

CPULATOR Quick Reference

Click the Compile and Load button at the top, then use Continue (F3) to run or Step Into (F2) to execute one instruction at a time; Stop (F4) halts execution and Restart (Ctrl-R) resets the program state. The toolbar has Step Over (Ctrl-F2) and Step Out (Shift-F2) for function-aware stepping, plus Reload (Ctrl-Shift-L) to reassemble after edits.

Basic run steps

- Paste or open code in the Editor area, and set Language to ARMv7 if writing assembly.
- Click Compile and Load to assemble and load into the simulator; the Disassembly pane will show the loaded instructions.
- Click Continue (F3) to run until a breakpoint or infinite loop; use Stop (F4) to halt. Prefer Step Into (F2) for learning/debugging.

Buttons to know

- Step Into (F2): executes the next instruction; best for seeing each state change.
- Step Over (Ctrl-F2): runs a call fully and stops at the next instruction on the same stack depth.
- Step Out (Shift-F2): runs until returning from the current function.
- Continue (F3): runs continuously until a breakpoint or stop.
- Stop (F4): stops the currently running program.
- Restart (Ctrl-R): resets PC and state to program start.
- Reload (Ctrl-Shift-L): rebuilds/loads after editing source.

Breakpoints

- Click in the leftmost gray gutter of the Disassembly pane to toggle a breakpoint on an instruction; then use Continue to run to it.
- View addresses of breakpoints in the Breakpoints tab under Registers.

Loading a prebuilt ELF

- If assembling elsewhere, use File → Load ELF Executable to load .axf/.elf and then use the same run/step buttons.

If it “hangs” on run

- Many beginner examples end with an infinite branch (e.g., b . or b _stop); Step Into will loop forever unless a breakpoint is set; use Stop (F4).
- Use Restart, set a breakpoint on a later instruction, then Continue to reach it.

Where to start

- Help → Sample shows sample programs to load and explore, then run with the toolbar buttons listed above.