Benchmark	Used[1]	Source	Description	Code structure & characteristics							
				NCSL	T	L	N	A	B	CP	MF	C
base64	Both	Online[2]	Computes the base64 encoding	32	1	✓		✓	✓			
radix4Div	Both	Online[3]	Optimized Software Division for platforms without hardware support	63	1	✓			✓	✓		
B.Radix4Div	Both	Online[3], modified	Software Division for platforms without hardware support	37	1	✓			✓	✓		
dijkstra	Prof.	Online[4]	Find shortest paths from source to all vertices in the given graph	34	1	✓	✓	✓	✓	✓	✓	
sha256	Prof.	Online[5]	Implementation of the SHA-256 hashing algorithm	168	1	✓		✓	✓		✓	
FIR	Both	XMOS	Finite Impulse Response filter	40	1	✓		✓	✓			
P. FIR_7T	Both			103	7	✓		✓	✓			✓
biquad	Both	XMOS	Signal equaliser using biquad filtering	49	1			✓	✓			
biquad_2T	RSA			55	2			✓	✓		✓	
biquad_4T	RSA			57	4			✓	✓		✓	
P. Biquad_7T	Both			94	7			✓	✓		✓	✓
SFloatAdd32bit	Prof.	SoftFloat	Single point F. addition for platforms without hardware support	359	1				✓	✓	✓	
SFloatSub32bit	Prof.	SoftFloat	Single point F. subtraction for platforms without hardware support	371	1				✓	✓	✓	
matMul	Both	MDH WCET	Matrix multiplication of two square matrices	15	1	✓	✓	✓				
matMul_2T	Both			25	2	✓	✓	✓				
matMul_4T	Both			27	4	✓	✓	✓				
jpegdct	Both	MDH WCET	Performs a JPEG discrete cosine transform	35	1	✓	✓	✓	✓			
jpegdct_2T	Both			43	2	✓	✓	✓	✓		✓	
jpegdct_4T	Both			45	4	✓	✓	✓	✓		✓	
ndes	Prof.	MDH WCET	Complex embedded code	253	1	✓		✓	✓		✓	
qsort	Prof.	MDH WCET	Non-recursive version of quick sort algorithm	101	1	✓	✓	✓			✓	
bs	Prof.	MDH WCET	Binary search	90	1	✓		✓	✓			
minver	Prof.	MDH WCET	Inversion of Matrix	155	1	✓	✓	✓	✓		✓	
ludcmp	Prof.	MDH WCET	LU decomposition algorithm	89	1	✓	✓	✓			✓	
nsichneu	Prof.	MDH WCET	Simulate an extended Petri Net	1469	1	✓						
cnt	Both	MDH WCET	Counts non-negative numbers in a matrix	29	1	✓	✓	✓				
st	Both	MDH WCET	Statistics program	85	1			✓	✓		✓	
mac	Both	MDH WCET	Dot product of two vectors and sum of squares	11	1	✓		✓				
crc	Prof.	BEEBS	Cyclic redundancy check computation on 40 bytes of data	18	1	✓		✓	✓			
recursion	Prof.	BEEBS	A simple example of recursive code	50	1						✓	
bsort100	Both	BEEBS	Bubblesort program	66	1	✓	✓		✓			
levenshtein	Both	BEEBS	Measures the difference between two strings	26	1	✓	✓	✓	✓		✓	

NCSL: Non-comment source-lines **T**: Number of threads **L**: Contains loops **N**: Nested loops **A**: Uses arrays/matrices **B**: Bitwise operations **CP**: Complex CFG structure **MF**: Multiple functions **C**: Contains thread communications.

[1] Indicates the energy estimation method(s) applied on the benchmark: **RSA**: Used only with Bound Static Analysis, **Prof.** Used only with Profiling, **Both**: Used with both RSA and profiling
[2] Retrieved from http://stackoverflow.com/questions/342409, Nov 2014.
[3] Retrieved from http://tinyurl.com/ld7exmd, Nov 2014.
[4] Retrieved from http://www.geeksforgeeks.org/greedy-algorithms-set-6-dijkstras-shortest-path-algorithm/, Nov 2014.
[5] Retrieved from https://github.com/B-Con/crypto-algorithms, Jul 2016.

Table I: Description and attributes of benchmarks.