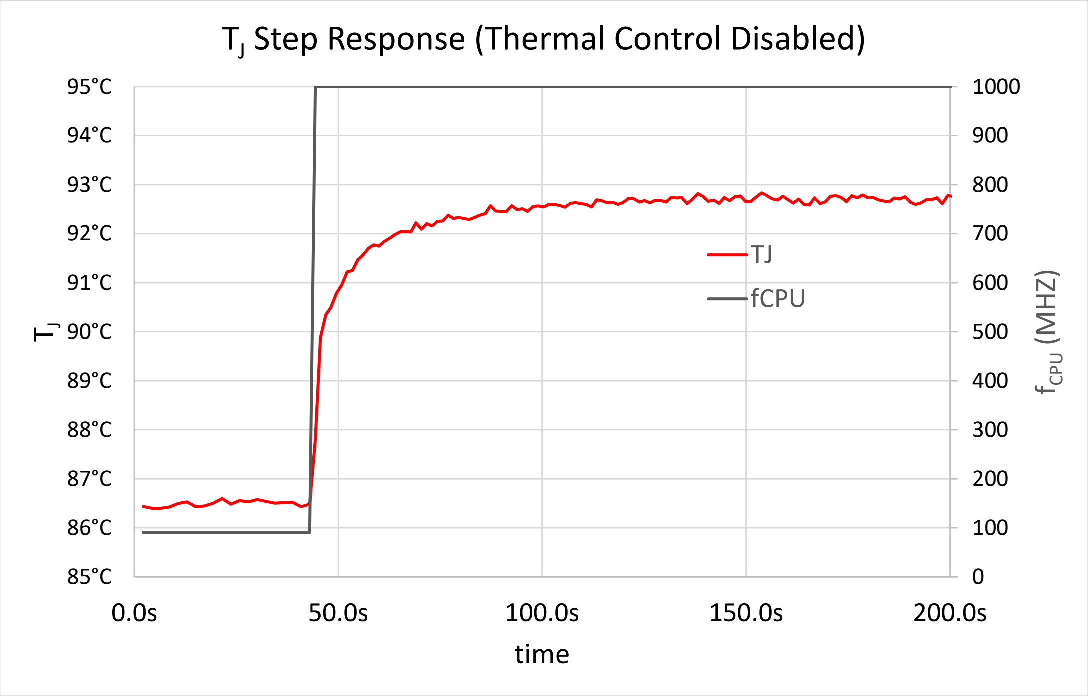
The system settles as follows:

- f_{CPU} = 90 MHz
- VDDCPU = 1.05V
- CPU load = 48% (as reported by the “top” monitoring program)
- T_J = 86.5°C

During the second state, the frequency scaling governor switches to the “performance” policy. The system variables are as follows:

- f_{CPU} = 1GHz
- CPU load = 100% (as reported by the “top” monitoring program)
- T_J increases up to 92.6°C.

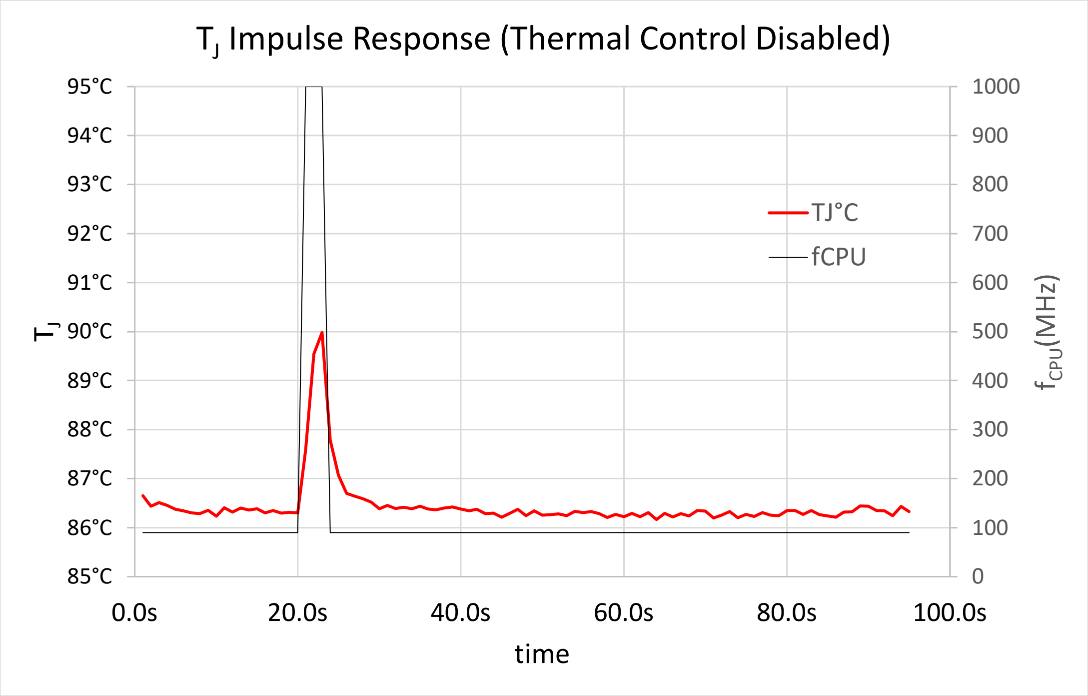
Figure 4-3. Junction Temperature Response to a CPU Load Step



As shown in the figure above, the SAMA7G54 junction temperature varies very slowly. It takes around six seconds for T_J to reach 60% of its final value. This time constant should be compared to the application activity burst duration. If activity bursts are shorter than the thermal time constant, these will be “filtered out” by the thermal inertia of the system and T_J will never settle to its final value.

To illustrate this, the following figure shows T_J impulse response to a CPU processing burst. From the Linux Idle state, the CPU executes a 3-second long processing burst, and then resumes its initial state.

Figure 4-4. Junction Temperature Response to a CPU Load Burst



The two responses above show how important dynamics are in a system. Studying a power budget in steady state conditions without accounting for response times may result in erroneous conclusions. It all depends on how the system rate of change compares to its thermal time constant.

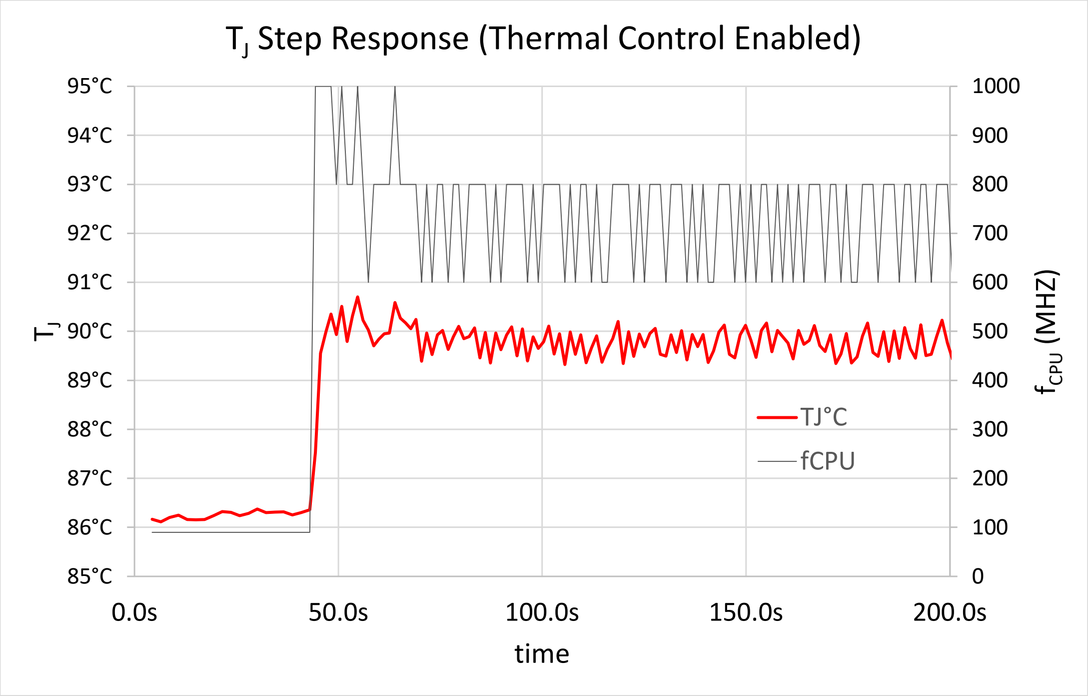
4.3.3 Linux Thermal Control Enabled

This second experiment is the same as the previous one, except that Linux thermal governor is enabled, as follows:

```
echo enabled > /sys/class/thermal/thermal_zone0/mode
```

Here, the Linux thermal control governor tries to cool down the processor when a specific temperature threshold (temperature trip point) is crossed. In this specific experiment, trip_point0 is set to 90°C and is associated to a passive cooling type, which means that Linux can decrease the CPU frequency to cool the device.

Figure 4-5. Junction Temperature Response to a CPU Load Step with Linux Thermal Control Enabled



The above figure shows how the Linux thermal governor navigates the P-states to decrease the CPU voltage and frequency to cool down the processor.

5. SAMA7G5 Series System-in-Package (SiP) Measurement Results

Integrating an SDRAM memory and an MPU in a System-in-Package (SiP) eliminates the high-speed memory interface constraints imposed on the PCB and reduces the PCB size. Junction temperature and power dissipation are also significant advantages. The larger package size of the SiP results in improved thermal performance compared to the SAMA7G5, improving the thermal resistance by about 30%. This leads to a lower operating junction temperature and hence a lower leakage power, and more generally a lower overall power consumption, particularly at high temperatures.

Use cases measurement conditions

Linux is started and reports a close to 100% CPU load for both of the following cases.

Case 1: Linux in idle mode, CPU frequency = 90 MHz, Linux governor in Power Save mode

Case 2: Linux in IPerf mode, CPU frequency = 1 Ghz, Linux governor in Performance mode

5.1 Power and Thermal Measurements

The board used for the measurements is an evaluation board with the same characteristics and size as the SAMA7G54 Evaluation Kit.

Figure 5-1 and Figure 5-2 report results from a selected part having maximum power consumption characteristics.

Figure 5-1 shows a 300 mW power difference between Case 1 and Case 2.

⚠WARNING For characterization purposes, the oven temperature is increased up to T_J = 125°C. This temperature exceeds the maximum specified range (105°C) and should never be applied during normal operation.

Figure 5-1. Total SiP Power versus CPU Junction Temperature

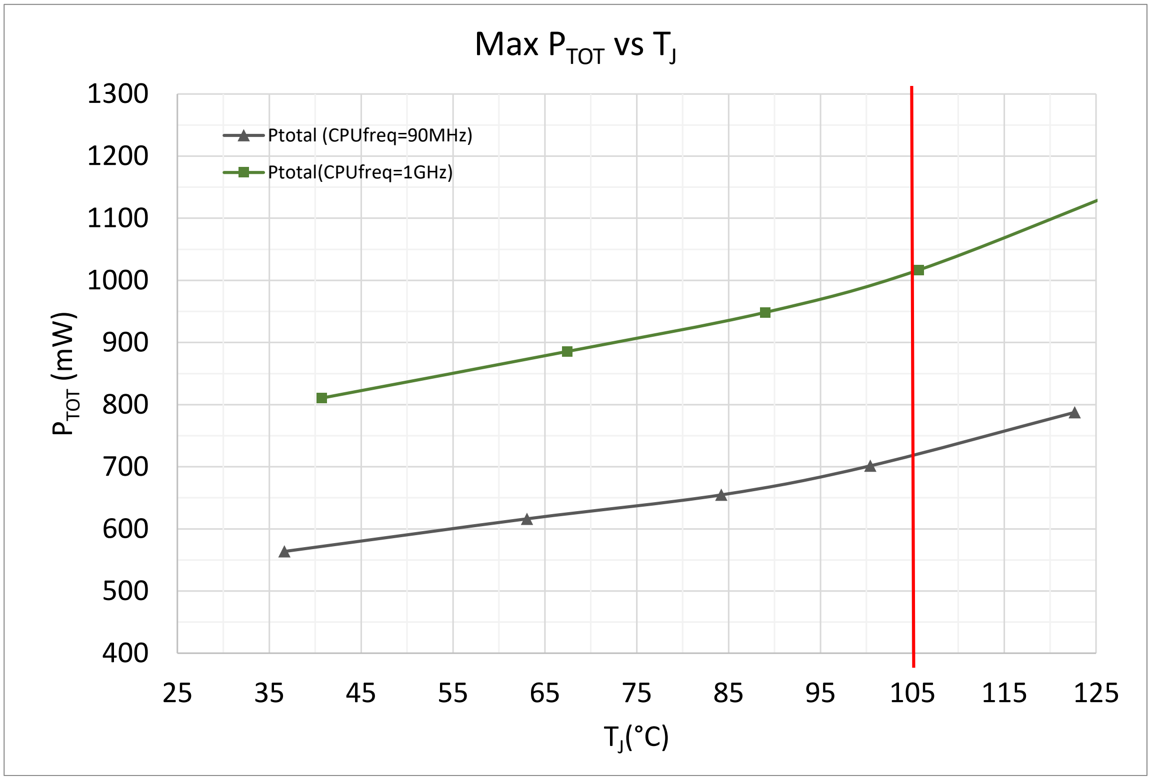


Figure 5-2 shows a 5°C temperature difference between Case 1 and Case 2.

Figure 5-2. SiP Ambient Temperature versus Junction Temperature

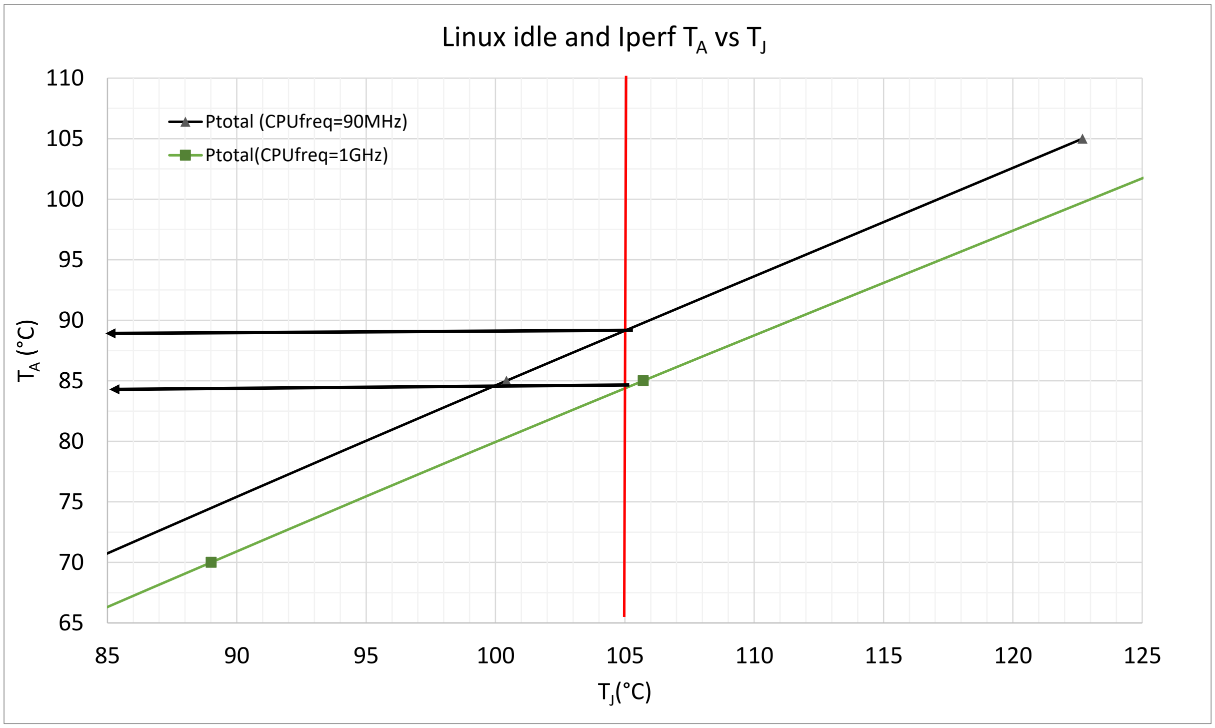
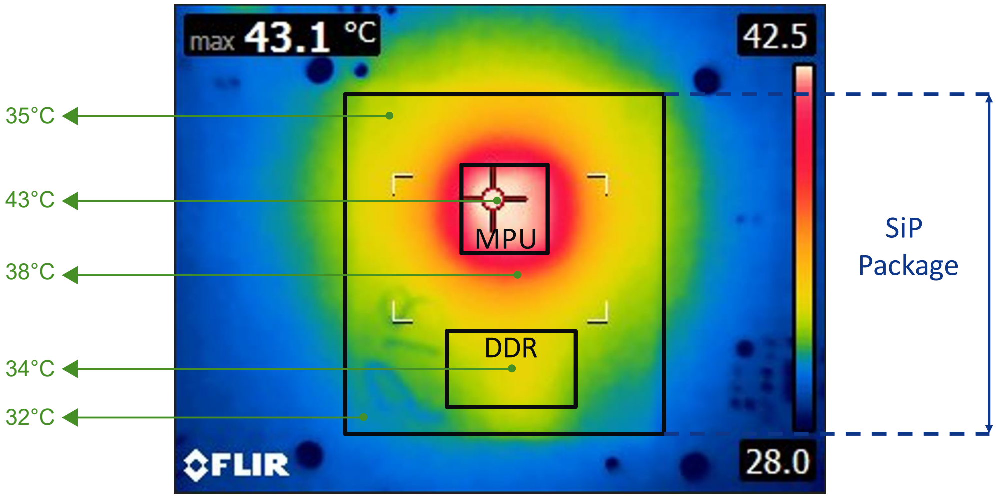


Figure 5-3 is a thermal image of the SiP (with CPU 100% loaded and ambient temperature at 21°C) showing the CPU hot spot. The DDR is located in a cold area of the SiP. The embedded temperature sensor is close to the CPU, and so measures the worst case temperature of the SiP.

Figure 5-3. SiP Thermal View



5.2 Thermal Resistance Measurements

The following thermal resistance measurements were carried out in an enclosure with controlled air flow (0 m/s) and controlled temperature conditions.

Respecting the JEDEC measurement method, which implies creating a delta power and measuring a delta junction temperature, the ambient temperature variations are considered negligible:

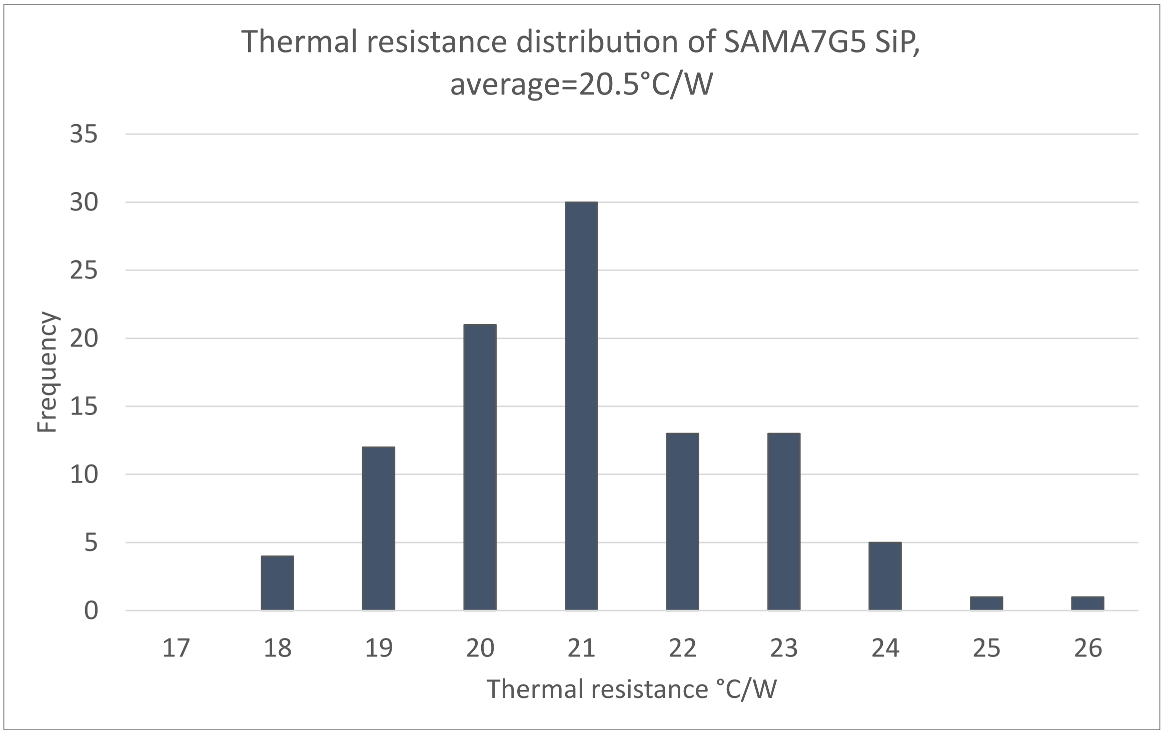
R_{th}=delta_T_J/delta_power

The two states of measurement are produced under Linux by changing the governor mode from Power Save to Performance. This leads to two power measurements and two junction temperature measurements.

To increase the measurement accuracy and get the correct thermal resistance, it is necessary to repeat the measurement more than 50 times and to average the results. In this case, the average value is 20.5 °C/W, which is close to the simulated target of 21 °C/W on a JEDEC board.

The measurements standard deviation is about 1.5°C/W. It is mainly dependent on the power measurement dispersion on the CPU power line. It is important to strongly average the power measurement when the temperature settles to a stable value.

Figure 5-4. Distribution of Junction-to-Air Thermal Resistance Measurement Results



Compared to the SAMA7G5 Series device, the SAMA7G5 Series SiP device offers improved thermal performance through a lower thermal resistance.

6. Hardware and Software Recommendations for High Temperature Applications

6.1 Software Recommendations

The following list compiles the good practices that can be adopted at software level in high temperature applications. In Linux systems, the kernel implements most of these by default.

- Temperature control
  - Enable the temperature sensor to periodically read the die temperature. Make sure to use the calibration data provided by Microchip to improve the initial temperature reading accuracy and mind that this reading has a ±5°C accuracy. This means that the system should be shut down when the temperature sensor reports T_J higher than 100°C.
    - The Linux kernel reports the die temperature in thermal zone 0. See Appendix A.
  - Apply a temperature control policy to your application to make sure corrective actions are taken when the temperature rises.
    - The Linux kernel defines several temperature trip points to trigger various actions ranging from decreasing the CPU frequency to enabling an external fan or stopping the system. See Appendix A.
- Dynamic power reduction
  - As previously discussed, reducing at any time the dynamic power in an application reduces the device junction temperature, and therefore the leakage power that depends on T_J. This creates a virtuous circle.
  - Disable any “unused” resource in the system. Pay attention to all analog blocks like USB transceivers, PLLs, MIPI receiver, etc. Most of these can be enabled only when needed.
  - Implement a CPU frequency control policy to reduce or stop the CPU clock when possible and couple this frequency control with a voltage control on VDDCPU (DVFS technique).
    - The Linux kernel implements the CPUfreq framework for which Microchip has defined several Power performance states (P-states).
  - Reduce MCKx frequency to a lower value when possible. This reduces the dynamic power consumption on VDDCORE. The maximum speed specified for each MCKx is designed to sustain the internal bandwidth requirements when the device runs a worst case scenario. From a data bandwidth perspective, that means using all peripherals at the same time to their maximum data rate. In some systems having reduced bandwidth needs, it may be possible to set these MCKx to half their maximum value, for example.

6.2 Hardware Recommendations

The following list presents the recommendations and measures that can be taken to minimize the junction-to-air thermal resistance of the device mounted in a specific application. These have a first order impact on the permissible ambient temperature at which a system can operate. Some are easy to implement at no cost, while others may generate a small cost adder in the application.

- Maximize the thermal inertia and thermal conduction of the PCB. When no heatsink is attached to the top of the device case, more than 70% of the heat generated by the processor is conducted to the PCB.
  - When possible, use larger PCBs and maximize the copper mass of the PCB. This can be done by pouring layout voids with copper shapes connected to ground and/or adding ground layers to the PCB stack.
- Pay attention to surrounding components that would conduct their heat to the processor. When possible, keep power dissipating devices to a reasonable distance from the processor.

- Add a heatsink or heat spreader over the processor case.
  - Some very cheap parts can bring a great improvement even when no air flow is present in the system.
  - In some cases, it may also be possible to thermally connect the device to the system enclosure when this enclosure is thermally conductive.
  - Create an airflow in the system with a fan. When a heatsink is mounted on the device top, forcing an airflow is a very effective way to decrease the device thermal resistance.
  - In Linux, a fan can be declared as a cooling device for a thermal zone and therefore be activated when temperature trip points are crossed.
- When possible, choose a SiP device that brings improved thermal performance over a standalone device.

7. Appendix A: Linux-Specific Information

The examples shown in this application note where measured on the SAMA7G54-EK board running the Microchip Linux distribution (kernel 5.15.x). This appendix describes the most useful Linux commands and concepts to run these tests.

7.1 CPU Frequency Scaling Framework

The Linux kernel supports CPU performance scaling by means of the CPUFreq (CPU Frequency scaling) subsystem. Changing the CPU frequency can save power consumption (when decreased) or boost performance (when increased). This subsystem navigates through predefined Power-performance states (P-states) and therefore VDDCPU is modified when the CPU frequency is adjusted.

The following governors are available to control the CPU frequency:

- “powersave” causes the lowest frequency, within the scaling_min_freq policy limit, to be requested for that policy.
- “performance” causes the highest frequency, within the scaling_max_freq policy limit, to be requested for that policy.
- “userspace” has no direct action, but allows the user space to set the CPU frequency for the policy.
- “ondemand” uses the CPU load as a CPU frequency selection metric.
- “conservative” has the same effect as “ondemand”, but avoids changing the frequency significantly over short time intervals, which may not be suitable for systems with limited power supply capacity.

To read the governor setting:

```
# cat /sys/devices/system/cpu/cpufreq/policy0/scaling_governor
powersave
```

The set of predefined frequencies used by the governor policies is listed in the file system:

```
# cat /sys/devices/system/cpu/cpufreq/policy0/scaling_available_frequencies
90000 250000 600000 800000 1000000
```

7.2 Thermal Management Framework

Linux divides a System-on-Chip (SoC) into multiple thermal zones that correspond to an area in the silicon die where the temperature is deemed uniform. On SAMA7G54, one single thermal zone is available (thermal_zone0), which means that the temperature is uniform across the die. The thermal zone properties can be displayed with the following command:

```
# ls /sys/class/thermal/thermal_zone0/
available_policies...........k_pu.............................trip_point_0_type
cdev0........................mode.............................trip_point_1_hyst
cdev0_trip_point.............offset...........................trip_point_1_temp
cdev0_weight.................policy...........................trip_point_1_type
cdev1........................power............................trip_point_2_hyst
cdev1_trip_point.............slope............................trip_point_2_temp
cdev1_weight.................subsystem........................trip_point_2_type
integral_cutoff..............sustainable_power................type
k_d..........................temp.............................uevent
k_i..........................trip_point_0_hyst
k_po.........................trip_point_0_temp
```

The SAMA7G54 die temperature can be read from the “temp” file:

```
# cat /sys/class/thermal/thermal_zone0/temp
55201
```

This result expressed in millidegrees Celsius provides the SAMAG54 die temperature (55.201°C).

Among the important attributes of a thermal zone are the temperature trip points. These points define when the Linux thermal governor makes decisions to start and stop cooling down the device (CPU frequency decrease, fan activation, etc.). These trip points may be of the following types:

- "active": a trip point to enable active cooling (ex.: external fan)
- "passive": a trip point to enable passive cooling (ex.: reduced CPU frequency)
- "hot": a trip point to notify emergency
- "critical": hardware not reliable. The CPU goes into Power-down mode.

The policies available to the thermal governor are listed below. By default, the policy is set to “step wise”.

- step wise: open-loop control, based on temperature threshold and trend. Walks through each cooling device cooling state, step by step.
- fair share: weight-based. Determines the cooling device state, based on assigned weight partitioning.
- bang bang: uses a hysteresis to abruptly switch on or off a cooling device. It is intended to control fans which cannot be throttled but just switched on or off.
- power allocator: closed-loop control, based on power budget, temperature, and current power consumption of each involved device.
- user space: hands off the control of a thermal zone to the user space.

As an example, the following settings can be used as a starting point for an industrial application. They may be adjusted depending on each system specificities:

- Trip point 0 “passive” (DVFS CPUfreq) at 90°C
- Trip point 1 “hot” at 95°C
- Trip point 2 “critical” at 100°C

These values account for the temperature sensor accuracy (±5°C).

8. Revision History

8.1 Rev. C - 11/2024

Throughout: added extended industrial grade content

Power and Thermal Measurements : added Warning

8.2 Rev. B - 10/2023

Added section SAMA7G5 Series System-in-Package (SiP) Measurement Results and references to the SAMA7G5 Series SiP device throughout

Added references to the SAMA7G5 Series automotive devices throughout

Experiment Description , Experiment Description: added Caution

8.3 Rev. A - 10/2022

First issue.

Microchip Information

Trademarks

The “Microchip” name and logo, the “M” logo, and other names, logos, and brands are registered and unregistered trademarks of Microchip Technology Incorporated or its affiliates and/or subsidiaries in the United States and/or other countries (“Microchip Trademarks”). Information regarding Microchip Trademarks can be found at https://www.microchip.com/en-us/about/legal-information/microchip-trademarks.

ISBN: 979-8-3371-0123-1

Legal Notice

This publication and the information herein may be used only with Microchip products, including to design, test, and integrate Microchip products with your application. Use of this information in any other manner violates these terms. Information regarding device applications is provided only for your convenience and may be superseded by updates. It is your responsibility to ensure that your application meets with your specifications. Contact your local Microchip sales office for additional support or, obtain additional support at www.microchip.com/en-us/support/design-help/client-support-services.

THIS INFORMATION IS PROVIDED BY MICROCHIP “AS IS”. MICROCHIP MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WHETHER EXPRESS OR IMPLIED, WRITTEN OR ORAL, STATUTORY OR OTHERWISE, RELATED TO THE INFORMATION INCLUDING BUT NOT LIMITED TO ANY IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY, AND FITNESS FOR A PARTICULAR PURPOSE, OR WARRANTIES RELATED TO ITS CONDITION, QUALITY, OR PERFORMANCE.

IN NO EVENT WILL MICROCHIP BE LIABLE FOR ANY INDIRECT, SPECIAL, PUNITIVE, INCIDENTAL, OR CONSEQUENTIAL LOSS, DAMAGE, COST, OR EXPENSE OF ANY KIND WHATSOEVER RELATED TO THE INFORMATION OR ITS USE, HOWEVER CAUSED, EVEN IF MICROCHIP HAS BEEN ADVISED OF THE POSSIBILITY OR THE DAMAGES ARE FORESEEABLE. TO THE FULLEST EXTENT ALLOWED BY LAW, MICROCHIP’S TOTAL LIABILITY ON ALL CLAIMS IN ANY WAY RELATED TO THE INFORMATION OR ITS USE WILL NOT EXCEED THE AMOUNT OF FEES, IF ANY, THAT YOU HAVE PAID DIRECTLY TO MICROCHIP FOR THE INFORMATION.

Use of Microchip devices in life support and/or safety applications is entirely at the buyer’s risk, and the buyer agrees to defend, indemnify and hold harmless Microchip from any and all damages, claims, suits, or expenses resulting from such use. No licenses are conveyed, implicitly or otherwise, under any Microchip intellectual property rights unless otherwise stated.

Microchip Devices Code Protection Feature

Note the following details of the code protection feature on Microchip products:

- Microchip products meet the specifications contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is secure when used in the intended manner, within operating specifications, and under normal conditions.
- Microchip values and aggressively protects its intellectual property rights. Attempts to breach the code protection features of Microchip products are strictly prohibited and may violate the Digital Millennium Copyright Act.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of its code. Code protection does not mean that we are guaranteeing the product is “unbreakable”. Code protection is constantly evolving. Microchip is committed to continuously improving the code protection features of our products.