Z-Ant Model Performance Comparison

	beer_model		**new2_model**		**darknet_s_model**		**fomo8_model**		**mnist-8_mode**
	ReleaseFast	**ReleaseSmall**	**ReleaseFast**	**ReleaseSmall**	**ReleaseFast**	**ReleaseSmall**	**ReleaseFast**	**ReleaseSmall**	**Release**
Total Instructions	87,791,919	130,930,476	2,376,299,678	5,638,188,336	72,170,395	106,000,990	5,801,221	5,724,155	15,218,7
Exec Time massif	605 ms	558 ms	2,420 ms	4,920 ms	697 ms	608 ms	434 ms	428 ms	402 ms
Primary alloc	36,960B	73,824B	102,208B	112,192B	45,120B	45,120B	110,652B	110,652B	50,240B
Primary alloc %	99.91%	99.95%	99.96%	99.96%	99.93%	99.93%	99.95%	99.95%	99.94%
Total heap usage	2,494,460B	2,494,492B	1,554,336B	6,199,296B	650,352B	650,416B	286,025B	286,025B	113,744
Memory leaks	1,728B	1,728B	320B	320B	16B	16B	24B	24B	40B
Executable Size	1.7M	88K	9.5M	9.1M	19M	17M	1.2M	182K	1.4M
Peak Memory	36.96 KB	73.82 KB	102.2 KB	112.2 KB	45.12 KB	45.12 KB	110.7 KB	110.7 KB	50.24 KE
Memory Timeline	0→36K	0→73K	0→102K	0→112K	0→45K	0→45K	0→110K	0→110K	0→50K
Allocations	163 allocs	164 allocs	33,353 allocs	323,660 allocs	259 allocs	267 allocs	17 allocs	17 allocs	1,877 al
nicla Vision	1586753	1657644							
RaspberryPy									
RaspberryPy									

FQBN="arduino:mbed_nicla:nicla_vision"

arduino-cli compile
--fqbn "$FQBN"
--export-binaries
--libraries ~/Arduino/libraries
--build-property "compiler.c.elf.extra_flags=-Wl,-T$PWD/custom.ld"

Key Observations

Performance Characteristics:

- **ReleaseFast** generally produces larger executables but with better runtime performance (fewer instructions for most models)
- **ReleaseSmall** produces significantly smaller executables but may have performance trade-offs

Model Complexity:

- **new2_model**: Most complex with highest instruction count and memory usage
- **fomo8_model**: Most memory-efficient with lowest allocation count
- **mnist-8_model**: Smallest peak memory footprint
- **darknet_s_model**: Largest executable size in both optimization modes
- **mobilenet_v2_model**: Moderate complexity with consistent memory usage across optimizations

Memory Behavior:

- All models show consistent memory leak patterns across optimization modes
- Memory allocation efficiency is very high (>99.9%) across all models
- ReleaseSmall sometimes increases memory usage (beer, new2) but reduces executable size significantly
- mobilenet_v2 shows identical memory behavior between optimization modes

Optimization Trade-offs:

- **Size reduction**: ReleaseSmall achieves 19x-28x size reduction for some models (beer: 1.7M→88K, mnist-8: 1.4M→50K)
- **Performance impact**: Varies by model - some show instruction count increases with ReleaseSmall
- **Memory consistency**: Peak memory usage remains similar between optimization modes for most models

- **mobilenet_v2**: Shows 38% increase in instructions with ReleaseSmall but maintains identical memory profile