CS23M117

V.NIKHITHA

PROJECT DETAILS

- Tested the parser and simulator using dhrystone SPEC bench mark and removed all the printf statements, simplified and corrected the code which is uploaded in repository as dhrystone.c.
- We cross compiled dhrystone.c to mips64 gcc 14.1.0 dhrystone.s file using an online cross compiler "compiler explorer"

  https://godbolt.org/

PARSER/ SCHEDULER DESIGN

1. **Instruction Class**

   Represents an assembly instruction.

   Attributes: opcode, operands, dependencies.

2. **BasicBlock Class:**

   Represents a basic block of assembly instructions.

3. **Function Class:**

   Represents a function consisting of basic blocks.

4. **parse_assembly_file(filename):**

   Reads assembly code from a file and returns it as a string.

5. **parse_instructions(assembly_code):**

   Parses assembly code string and extracts instructions.

   Creates Instruction objects for each instruction.

6. **detect_dependencies(instructions):**

   Detects data dependencies between instructions.

   Updates the dependencies attribute of each Instruction object accordingly.

7. optimize_ilp(function, fetch_width):

   Optimizes the code using Instruction-Level Parallelism (ILP).

   Reorders instructions to maximize utilization of the fetch width.

   Ensures that instructions with no dependencies can be executed in parallel.

   Handles dependencies by rescheduling instructions when fetch width is exceeded.

8. **perform_loop_unrolling(function):**

   Identifies loops in the code and unrolls them.

   Replaces loop instructions with multiple copies of the loop body.

9. **write_rescheduled_program(optimized_instructions, output_file):**

   Writes the optimized instructions to an output file.

SUPERSCALAR SIMULATOR DETAILS:

Code for super sclar simulator is written in python. Calculates number of cycles required to execute the mips program.

- Fetch width =4
- Number of decoders =4
- Depth =6
- Fetch, decode, issue, execute, reorder buffer and commit are the super scalar pipeline stages.

- Uses a 2 bit branch predictor and introduced stalls during deoendencies

The parser and superscalar processor are tested on dhrystone SPEC bench mark

Superscalar simulateor takes 102cycles to execute original mips file of dhrystone and 93 cycles to execute resheduled mips file using parser.

Hence the resheduler improves performance by loop unrolling, resheduling during dependecies, shedules according to the fetch width 4.