OCLAcc
OpenCL™ for FPGA-Accelerators

Chair of Computer Science 3
Computer Architecture
Prof. Dr.-Ing. Dietmar Fey
dietmar.fey@fau.de

FRIEDRICH-ALEXANDER
UNIVERSITÄT
ERLANGEN-NÜRNBERG
FACULTY OF ENGINEERING

Project

- General Purpose Processors are more and more supported by accelerators with different programming environments. OpenCL tackles the divergence by offering a uniform, device-independent platform model to the user, leaving the mapping to physical devices to the OpenCL driver, often delivered by the device's vendor.

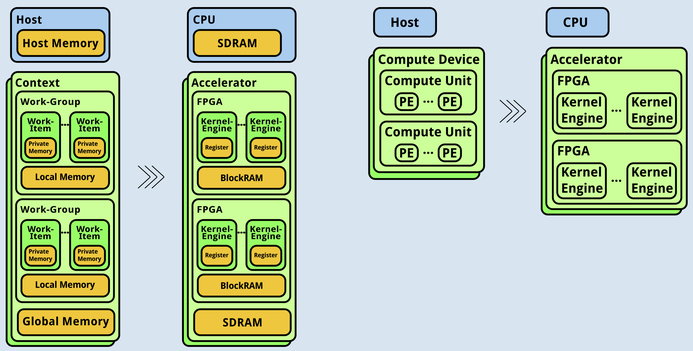
- OCLAcc tries to make FPGAs comparably easy to use to GPGPUs by developing an Open-Source OpenCL Driver for FPGA-based accelerators from different vendors.

Applications

- Scientific computing (e.g. Stencil Codes) on Altera Stratix, Xilinx Virtex via PCIe

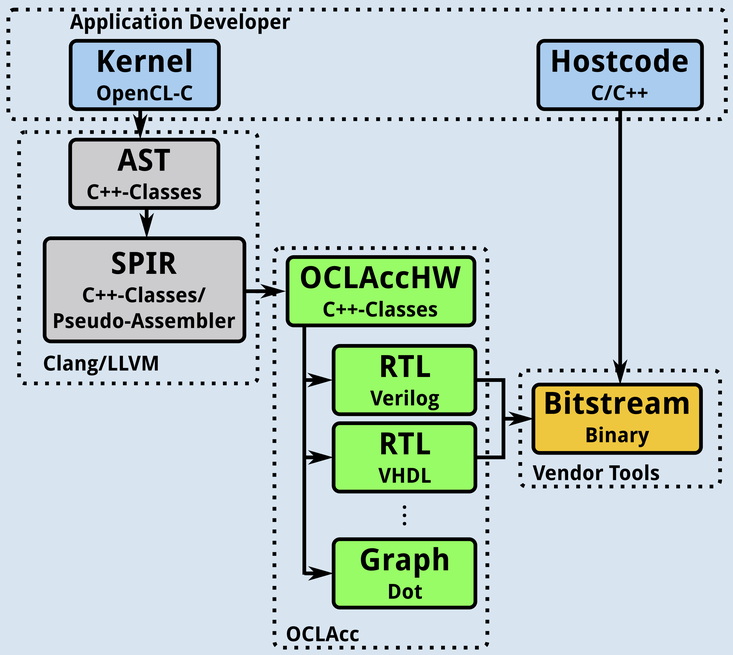
- Embedded signal and image processing on Altera SoC, Xilinx SoC

OpenCL-Mapping

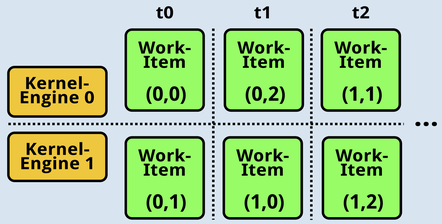


Overview

Our LLVM target machine transforms SPIR to a hardware description



Work Item Scheduling



- One or more kernel engines process work items in parallel

- Number of kernel engines depends on kernel complexity and FPGA ressources

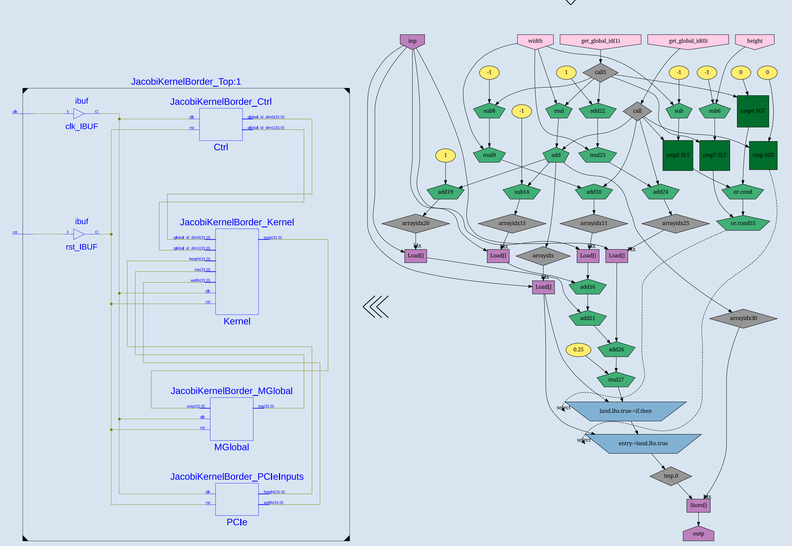
Transformations

```
__kernel void JacobiKernelBorder(
 __global float*inp, __global float*outp,
int width,int height)
{
int x =get_global_id(0);
int y =get_global_id(1);

float tmp = inp[y*width + x];
if((x >0) && (x < (width-1)) && (y >0) && (y < (height-1))) {
   tmp = ( inp[(y-1)*width + x] + inp[y*width + x-1] +
   inp[y*width + x+1] + inp[(y+1)*width + x] ) *.25f;
}

outp[y*width + x] = tmp;
}
```

```
...
%sub8 = addnsw i32 %call1, -1, !dbg !33
%mul9 = mulnsw i32 %sub8, %width, !dbg !33
%add10 = addnsw i32 %mul9, %call, !dbg !33
%arrayidx11 = getelementptr inbounds float addrspace(1)* %inp, i32 %add10, !dbg !33
%1 = load float addrspace(1)* %arrayidx11, align 4, !dbg !33, !tbaa !27
%sub14 = addnsw i32 %add, -1, !dbg !33
%arrayidx15 = getelementptr inbounds float addrspace(1)* %inp, i32 %sub14, !dbg !33
%2 = load float addrspace(1)* %arrayidx15, align 4, !dbg !33, !tbaa !27
%add16 = fadd float %1, %2, !dbg !33
%add19 = addnsw i32 %add, 1, !dbg !33
%arrayidx20 = getelementptr inbounds float addrspace(1)* %inp, i32 %add19, !dbg !33
%3 = load float addrspace(1)* %arrayidx20, align 4, !dbg !33, !tbaa !27
%add21 = fadd float %add16, %3, !dbg !33
%add22 = addnsw i32 %call1, 1, !dbg !33
%mul23 = mulnsw i32 %add22, %width, !dbg !33
%add24 = addnsw i32 %mul23, %call, !dbg !33
%arrayidx25 = getelementptr inbounds float addrspace(1)* %inp, i32 %add24, !dbg !33
%4 = load float addrspace(1)* %arrayidx25, align 4, !dbg !33, !tbaa !27
%add26 = fadd float %add21, %4, !dbg !33
%mul27 = fmul float %add26, 2.500000e-01, !dbg !33
tail call void @llvm.dbg.value(metadata !{float %mul27}, i64 0, metadata !18), !dbg !33
br label %if.end, !dbg !35
...
```



Contact

- Franz Richter-Gottfried
franz.richter-gottfried@fau.de
http://www3.cs.fau.de/Persons/richter/

- More information available on
http://www3.cs.fau.de