

## 1. Fichier de Configuration de l'Instance

# Fichier de Configuration de l'Instance

Afin de générer une instance, le générateur a besoin des directives de l'utilisateur. C'est l'objectif du **fichier de configurations de l'Instance**

```
<?xml version="1.0" encoding="ISO-8859-1" ?>

<cpu name="nom_instance" >

  <!-- *****
  * Définition du processeur *
  ***** -->

  <core size_data="..." > <!-- 1 -->

    <cache_unit> <!-- 1 -->

      <icache id="..."
        arbiter="..."
        nb_port="..." /> <!-- 1+ -->

      <dcache id="..."
        arbiter="..."
        nb_port="..." /> <!-- 1+ -->

    </cache_unit>

    <front_end id="..."
      link_icache="..."
      link_out_of_order_engine="..."> <!-- 1+ -->

    <fetch_unit> <!-- 1 -->

      <ifetch id="..."
        nb_inst="..."
        size_queue="..."
        link_decod="..." /> <!-- 1+ -->

    </fetch_unit>

    <prediction_unit arbiter="..."
      nb_prediction="..."
      nb_branch_complete="..." > <!-- 1 -->

      <branch_target_buffer size_queue="..."
        associativity="..." /> <!-- 1 -->

      <return_address_stack size_queue="..." /> <!-- 1 -->

      <branch_context size_queue="..." /> <!-- 1 -->

      <meta_predictor have_meta_predictor="..."
        predictor_0_have_bht="..."
        predictor_0_bht_size_shifter="..."
        predictor_0_bht_nb_shifter="..."
        predictor_0_have_pht="..."
        predictor_0_pht_size_counter="..."
        predictor_0_pht_nb_counter="..."
        predictor_0_pht_size_address_share="..."
        predictor_1_have_bht="..."
        predictor_1_bht_size_shifter="..."
```

```

        predictor_1_bht_nb_shifter="..."
        predictor_1_have_pht="..."
        predictor_1_pht_size_counter="..."
        predictor_1_pht_nb_counter="..."
        predictor_1_pht_size_address_share="..."
        predictor_2_have_bht="..."
        predictor_2_bht_size_shifter="..."
        predictor_2_bht_nb_shifter="..."
        predictor_2_have_pht="..."
        predictor_2_pht_size_counter="..."
        predictor_2_pht_nb_counter="..."
        predictor_2_pht_size_address_share="..." /> <!-- 1 -->
</prediction_unit>

<decod_unit> <!-- 1 -->

    <decod id="..."
        arbiter="..."
        nb_branch_decod="..."
        nb_inst="..."
        size_queue="..."
        link_rename="..." /> <!-- 1+ -->

</decod_unit>

</front_end>

<out_of_order_engine id="..."
    link_execution_loop="..."> <!-- 1+ -->

    <rename arbiter="..."
        nb_inst="..."
        size_queue="..."
        size_windows="..."
        nb_gpr_physical="..."
        nb_gpr_free="..."
        nb_spr_physical="..."
        nb_spr_free="..." /> <!-- 1+ -->

    <commit arbiter="..."
        nb_inst="..."
        size_queue="..."
        size_windows="..."
        commit_out_of_order_thread="..." /> <!-- 1 -->

</out_of_order_engine>

<execution_loop id="..."
    link_dcache="..." > <!-- 1+ -->

    <read_unit> <!-- 1 -->

        <reservation_station id="..."
            size_queue="..."
            one_queue="..." > <!-- 1+ -->

            <link id="..."> <!-- 1+ -->

        </reservation_station>

    </read_unit>

    <memory_unit size_queue="..."
        one_queue="..."
        nb_inst="..."
        size_windows="..."

```

```

        bypass_memory_out="..."
        keep_data="..."
        speculated_memory_read="..." > <!-- 1 -->

    <link id="..."> <!-- 1+ -->

</memory_unit>

<execution_unit> <!-- 1 -->

    <execution id="..."
        type_mul="..."
        type_div="..."
        type_ext="..."
        type_ffl="..."
        type_fll="..." /> <!-- 1+ -->

</execution_unit>

<write_unit> <!-- 1 -->

    <writeback id="..."
        size_queue="..."
        one_queue="..."
        bypass_execute_gpr="..."
        bypass_execute_spr="..."
        bypass_memory_in="..." > <!-- 1+ -->

        <link id="..."> <!-- 1+ -->

    </writeback>

</write_unit>

</execution_loop>

</core>

</cpu>

```

### Un petit exemple :

```

<?xml version="1.0" encoding="ISO-8859-1" ?>

<cpu name="nom_instance" >

    <core size_data="32" >

        <cache_unit>
            <icache id="0" arbiter="0" nb_port="1" />
            <dcache id="0" arbiter="0" nb_port="1" />
        </cache_unit>

        <front_end id="0" link_icache="0" link_out_of_order_engine="0">
            <fetch_unit>
                <ifetch id="0" nb_inst="1" size_queue="4" link_decod="0" />
                <ifetch id="1" nb_inst="1" size_queue="4" link_decod="0" />
            </fetch_unit>

            <prediction_unit arbiter="0" nb_prediction="1" nb_branch_complete="1" >
                <branch_target_buffer size_queue="32" associativity="4" />
                <return_address_stack size_queue="8" />
                <branch_context size_queue="8" />
                <meta_predictor have_meta_predictor="1"

```

```

        predictor_0_have_bht="1"
        predictor_0_bht_size_shifter="10"
        predictor_0_bht_nb_shifter="16"
        predictor_0_have_pht="0"
        predictor_1_have_bht="0"
        predictor_1_have_pht="1"
        predictor_1_pht_size_counter="2"
        predictor_1_pht_nb_counter="64"
        predictor_1_pht_size_address_share="3"
        predictor_2_have_bht="1"
        predictor_2_bht_size_shifter="4"
        predictor_2_bht_nb_shifter="32"
        predictor_2_have_pht="1"
        predictor_2_pht_size_counter="2"
        predictor_2_pht_nb_counter="32"
        predictor_2_pht_size_address_share="4" />
</prediction_unit>

<decod_unit>
  <decod id="0" arbiter="0" nb_branch_decod="1" nb_inst="1" size_queue="4" link_rename="0"
</decod_unit>

</front_end>

<out_of_order_engine id="0" link_execution_loop="0"> <!-- 1+ -->

  <rename arbiter="1"
    nb_inst="1"
    size_queue="8"
    size_windows="3"
    nb_gpr_physical="64"
    nb_gpr_free="2"
    nb_spr_physical="16"
    nb_spr_free="1" />

  <commit arbiter="0"
    nb_inst="1"
    size_queue="32"
    size_windows="2"
    commit_out_of_order_thread="1" />

</out_of_order_engine>

<execution_loop id="0">

  <read_unit>
    <reservation_station id="0" size_queue="4" one_queue="1" >
      <link id="0">
    </reservation_station>
  </read_unit>

  <memory_unit size_queue="8"
    one_queue="1"
    nb_inst="1"
    size_windows="2"
    bypass_memory_out="1"
    keep_data="1"
    speculated_memory_read="1" >
    <link id="0">
  </memory_unit>

  <execution_unit>
    <execution id="0"
      type_mul="1"
      type_div="0"
      type_ext="0"

```

```

        type_ffl="1"
        type_fll="1" />
</execution_unit>

<write_unit>
  <writeback id="0"
    size_queue="4"
    one_queue="1"
    bypass_execute_gpr="1"
    bypass_execute_spr="1"
    bypass_memory_in="0" >
    <link id="0">
  </writeback>
</write_unit>

</execution_loop>
</core>
</cpu>

```

### Autres exemple :

```

<?xml version="1.0" encoding="ISO-8859-1" ?>

<cpu name="nom_instance" >

  <core>
    <parameter name="size_data" value="32" />

    <cache_unit>
      <icache id="0">
        <parameter name="arbiter" value="0" />
        <parameter name="nb_port" value="4" />
      </icache>
      <icache id="1">
        <parameter name="arbiter" value="0" />
        <parameter name="nb_port" value="1" />
      </icache>
      <dcache id="0">
        <parameter name="arbiter" value="0" />
        <parameter name="nb_port" value="1" />
      </dcache>
    </cache_unit>

    <front_end id="0">
      <fetch_unit>
        <context id="0">
          <parameter name="nb_inst" value="1" />
          <parameter name="size_queue" value="4" />
          <link dest="icache" id="0" slot="1,2,3" />

          <!--
          NOTE :
          l'attribut slot est optionnel, dans ce cas la valeur par défaut est 0

          La notation :

          <link dest="icache" id="0" slot="1,2,3" />

          est équivalente à :

          <link dest="icache" id="0" slot="1" />
          <link dest="icache" id="0" slot="2" />
          <link dest="icache" id="0" slot="3" />
          -->

```

```
    <link dest="icache" id="1" slot="1" />
  </context>
  <context id="1">
    <parameter name="nb_inst" value="4" />
    <parameter name="size_queue" value="8" />
    <link dest="icache" id="0" slot="1,2,3,4" />
  </context>
</fetch_unit>
</front_end>

</core>

</cpu>
```