Runtime options of the generator						
S. No.	**Option**	**Description**	**Default**	**Arguments we need to use for generating instructions depending on supported features**	**1) x (cross) means using defaut values 2) (tick) means passing values for +args (otherwise default values is used)**	
1	num_of_tests	Number of assembly tests to be generated	1	✅	x	
2	num_of_sub_program	Number of sub-program in one test	5	✅	✅	
3	instr_cnt	Instruction count per test	200	✅	✅	
4	enable_page_table_exception	Enable page table exception	0	x	x	
5	enable_unaligned_load_store	Enable unaligned memory operations	0	✅	✅	
6	no_ebreak	Disable ebreak instruction	1	✅	✅	
7	no_wfi	Disable WFI instruction **with interupts**	1	✅	✅	
8	set_mstatus_tw	Enable WFI to be treated as illegal instruction **without interupts**	0	✅	✅	
9	no_dret	Disable dret instruction	1	✅	✅	
10	no_branch_jump	Disable branch/jump instruction	0	✅	✅	
11	no_csr_instr	Disable CSR instruction	0	✅	✅	
12	enable_illegal_csr_instruction	Enable illegal CSR instructions	0	✅	✅	
13	enable_access_invalid_csr_level	Enable accesses to higher privileged CSRs	0	✅	✅	
14	enable_dummy_csr_write	Enable some dummy CSR writes in setup routine	0	?	✅	
15	enable_misaligned_instr	Enable jumps to misaligned instruction addresses **with compressed enable**	0	✅	✅	
16	no_fence	Disable fence instruction	0	✅	✅	
17	no_data_page	Disable data page generation	0	x	✅	
18	disable_compressed_instr	Disable compressed instruction generation	0	✅	✅	
19	illegal_instr_ratio	Number of illegal instructions every 1000 instr	0	✅	✅	
20	hint_instr_ratio	Number of HINT instructions every 1000 instr	0	x	✅	
21	boot_mode	m:Machine mode, s:Supervisor mode, u:User mode, **we only have boot for machine mode**	m	✅	✅	
22	no_directed_instr	Disable directed instruction stream	0	✅	x	
23	require_signature_addr	Set to 1 if test needs to talk to testbench	0	x	✅	
24	signature_addr	Write to this addr to send data to testbench	0	x	✅	
25	enable_interrupt	Enable MStatus.MIE, used in interrupt test	0	✅	✅	
26	enable_timer_irq	Enable mIE.mTIE, used to enable timer interrupts	0	✅	✅	
27	gen_debug_section	Enables randomized debug_rom section	0	?	✅	
28	num_debug_sub_program	Number of debug sub-programs in test	0	?	✅	
29	enable_ebreak_in_debug_rom	Generate ebreak instructions inside debug ROM	0	✅	✅	
30	set_dcsr_ebreak	Randomly enable dcsr.ebreak(m/s/u) **we have m & u**	0	✅	✅	
31	enable_debug_single_step	Enable debug single stepping functionality	0	✅	✅	
32	randomize_csr	Fully randomize main CSRs (xSTATUS, xIE)	0	x	✅	only randomization between m and u
33	floating point	enable floating point instructions	0	✅	✅	
34	enable_nested_interrupt	Enable nested interrupt	0	x	✅	
35	no_load_store	No load store instructions	0	✅	✅	
36	fix_sp	Use SP as stack pointer	0	✅	✅	
37	bare_program_mode	Generate a bare program without any init/exit/error handling/page table routines The generated program can be integrated with a larger program. Note that the bare mode program is not expected to run in standalone mode	0	?	x	
38	use_push_data_section	push/pop section for data pages	0	x	x	
39	allow_sfence_exception	sfence support	0	x	x	
40	no_delegation	Interrupt/Exception Delegation	1	x	x	Add test for no delegation = 0
41	num_of_harts	Number of harts to be simulated **we have 1 hart**	NUM_HARTS	✅	x	
42	force_m_delegation	force m delegation	0	x	x	
43	force_s_delegation	force s delegation	0	x	x	
44	set_mstatus_mprv	Enable users to set mstatus.mprv to enable privilege checks on memory accesses.	0	✅	✅	
45	enable_vector_extension	Vector extension support	0	x	x	
46	enable_b_extension	Bit manipulation extension support	0	x	x	