Apex LLVM Release Notes

Please refer to ApexReadme.pdf for usage details and known issues.

Apex LLVM Release 13-August-2018

Build #530, based on clang 4.0.1

Compiler

- Fixed an issue where the end of vif region is not being detected
- Improved error handling of unsupported vector predication control flow

Apex LLVM Release 03-July-2018

Build #519, based on clang 4.0.1

Compiler

- Added code coverage metric generation using llvm profiler and coverage tools. Please see ApexReadme.pdf for usage details.
- Improved error message for invalid VLIW packet
- Fixed debug_frame entries to use word addresses
- Fixed large frame offset generation
- Fixed block copy of vector structure
- Fixed vector condition CU element access
- Fixed error when compiling preprocessable .S
- The use of unsupported -fPIC -fPIE option now results in error
- The tool opt, llc, llvm-mc, llvm-symbolizer has been removed from the layout. clang is the single invocation point for compilation going foward.

Libraries

- Bug fixes for SuperTests C11 conformance
- Added profiling runtime libraries and lcf

Documentation

- Documentation updates related to ISO26262 requirements
- New document ApexRuntime.pdf describes APEX specific runtime routines

Apex LLVM Release 27-March-2018

Build #486, based on clang 4.0.1

Compiler

- Fixed missing CFA info in debug frame
- Fixed dwarf location offset mismatch

Apex LLVM Release 08-Februrary-2018

Build #483, based on clang 4.0.1

Compiler

- Added support for attribute apex_hw_reg
- Improved DAG and MI scheduler heuristic
- Updated license for Design Studio 2018.R1
- Updated linker script to move stack to after user data
- Fixed CMPAPEX-792: error during register allocation

Apex LLVM Release 18-November-2017

Build #471, based on clang 4.0.1

Compiler

- Updated LLVM framework to 4.0
- Updated ISA with HDD-10294-14-11
- Added support for loop attribute apex_min_loop_count See ApexReadme for usage details
- Fixed sizeof() of vector aggregates

Libraries

- Added vmul, vmac, vmad intrinsics that return full 32-bit results
- Update intrinsics doxygen content
- Fixes for vsr, vselect instrinsics
- Updated link script for eh_frame sections

Apex LLVM Release 22-August-2017

Build #454, based on clang 3.9.1

Compiler

- **Added -fdwarf-cfi option for enabling call frame directive in debug info;**
  default off
- Updated license check for Design Studio v2.0

Apex LLVM Release 5-June-2017

Build #446, based on clang 3.9.1

Compiler

- Improved bundling of vmul / vacc instructions
- Fixed memset conversion using wrong address space
- Fixed assembling of vswap bundles

Libraries

- Added optimized vec16 divide routines
- Fixed vsube intrinsic

Kernels

- Ported chess_loop_range() and chess_unroll_loop() for vsdk kernels

Apex LLVM Release 4-May-2017

Build #444, based on clang 3.9.1

Compiler

- Added structural hazard handling for vector condition write port, between vector condition loads and vector compares.
- Fixed issue of -g generating unneeded nops.
- Bug fixes for apexcv_pro_canny, apexcv_pro_aggcf, gdc_ldw, gdc_cnn
- clang options cleanup for ISO26262 compliance.

Libraries

- Added vclb intrinsics of vec32u/s to intrinsics.hpp

Apex LLVM Release 24-March-2017

Build #439, based on clang 3.9.1

- Update license check version for Design Studio v1.1

Apex LLVM Release 24-Feburary-2017

Build #435, based on clang 3.9.1

Compiler

- Implemented VLIW Scheduler to generate more VLIW bundles opportunities.
- Added support for ptr_mod instructions and associated scalar-vector bundles.
- Improvements to delay slot filler to allow bundles.
- Bug fixes and integration for vsdk 0.9.7 demos

Apex LLVM Release 23-January-2017

Build #424, based on clang 3.9.1

Compiler

- The LLVM framework has been updated to 3.9.1
- Vector builtins now use the Extended vector type matching the headers.
- Optimization for scalar and vector memcpy and memset.
- Added ApexIntrinsics.pdf intrinsics reference document.
- Define __OPENCL_C_VERSION__ for OpenCL (cherry-picked from release_39)
- Bugfixes for for OpenCL vectors.

Apex LLVM Release 18-November-2016

Build #396, based on clang 3.6.2

Compiler

- Bugfixes for vec32s support
- Remove unneeded vector builtins
- Bugfixes for ApexCV Pro demos cppunitapp all passes except harris corner
- Initial OpenCL vector support

Libraries

- Cleanup of intrinsics.hpp

Apex LLVM Release 25-October-2016

Build #373, based on clang 3.6.2

Compiler

- Codegen improvements for branches
- Improvements to VLIW packetizer
- Bugfixes for ApexCV Pro demos

Linker

- Updated to binutils version 2.27

Libraries

- Added new tool for creating archives: llvm-ar

Apex LLVM Release 26-September-2016

Build #349, based on clang 3.6.2

Compiler

- Improvements for **vif** code generation.
- Added __LLVM_APEX__ and __LLVM_APEX_VERSION__ macros
- Updated relocations

Assembler

- Support for all VLIW encodings (32-bits and 64-bits)

Linker

- Updated relocations

Libraries

- Added vector intrinsics for pro kernels

Apex LLVM Release 31-August-2016

Build #306, based on clang 3.6.2

Compiler

- Implemented VLIW bundling of vector instructions
- Implemented assembler and disassembler for VLIW bundle
- Fixes to debug info for Lauterbach debugger compatibility
- Codegen improvements for sign extension and instruction schedule.

Apex LLVM Release 7-August-2016

Build #286, based on clang 3.6.2

Compiler

- Optimizations for vec32 multiply and carry
- Bugfix for vector alignment attribute
- Bugfix for vector predication in VIF/VELSE
- Various bugfixes for VSDK apex demos. All demos under vsdk/demos/apex standalone config are now passing

Libraries

- Added vmemset runtime routine

Apex LLVM Release 29-June-2016

Build #265, based on clang 3.6.2

Compiler

- Disabled jump table generation to workaround hardware bug.
- Updated license check

Linker

- Fix tctmemtab allocation issue

Libraries

- Added lcf for Treerunner hardware

Apex LLVM Release 11-June-2016

Build #257, based on clang 3.6.2

Compiler

- Added support for bundling scalar and vector instructions.
- Bugfixes

Apex LLVM Release 29-May-2016

Build #255, based on clang 3.6.2

Compiler

- Succesful validation with new Synopsys L-2016.03 release
- Bug fixes for hardware loops
- Fix debug relocation symbol mismatch

Linker

- Support for overlapping address space (VMb, DMb, PMh)

Apex LLVM Release 06-May-2016

Build #252, based on clang 3.6.2

Compiler

- Improvements to hardware loops generation

Apex LLVM Release 15-April-2016

Build #247, based on clang 3.6.2

Compiler

- More APU kernel fixes.
- All APU kernels from VSDK 0.9.0 are now passing under -O3 and -Os
- Updated license check for external use.

Apex LLVM Release 24-March-2016

Build #240, based on clang 3.6.2

Compiler

- VSDK kernel fixes
- Codegen improvements for hardware loops, fold immediate, and vector carry

Linker

- Update LCF to avoid stack overflow on simulator

Apex LLVM Release 27-February-2016

Build #217, based on clang 3.6.2

Compiler

- Updated to Clang/LLVM 3.6.2 (from 3.6.1)
- Each variable/function is implicitly placed in its own section (for stripping support)
- VSDK kernel fixes and improvements
- Activated Flexera license check

Linker

- Updated to binutils version 2.26

Libraries

- Additional intrinsics for Synopsys compatibility

Apex LLVM Release 30-January-2016

Build #178, based on clang 3.6.1

Compiler

- Bugfixes for vector kernels

Linker

- Bugfixes

Libraries

- Startup is now included in the libstartc.a and libstartcpp.a archives
- Added more vector intrinsics
- Built all Vision SDK kernels except apexcl
- Support Synopsys 15R6a simulator

Apex LLVM Release 17-December-2015

Build #144, based on clang 3.6.1

Compiler

- Bugfixes

Linker

- Bugfixes

Libraries

- Added libhosted.a and libhosted_syn.a

Apex LLVM Release 04-December-2015

Build #132, based on clang 3.6.1

Compiler

- Implements Apex ISA version HDD-10294-14-08
- Added support for vif/velse/vendif
- Added support for emulation of vdiv/vrem
- Various dwarf changes for Synopsys compatibility

Disassembler

- llvm-objdump provided, which is similar to GNU objdump
- llvm-mc supports `-disassemble` option

Linker

- Added support for tcthostedclib segment. To be used with Apex checker version 15.2.29.

Apex LLVM Release 10-November-2015

Build #115, based on clang 3.6.1

Compiler

- 32-bit vector instructions modeled in TD files
- Basic vec32s support
- Allow C-style casts between integer vectors of different element size
- Implementation of the intrinsics in `apex/intrinsics.hpp`
- Bugfixes in various places

Linker

- GOLD linker provided as `bin/ld-new`

Apex LLVM Release 30-September-2015

Build #81, based on clang 3.6.1

This is the initial release of LLVM toolchain for Apex.

Compiler

- Clang + LLVM 3.6 based
- Implements Apex ISA version HDD-10294-14-05
- Implements Apex ABI version HDD-10294-15-01
- 32 and 64-bit scalar code generation
- Pipeline model with delay slots
- Delay slot scheduling
- Tail call optimization
- Software floating point emulation
- GCC inline assembly syntax support
- Vector intrinsics support through `intrinsics.hpp`

Libraries

- EWL2 (libc, libc99, libm, librt) port for APEX
- Hosted IO support for simulator (using Synopsys hosted_io)
- LLVM compiler-rt builtins support

Assembler

- Implements GNU syntax assembler