Intel XTU Cryo Cooler Procedure

I've created a complete walkthrough on how to use the Intel Extreme Tuning Utility (XTU) and it's new Thermal Velocity Boost (TVB) to overclock your CPU and get the best results with any 2nd generation Intel Cryo Cooler (MasterLiquid ML360 SUB-ZERO V2 and EK-QuantumX Delta TEC EVO E2). This is the procedure that I followed using my MasterLiquid ML360 SUB-ZERO V2 to boost my performance in games such as Civilization VI, War Thunder, CS:GO, Shadow of the Tomb Raider, F1 22, Metro Exodus, and more. Disclaimer: Overclocking will void your warranty. It is safe if done correctly, but pushing voltages too high can potentially damage the chip.

Let's get started!

Before continuing with this tutorial please make sure you have completed these requirements:
- We will be using Intel Extreme Tuning Utility (XTU) to overclock your system. Install the latest version of XTU and make sure it is running properly.
- Install your 2nd generation generation Intel Cryo Cooler onto your system and make sure you have installed the latest version of Intel Cryo Cooling Technology Software. Once installed on your system it should look like this:
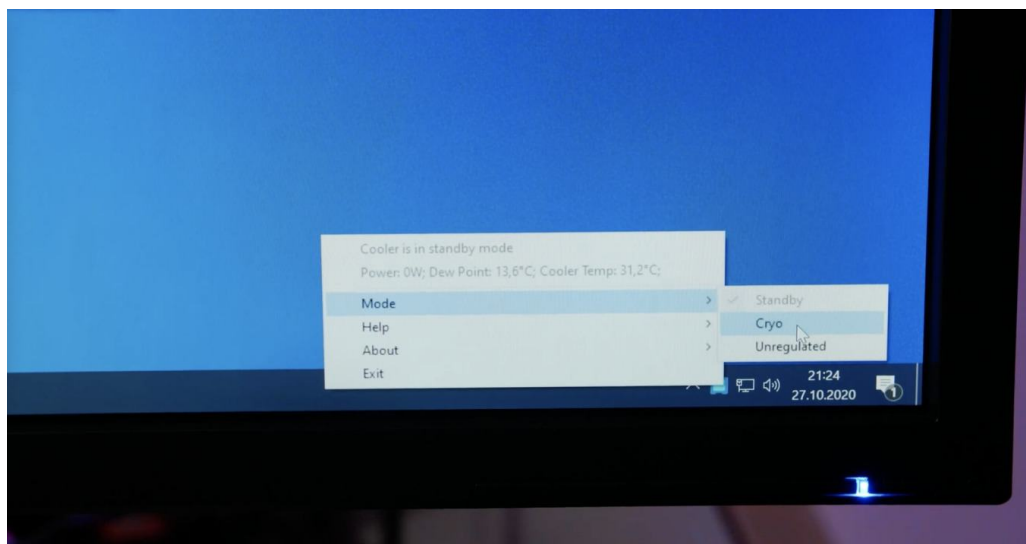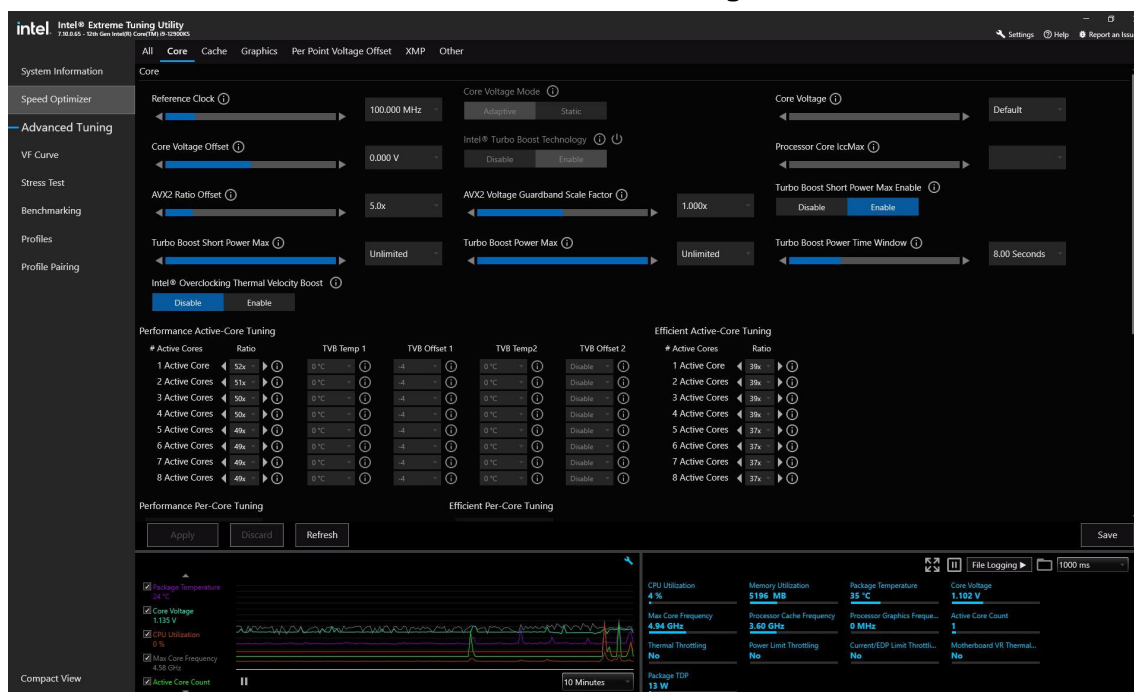
- Make sure your BIOS doesn't have any changed settings. Refer to this link on how to clear your CMOS.

Chapter 1: Getting Setup

*If you have never used XTU before, I highly recommend reading this tutorial.*

Launch XTU and click on the Advanced Tuning section.

Here's what I recommend changing to get the best performance from your overclock:
- Core Voltage Mode: **Adaptive**.
    - Can only be changed once Core Voltage isn't set to default.
- Intel Overclocking Thermal Velocity Boost **Enabled**.
- AVX2 Offset Ratio: **0**

Once these settings have been applied, we will mainly be focusing on changing the **Core Voltage** and **Active-Core Frequencies.**
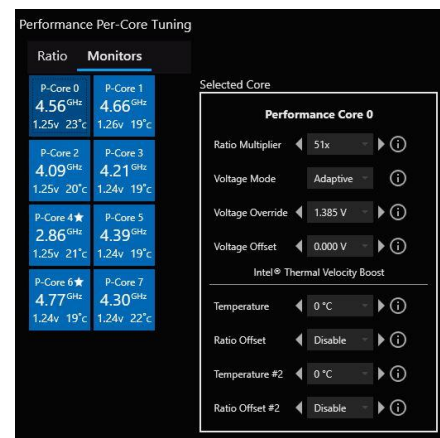
If you have never overclocked your PC before, [I recommend reading through this tutorial](#)

Make sure that your Cryo Cooler **mode is set to: "Cryo Mode."** This can be done by clicking the cryo icon in the lower right corner of your screen and navigating to the mode selection menu.
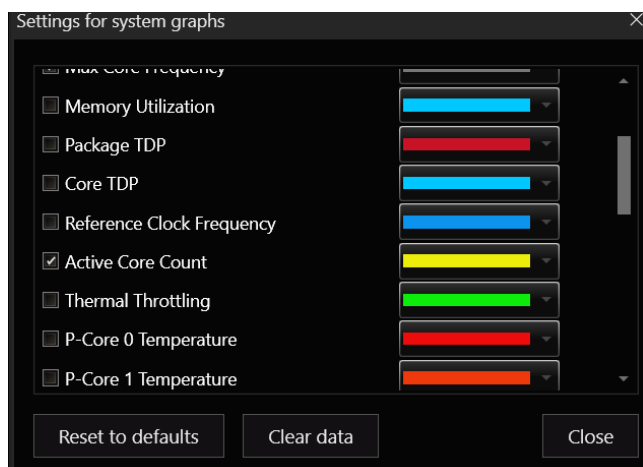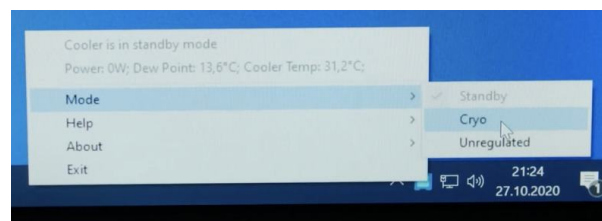
Next, we will enable our Active Core Count so that we can monitor it within XTU. Select the wrench icon in the system monitor box towards the bottom of XTU. This will be helpful later in the tutorial for tuning our overclocking profile.
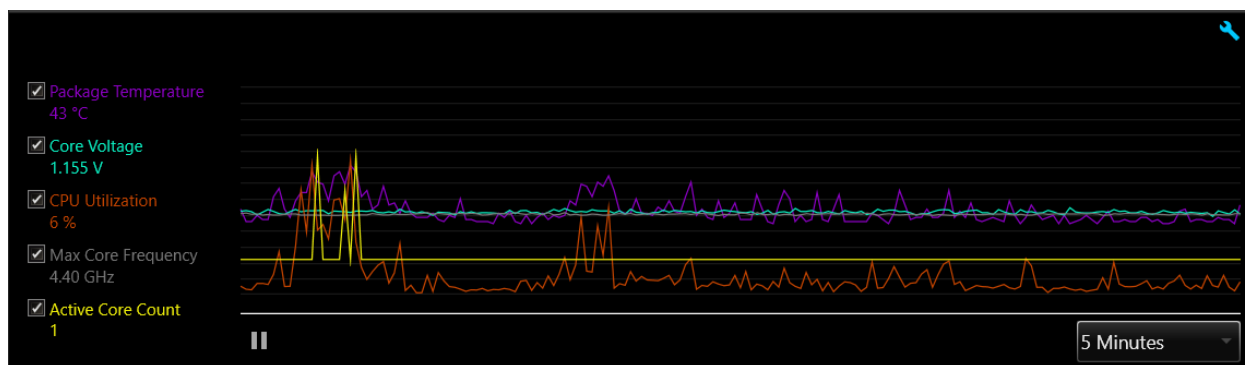
Here you can see in the lower left corner that Active Core Count is enabled. This shows us how many cores are actively being used.


Chapter 1: Testing Your CPU

XTU has a built-in benchmark that we will be using for this next step. Click on Stress Test in the side menu of XTU.
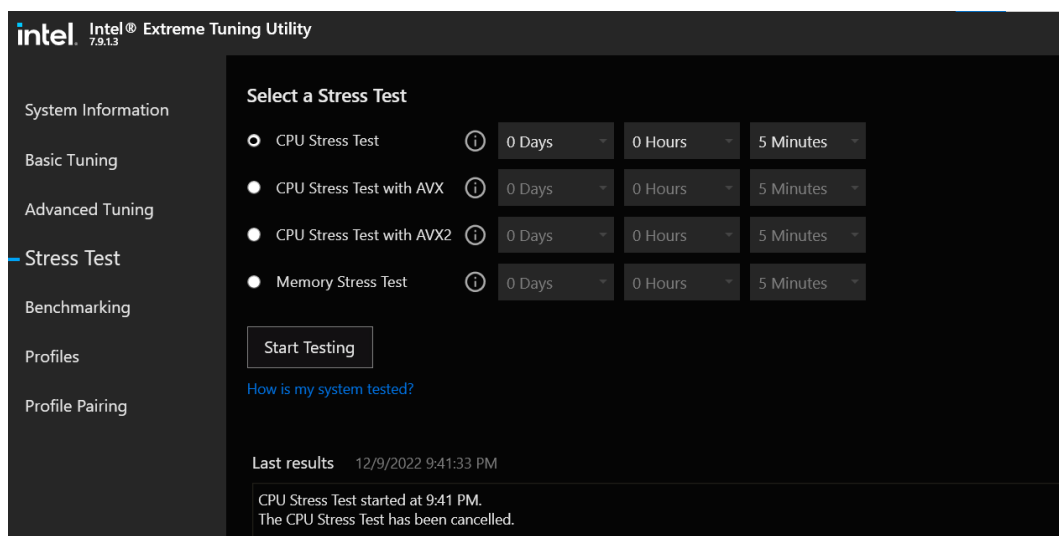
Select CPU stress test and click Start Testing. We are now running a multi-core benchmark on your system.

To see how many cores are running click on Advanced Tuning. You should see the full amount of cores on your CPU being used (12th gen and above will see P-cores being used, this is normal).
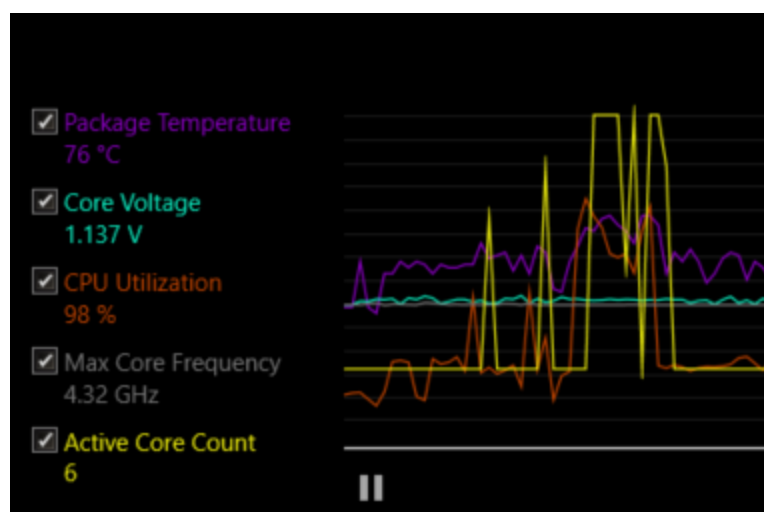
In your Advanced Tuning section, we will now start to overclock your system. The goal is to find the highest multicore frequency with the lowest stable voltage. Your system might crash once you have hit an unstable overclock. This won't do any damage but you will likely need to restart your system and re-apply the settings discussed earlier. I highly recommend writing down the multicore frequency and core voltage before each change.
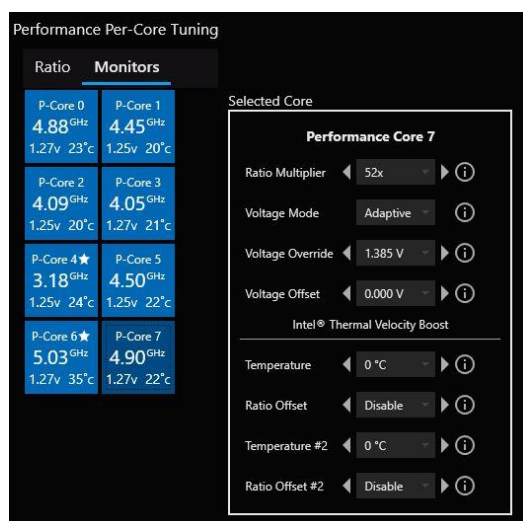
In the Active-Core Tuning section, we will only be changing the last Active Performance Core Ratio (in this case it's P-Core 7). This is done

by increasing the multiplier with the arrow on the right side of the ratio box.

I recommend looking up your CPU's default voltage as this will be the baseline for our core voltage setting. Instead of the button being

"default", change your Core Voltage to the actual voltage of your CPU by selecting the core voltage drop-down menu. **Make sure your Core Voltage Mode is set to Adaptive.**

If you were to keep increasing the Active Core Ratio with the same voltage, we would eventually crash. Meaning we need to increase Core Voltage when increasing our Active Core Ratio. As a general rule of thumb, do not go higher than 1.5V for most Intel CPUs.

If you're on a 10th-13th gen Intel CPU set your CPU voltage to 1.5V (don't worry we will change this). If you're on a previous gen CPU, add +.15V to the default voltage. Your benchmark should still be running, increase the last Active Core Ratio by increments of 1. Hit apply with each change. Write down your Active Core Ratio; this is your multicore frequency. Wait about 30 seconds between each increment.

Your system will likely crash (or crash XTU) once you have hit an unstable multi core frequency. Make sure you take note of your last stable multicore frequency.

Wonderful! We have found your highest stable multi core frequency. We will now move on to testing your single core frequency.

We will be using Cinebench to test single core performance. [Make sure Cinebench R23 is installed before continuing.](#)

Launch Cinebench R23. You should see a window that looks similar to this
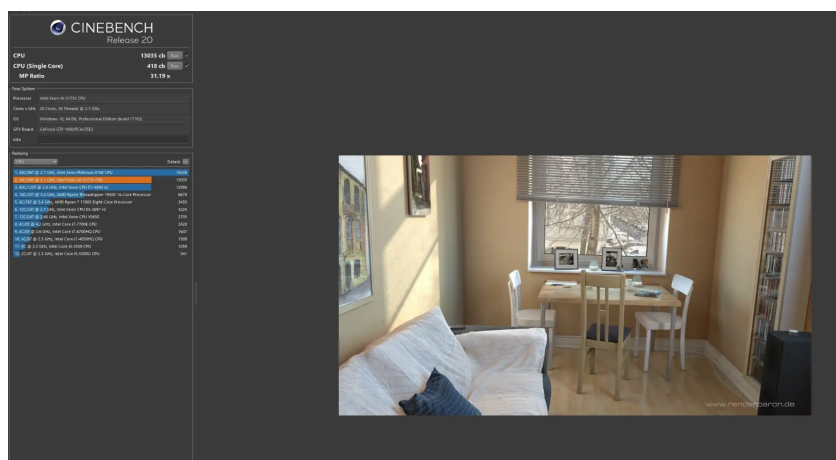
Select file, in the top left and click advanced settings. Set the minimum duration to 10 minutes. We will be using the CPU (Single Core) test. Before clicking run, load XTU.

Your settings should not have saved from our last test. Make sure these settings are applied:
- Voltage: **Adaptive and set to 1.5V (older than 10th gen add .15V to default voltage)**
- Intel Overclocking Thermal Velocity Boost: **Enabled**
- AVX2 Offset Ratio: **0**

Similar to the multi-core test, we will be changing the first Active Core Ratio by increments of one. Run the Cinebench R23 CPU (Single Core) test. Increase the Active Core Ratio by 1, hit apply, and wait 30
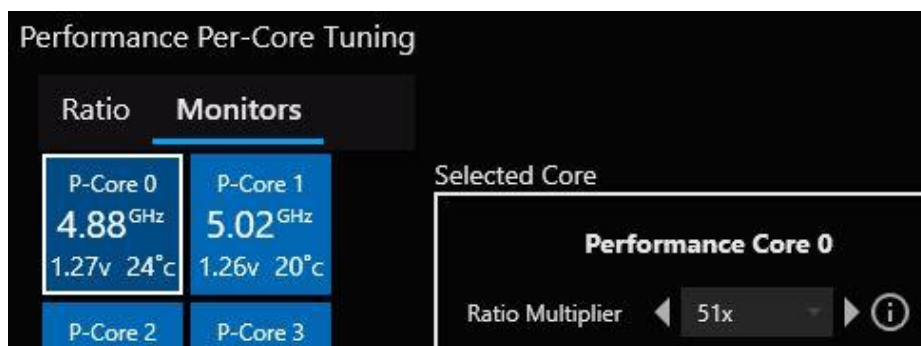
seconds. Continue documenting your last stable frequency until you crash (or Cinebench crashes). Record your last stable single core frequency.

Great, we have now found our highest single core frequency. You should have two stable frequencies, our single core, and multi core frequency. Next, we will tune our overclocking profile.

Chapter 2: Tuning Your Profile & Game Benchmark Optimization

Load XTU, we are now going to save a profile for the settings we just found.

Once again, make sure you have these settings applied:
- Voltage: **Adaptive and set to 1.5V (older than 10th gen add .15V to default voltage)**
- Intel Overclocking Thermal Velocity Boost: **Enabled**
- AVX2 Offset Ratio: **0**

Set your Active-Core Tuning for both all core (multi core/your last active core) and single core to the last stable frequencies you recorded.
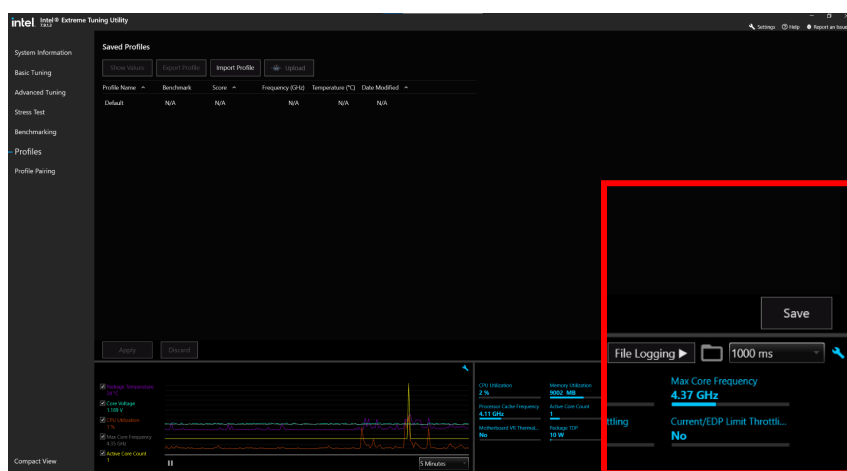
Once you have finished applying these settings, click on "Profiles" on the side menu of XTU.

In the bottom right, click the "Save" button. Name the saved profile: "XTU_V1".

If your system crashes, you will not have to reapply these settings. Simply, go back to the "Profiles" tab, select "XTU_V1" and click "Show Values" under Saved Profiles. Go back to the Advanced Tuning tab and click Apply.

Next, we will change the clock speeds for the rest of our cores. Essentially what we found earlier (all core and single core clock speed) were the lower and upper bounds of our CPU's clock speed. Here's an example of what this might look like for the 12900k (8 P-cores):

1 Active Core: 5.6 ← Our last stable single core clock speed.
2 Active Core: 5.5
3 Active Core: 5.4
4 Active Core: 5.4
5 Active Core: 5.3

6 Active Core: 5.2
7 Active Core: 5.1
8 Active Core: 5.1 ← Our last stable multi core clock speed.

For now, set your middle core count's clock speed ratio (aka everything except your first and last core) to +1 the previous core count. Do this until you reach 1 less than your single core clock speed. Here's an example for the 12900k that had a single core frequency of 5.5 GHz and multicore frequency of 5 GHz:
1 Active Core: 5.5
2 Active Core: 5.4
3 Active Core: 5.4
4 Active Core: 5.4
5 Active Core: 5.3
6 Active Core: 5.2
7 Active Core: 5.1
8 Active Core: 5

Before continuing to the next step, go through the process of saving our overclocking profile as discussed above and call it "XTU_V2".

Next, we will be testing our overclocking profile on any game of your choosing. It doesn't matter which one you choose, however, it needs to have a built-in benchmark. Quickly look up if the game you want to test with has a built-in benchmark. Once you confirm it has a benchmark feature load your game.
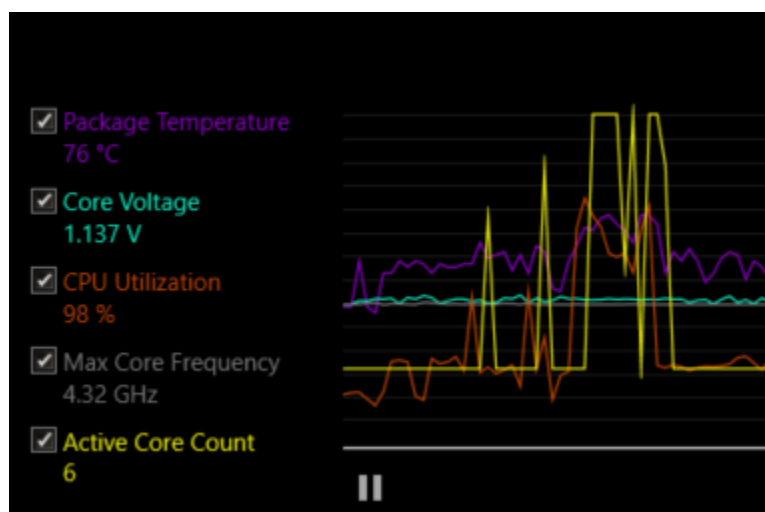
Run your first benchmark with your new overclocking profile. XTU should be open. If the game crashes it is because one of our core's clock speed is too high. XTU will give you a great visual indicator as to which core is problematic. Notice in the picture to the left, the last Active Core was #5 before the crash. Lower core #5 by -1.

Run the benchmark again, if it still doesn't work and the last active core was #5, lower it once more. Repeat this until you have a stable benchmark. Continue benchmarking and figuring out which core's have unstable clock speeds. Do this for multiple games, not just one. Continue saving your profiles as "XTU_V#" so that in case your system crashes you will have updated stable iterations of your profiles. The cores that aren't making the game crash are either stable or have potential to be increased. Increase them by +1, if it crashes lower it back so it is stable again.

Chapter 3: Intel Thermal Velocity Boost & More Optimization

Alright, we're almost there!

We are now going to change our Core Voltage which was set to 1.5V. Lower this voltage by

.025V and run your game benchmarks on your optimized profile. If they still run and are stable, update this as your new voltage. Go until you hit an unstable voltage and use the previous stable voltage.

Next, our final steps will be using Intel's Thermal Velocity Boost tool in XTU to continue optimizing our profile. The purpose of applying the TVB setting is to increase core performance and prevent overheating. This tool is best explained by an example:

- In the picture to the right, if the first core's temperature is above 40°C, then our ratio offset of -100MHz will be applied to our clock speed. If the first core's temperature is

above 80°C, then our second ratio offset of -100MHz will be applied. So if our first core is at 5.2 GHz and its temperature is above 80°C then our clock speed will be 5 GHz. This tool is to help the thermal performance of your system by slowing down your CPUs clock speed.

During my testing, I found that TVB is very effective in increasing the performance of the cores clock speeds.

Leave XTU open, let your system run while doing standard tasks. Make sure your mode is set to "Standby". Don't put your system under heavy load. After a minute or two, you should see a pretty standard package temperature with one active core enabled. Add +10°C to this number. The new number is going to be our single core temperature in TVB. As an example, in the scenario above our average package temperature was 25°C. Set the first TVB temperature to 35°C and set a ratio offset of -1:
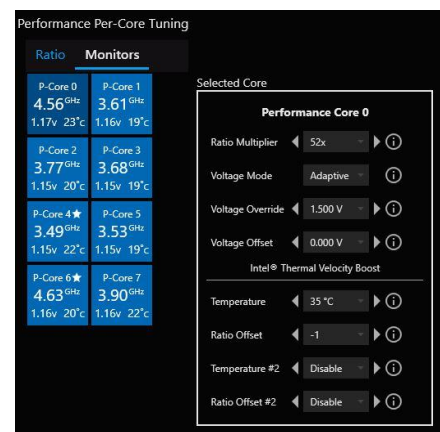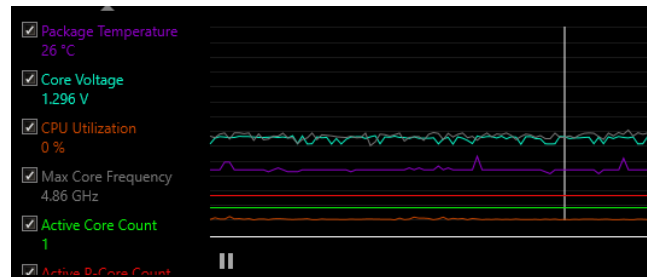
Change this by selecting the first core temperature drop down menu and the ratio offset menu. We will now try to get the highest stable single core frequency with the lowest ratio offset. You may crash here so I recommend saving your profiles after each stable iteration. Change back to Cryo Mode. Play your video game and continue
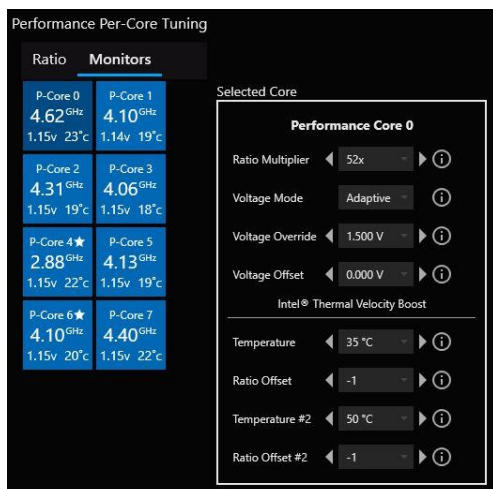
incrementing your frequency +1 about every minute. Once you crash your video game go back to the last stable frequency. Apply +15°C to your second temperature and add an offset of -1. Play your video game and make sure it works.

The second ratio offset when applied will only slow down the frequency. You can leave it disabled if you'd like, it mainly should be used for thermal management issues and making sure we don't crash at higher temperatures. If you still crash, I would recommend repeating the same procedure with your video game but decreasing your temperature.

For this next part, we will add +5°C (to the next core) for both my first and second temperature. As an example, if we were tuning a 12900k (8 P-cores), the first 4 cores would look like this:

Frequency	Temperature #1	Offset Ratio #1	Temperature #2	Offset Ratio #2
56x	35°C	-1	50°C	-1
55x	40°C	-1	55°C	-1
54x	45°C	-1	60°C	-1
53x	50°C	-1	65°C	-1

These will change, don't worry. Run your game benchmark and make sure your system is stable.

Similar to what we did above, we will be changing settings, seeing if they're stable with our game benchmark, and saving our stable iterations. In general, for the best performance, we will try to have the least amount of TVB offsets affecting our system. First try to improve the cores clock speeds by increasing them individually by +1x. Once you have hit an unstable clock speed for an individual core go back to the last stable profile and try to take away your TVB temperature offset (always start with your second). If you notice that your system is running too hot for some reason, you either have to lower your offset or decrease your temperature number (again start with your second temperature). Continue doing this until you have a stable and improved TVB offset profile.

As an example, my 12900k using this procedure ended up with these results (first 4 cores):

Frequency	Temperature #1	Offset Ratio #1	Temperature #2	Offset Ratio #2
57x	25°C	-1	50°C	-1
56x	30°C	-1	55°C	-2
56x	35°C	-1	60°C	-2
55x	40°C	-1	65°C	-1

Continue this process for the rest of your cores. One trick that I found helpful was mentioned above, where you monitor which core was problematic in XTU after a crash (refer to page 8).

Chapter 4: The End!

Great! We have effectively found the best overclocking profile for your system.

If you run into issues with the stability of your system, I recommend going back and tracking which core's clock speed in XTU is problematic. Do this by referring back to Chapter 2's section about monitoring Active Core Count.

Hope you found this tutorial helpful! Post any questions you have and I'll try to answer them, thanks.

Acknowledgements: