coreboot 4.20 release

The 4.20 release was done on May 15, 2023. Unfortunately, a licensing issue was found immediately after the release was completed, and it was decided to hold the release until that was fixed.

Please do not use the 4.20 tag, and use the 4.20.1 git tag instead. The 4.20_branch will contain all code for 4.20, 4.20.1, and any further changes required for this release.

The coreboot community has done a tremendous amount of work on the codebase over the last three and a half months. We've had over 1600 commits in that time period, doing ongoing cleanup and improvement.

It can be hard to remember at times how much the codebase really has improved, but looking back at coreboot code from previous years, it's really impressive the changes that have happened. We'd like to thank everyone who has been involved in these changes. It's great to work with everyone involved, from the people who make the small cleanup patches and review all of the incoming changes to the people working on new chipsets and SoCs. We'd additionally like to thank all of those individuals who make the effort to become involved and report issues or push even a single patch to fix a bug that they've noticed.

Many thanks to everyone involved!

We plan to get the 4.21 release done in mid August, 2023.

Significant or interesting changes

cpu/mp_init.c: Only enable CPUs once they execute code

On some systems the BSP cannot know how many CPUs are present in the system. A typical use case is a multi socket system. Setting the enable flag only on CPUs that actually exist makes it more flexible.

cpu/x86/smm: Add PCI resource store functionality

In certain cases data within protected memory areas like SMRAM could
be leaked or modified if an attacker remaps PCI BARs to point within
that area. Add support to the existing SMM runtime to allow storing
PCI resources in SMRAM and then later retrieving them.

This helps prevent moving BARs around to get SMM to access memory in
areas that shouldn't be accessed.

acpi: Add SRAT x2APIC table support

For platforms using X2APIC mode add SRAT x2APIC table
generation. This allows the setup of proper SRAT tables.

drivers/usb/acpi: Add USB _DSM method to enable/disable USB LPM per port

This patch supports projects to use _DSM to control USB3 U1/U2
transition per port.

More details can be found in
https://web.archive.org/web/20230116084819/https://learn.microsoft.com/en-us/windows-hardware/drivers/bringup/usb-device-specific-method---dsm-

The ACPI and USB driver of linux kernel need corresponding functions
to support this feature. Please see
https://git.kernel.org/pub/scm/linux/kernel/git/mnyman/xhci.git/log/?h=port_check_acpi_dsm

drivers/efi: Add EFI variable store option support

Add a driver to read and write EFI variables stored in a region device.
This is particularly useful for EDK2 as payload and allows it to reuse
existing EFI tools to set/get options used by the firmware.

The write implementation is fault tolerant and doesn't corrupt the
variable store. A faulting write might result in using the old value
even though a 'newer' had been completely written.

Implemented basic unit tests for header corruption, writing existing

data and append new data into the store.

Initial firmware region state:
Initially the variable store region isn't formatted. Usually this is
done in the EDK2 payload when no valid firmware volume could be found.
It might be useful to do this offline or in coreboot to have a working
option store on the first boot or when it was corrupted.

Performance improvements:
Right now the code always checks if the firmware volume header is valid.
This could be optimised by caching the test result in heap. For write
operations it would be good to cache the end of the variable store in
the heap as well, instead of walking the whole store. For read
operations caching the entire store could be considered.

Reclaiming memory:
The EFI variable store is append write only. To update an existing
variable, first a new is written to the end of the store and then the
previous is marked invalid. This only works on PNOR flash that allow to
clear set bits, but keep cleared bits state.
This mechanisms allows a fault tolerant write, but it also requires to
"clean" the variable store from time to time. This cleaning would remove
variables that have been marked "deleted".
Such cleaning mechanism in turn must be fault tolerant and thus must use
a second partition in the SPI flash as backup/working region.
For now, cleaning is done in coreboot.

Fault checking:
The driver should check if a previous write was successful and if not
mark variables as deleted on the next operation.

drivers/ocp/ewl: Add EWL driver for EWL type 3 error handling

Add EWL (Enhanced Warning Log) driver which handles Intel EWL HOB
and prints EWL type 3 primarily associated with MRC training failures.

Toolchain updates

* Upgrade MPC from version 1.2.1 to 1.3.1
* Upgrade MPFR from version 4.1.1 to 4.2.0

* Upgrade CMake from version 3.25.0 to 3.26.3
* Upgrade LLVM from version 15.0.6 to 15.0.7
* Upgrade GCC from version 11.2.0 to 11.3.0
* Upgrade binutils from version 2.37 to 2.40

Additional coreboot changes
----

* Remove Yabits payload. Yabits is deprecated and archived.
* Add DDR2 support to Intel GM45 code.
* Fix superiotool compilation issues when using musl-libc.
* Drop the Python 2 package from the coreboot-sdk.
* Drop the Zephyr SDK from coreboot-sdk since the packaged version
  was quite old and wasn't really used.
* Add inteltool support for the Intel "Emmitsburg" PCH.
* Work to improve cache hit percentage when rebuilding using ccache.
* Adding Sound-Open-Firmware drivers to chromebooks to enable audio on
  non-chrome operating systems.
* Improve and expand ACPI generation code.
* Fix some issues for the RISC-V code.
* Continue upstreaming the POWER9 architecture.
* Add documentation for SBOM (Software Bill of Materials).
* Add SimNow console logging support for AMD.
* Do initial work on Xeon SPR
* CMOS defaults greater than 128 bytes long now extend to bank 1.

New Mainboards
----

* Asrock: B75M-ITX
* Dell: Latitude E6400
* Google: Aurash
* Google: Boxy
* Google: Constitution
* Google: Gothrax
* Google: Hades
* Google: Myst
* Google: Screebo
* Google: Starmie
* Google: Taranza

* Google: Uldren
* Google: Yavilla
* HP: EliteBook 2170p
* Intel: Archer City CRB
* Intel: DQ67SW
* Protectli: VP2420
* Protectli: VP4630/VP4650
* Protectli: VP4670
* Siemens: MC EHL4
* Siemens: MC EHL5
* System76: lemp11
* System76: oryp10
* System76: oryp9

Removed Mainboards
----

* Intel Icelake U DDR4/LPDDR4 RVP
* Intel Icelake Y LPDDR4 RVP
* Scaleway TAGADA

Updated SoCs
----

* Removed soc/intel/icelake

Plans to move platform support to a branch

Intel Quark SoC & Galileo mainboard

The SoC Intel Quark is unmaintained and different efforts to revive it
have so far failed. The only user of this SoC ever was the Galileo
board.

Thus, to reduce the maintenance overhead for the community, support for
the following components will be removed from the master branch and will
be maintained on the release 4.20 branch.

  * Intel Quark SoC
  * Intel Galileo mainboard

Statistics from the 4.19 to the 4.20 release

- Total Commits: 1630
- Average Commits per day: 13.72
- Total lines added: 102592
- Average lines added per commit: 62.94
- Number of patches adding more than 100 lines: 128
- Average lines added per small commit: 37.99
- Total lines removed: 34824
- Average lines removed per commit: 21.36
- Total difference between added and removed: 67768
- Total authors: ~170
- New authors: ~35

Significant Known and Open Issues

Issues from the coreboot bugtracker: https://ticket.coreboot.org/
```eval_rst
+-----+-----------------------------------------------------------------+
| #   | Subject                                                         |
+=====+=================================================================+
| 478 | X200 booting Linux takes a long time with TSC                   |
+-----+-----------------------------------------------------------------+
| 474 | X200s crashes after graphic init with 8GB RAM                   |
+-----+-----------------------------------------------------------------+
| 457 | Haswell (t440p): CAR mem region conflicts with CBFS_SIZE > 8mb  |
+-----+-----------------------------------------------------------------+
| 453 | Intel HDMI / DP Audio device not showing up after libgfxinit    |
+-----+-----------------------------------------------------------------+
| 449 | ThinkPad T440p fail to start, continuous beeping & LED blinking |
+-----+-----------------------------------------------------------------+
| 448 | Thinkpad T440P ACPI Battery Value Issues                        |
+-----+-----------------------------------------------------------------+
| 446 | Optiplex 9010 No Post                                           |
+-----+-----------------------------------------------------------------+
```

```
| 439 | Lenovo X201 Turbo Boost not working (stuck on 2,4GHz)           |
+-----+-----------------------------------------------------------------+
| 427 | x200: Two battery charging issues                               |
+-----+-----------------------------------------------------------------+
| 414 | X9SAE-V: No USB keyboard init on SeaBIOS using Radeon RX 6800XT |
+-----+-----------------------------------------------------------------+
| 412 | x230 reboots on suspend                                         |
+-----+-----------------------------------------------------------------+
| 393 | T500 restarts rather than waking up from suspend                |
+-----+-----------------------------------------------------------------+
| 350 | I225 PCIe device not detected on Harcuvar                       |
+-----+-----------------------------------------------------------------+
| 327 | OperationRegion (OPRG, SystemMemory, ASLS, 0x2000) causes BSOD  |
+-----+-----------------------------------------------------------------+
```