Seal5 passes: execution order and scope

Read T01–T38, then G01–G11, then the expanded G12 pipeline P01–P04.

M = once per input model G = once globally * = conditional / opt-in

Transform · part 1

T01	convert_models	M
T02	filter_models	M
T03	drop_unused	M
T04	eliminate_mod_rfs	M
T05	eliminate_rd_cmp_zero	M
T06	drop_unused2	M
T07	inline_functions	M
T08	drop_unused3	M
T09	optimize_model	M
T10	infer_types	M
T11	simplify_trivial_slices	M
T12	explicit_truncations	M
T13	unroll_loops	M
T14	infer_types1	M
T15	detect_behavior_constraints	M
T16	detect_registers	M
T17	collect_register_operands	M
T18	collect_immediate_operands	M
T19	collect_operand_types	M

Continue at T20 →

Transform · part 2

T20	process_settings	M
T21	write_yaml2	M
T22	process_settings3	M
T23	detect_side_effects	M
T24	detect_inouts	M
T25	detect_imm_leafs	M
T26	detect_calls	M
T27	detect_loops	M
T28	annotate_opcodes	M
T29	handle_auto_unroll_operands	M
T30	gen_auto_intrinsics	M
T31	write_yaml3	M
T32	infer_types2	M
T33	explicit_memory_names *	M
T34	explicit_register_names *	M
T35	normalize_subscripts *	M
T36	detect_branches *	M
T37	check_pattern_support	M
T38	write_cdsl_full	M

Continue at G01 →

Generate

G01	riscv_gnu *	G
G02	seal5_td	G
G03	set_td	M
G04	riscv_features	M
G05	riscv_isa_infos	M
G06	riscv_register_info	M
G07	riscv_field_types	G
G08	riscv_instr_info	M
G09	riscv_intrinsics	M
G10	riscv_disass	M
G11	riscv_gisel_legalizer	G
G12	pattern_gen	M

G12 expands into:

P01	write_cdsl_compat	M
P02	behav_to_llvmir	M
P03	llvmir_to_gmir	M
P04	behav_to_pat	M

Default wrapper: G01–G11 run before phases 1–2 are applied; G12 runs after PatternGen / llc are built.
Order follows the registered lists, not Seal5Pass.order. Passes are serial; model instances may run concurrently.
All transforms and P01–P04 have MODEL scope. Four generators have GLOBAL scope. INSTR scope is not implemented;
SET is marked unsupported in the API and is not used by any registered pass. Splitting files per instruction does not change scope.
* T33–T36 require enabled=true; G01 needs a GNU source tree. Filters and PatternGen eligibility can skip additional work.
gen_model_td is an available handler but is not registered; it has no default execution position (MODEL, phase 1).
Source: seal5/flow.py (pass maps + create_passes) · seal5/passes.py · seal5/pass_list.py (pattern_gen_pass)