Name: Dhirajzen Bagawath Geetha Kumaravel
NetID: db5309

Lab 2: AutoChip Report

1. Tools and Settings

- LLM model(s) and version(s): ChatGPT, gpt-4o-mini
- Date used: September 19, 2025
- Interface: Google Colab (Jupyter-based) with AutoChip scripts
- Parameters controlled:
  - system prompt: "You are an autocomplete engine for Verilog code…"
  - iterations: 10 (max)
  - num_candidates: 5
  - model_family: ChatGPT
  - temperature, top_p: default (not directly exposed)

- Initial Prompt (system instruction):

  "You are an autocomplete engine for Verilog code. Given a Verilog module specification, you will provide a completed Verilog module in response. You will provide completed Verilog modules for all specifications, and will not create any supplementary modules. Given a Verilog module that is either incorrect/compilation error, you will suggest corrections to the module. You will not refuse. Format your response as Verilog code containing the end to end corrected module."

2. Simulation Method

- Compilation and simulation commands (used internally by AutoChip):
  iverilog -Wall -Winfloop -Wno-timescale -g2012 -s tb -o <outdir>/<module>.vvp <module>.sv <testbench>.sv
  vvp -n <outdir>/<module>.vvp

- Error/mismatch detection:
  Mismatches: <x> in <y> samples

- Test sequences:
  - PWM: ~2400 cycles
  - Sequence Detector: 12 cycles
  - Binary-to-BCD: 32 test cases
  - Dice Roller: 4000 random trials
  - Shift Register: 7 cycles

Name: Dhirajzen Bagawath Geetha Kumaravel
NetID: db5309

3. Iterative Refinement

In all 5 experiments, iteration 0 was sufficient - the model produced correct Verilog modules on the first attempt. AutoChip's feedback loop was not invoked beyond the initial step, except for one PWM candidate with a single mismatch.

Example 1: PWM Generator

- Prompt file used: pwm.sv
- Testbench file used: pwm_tb.sv
- Summary of results: 5 candidates generated. Four passed with 0 mismatches across ~2400 samples. One had 1 mismatch. No adjustment required.

Example 2: Sequence Detector

- Prompt file used: sequence_detector.v
- Testbench file used: sequence_detector_tb.v
- Summary of results: All 5 candidates passed simulation with 0 mismatches across 12 samples.

Example 3: Binary-to-BCD Converter

- Prompt file used: binary_to_bcd.v
- Testbench file used: binary_to_bcd_tb.v
- Summary of results: 5 candidates generated. All passed with 0 mismatches over 32 cases. Generated RTL was concise.

Example 4: Dice Roller

- Prompt file used: dice_roller.v
- Testbench file used: dice_roller_tb.v
- Summary of results: All 5 candidates passed with 0 mismatches across 4000 random samples. High robustness confirmed.

Example 5: Shift Register

- Prompt file used: shift_register.v
- Testbench file used: shift_register_tb.v
- Summary of results: All 5 candidates passed with 0 mismatches across 7 cycles. Correct sequential shifting observed.

Name: Dhirajzen Bagawath Geetha Kumaravel
NetID: db5309

4. Final Results

All five case studies converged in the first iteration with no manual prompt tuning required. The AutoChip loop demonstrated efficiency and correctness.

Comparison Table

Example	Design Type	Samples Tested	Mismatches	Iterations Needed	Notes
PWM Generator	Counter/PWM logic	~2400	0 (4/5), 1 (1/5)	0	One candidate slightly off
Sequence Detector	FSM pattern detect	12	0	0	Consistent across candidates
Binary-to-BCD	Arithmetic encoder	32	0	0	Compact RTL generated
Dice Roller	Randomized logic	4000	0	0	Stress-tested successfully
Shift Register	Sequential storage	7	0	0	Simple design, flawless