Evaluating GPP Predictors for Software Based Waveform Performance

William “BJ” Blair WBLAIR@ANDROCS.COM
ANDRO Computational Solutions, 111 Dart Circle, Rome, NY 13440 USA

Dr. Ashwin Amanna AAMANNA@ANDROCS.COM
ANDRO Computational Solutions, 111 Dart Circle, Rome, NY 13440 USA

Timothy Reichert TREICHERT@ANDROCS.COM
ANDRO Computational Solutions, 111 Dart Circle, Rome, NY 13440 USA

Michael Gudaitis MICHAEL.GUDAITIS@US.AF.MIL
Air Force Research Laboratory, Rome, NY 13440 USA

Abstract

Making full use of a computer’s capabilities today is a challenging task due to increased hardware and software complexity, requiring the use of multithreading, SIMD intrinsics, and overclocking to squeeze as much performance out of a system as possible. A challenge is predicting how a software-based waveforms will perform based on published benchmarks on a general purpose processor (GPP) of interest and where the key limiters exist. This is valuable insight to determine implementation and optimization strategies for software-based waveforms. This paper attempts to identify key indicators of modern GPP performance for usage with waveform software, using LDPC and DVB-S2 waveform benchmarks on two consumer grade desktops. We find a correlation in software performance between GPP or memory reliance and GPP clock speed and cache, as well as the importance of system tuning and overclocking.

This material is based upon work supported by Air Force Research Laboratory (AFRL/RITE) under contract FA8750-20-C-0529. Any opinions, findings, conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of AFRL.

1. Introduction

At ANDRO Computational Solutions, it is a common need for us to decide between different computer systems for use on a given GPP-based software waveform project. Thanks to Moore’s law and constant advancements in CPU technology, since 2015 for us this has been a recurring event, with the top available GPPs changing and being introduced constantly. For this paper the two particular CPUs in question are the Intel i9-13900K and the AMD Ryzen 9 7900X, which we test using three LDPC and DVB-S2 implementations: ANDRO’s proprietary implementation, AFF3CT (Cassagne et al., 2021), and gr-dvbs2rx (Freire & Economos, 2023), with LDPC from (Inan & Schiefer, 2023). The full system specifications for both systems are given in tables 1 and 2. As pure software-based waveforms run entirely on the GPP, maximum waveform rates and throughput are largely tied to and dependant on its performance. Software defined waveforms usually act as an intensive load on the system, so for our success we want to ensure we know of and select the highest performing GPPs and other computer components available. Our goal then is to provide guidance and draw conclusions about predicting said performance.

Both systems were tested with and without overclocking. For the i9 system, we were able to use a built-in AI overclocking feature, which involved training the system via a stress test, in our case Prime95 (Mersenne Research, 2023). This resulted in an increase in CPU frequency from 5.5 to 5.9 GHz, and a memory frequency increase from 4800 to 7200 MHz. We also disabled e-cores in the BIOs based on past testing which had showed improved performance with only p-cores enabled. On the Ryzen system, the AI feature was unavailable, and instead used the BIOs automatic overclocking, which increased the CPU frequency from 5.73 to 5.82 GHz and the memory from 4800 to 6400 MHz.

For full DVB-S2 waveform testing, we used two Ettus X300 SDRs connected to a single desktop via a 10 Gb ethernet card with two ports. They were wired together with SMA cables and 30dB of attenuation. One radio was used for transmit, the second for receive.

Initially, at a first glance, we expect the i9 to outperform the Ryzen based on CPU information and available benchmarks online. The i9 has a lower base clock speed but higher overclocked speed, and shows a higher PassMark CPU Mark (59,894 vs. 52,412) and single thread rating (4,678 vs. 4,327) (PassMark Software, 2023).

OS	Ubuntu 20.04.6 LTS
Linux Kernel	5.15.0-72-generic
Compiler	GCC 11.1.0
CPU	Intel i9-13900K
RAM	32 GB
CPU Power Governor	Performance
Active CPU Cores	16
Max CPU Speed	5.5/5.9 GHz
RAM Speed	4800/7200 MHz

Table 1. System specifications for i9 desktop.

OS	Ubuntu 20.04.6 LTS
Linux Kernel	5.15.0-72-generic
Compiler	GCC 11.1.0
CPU	AMD Ryzen 9 7900X
RAM	32 GB
CPU Power Governor	Performance
Active CPU Cores	24
Max CPU Speed	5.73/5.82 GHz
RAM Speed	4800/6400 MHz

Table 2. System specifications for Ryzen 9 desktop.

2. LDPC Benchmarks

We first ran throughput tests on three different LDPC implementations for use with DVB-S2, on both the i9 and Ryzen systems with and without overclocking. A code rate of $\frac{1}{2}$ and normal frame length (64800) were chosen arbitrarily. Early termination was disabled, all running with a fixed iteration count of ten. We used the separate, standalone benchmark programs included with each version, with parameters matching as closely as possible between them. AFF3CT ran with 4 threads, as this seemed to result in the maximum throughput out of testing with 1,4,5,6, and 16 threads. The XDSOPL program was modified to use a different throughput calculation to more closely match the ANDRO and AFF3CT calculations, and have early termination hard-coded to be disabled. The benchmarks used an EbN0 of 6.9897 dB (the default for the XDSOPL benchmark) to generate random, AWGN modified input. The exact benchmarking syntax for AFF3CT and XDSOPL is given in the appendix. The results are shown in table 3.

System	ANDRO	AFF3CT	XDSOPL
i9-13900K	160	113	139
i9-13900K Ovr.	168	129	139
Ryzen 9	217	116	110
Ryzen 9 Ovr.	222	119	112

Table 3. LDPC Benchmark Throughputs, in Mbps.

These results show that the ANDRO LDPC test performed better on the Ryzen, while the i9 gave higher results for the XDSOPL. For AFF3CT, the Ryzen performed slightly better without overclocking, but with overclocking gave higher throughput on the i9.

To explore why the tests performed differently between the two systems, we analyzed the benchmarks using using *perf*, *cachegrind*, *Intel VTune Profiler*, and *gprof*. Perf is a tool available in Linux to monitor CPU event data during an application's runtime, and requires no special usage or compilation flags of the target application. Cachegrind is a tool included with Valgrind which profiles a program's cache usage in simulation, and requires the application to be compiled with debug flags (-g in GCC). Finally, VTune and gprof are both profilers which trace the most time-consuming functions within an application's code. Gprof requires a special complication flag (-pg) to generate tracing information. VTune additionally can provide information such as memory, micro-architecture usage and flame graphs, with the downside being it needs to be ran on an Intel CPU to work fully. Thus for our tests we ran VTune on the i9 system. We were unable to use cachegrind on the XDSOPL and AFF3CT tests due to errors, as well as VTune on AFF3CT, for which we used gprof instead. In cachegrind with XDSOPL, we found the benchmark would exit with an *unhandled instruction/illegal opcode* message. When running AFF3CT in both cachegrind and VTune, the benchmark failed to allocate required memory. The perf output for each benchmark we did not find particularly useful or draw conclusions from when compared to the other programs and is therefore not discussed here.

We speculate that the ANDRO LDPC likely was faster on the AMD system due to the Ryzen having twice the amount of L3 cache-memory as the i9, as well as being direct-mapped as opposed to associative (table 4), which from our research is faster (Ho, 2018). As shown in table 5, the ANDRO implementation is more memory dependent, and thus is more affected by memory and cache speed, while the XDSOPL LDPC is more GPP dependent. The "Top AVX intrinsic" refers to the VTune hotspot summary results, indicating the CPU spent the most time executing said AVX call. The ANDRO intrinsic *_mm256_store_si256* is a memory storing operation, whereas the XDSOPL *_mm256_max_epi8* is a logical operation stressing the CPU. The bound percentages refer to how often the CPU was stalled on different memory locations during execution. According to (Areej, 2022), the Ryzen 9 has a 50% higher cache bandwidth than previous Ryzens and other 12th generation Intel CPUs. It is also worth nothing that, in similar work (Grayver, 2019), the author found that the amount of cache available is a limiting factor of waveform performance on a given CPU, as opposed to the number of cores available to add more processing threads.

For the AFF3CT version, we see that the Ryzen gave higher throughput at base clock speed, while the i9 throughput was higher when overclocked. This seems to match the Ryzen having a higher base frequency, but the i9 having a higher overclocked frequency. Finally, examining the gprof output for AFF3CT, we find the top reported function is *_decode_single_ite<0>*, which, examining the source code, seems to be composed of mostly logical operations over loading and storing operations. This indicates the AFF3CT LDPC is more CPU bound, going along with the base and overclocked results.

Metric	i9	Ryzen 9
I1 cache size	32768 B	32768 B
I1 cache type	8-way associative	8-way associative
D1 cache size	49152 B	32768 B
D1 cache type	12-way associative	8-way associative
LL cache size	37748736 B	67108864 B
LL cache type	18-way associative	direct-mapped

Table 4. ANDRO LDPC Cachegrind Report Output.

Metric	ANDRO	XDSOPL
Top AVX intrinsic	store	max
Back-End Bound	63.3%	28.7%
Memory Bound	33.3%	23.3%
L3 Bound	6.3%	0.6%

Table 5. LDPC Benchmark Selected VTune Results.

3. DVB-S2 Benchmarks

To benchmark the full DVB-S2 waveform, the ANDRO implementation ran in real time using X300s, and we increased the configured sample rate and symbol rate until the waveform would not stably synchronize. We ran at a maximum sample rate of 100 Msps and increased symbol rate from there. Due to the discrete available sample rates of the X300, to increase further we would have had to move directly to 200 Msps, which is why the reported symbol rates are so similar compared to AFF3CT and gr-dvbs2rx.

The AFF3CT and gr-dvbs2rx waveforms ran in simulation, as fast as they could on the given CPU. These two were simulated as we were unable to fully synchronize when using X300s with the AFF3CT waveform, while for gr-dvbs2rx we were unable to fully synchronize above 3 Msym/s, although HTOP reported CPU usage was low (50%). The gr-dvbs2rx real-time results are in line with the author's presentation in (Freire, 2022), which reports 1 Msym/s usage. The AFF3CT waveform initially would crash upon startup due to using up the entire system's RAM (32 GB), which was resolved by reducing the number of RX threads down from 28 to 8, based on the author's recommendation. We additionally compiled AFF3CT with *hwloc* disabled, as the receiver would crash and fail on thread pinning. For the synchronization issue, we contacted the authors of AFF3CT but received no response.

The results for symbol rate are shown in table 6. ANDRO and gr-dvbs2rx ran using DVB-S2 configuration QPSK, rate $\frac{1}{2}$, normal frame length, while AFF3CT used QPSK, rate $\frac{3}{5}$, short frames as the $\frac{1}{2}$ normal frame length configuration was unavailable. The result symbol rates for AFF3CT and gr-dvbs2rx were calculated based on the number of reported processed frames over the duration of the benchmark.

Unlike the LDPC benchmark results, the i9 seems to universally outperform the Ryzen 9 running the full DVB-S2 waveform. For the ANDRO implementation, running the full waveform as opposed to just LDPC appears to become more CPU bound rather than memory bound. In VTune profiler, the memory store AVX intrinsic moved from the top hotspot down to 4th, and is instead replaced with phase correction. Back-end, memory, L3 latency, and DRAM bound percentages all dropped from the ANDRO LDPC to DVB-S2, as shown in table 7. The AFF3CT and gr-dvbs2rx full waveform results both match their LDPC counterparts in that the i9 outperformed the Ryzen 9.

System	ANDRO	AFF3CT	gr-dvbs2rx
i9-13900K	65.000	52.116	14.502
i9-13900K Ovr.	75.000	54.704	15.033
Ryzen 9	65.000	50.816	12.512
Ryzen 9 Ovr.	65.000	51.283	12.966

Table 6. DVB-S2 Benchmark Max Symbol Rates, in Msps.

4. Conclusion and Future Work

When deciding between GPPs and different system components for use with waveform development, we draw the following conclusions. First, you cannot always rely on

Metric	LDPC	DVB-S2
Top Hotspot	_mm256_store_si256	correctPhase
Back-End Bound	63.3%	52.2%
Memory Bound	33.3%	25.8%
L3 Bound	6.3%	6.9%
L3 Latency	35.7%	12.6%
DRAM Bound	11.3%	8.8%

Table 7. ANDRO Benchmarks Selected VTune Results.

available consumer benchmarks to compare CPUs. Each application is unique and may interact differently with the hardware than a general stress-test program does, which tends to focus on stressing all of the available cores with as many threads as possible, whereas we tend to find single thread performance ratings more helpful. Note again that (Grayver, 2019) found cache as a limiting/key factor for scaling performance on a cpu, as opposed to the number of available cores. Similarly, in this paper we found higher AFF3CT throughput with only four threads, as opposed to as many threads as there are cores. Therefore, when comparing GPPs, one having more cores than another does not necessarily mean it will give better results for your application.

Second, it is important to consider your application's specific behavior when looking for CPU traits. Analyzing the waveform software with the tools from this paper or others available can show if the code is generally memory dependant or processing dependant. Knowing this, you can compare two systems and see which is stronger in the given area. In our case, the Ryzen seems better for cache and memory, while the i9 is better for CPU computational speed. Finally, overclocking ability is worth considering also. For the DVB-S2 benchmarks, the i9, which we were able to overclock much farther than the Ryzen, resulted in both AFF3CT and ANDRO outperforming non-overclocked Ryzen results, of which the Ryzen was superior initially.

For future work, we would like to investigate deeper into the real-time synchronization issues encountered with gr-dvbs2rx and AFF3CT. Running the ANDRO DVB-S2 implementation in simulation and the other two waveforms over RF, we could compare the simulated performance to real time performance. Additionally, investigating the AFF3CT hwloc issue could help improve throughput by pinning threads to GPP cores, reducing context switching by the OS.

References

Areej. Amd ryzen 9 7900x delivers nearly 50% more cache bandwidth than the 12th gen intel core cpus, August 2022. URL https://tinyurl.com/3fhejcnc. Last Accessed July 26, 2023.

Cassagne, Adrien, Leonardon, Mathieu, Tajan, Romain, Leroux, Camille, Jégo, Christophe, Aumage, Olivier, and Barthou, Denis. A Flexible and Portable Real-time DVB-S2 Transceiver using Multicore and SIMD CPUs. In *The 11th IEEE International Symposium on Topics in Coding (ISTC 2021)*, Montréal, Canada, August 2021. doi: 10.1109/ISTC49272.2021.9594063. URL https://hal.science/hal-03336450.

Freire, Igor. Grcon22: gr-dvbs2rx: An overview of the project state and path forward, Oct 2022. URL https://www.youtube.com/watch?v=qcqvfElQUVk. Last accessed 18 July 2023.

Freire, Igor and Economos, Ron. gr-dvbs2rx: a Software-Defined DVB-S2 Receiver based on GNU Radio, March 2023. URL https://github.com/igorauad/gr-dvbs2rx.

Grayver, Eugene. Scaling the fast x86 dvb-s2 decoder to 1 gbps. In *2019 IEEE Aerospace Conference*, pp. 1–9, 2019. doi: 10.1109/AERO.2019.8742225.

Ho, Steven. Direct-mapped and set associative caches, July 2018. URL https://inst.eecs.berkeley.edu/~cs61c/resources/su18_lec/Lecture15.pdf.

Inan, Ahmet and Schiefer, Jan. Ldpc: Playing with low-density parity-check codes, 2023. URL https://github.com/xdsopl/LDPC.

Mersenne Research, Inc. Gimps: Great internet mersenne prime search, 2023. URL https://www.mersenne.org/.

PassMark Software. Cpu benchmarks, 2023. URL https://www.cpubenchmark.net/compare/5022vs5027/Intel-i9-13900K-vs-AMD-Ryzen-9-7900X.

A. Benchmark/Waveform Syntax

XDSOPL LDPC: testbench 10 T2 A1 QAM16 32

AFF3CT LDPC: aff3ct -C LDPC -K 32400 -N 64800 –enc-type LDPC_ DVBS2 -m 6.9897 -M 7.0 –dec-simd INTER –mdm-type BPSK –mdm-implem FAST –src-type RAND –src-implem FAST –chn-implem FAST –dec-type BP_HORIZONTAL_LAYERED –dec-implem NMS –dec-norm 0.75 –dec-ite 10 –sim-threads 4 –dec-no-synd

gr-dvbs2rx DVB-S2: ./dvbs2-tx –source file –in-file ./example.ts –in-repeat –frame-size normal –modcod QPSK1/2

–sym-rate 5000000 –samp-rate 10000000 –snr 10.5 –freq-offset 0 — ./dvbs2-rx –frame-size normal –modcod QPSK1/2 –ldpc-iterations 10 –log-all –sym-rate 5000000 –samp-rate 10000000 –out-fd 3 3¿ /dev/null

AFF3CT DVB-S2: ./bin/dvbs2_tx_rx -F 8 –src-type USER_BIN –src-path ./example.ts –snk-path output_stream_fifo.ts –mod-cod QPSK-S_3/5 –dec-implem NMS –dec-ite 10 –dec-simd INTER