MID SEM PROJECT

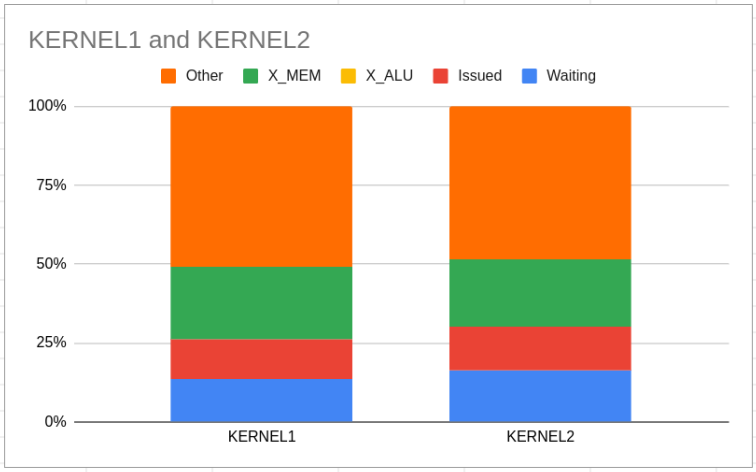
Group 15

1.Vivekanand (21CS02002)
2.R Saketh Kumar (21CS01017)
3.Macha Rushikeswara (21CS01012)
4.M Srinivas (21CS01070)
5.Y Sashank (21CS01065)

WARPS STATE BREAKDOWN FOR EACH KERNEL IN BFS

- 2 kernels in BFS.cu

	KERNEL1	KERNEL2
Waiting	13.47	16.53
Issued	12.59	13.7
X_ALU	0.007	0.017
X_MEM	23.16	21.37
Other	50.76	48.37



Code:

Others:

```
if (warp(warp_id).ibuffer_empty() || warp(warp_id).waiting()){
  other_count++;
  SCHED_DPRINTF(
      "Warp (warp_id %u, dynamic_warp_id %u) fails as ibuffer_empty\n",
      (*iter)->get_warp_id(), (*iter)->get_dynamic_warp_id());

  SCHED_DPRINTF(
      "Warp (warp_id %u, dynamic_warp_id %u) fails as waiting for "
      "barrier\n",
      (*iter)->get_warp_id(), (*iter)->get_dynamic_warp_id());
  }
```

Xmem:

```
if ((pI->op == LOAD_OP) || (pI->op == STORE_OP) ||
    (pI->op == MEMORY_BARRIER_OP) ||
    (pI->op == TENSOR_CORE_LOAD_OP) ||
    (pI->op == TENSOR_CORE_STORE_OP)) {
  if (m_mem_out->has_free(m_shader->m_config->sub_core_model,
                          m_id) &&
      (!diff_exec_units ||
       previous_issued_inst_exec_type != exec_unit_type_t::MEM)) {
    m_shader->issue_warp(*m_mem_out, pI, active_mask, warp_id,
                         m_id);
    issued++;
    issued_inst = true;
    warp_inst_issued = true;
    previous_issued_inst_exec_type = exec_unit_type_t::MEM;
  }

  else{
      xmem_count++;
  }
```

Xalu:

```
        if (execute_on_SP) {
          m_shader->issue_warp(*m_sp_out, pI, active_mask, warp_id,
                               m_id);
          issued++;
          issued_inst = true;
          warp_inst_issued = true;
          previous_issued_inst_exec_type = exec_unit_type_t::SP;
        } else if (execute_on_INT) {
          m_shader->issue_warp(*m_int_out, pI, active_mask, warp_id,
                               m_id);
          issued++;
          issued_inst = true;
          warp_inst_issued = true;
          previous_issued_inst_exec_type = exec_unit_type_t::INT;
        }
        else{
          xalu_count++;
        }

      } else if ((m_shader->m_config->gpgpu_num_dp_units > 0) &&
                 (pI->op == DP_OP) &&
                 !(diff_exec_units && previous_issued_inst_exec_type ==
                                      exec_unit_type_t::DP)) {
        if (dp_pipe_avail) {
          m_shader->issue_warp(*m_dp_out, pI, active_mask, warp_id,
                               m_id);
          issued++;
          issued_inst = true;
          warp_inst_issued = true;
          previous_issued_inst_exec_type = exec_unit_type_t::DP;
        }

        else{
          xalu_count++;
        }

      }
      else if (((m_shader->m_config->gpgpu_num_dp_units == 0 &&
                 pI->op == DP_OP) ||
                (pI->op == SFU_OP) || (pI->op == ALU_SFU_OP)) &&
               !(diff_exec_units && previous_issued_inst_exec_type ==
                                    exec_unit_type_t::SFU)) {
        if (sfu_pipe_avail) {
          m_shader->issue_warp(*m_sfu_out, pI, active_mask, warp_id,
                               m_id);
          issued++;
          issued_inst = true;
          warp_inst_issued = true;
          previous_issued_inst_exec_type = exec_unit_type_t::SFU;
        }

        else{
          xalu_count++;
        }
```

```
} else if ((pI->op == TENSOR_CORE_OP) &&
           !(diff_exec_units && previous_issued_inst_exec_type ==
                                  exec_unit_type_t::TENSOR)) {
  if (tensor_core_pipe_avail) {
    m_shader->issue_warp(*m_tensor_core_out, pI, active_mask,
                         warp_id, m_id);
    issued++;
    issued_inst = true;
    warp_inst_issued = true;
    previous_issued_inst_exec_type = exec_unit_type_t::TENSOR;
  }

  else{
    xalu_count++;
  }

} else if ((pI->op >= SPEC_UNIT_START_ID) &&
           !(diff_exec_units &&
             previous_issued_inst_exec_type ==
                 exec_unit_type_t::SPECIALIZED)) {
  unsigned spec_id = pI->op - SPEC_UNIT_START_ID;
  assert(spec_id < m_shader->m_config->m_specialized_unit.size());
  register_set *spec_reg_set = m_spec_cores_out[spec_id];
  bool spec_pipe_avail =
      (m_shader->m_config->m_specialized_unit[spec_id].num_units >
       0) &&
      spec_reg_set->has_free(m_shader->m_config->sub_core_model,
                             m_id);

  if (spec_pipe_avail) {
    m_shader->issue_warp(*spec_reg_set, pI, active_mask, warp_id,
                         m_id);
    issued++;
    issued_inst = true;
    warp_inst_issued = true;
    previous_issued_inst_exec_type =
        exec_unit_type_t::SPECIALIZED;
  }

  else{
    xalu_count++;
  }
```

Waiting:

```
    } else {
      SCHED_DPRINTF(
          "Warp (warp_id %u, dynamic_warp_id %u) fails scoreboard\n",
          (*iter)->get_warp_id(), (*iter)->get_dynamic_warp_id());

      wait_count++;
    }
  }
```