

XBX Benchmarking Results August 2010

Christian Wenzel-Benner¹ and Jens Gräff²

¹ ITK Engineering AG
Software Center 1, 35037 Marburg, Germany
Christian.Wenzel-Benner@itk-engineering.de,
WWW home page: <http://www.itk-engineering.de>

² LiNetCo GmbH
Hauptstrasse 17a, 35684 Dillenburg, Germany
jgraef@linetco.com,
WWW home page: <http://www.linetco.com>

1 Introduction

We benchmarked many implementations of most SHA-3 candidate algorithms on several platforms. The benchmarking method is called XBX, short for eXternal Benchmarking eXtension, an extension of the SUPERCOP-eBASH framework that allows benchmarking small devices. For details on how XBX works, please see [1]. The main sources of implementations were SUPERCOP³ (supercop-20100712.tar.bz2), the avr-crypto-lib⁴, its derivate arm-crypto-lib and sphlib⁵. We wrote converter scripts to interface them to the SUPERCOP API. Some implementations had to be adapted manually, we succeeded in doing so for a fast AVR assembly Grøstl⁶ and a fast AVR assembly Skein⁷. The Skein implementation however failed test vectors above 32 byte input length and hence does not pass the XBX checksum test. Because we were unable to resolve the issue in time for the 2nd SHA-3 candidate conference we can not list an official benchmark result for this implementation.

Note: The previous version of this document missed some benchmarking numbers, namely keccak256 and hamsi due to problems in the table generation process. This has been fixed.

2 Result Data

The following pages contain the full results from the benchmarking runs. Please be aware that as per SUPERCOP benchmarking flow, these are only the best in class for each platform. Readers who would like to see how slower or larger implementations performed should look up the results of the 'try' runs on our

³ (see <http://bench.cr.yp.to/supercop.html>)

⁴ (see <http://avrcryptolib.das-labor.org/trac>)

⁵ (see <http://www.saphir2.com/sphlib/>)

⁶ (see <http://www.groestl.info/implementations.html>)

⁷ (see <http://www.syntax-k.de/projekte/fhreefish/>)

website <http://xbx.das-labor.org/trac/wiki>. The speed tables are ordered by try speed (1536 byte input length), the slides show speed ordered by long message speed. Thus the ranking may differ slightly. Readers interested in the arguably more realistic input lengths of 64 or 512 bytes should look at the website and sort the result tables there by the criteria of their choice.

Nomenclature

1024 Bytes	Median speed in cpb using a message length of 1024 bytes
2048 bytes	Median speed in cpb using a message length of 2048 bytes
512 Bytes	Median speed in cpb using a message length of 512 bytes
64 Bytes	Median speed in cpb using a message length of 64 bytes
8 Bytes	Median speed in cpb using a message length of 0 bytes
Compiler	Identifier of compiler options used
cpb	Cycles per byte
Empty Message ...	Median speed in cpb using a message length of 0 bytes
Hash	Identifier of a hash function
Implementation ...	Name of a specific hash function implementation
Long Messages	Speed estimation for message lengths greater than 2048 bytes, calculated from the speed difference between 2048 bytes and 1024 bytes message length
RAM Usage	Sum of data and bss segment sizes and stack usage minus data and bss segment sizes and stack usage of 0hash
ROM Usage	Sum of text and data segment sizes minus text and data segment sizes of 0hash
Source	Identifier of the original data set containing the measurement
Try Cycles (1536) .	Median speed in cpb using a message length of 1536 bytes
Version	Version of compiler used
WRIR	Worst relative interquartile range, worst relative interquartile range of all measurements

2.1 Atmel ATmega1281

8 bit microcontroller manufactured by Atmel. RISC design, one of the most powerful 8 bit designs currently in use. 128kiB Flash, 8kiB RAM, 16MHz. See http://www.atmel.com/dyn/products/product_card.asp?part_id=3630 for details.

Table 1. Fastest Hashes on the ATmega1281 Platform

Hash	Implementation	Try Cycles (1536)	Empty Mes- sage	8 Bytes	64 Bytes	512 Bytes	1024 Bytes	2048 Bytes	Long Mes- sages	WRIR [%]	ROM Usage	RAM Usage	Compiler	Version	Source
shabal256	avrcryptolib-asm	374708	52325	6542	1027	312	261	235	210	0.0	1900	298	gcc 1281 -Os	4.3.4	R4
shabal512	avrcryptolib-asm	374930	52547	6570	1031	313	261	236	210	0.0	1898	330	gcc 1281 -Os	4.3.4	R4
bmw256	avrcryptolib-asm	809047	64287	8038	1489	610	548	516	485	0.0	11776	3166	gcc 1281 -Os	4.3.4	R1
fugue256	sphlib	1358657	113354	15237	2611	1035	922	866	810	0.0	41788	5326	gcc 1281 -O3	4.3.4	R4
blake32	sphlib	1419667	60206	7521	1841	1004	944	914	884	0.0	39902	1764	gcc 1281 -O3	4.3.4	R2
skein256	avrcryptolib-asm	1929005	117924	14754	2451	1360	1282	1243	1204	0.0	2398	268	gcc 1281 -Os	4.3.4	R2
echo256	avrcryptolib	1964790	220677	27587	3455	1285	1280	1173	1065	0.0	3534	704	gcc 1281 -O2	4.3.4	R3
groestl256	opt32-avr	2147656	136362	17155	3440	1576	1443	1376	1309	0.0	24596	487	gcc 1281 -Os	4.3.4	R4
skein512	avrcryptolib-asm	2405522	279950	35007	4387	1811	1627	1535	1443	0.0	2500	429	gcc 1281 -O3	4.3.4	R2
fugue512	sphlib	2645590	263856	34797	5704	2069	1809	1679	1549	0.0	50946	5322	gcc 1281 -O1	4.3.4	R4
echo512	avrcryptolib	3511553	272838	34108	4270	2642	2375	2242	2108	0.0	3598	742	gcc 1281 -O2	4.3.4	R4
simd256	sphlib	3766022	152312	37855	4733	2650	2501	2427	2353	0.0	91276	5074	gcc 1281 -O3	4.3.4	R2
jh256	sphlib-small	4232010	179152	43533	5470	2991	2814	2726	2637	0.0	28464	2269	gcc 1281 -O3	4.3.4	R4
jh512	sphlib-small	4232265	179409	43565	5474	2992	2814	2726	2637	0.0	28464	2301	gcc 1281 -O3	4.3.4	R4
luffa256	sphlib	4358654	176478	22062	5501	3069	2896	2809	2722	0.0	20972	770	gcc 1281 -O3	4.3.4	R4
bmw512	avrcryptolib	5326397	766270	95792	11981	4466	3717	3343	2969	0.0	17896	1140	gcc 1281 -O2	4.3.4	R2
jh224	bitslice_opt32	5333129	217375	53900	6727	3755	3543	3437	3331	0.0	24038	2295	gcc 1281 -O2	4.3.4	R4
jh384	bitslice_opt32	5333275	217524	53919	6729	3755	3543	3437	3331	0.0	24038	2315	gcc 1281 -O2	4.3.4	R4
groestl512	sphlib-small	5825341	654499	81807	10220	4647	4006	3686	3365	0.0	23494	5326	gcc 1281 -Os	4.3.4	R4
blake64	armcryptolib	6326408	497821	62252	7788	4767	4281	4038	3795	0.0	6834	971	gcc 1281 -O2	4.3.4	R2
keccak256	sphlib-small	7042589	601938	75236	9399	4610	4591	4582	4572	0.0	74616	3317	gcc 1281 -O1	4.3.4	R4
luffa512	sphlib	7658056	454899	56864	11835	5581	5135	4911	4688	0.0	46948	1476	gcc 1281 -O3	4.3.4	R4
hamsi	sphlib-small	8312273	89949	16595	6758	5529	5441	5397	5353	0.0	50840	5065	gcc 1281 -O3	4.3.4	R4
cubehash1632	avrcryptolib-cwbasm	9197589	2795615	349460	47850	9628	6898	5533	4168	0.0	1648	249	gcc 1281 -O3	4.3.4	R3
keccak512	sphlib-small	12811402	598244	74774	9341	9124	8537	8243	7949	0.0	74616	3285	gcc 1281 -O1	4.3.4	R4

Table 2. Smallest Hashes Considering RAM Usage on the ATmega1281 Platform

Hash	Implementation	Try Cycles (1536)	ROM Usage	RAM Usage	Compiler	Version	Source
cubehash1632	avrcryptolib-cwbasm	9197607	1648	249	gcc 1281 -O3	4.3.4	R4
skein256	avrcryptolib-asm	1929028	2414	266	gcc 1281 -O1	4.3.4	R4
blake32	avrcryptolib	1797346	3586	269	gcc 1281 -O1	4.3.4	R4
shabal256	avrcryptolib-asm	374726	1916	296	gcc 1281 -O1	4.3.4	R4
shabal512	avrcryptolib-asm	374948	1914	328	gcc 1281 -O1	4.3.4	R4
bmw256	avrcryptolib-tinyasm	1369989	1926	333	gcc 1281 -O1	4.3.4	R4
groestl256	avrcryptolib	13312925	2214	346	gcc 1281 -Os	4.3.4	R4
skein512	avrcryptolib-asm	2405547	2516	427	gcc 1281 -O1	4.3.4	R4
blake64	avrcryptolib	6752174	6472	527	gcc 1281 -O1	4.3.4	R4
keccak512	avrcryptolib	26345105	3928	638	gcc 1281 -O1	4.3.4	R4
groestl512	avrcryptolib	19713892	2306	642	gcc 1281 -Os	4.3.4	R4
keccak256	avrcryptolib	14380336	3928	669	gcc 1281 -O1	4.3.4	R2
echo256	avrcryptolib	2002263	3586	697	gcc 1281 -Os	4.3.4	R4
echo512	avrcryptolib	3579357	3602	735	gcc 1281 -Os	4.3.4	R4
luffa256	sphlib	4451917	21840	762	gcc 1281 -O1	4.3.4	R4
bmw512	avrcryptolib	5352763	17594	1136	gcc 1281 -Os	4.3.4	R4
luffa512	sphlib	7713006	45252	1402	gcc 1281 -Os	4.3.4	R4
jh256	bitslice_ref32	5877072	36620	1631	gcc 1281 -O3	4.3.4	R4
jh512	bitslice_ref32	5877375	36620	1663	gcc 1281 -O3	4.3.4	R4
jh224	bitslice_ref32	5877048	41410	1883	gcc 1281 -O3	4.3.4	R4
jh384	bitslice_ref32	5877228	41410	1903	gcc 1281 -O3	4.3.4	R4
simd256	sphlib-small	4706213	35108	2970	gcc 1281 -O1	4.3.4	R4
fugue256	ref-xbx	5756548	10212	3024	gcc 1281 -Os	4.3.4	R4
fugue512	ref-xbx	11453411	10212	3056	gcc 1281 -Os	4.3.4	R4
hamsi	sphlib-small	8312273	50840	5065	gcc 1281 -O3	4.3.4	R4

Table 3. Smallest Hashes Considering ROM Usage on the ATmega1281 Platform

Hash	Implementation	Try Cycles (1536)	ROM Usage	RAM Usage	Compiler	Version	Source
cubehash1632	avrcryptolib-cwbasn	9197833	1470	251	gcc 1281 -Os	4.3.4	R4
shabal512	avrcryptolib-asm	374930	1898	330	gcc 1281 -Os	4.3.4	R4
shabal256	avrcryptolib-asm	374708	1900	298	gcc 1281 -Os	4.3.4	R4
bmw256	avrcryptolib-tinyasm	1369971	1910	335	gcc 1281 -Os	4.3.4	R4
groestl256	avrcryptolib	13312925	2214	346	gcc 1281 -Os	4.3.4	R4
groestl512	avrcryptolib	19713892	2306	642	gcc 1281 -Os	4.3.4	R4
skein256	avrcryptolib-asm	1929010	2398	268	gcc 1281 -Os	4.3.4	R4
skein512	avrcryptolib-asm	2405529	2500	429	gcc 1281 -Os	4.3.4	R4
echo256	armcryptolib	5365715	3296	969	gcc 1281 -Os	4.3.4	R4
echo512	armcryptolib	9647531	3312	1007	gcc 1281 -Os	4.3.4	R4
blake32	armcryptolib	1766149	3528	611	gcc 1281 -O1	4.3.4	R4
keccak512	armcryptolib	26408439	3870	844	gcc 1281 -O1	4.3.4	R4
keccak256	armcryptolib	14414762	3870	876	gcc 1281 -O1	4.3.4	R4
blake64	armcryptolib	6600783	6398	972	gcc 1281 -O1	4.3.4	R4
jh256	bitslice_ref32	6041068	8760	1996	gcc 1281 -Os	4.3.4	R4
jh512	bitslice_ref32	6041320	8760	2028	gcc 1281 -Os	4.3.4	R4
fugue256	ref-xbx	5756548	10212	3024	gcc 1281 -Os	4.3.4	R4
fugue512	ref-xbx	11453411	10212	3056	gcc 1281 -Os	4.3.4	R4
jh224	bitslice_ref32	6041055	13528	2248	gcc 1281 -Os	4.3.4	R4
jh384	bitslice_ref32	6041202	13528	2268	gcc 1281 -Os	4.3.4	R4
bmw512	armcryptolib	5758680	16724	1289	gcc 1281 -O1	4.3.4	R4
luffa256	sphlib	4390245	20380	768	gcc 1281 -Os	4.3.4	R4
simd256	sphlib-small	4587521	31930	3036	gcc 1281 -Os	4.3.4	R4
luffa512	sphlib	7713006	45252	1402	gcc 1281 -Os	4.3.4	R4
hamsi	sphlib-small	8689056	46478	5326	gcc 1281 -Os	4.3.4	R4

2.2 Atmel ATmega1284P

8 bit microcontroller manufactured by Atmel. RISC design, one of the most powerful 8 bit designs currently in use. 128kiB Flash, 16kiB RAM, 16MHz. Should create almost the same results as the 1281, with two differences. SHAvite runs on the 1284P, not on the 1281 because of RAM usage and the 1284P runs included a new high speed assembly Grøstl implementation. See http://www.atmel.com/dyn/products/product_card.asp?part_id=4331 for details.

Table 4. Fastest Hashes on the ATmega1284P Platform

Hash	Implementation	Try Cycles (1536)	Empty Mes- sage	8 Bytes	64 Bytes	512 Bytes	1024 Bytes	2048 Bytes	Long Mes- sages	WRIR [%]	ROM Usage	RAM Usage	Compiler	Version	Source
shabal256	avrcryptolib-asm	374708	52326	6543	1027	312	261	235	210	0.0	1900	298	gcc 1284p -O3	4.3.4	R22
shabal512	avrcryptolib-asm	374930	52547	6570	1031	313	261	236	210	0.0	1898	330	gcc 1284p -Os	4.3.4	R22
groestl256	avr-highspeed-atmega-asm-xbx	745945	53494	6685	1286	555	503	477	451	0.0	5506	2593	gcc 1284p -Os	4.3.4	R22
bmw256	avrcryptolib-asm	809058	64288	8038	1489	610	548	516	485	0.0	3594	339	gcc 1284p -Os	4.3.4	R21
shavite3256	256-nosalt-smallc	1025991	50874	6375	1430	734	685	660	635	0.0	66566	7297	gcc 1284p -O1	4.3.4	R22
fugue256	sphlib	1358659	113354	15237	2611	1035	922	866	810	0.0	41788	5457	gcc 1284p -O3	4.3.4	R22
blake32	sphlib	1419678	60206	7521	1841	1004	944	914	884	0.0	39902	1764	gcc 1284p -O3	4.3.4	R22
skein256	avrcryptolib-asm	1929012	117925	14754	2451	1360	1282	1243	1204	0.0	2398	268	gcc 1284p -Os	4.3.4	R22
echo256	avrcryptolib	1964793	220677	27587	3455	1285	1280	1173	1065	0.0	3598	704	gcc 1284p -O2	4.3.4	R22
skein512	avrcryptolib-asm	2405533	279951	35007	4387	1811	1627	1535	1443	0.0	2500	429	gcc 1284p -O3	4.3.4	R22
fugue512	sphlib	2645595	263857	34797	5704	2069	1809	1679	1549	0.0	50946	5681	gcc 1284p -O1	4.3.4	R22
echo512	avrcryptolib	3511558	272838	34108	4270	2642	2375	2242	2108	0.0	3662	742	gcc 1284p -O2	4.3.4	R22
simd256	sphlib	3766057	152312	37856	4733	2650	2501	2427	2353	0.0	91276	5074	gcc 1284p -O3	4.3.4	R22
jh256	sphlib-small	4232018	179153	43534	5470	2991	2814	2726	2637	0.0	28464	2269	gcc 1284p -O3	4.3.4	R22
jh512	sphlib-small	4232274	179409	43566	5474	2992	2814	2726	2637	0.0	28464	2301	gcc 1284p -O3	4.3.4	R22
luffa256	sphlib	4358668	176479	22062	5501	3069	2896	2809	2722	0.0	20972	770	gcc 1284p -O3	4.3.4	R22
bmw512	avrcryptolib	5326436	766276	95793	11981	4466	3717	3343	2969	0.0	17896	1140	gcc 1284p -O2	4.3.4	R21
jh224	bitslice_opt32	5333137	217374	53900	6727	3755	3543	3437	3331	0.0	24038	2295	gcc 1284p -O2	4.3.4	R22
jh384	bitslice_opt32	5333285	217524	53919	6729	3755	3543	3437	3331	0.0	24038	2315	gcc 1284p -O2	4.3.4	R22
groestl512	sphlib-small	5825349	654501	81807	10220	4647	4006	3686	3365	0.0	23494	6062	gcc 1284p -Os	4.3.4	R22
blake64	avrcryptolib	6601087	519085	64910	8120	4974	4467	4213	3960	0.0	7200	541	gcc 1284p -O2	4.3.4	R21
keccak256	sphlib-small	7042607	601940	75237	9399	4610	4591	4582	4572	0.0	74616	3317	gcc 1284p -O1	4.3.4	R22
luffa512	sphlib	7658080	454900	56865	11835	5581	5135	4911	4688	0.0	46948	1476	gcc 1284p -O3	4.3.4	R22
hamsi	sphlib-small	8312293	89950	16595	6758	5529	5441	5397	5353	0.0	50840	5065	gcc 1284p -O3	4.3.4	R22
cubehash1632	avrcryptolib-cwbasm	9197614	2795621	349461	47850	9628	6898	5533	4168	0.0	1648	249	gcc 1284p -O3	4.3.4	R21
keccak512	sphlib-small	12811434	598245	74775	9341	9124	8537	8243	7949	0.0	74616	3285	gcc 1284p -O1	4.3.4	R22

Table 5. Smallest Hashes Considering RAM Usage on the ATmega1284P Platform

Hash	Implementation	Try Cycles (1536)	ROM Usage	RAM Usage	Compiler	Version	Source
cubehash1632	avrcryptolib-cwbasn	9197615	1648	249	gcc 1284p -O3	4.3.4	R22
skein256	avrcryptolib-asm	1929031	2414	266	gcc 1284p -O1	4.3.4	R22
blake32	avrcryptolib	1797348	3586	269	gcc 1284p -O1	4.3.4	R22
shabal256	avrcryptolib-asm	374727	1916	296	gcc 1284p -O1	4.3.4	R22
shabal512	avrcryptolib-asm	374948	1914	328	gcc 1284p -O1	4.3.4	R22
bmw256	avrcryptolib-tinyasm	1369989	1926	333	gcc 1284p -O1	4.3.4	R22
groestl256	avrcryptolib	13312953	2214	346	gcc 1284p -Os	4.3.4	R22
skein512	avrcryptolib-asm	2405551	2516	427	gcc 1284p -O1	4.3.4	R22
blake64	avrcryptolib	6752178	6472	527	gcc 1284p -O1	4.3.4	R22
keccak512	avrcryptolib	26345169	3928	638	gcc 1284p -O1	4.3.4	R22
groestl512	avrcryptolib	19713929	2306	642	gcc 1284p -Os	4.3.4	R22
keccak256	avrcryptolib	14380254	3928	670	gcc 1284p -O1	4.3.4	R22
echo256	avrcryptolib	2002265	3650	697	gcc 1284p -Os	4.3.4	R22
echo512	avrcryptolib	3579361	3666	735	gcc 1284p -Os	4.3.4	R22
luffa256	sphlib	4451932	21840	762	gcc 1284p -O1	4.3.4	R22
bmw512	avrcryptolib	5352769	17594	1136	gcc 1284p -Os	4.3.4	R22
luffa512	sphlib	7713030	45252	1402	gcc 1284p -Os	4.3.4	R22
jh256	bitslice_ref32	5877084	36620	1631	gcc 1284p -O3	4.3.4	R22
jh512	bitslice_ref32	5877389	36620	1663	gcc 1284p -O3	4.3.4	R22
jh224	bitslice_ref32	5877060	41410	1883	gcc 1284p -O3	4.3.4	R22
jh384	bitslice_ref32	5877240	41410	1903	gcc 1284p -O3	4.3.4	R22
simd256	sphlib-small	4706307	35108	2970	gcc 1284p -O1	4.3.4	R22
fugue256	ref-xbx	5756562	10212	3024	gcc 1284p -Os	4.3.4	R22
fugue512	ref-xbx	11453439	10212	3055	gcc 1284p -Os	4.3.4	R22
hamsi	sphlib-small	8312293	50840	5065	gcc 1284p -O3	4.3.4	R22
shavite3256	256-nosalt-smallc	1052389	80452	6663	gcc 1284p -O3	4.3.4	R22

Table 6. Smallest Hashes Considering ROM Usage on the ATmega1284P Platform

Hash	Implementation	Try Cycles (1536)	ROM Usage	RAM Usage	Compiler	Version	Source
cubehash1632	avrcryptolib-cwbasm	9197843	1470	251	gcc 1284p -Os	4.3.4	R22
shabal512	avrcryptolib-asm	374930	1898	330	gcc 1284p -Os	4.3.4	R22
shabal256	avrcryptolib-asm	374709	1900	298	gcc 1284p -Os	4.3.4	R22
bmw256	avrcryptolib-tinyasm	1369972	1910	335	gcc 1284p -Os	4.3.4	R22
groestl256	avrcryptolib	13312953	2214	346	gcc 1284p -Os	4.3.4	R22
groestl512	avrcryptolib	19713929	2306	642	gcc 1284p -Os	4.3.4	R22
skein256	avrcryptolib-asm	1929012	2398	268	gcc 1284p -Os	4.3.4	R22
skein512	avrcryptolib-asm	2405534	2500	429	gcc 1284p -Os	4.3.4	R22
blake32	avrcryptolib	1797348	3586	269	gcc 1284p -O1	4.3.4	R22
echo256	avrcryptolib	1964793	3598	704	gcc 1284p -O2	4.3.4	R22
echo512	avrcryptolib	3511558	3662	742	gcc 1284p -O2	4.3.4	R22
keccak512	avrcryptolib	26345169	3928	638	gcc 1284p -O1	4.3.4	R22
keccak256	avrcryptolib	14380254	3928	670	gcc 1284p -O1	4.3.4	R22
blake64	avrcryptolib	6752178	6472	527	gcc 1284p -O1	4.3.4	R22
jh256	bitslice_ref32	6041081	8760	1996	gcc 1284p -Os	4.3.4	R22
jh512	bitslice_ref32	6041335	8760	2028	gcc 1284p -Os	4.3.4	R22
fugue256	ref-xbx	5756562	10212	3024	gcc 1284p -Os	4.3.4	R22
fugue512	ref-xbx	11453439	10212	3055	gcc 1284p -Os	4.3.4	R22
jh224	bitslice_ref32	6041066	13528	2248	gcc 1284p -Os	4.3.4	R22
jh384	bitslice_ref32	6041215	13528	2268	gcc 1284p -Os	4.3.4	R22
bmw512	avrcryptolib	5791598	16812	1159	gcc 1284p -O1	4.3.4	R22
luffa256	sphlib	4390258	20380	768	gcc 1284p -Os	4.3.4	R22
simd256	sphlib-small	4587465	31930	3036	gcc 1284p -Os	4.3.4	R22
luffa512	sphlib	7713030	45252	1402	gcc 1284p -Os	4.3.4	R22
hamsi	sphlib-small	8689077	46478	5515	gcc 1284p -Os	4.3.4	R22
shavite3256	256-nosalt-smallc	1052890	65922	7001	gcc 1284p -Os	4.3.4	R22

2.3 Texas Instruments AR7

The AR7 is a MIPS based system-on-chip (SoC) manufactured by Texas Instruments. It is not sold to end customers and specifications are not generally public. However, there is a linux kernel available. See <http://www.linux-mips.org/wiki/AR7>. The XBX team got their hands on an AR7 by modifying a DSL router http://www.avm.de/de/Produkte/FRITZBox/FRITZ_Box_Fon_WLAN/index.php.

Table 7. Fastest Hashes on the AR7 Platform

Hash	Implementation	Try Cycles (1536)	Empty Mes- sage	8 Bytes	64 Bytes	512 Bytes	1024 Bytes	2048 Bytes	Long Mes- sages	WRIR [%]	ROM Usage	RAM Usage	Compiler	Version	Source
bmw256	armcryptolib-speed	106795	50350	6324	816	135	87	62	37	2.1	10211	812	mips-gcc -Os	4.2.1	R11
shabal256	sphlib	123437	64037	8044	1316	188	108	68	27	1.9	20820	1160	mips-gcc -Os	4.2.1	R12
blake32	sphlib-small	123649	35099	4394	605	126	92	75	58	1.5	5938	568	mips-gcc -O1	4.2.1	R11
shabal512	sphlib	123941	64938	8131	1316	189	108	68	27	0.8	20566	1168	mips-gcc -O3	4.2.1	R12
bmw512	sphlib-small	185778	79540	10067	1265	226	148	108	68	0.9	14457	1920	mips-gcc -O1	4.2.1	R12
luffa256	opt32	202579	34159	4295	702	182	144	126	107	1.3	21404	988	mips-gcc -O3	4.2.1	R9
fugue256	sphlib	237877	69430	10090	1447	268	183	141	99	1.3	17134	888	mips-gcc -Os	4.2.1	R12
blake64	armcryptolib	238540	39538	4995	630	208	169	149	129	1.4	3618	572	mips-gcc -Os	4.2.1	R10
skein512	sphlib-small	240766	51834	6519	818	237	177	147	116	1.4	11719	1132	mips-gcc -O1	4.2.1	R10
skein256	sphlib-small	247695	74690	9376	1687	295	195	145	95	1.6	27847	1564	mips-gcc -O1	4.2.1	R10
luffa384	opt32	280171	46905	5888	948	249	199	174	149	1.1	23023	1080	mips-gcc -O3	4.2.1	R11
keccak256	opt32n2	309441	75671	9784	1259	289	225	190	156	1.0	30701	263480	mips-gcc -O1	4.2.1	R11
echo256	sphlib-small	343533	73683	9214	1159	277	238	201	164	1.6	8146	864	mips-gcc -Os	4.2.1	R12
cubehash1632	sphlib-small	357512	90948	11387	1858	374	268	215	162	0.8	11939	752	mips-gcc -O1	4.2.1	R9
luffa512	opt32	365687	57094	7201	1148	318	258	228	199	0.8	21404	1132	mips-gcc -O3	4.2.1	R9
fugue512	sphlib	396308	103072	14289	1946	405	295	240	185	0.8	19958	880	mips-gcc -Os	4.2.1	R9
hamsi	sphlib-small	399196	52391	7009	1172	347	283	250	218	1.4	15267	512	mips-gcc -O2	4.2.1	R12
simd256	sphlib-small	402469	65654	10164	1270	350	284	251	218	1.3	12386	1260	mips-gcc -O2	4.2.1	R11
jh224	bitslice_opt32	440828	48164	8233	1018	351	303	279	256	1.9	9773	436	mips-gcc -O2	4.2.1	R9
jh384	bitslice_opt32	441040	48681	8306	1025	352	304	279	255	0.7	9773	456	mips-gcc -O2	4.2.1	R9
jh256	bitslice_opt32	441079	48243	8187	1021	352	303	279	256	1.4	8170	436	mips-gcc -O2	4.2.1	R10
jh512	bitslice_opt32	441570	48893	8305	1030	353	304	280	256	0.7	8170	468	mips-gcc -O2	4.2.1	R10
simd512	opt	488832	88735	15354	1916	436	348	304	260	0.8	18611	1904	mips-gcc -O2	4.2.1	R11
echo512	sphlib-small	580549	82243	10310	1287	486	405	365	324	1.5	8162	936	mips-gcc -Os	4.2.1	R10
groestl256	opt32-xbx-rolled	609924	174728	21995	3084	631	456	369	281	1.3	46647	496	mips-gcc -O1	4.2.1	R10
echosp256	generic_opt32	688165	156191	19855	2493	672	504	461	419	0.7	30128	944	mips-gcc -O1	4.2.1	R9
keccak512	sphlib-small	778782	100925	12644	1579	642	541	491	441	0.9	18184	2120	mips-gcc -O1	4.2.1	R10
groestl512	sphlib-small	936682	157225	19640	2463	815	661	585	508	0.8	13254	1664	mips-gcc -O1	4.2.1	R12
echosp512	generic_opt32	985588	172966	21854	2735	757	672	581	489	1.5	30128	976	mips-gcc -O1	4.2.1	R11
shavite3256	256-nosalt-smallc	1640999	118694	14862	2874	1225	1108	1058	1008	0.8	31671	1728	mips-gcc -O1	4.2.1	R11

Table 8. Smallest Hashes Considering RAM Usage on the AR7 Platform

Hash	Implementation	Try Cycles (1536)	ROM Usage	RAM Usage	Compiler	Version	Source
fugue256	ANSI_opt32	267624	65172	220	mips-gcc -O1	4.2.1	R12
hamsi	sphlib	468560	43189	296	mips-gcc -O1	4.2.1	R12
cubehash1632	simple	1838570	2375	335	mips-gcc -O2	4.2.1	R11
blake32	armcryptolib	186017	3106	340	mips-gcc -Os	4.2.1	R12
jh224	bitslice_opt32	9180157	28102	388	mips-gcc -O0	4.2.1	R12
jh256	bitslice_opt32	9182820	28102	392	mips-gcc -O0	4.2.1	R12
shabal256	armcryptolib	207601	2884	404	mips-gcc -O1	4.2.1	R12
jh384	bitslice_opt32	9184370	28102	408	mips-gcc -O0	4.2.1	R12
luffa256	sphlib	215975	8369	420	mips-gcc -O1	4.2.1	R12
jh512	bitslice_opt32	9181137	28102	424	mips-gcc -O0	4.2.1	R12
shabal512	armcryptolib	207813	2884	436	mips-gcc -O1	4.2.1	R12
groestl256	opt32-xbx-rolled	3338669	109411	452	mips-gcc -O0	4.2.1	R12
skein256	armcryptolib	30054061	5063	520	mips-gcc -O1	4.2.1	R12
blake64	armcryptolib	635258	5930	548	mips-gcc -O0	4.2.1	R12
bmw256	armcryptolib-speed	1567979	22975	616	mips-gcc -O0	4.2.1	R12
skein512	armcryptolib	34635871	5047	704	mips-gcc -O1	4.2.1	R12
keccak512	armcryptolib	1505266	3755	772	mips-gcc -O2	4.2.1	R12
fugue512	sphlib	417720	21566	784	mips-gcc -O1	4.2.1	R12
echosp256	powerpc_pp32cv1	1063578	25033	788	mips-gcc -O3	4.2.1	R12
groestl512	armcryptolib	42710275	4790	788	mips-gcc -O0	4.2.1	R12
luffa512	sphlib	917549	32945	796	mips-gcc -O0	4.2.1	R12
keccak256	armcryptolib	841680	3755	804	mips-gcc -O2	4.2.1	R12
echosp512	powerpc_pp32cv1	1650513	25033	820	mips-gcc -O3	4.2.1	R12
echo256	sphlib-small	368496	8218	840	mips-gcc -O1	4.2.1	R12
echo512	armcryptolib	18488586	6622	872	mips-gcc -O0	4.2.1	R12
luffa384	opt32	357591	18876	919	mips-gcc -O1	4.2.1	R9
simd256	sphlib	1819371	27532	1028	mips-gcc -O1	4.2.1	R12
bmw512	armcryptolib	1016606	13017	1248	mips-gcc -O0	4.2.1	R12
shavite3256	256-nosalt-smallc	2933563	50757	1424	mips-gcc -O0	4.2.1	R12
simd512	opt	618152	21011	1856	mips-gcc -O1	4.2.1	R12

Table 9. Smallest Hashes Considering ROM Usage on the AR7 Platform

Hash	Implementation	Try Cycles (1536)	ROM Usage	RAM Usage	Compiler	Version	Source
cubehash1632	simple	2146725	2167	336	mips-gcc -O1	4.2.1	R12
shabal256	armcryptolib	176861	2500	428	mips-gcc -Os	4.2.1	R12
shabal512	armcryptolib	177007	2500	460	mips-gcc -Os	4.2.1	R12
groestl256	armcryptolib	12228385	2963	588	mips-gcc -O3	4.2.1	R12
blake32	armcryptolib	186017	3106	340	mips-gcc -Os	4.2.1	R12
groestl512	armcryptolib	18283556	3310	836	mips-gcc -Os	4.2.1	R12
blake64	armcryptolib	238606	3618	572	mips-gcc -Os	4.2.1	R12
keccak512	armcryptolib	1505266	3755	772	mips-gcc -O2	4.2.1	R12
keccak256	armcryptolib	841680	3755	804	mips-gcc -O2	4.2.1	R12
echo512	armcryptolib	6390661	4550	1016	mips-gcc -Os	4.2.1	R12
echo256	armcryptolib	3571922	4598	992	mips-gcc -Os	4.2.1	R12
skein256	armcryptolib	30146055	4615	544	mips-gcc -Os	4.2.1	R12
skein512	armcryptolib	34724182	4679	736	mips-gcc -Os	4.2.1	R12
jh256	bitslice_ref32	540189	6058	444	mips-gcc -Os	4.2.1	R12
jh512	bitslice_ref32	540680	6058	476	mips-gcc -Os	4.2.1	R12
bmw256	armcryptolib	146956	6538	628	mips-gcc -O1	4.2.1	R12
simd512	ref	15280907	6846	3576	mips-gcc -Os	4.2.1	R12
bmw512	armcryptolib	258508	7117	1308	mips-gcc -Os	4.2.1	R12
jh224	bitslice_ref32	540825	7693	444	mips-gcc -Os	4.2.1	R12
jh384	bitslice_ref32	541090	7693	464	mips-gcc -Os	4.2.1	R12
luffa256	sphlib	207601	7862	492	mips-gcc -O3	4.2.1	R12
simd256	sphlib-small	403039	12386	1260	mips-gcc -O2	4.2.1	R12
luffa512	sphlib	376830	15205	952	mips-gcc -Os	4.2.1	R12
hamsi	sphlib-small	399196	15267	512	mips-gcc -O2	4.2.1	R12
fugue256	sphlib	237877	17134	888	mips-gcc -Os	4.2.1	R12
luffa384	opt32	331899	17500	952	mips-gcc -Os	4.2.1	R12
fugue512	sphlib	396957	19958	880	mips-gcc -Os	4.2.1	R12
echosp256	powerpc_pp32cv1	1104467	22792	796	mips-gcc -Os	4.2.1	R12
echosp512	powerpc_pp32cv1	1704228	22792	828	mips-gcc -Os	4.2.1	R12
shavite3256	256-nosalt-smallc	1671183	31480	3172	mips-gcc -Os	4.2.1	R12

2.4 Atmel AT91RM9200

ARM920T based single PCB computer. http://www.artila.com/p_sbc.html#m_501

Table 10. Fastest Hashes on the AT91RM9200 Platform

Hash	Implementation	Try Cycles (1536)	Empty Mes- sage	8 Bytes	64 Bytes	512 Bytes	1024 Bytes	2048 Bytes	Long Mes- sages	WRIR [%]	ROM Usage	RAM Usage	Compiler	Version	Source
bmw256	optc04	59220	20453	2569	347	66	45	35	25	0.9	11124	752	arm-gcc -O1	3.3.2	R13
shabal256	sphlib	69716	21668	2704	517	86	56	40	25	1.2	17288	668	arm-gcc -Os	3.3.2	R15
shabal512	sphlib	69896	21825	2728	523	87	56	41	25	2.9	17288	700	arm-gcc -Os	3.3.2	R16
blake32	sphlib	92475	19811	2502	358	86	67	57	47	1.5	22072	652	arm-gcc -O1	3.3.2	R15
luffa256	sphlib	151785	15615	1952	385	123	105	96	86	1.5	28268	364	arm-gcc -O1	3.3.2	R16
bmw512	sphlib-small	160718	35314	4433	555	151	116	99	81	1.7	15612	1388	arm-gcc -O1	3.3.2	R14
fugue256	sphlib	208024	37159	5345	805	194	150	128	107	1.6	47550	1352	arm-gcc -Os	3.3.2	R16
blake64	sphlib	216968	33683	4216	529	185	153	136	119	6.1	22072	1076	arm-gcc -O1	3.3.2	R16
skein256	sphlib-small	231525	29160	3656	798	207	165	144	122	0.4	38100	924	arm-gcc -Os	3.3.2	R15
skein512	sphlib-small	266783	29722	3713	468	208	182	169	156	0.8	35176	688	arm-gcc -O1	3.3.2	R14
keccak256	opt32u6	269978	47306	6091	801	225	189	169	150	1.1	55372	263140	arm-gcc -O1	3.3.2	R16
luffa384	opt32	276998	23895	3005	543	212	188	177	165	3.3	11216	828	arm-gcc -O2	3.3.2	R14
cubehash1632	sphlib-small	282791	57038	7145	1107	264	204	174	144	3.7	8388	808	arm-gcc -O1	3.3.2	R16
echosp256	generic_opt32	282893	68108	8743	1094	282	209	205	201	16.5	23350	4856	arm-gcc -O1	3.3.2	R16
luffa512	sphlib	288383	29228	3655	748	236	200	189	178	4.2	28268	544	arm-gcc -O1	3.3.2	R13
echo256	sphlib	331796	63686	7996	1008	258	227	211	196	8.2	33820	1028	arm-gcc -O1	3.3.2	R15
fugue512	sphlib	374400	63731	8920	1290	334	267	233	198	1.0	47550	1364	arm-gcc -Os	3.3.2	R14
simd256	sphlib-small	398194	31995	5970	745	302	270	254	238	1.1	32548	1440	arm-gcc -O2	3.3.2	R15
groestl256	sphlib-small	410423	53111	6719	1075	337	285	259	234	2.6	26100	1732	arm-gcc -O2	3.3.2	R16
hamsi	sphlib	413696	19485	2734	585	304	278	272	266	13.3	221992	212	arm-gcc -O1	3.3.2	R13
echosp512	generic_opt32	421268	76646	9762	1229	329	288	264	240	6.3	23350	4888	arm-gcc -O1	3.3.2	R16
jh224	bitslice_opt32	542565	31646	6920	828	394	364	354	345	21.5	6684	324	arm-gcc -O1	3.3.2	R15
jh256	bitslice_opt32	542599	31590	6631	822	394	364	354	345	1.8	6684	328	arm-gcc -O1	3.3.2	R16
jh512	bitslice_opt32	542621	31534	6636	823	394	364	348	333	3.5	6684	360	arm-gcc -O1	3.3.2	R15
jh384	bitslice_opt32	542678	31489	6629	824	395	376	354	333	3.4	6684	344	arm-gcc -O1	3.3.2	R16
echo512	sphlib	568361	72439	9096	1145	466	394	375	355	5.6	33820	1116	arm-gcc -O1	3.3.2	R16
simd512	opt	574751	64170	13303	1665	457	395	364	333	2.7	21174	1804	arm-gcc -O2	3.3.2	R16
keccak512	sphlib-small	922635	64811	8128	1011	686	640	606	571	4.8	17300	1700	arm-gcc -O1	3.3.2	R16
groestl512	opt32	1002713	137655	17315	2174	835	711	637	562	3.1	49348	804	arm-gcc -O1	3.3.2	R14

Table 11. Smallest Hashes Considering RAM Usage on the AT91RM9200 Platform

Hash	Implementation	Try Cycles (1536)	ROM Usage	RAM Usage	Compiler	Version	Source
hamsi	sphlib	414563	221992	212	arm-gcc -O1	3.3.2	R16
blake32	armcryptolib	680423	3992	272	arm-gcc -O0	3.3.2	R16
cubehash1632	armcryptolib	1527638	1612	272	arm-gcc -O3	3.3.2	R16
jh224	bitslice_opt32	555323	6684	324	arm-gcc -O1	3.3.2	R16
shabal256	armcryptolib	507814	5660	327	arm-gcc -O0	3.3.2	R14
jh256	bitslice_opt32	542599	6684	328	arm-gcc -O1	3.3.2	R16
jh384	bitslice_opt32	542678	6684	344	arm-gcc -O1	3.3.2	R16
groestl256	armcryptolib	36523991	3176	360	arm-gcc -O0	3.3.2	R16
jh512	bitslice_opt32	556020	6684	360	arm-gcc -O1	3.3.2	R16
shabal512	armcryptolib	139163	3232	360	arm-gcc -O1	3.3.2	R16
luffa256	sphlib	151785	28268	364	arm-gcc -O1	3.3.2	R16
skein256	armcryptolib	1250685	2880	380	arm-gcc -Os	3.3.2	R16
fugue256	ANSI_opt32	657686	139456	452	arm-gcc -O0	3.3.2	R16
blake64	armcryptolib	574459	5092	488	arm-gcc -O0	3.3.2	R16
skein512	sphlib	919204	83680	492	arm-gcc -O1	3.3.2	R16
bmw256	sphlib	1402886	81760	528	arm-gcc -O0	3.3.2	R16
luffa512	sphlib	288754	28268	544	arm-gcc -O1	3.3.2	R16
keccak512	armcryptolib	2785410	3332	648	arm-gcc -O3	3.3.2	R16
groestl512	armcryptolib	53976308	3428	656	arm-gcc -O0	3.3.2	R16
keccak256	armcryptolib	1537571	3332	680	arm-gcc -O3	3.3.2	R16
echo256	sphlib-small	351968	8544	728	arm-gcc -O1	3.3.2	R16
echo512	armcryptolib	19855946	5956	756	arm-gcc -O0	3.3.2	R16
echosp256	powerpc_pp32cv1	446456	20900	760	arm-gcc -O1	3.3.2	R16
fugue512	sphlib	446186	42342	764	arm-gcc -O1	3.3.2	R16
echosp512	powerpc_pp32cv1	667890	20900	792	arm-gcc -O1	3.3.2	R16
luffa384	opt32	298181	10852	808	arm-gcc -O1	3.3.2	R16
simd256	sphlib	2359969	114976	944	arm-gcc -O0	3.3.2	R16
bmw512	armcryptolib	657338	5108	1024	arm-gcc -O1	3.3.2	R16
simd512	opt	636593	21810	1764	arm-gcc -O1	3.3.2	R16

Table 12. Smallest Hashes Considering ROM Usage on the AT91RM9200 Platform

Hash	Implementation	Try Cycles (1536)	ROM Usage	RAM Usage	Compiler	Version	Source
cubehash1632	armcryptolib	1909350	1260	280	arm-gcc -Os	3.3.2	R16
groestl256	armcryptolib	12914708	1892	396	arm-gcc -Os	3.3.2	R16
groestl512	armcryptolib	19055194	1952	696	arm-gcc -Os	3.3.2	R16
blake32	armcryptolib	159525	2080	280	arm-gcc -Os	3.3.2	R16
keccak512	armcryptolib	3288870	2196	672	arm-gcc -Os	3.3.2	R16
keccak256	armcryptolib	1810013	2196	704	arm-gcc -Os	3.3.2	R16
skein256	armcryptolib	1250685	2880	380	arm-gcc -Os	3.3.2	R16
shabal256	armcryptolib	131760	3096	340	arm-gcc -Os	3.3.2	R16
shabal512	armcryptolib	131603	3096	372	arm-gcc -Os	3.3.2	R16
skein512	armcryptolib	1240391	3148	560	arm-gcc -Os	3.3.2	R16
echo256	armcryptolib	4142959	3168	756	arm-gcc -Os	3.3.2	R16
echo512	armcryptolib	7422413	3168	808	arm-gcc -Os	3.3.2	R16
bmw256	armcryptolib	117371	3784	584	arm-gcc -O1	3.3.2	R16
jh224	bitslice_ref32	592920	4284	384	arm-gcc -Os	3.3.2	R15
jh256	bitslice_ref32	592909	4284	388	arm-gcc -Os	3.3.2	R16
jh384	bitslice_ref32	593235	4284	404	arm-gcc -Os	3.3.2	R16
jh512	bitslice_ref32	592931	4284	420	arm-gcc -Os	3.3.2	R16
bmw512	armcryptolib	644861	5020	1044	arm-gcc -O2	3.3.2	R16
blake64	armcryptolib	574459	5092	488	arm-gcc -O0	3.3.2	R16
simd512	ref	14274000	9794	3496	arm-gcc -Os	3.3.2	R16
luffa256	opt32	172766	10852	696	arm-gcc -O1	3.3.2	R16
luffa384	opt32	298181	10852	808	arm-gcc -O1	3.3.2	R16
luffa512	opt32	440629	10852	856	arm-gcc -O1	3.3.2	R16
echosp256	powerpc_pp32cv1	446456	20900	760	arm-gcc -O1	3.3.2	R16
echosp512	powerpc_pp32cv1	667890	20900	792	arm-gcc -O1	3.3.2	R16
simd256	sphlib-small	402210	28416	1260	arm-gcc -O1	3.3.2	R16
fugue256	sphlib	235204	42342	664	arm-gcc -O1	3.3.2	R16
fugue512	sphlib	446186	42342	764	arm-gcc -O1	3.3.2	R16
hamsi	sphlib-small	415564	156408	216	arm-gcc -O1	3.3.2	R16

2.5 Intel XScale IXP420

ARMv5TE based Intel chip. We used an NAS server and changed the firmware to <http://www.nslu2-linux.org/wiki/Main/HomePage>.

Table 13. Fastest Hashes on the IXP429 Platform

Hash	Implementation	Try Cycles (1536)	Empty Mes- sage	8 Bytes	64 Bytes	512 Bytes	1024 Bytes	2048 Bytes	Long Mes- sages	WRIR [%]	ROM Usage	RAM Usage	Compiler	Version	Source
bmw256	sphlib	88395	23408	2957	411	88	65	54	42	2.2	7124	740	armeb-gcc -O1	4.1.2	R17
shabal512	ref	90523	33416	4181	707	115	73	52	30	1.2	18772	2432	armeb-gcc -O2	4.1.2	R20
shabal256	sphlib	94081	31438	3949	668	114	75	55	35	1.4	17840	1572	armeb-gcc -Os	4.1.2	R20
blake32	sphlib	99933	28229	3552	489	103	75	61	46	1.2	10896	1324	armeb-gcc -O3	4.1.2	R19
luffa256	sphlib	138121	19418	2409	438	121	98	86	75	2.3	6052	524	armeb-gcc -O2	4.1.2	R18
bmw512	sphlib	184272	66483	8319	1043	207	142	109	76	0.9	26048	3260	armeb-gcc -O1	4.1.2	R19
skein256	sphlib-small	218320	36691	4591	875	207	158	134	110	0.7	23912	1836	armeb-gcc -Os	4.1.2	R20
fugue256	sphlib	240048	41430	5919	872	220	173	148	124	2.8	14298	708	armeb-gcc -O1	4.1.2	R17
luffa512	sphlib	258369	34414	4302	807	225	182	161	140	1.5	13304	1256	armeb-gcc -O3	4.1.2	R18
shavite3256	256-nosalt- smallc	263357	65652	8207	1158	260	194	160	126	1.7	31688	5192	armeb-gcc -O2	4.1.2	R20
blake64	sphlib	266865	38071	4765	602	226	187	167	148	3.2	7104	1140	armeb-gcc -O1	4.1.2	R17
cubehash1632	sphlib-small	299034	65486	8198	1280	290	219	183	147	0.6	10028	540	armeb-gcc -O3	4.1.2	R20
skein512	sphlib	339233	80814	10025	1255	361	256	203	151	0.5	57836	5291	armeb-gcc -O1	4.1.2	R20
keccak256	opt32u2	359466	52252	6837	913	285	247	228	209	2.7	23148	263532	armeb-gcc -O1	4.1.2	R20
echosp256	generic_opt32	377437	95162	12265	1535	376	279	252	225	3.7	31302	2792	armeb-gcc -O3	4.1.2	R20
echo256	sphlib-small	382508	59351	7438	929	277	256	226	196	2.3	7056	716	armeb-gcc -O1	4.1.2	R20
simd256	sphlib	389939	55494	8718	1097	328	273	244	216	0.9	24332	1704	armeb-gcc -O1	4.1.2	R18
fugue512	sphlib	435292	70291	9566	1399	382	309	271	233	1.5	16766	732	armeb-gcc -O1	4.1.2	R19
hamsi	sphlib	475957	26667	3780	798	387	334	297	261	1.9	41576	1072	armeb-gcc -O1	4.1.2	R20
jh512	bitslice_opt32	487212	36475	6916	861	366	330	311	292	1.2	6576	524	armeb-gcc -Os	4.1.2	R20
jh256	bitslice_opt32	487229	36459	6929	866	366	330	311	292	1.6	6576	492	armeb-gcc -Os	4.1.2	R20
jh384	bitslice_opt32	487279	36741	7010	872	367	330	311	292	3.9	8132	512	armeb-gcc -Os	4.1.2	R17
jh224	bitslice_opt32	487412	36758	6939	869	367	330	311	292	5.3	8132	492	armeb-gcc -Os	4.1.2	R20
simd512	opt	521909	66118	12914	1611	428	362	329	296	2.8	15630	1744	armeb-gcc -O2	4.1.2	R17
echosp512	generic_opt32	559481	105419	13539	1696	438	383	329	274	4.3	31302	2824	armeb-gcc -O3	4.1.2	R18
groestl256	opt32-xbx- rolled	564169	107098	13497	1978	509	403	350	296	0.5	43524	675	armeb-gcc -O1	4.1.2	R19
echo512	sphlib-small	662972	68794	8647	1084	523	455	420	385	2.4	7016	776	armeb-gcc -O1	4.1.2	R20
groestl512	opt32	879379	185851	23250	2917	817	635	542	449	5.5	58520	1768	armeb-gcc -O1	4.1.2	R18
keccak512	sphlib-small	971282	78537	9838	1226	736	676	629	581	2.8	17976	4040	armeb-gcc -Os	4.1.2	R17

Table 14. Smallest Hashes Considering RAM Usage on the IXP420 Platform

Hash	Implementation	Try Cycles (1536)	ROM Usage	RAM Usage	Compiler	Version	Source
cubehash1632	armcryptolib	1815467	1196	260	armeb-gcc -O1	4.1.2	R20
hamsi	sphlib-small	1443349	29128	288	armeb-gcc -O0	4.1.2	R20
jh224	bitslice_ref32	742306	5836	312	armeb-gcc -O1	4.1.2	R20
jh256	bitslice_ref32	759497	4284	312	armeb-gcc -O1	4.1.2	R20
jh384	bitslice_ref32	742273	5836	332	armeb-gcc -O1	4.1.2	R20
jh512	bitslice_ref32	741724	4284	344	armeb-gcc -O1	4.1.2	R20
groestl256	armcryptolib	38513292	2992	352	armeb-gcc -O0	4.1.2	R20
luffa256	sphlib	160082	5412	356	armeb-gcc -O1	4.1.2	R20
blake32	light	327463	6492	380	armeb-gcc -O0	4.1.2	R20
bmw256	sphlib	224338	16236	580	armeb-gcc -O0	4.1.2	R20
shabal512	ref	155693	23956	644	armeb-gcc -O0	4.1.2	R20
groestl512	armcryptolib	57063351	3196	648	armeb-gcc -O0	4.1.2	R20
luffa512	sphlib	933993	23008	667	armeb-gcc -O0	4.1.2	R17
fugue256	sphlib	240930	14298	708	armeb-gcc -O1	4.1.2	R20
echo256	sphlib-small	382508	7056	716	armeb-gcc -O1	4.1.2	R20
shabal256	sphlib	163141	25216	716	armeb-gcc -O0	4.1.2	R20
fugue512	sphlib	435342	16766	732	armeb-gcc -O1	4.1.2	R20
echo512	sphlib-small	662972	7016	776	armeb-gcc -O1	4.1.2	R20
echosp256	powerpc_pp32cv1	513330	20932	812	armeb-gcc -O1	4.1.2	R20
echosp512	powerpc_pp32cv1	785249	20932	844	armeb-gcc -O1	4.1.2	R20
skein512	sphlib-small	617303	15692	972	armeb-gcc -O0	4.1.2	R20
blake64	sphlib-small	267047	7104	1140	armeb-gcc -O1	4.1.2	R20
simd256	sphlib-small	470471	10844	1192	armeb-gcc -O1	4.1.2	R20
shavite3256	256-nosalt-smallc	500296	32832	1412	armeb-gcc -O0	4.1.2	R20
skein256	sphlib-small	518501	41128	1508	armeb-gcc -O0	4.1.2	R20
simd512	opt	620994	15334	1720	armeb-gcc -O1	4.1.2	R20
bmw512	sphlib-small	363955	17980	1748	armeb-gcc -O0	4.1.2	R20
keccak256	opt64u6	793461	97712	2816	armeb-gcc -O0	4.1.2	R20
keccak512	sphlib-small	1110085	16528	2956	armeb-gcc -O1	4.1.2	R20

Table 15. Smallest Hashes Considering ROM Usage on the IXP420 Platform

Hash	Implementation	Try Cycles (1536)	ROM Usage	RAM Usage	Compiler	Version	Source
cubehash1632	simple	1958492	1312	264	armeb-gcc -Os	4.1.2	R20
groestl256	armcryptolib	16364752	1628	380	armeb-gcc -Os	4.1.2	R20
groestl512	armcryptolib	24103108	1696	680	armeb-gcc -Os	4.1.2	R20
blake32	light	177322	3736	412	armeb-gcc -O1	4.1.2	R20
jh256	bitslice_ref32	616189	4200	360	armeb-gcc -Os	4.1.2	R20
jh512	bitslice_ref32	616305	4200	392	armeb-gcc -Os	4.1.2	R20
simd512	ref	73052777	4570	3472	armeb-gcc -Os	4.1.2	R20
luffa256	sphlib	160082	5412	356	armeb-gcc -O1	4.1.2	R20
jh224	bitslice_ref32	616372	5756	360	armeb-gcc -Os	4.1.2	R20
jh384	bitslice_ref32	616389	5756	380	armeb-gcc -Os	4.1.2	R20
echo512	sphlib-small	662972	7016	776	armeb-gcc -O1	4.1.2	R20
bmw256	sphlib-small	89908	7032	1080	armeb-gcc -O1	4.1.2	R20
echo256	sphlib-small	382508	7056	716	armeb-gcc -O1	4.1.2	R20
blake64	sphlib-small	267047	7104	1140	armeb-gcc -O1	4.1.2	R20
simd256	sphlib-small	470471	10844	1192	armeb-gcc -O1	4.1.2	R20
skein512	sphlib-small	353980	10868	1256	armeb-gcc -Os	4.1.2	R20
luffa512	sphlib	289707	11852	964	armeb-gcc -O1	4.1.2	R20
bmw512	sphlib-small	215510	14076	2328	armeb-gcc -O1	4.1.2	R20
hamsi	sphlib-small	576671	14284	1476	armeb-gcc -O1	4.1.2	R20
fugue256	sphlib	240930	14298	708	armeb-gcc -O1	4.1.2	R20
keccak512	sphlib-small	1110085	16528	2956	armeb-gcc -O1	4.1.2	R20
keccak256	sphlib-small	632515	16528	2988	armeb-gcc -O1	4.1.2	R20
fugue512	sphlib	435342	16766	732	armeb-gcc -O1	4.1.2	R20
shabal512	ref	104505	17488	2140	armeb-gcc -Os	4.1.2	R20
shabal256	sphlib	94081	17840	1572	armeb-gcc -Os	4.1.2	R20
echosp256	powerpc_pp32cv1	513330	20932	812	armeb-gcc -O1	4.1.2	R20
echosp512	powerpc_pp32cv1	785249	20932	844	armeb-gcc -O1	4.1.2	R20
skein256	sphlib-small	218320	23912	1836	armeb-gcc -Os	4.1.2	R20
shavite3256	256-nosalt-smallc	278535	24880	1700	armeb-gcc -O1	4.1.2	R20

2.6 Luminary Micro (TI) lm3s811

ARM Cortex-M3 based microcontroller. The most modern CPU design in our current collection. See http://www.luminarymicro.com/products/lm3s811_microcontroller.html for details.

Table 16. Fastest Hashes on the lm3s811 Platform

Hash	Implementation	Try Cycles (1536)	Empty Mes- sage	8 Bytes	64 Bytes	512 Bytes	1024 Bytes	2048 Bytes	Long Mes- sages	WRIR [%]	ROM Usage	RAM Usage	Compiler	Version	Source
bmw256	armcryptolib-speed	43355	4012	509	88	33	30	28	26	0.0	10296	856	arm-elf-gcc -O3	4.3.2	R8
shabal256	sphlib	50150	7184	899	143	42	35	31	28	0.0	12380	884	arm-elf-gcc -O2	4.3.2	R8
shabal512	sphlib	50208	7242	906	144	42	35	31	28	0.0	12380	916	arm-elf-gcc -O2	4.3.2	R8
blake32	sphlib	66451	3654	459	99	48	44	43	41	0.0	7000	856	arm-elf-gcc -O2	4.3.2	R8
bmw512	sphlib	112922	17233	2161	265	96	79	71	62	0.0	15948	920	arm-elf-gcc -O3	4.3.2	R8
luffa256	opt32	125913	5606	714	166	89	84	81	78	0.0	11916	736	arm-elf-gcc -O3	4.3.2	R8
luffa384	opt32	206717	12875	1622	328	151	139	132	126	0.0	13628	852	arm-elf-gcc -O3	4.3.2	R8
fugue256	sphlib	215094	33729	4360	648	184	151	135	118	0.0	12632	884	arm-elf-gcc -O2	4.3.2	R8
skein256	sphlib-small	223898	10121	1274	232	153	148	145	142	0.0	21424	888	arm-elf-gcc -O3	4.3.2	R8
luffa512	sphlib	238706	15029	1880	383	175	160	153	146	0.0	8472	900	arm-elf-gcc -O3	4.3.2	R8
blake64	sphlib	240012	20095	2516	308	183	163	153	143	0.0	8536	920	arm-elf-gcc -O3	4.3.2	R8
echosp256	generic_opt32	252647	37998	4809	596	215	177	176	175	0.0	20520	904	arm-elf-gcc -O3	4.3.2	R8
cubehash1632	sphlib	254988	46087	5762	858	226	181	158	136	0.0	19664	916	arm-elf-gcc -O2	4.3.2	R8
skein512	sphlib	286187	23632	2963	370	202	190	184	178	0.0	47536	916	arm-elf-gcc -O2	4.3.2	R8
simd256	sphlib	295861	12224	3013	375	209	197	191	185	0.0	17496	884	arm-elf-gcc -Os	4.3.2	R8
echo256	sphlib	322760	37096	4637	575	211	210	192	174	0.0	16336	836	arm-elf-gcc -O3	4.3.2	R8
hamsi	sphlib	329483	3874	696	272	220	216	214	212	0.0	39476	800	arm-elf-gcc -O2	4.3.2	R8
keccak256	opt64u6	372799	32074	4032	504	245	243	242	241	0.0	48200	884	arm-elf-gcc -Os	4.3.2	R5
echosp512	generic_opt32	398955	46492	5870	729	263	261	238	214	0.0	20520	936	arm-elf-gcc -O3	4.3.2	R8
groestl256	sphlib	445559	27059	3384	696	325	299	286	272	0.0	28056	888	arm-elf-gcc -O3	4.3.2	R8
jh256	bitslice_opt32	447124	18334	4533	566	315	297	288	279	0.0	5164	416	arm-elf-gcc -O2	4.3.2	R8
jh224	bitslice_opt32	447138	18349	4535	566	315	297	288	279	0.0	6876	412	arm-elf-gcc -O2	4.3.2	R5
jh384	bitslice_opt32	447157	18367	4537	566	315	297	288	279	0.0	6876	432	arm-elf-gcc -O2	4.3.2	R8
jh512	bitslice_opt32	447177	18387	4539	566	315	297	288	279	0.0	5164	448	arm-elf-gcc -O2	4.3.2	R8
fugue512	sphlib	479769	125630	15963	2197	476	353	292	230	0.0	14728	916	arm-elf-gcc -O2	4.3.2	R8
echo512	sphlib	541323	42782	5348	664	408	366	345	325	0.0	15180	888	arm-elf-gcc -O2	4.3.2	R5
groestl512	sphlib	686979	78254	9784	1217	550	473	434	396	0.0	37264	916	arm-elf-gcc -O2	4.3.2	R8
keccak512	sphlib-small	733676	36176	4529	564	525	490	472	454	0.0	10564	916	arm-elf-gcc -Os	4.3.2	R8

Table 17. Smallest Hashes Considering RAM Usage on lm3s811 Platform

Hash	Implementation	Try Cycles (1536)	ROM Usage	RAM Usage	Compiler	Version	Source
cubehash1632	armcryptolib	1245857	672	272	arm-elf-gcc -Os	4.3.2	R8
blake32	armcryptolib	184763	1452	280	arm-elf-gcc -O2	4.3.2	R8
shabal256	armcryptolib	146185	952	336	arm-elf-gcc -Os	4.3.2	R8
jh224	bitslice_ref32	669651	5016	348	arm-elf-gcc -Os	4.3.2	R8
jh256	bitslice_ref32	669629	3324	352	arm-elf-gcc -Os	4.3.2	R8
skein256	armcryptolib	998777	1372	356	arm-elf-gcc -Os	4.3.2	R8
jh384	bitslice_ref32	669664	5016	368	arm-elf-gcc -Os	4.3.2	R8
shabal512	armcryptolib	146221	952	368	arm-elf-gcc -Os	4.3.2	R8
jh512	bitslice_ref32	669683	3324	384	arm-elf-gcc -Os	4.3.2	R8
luffa256	sphlib	157161	4228	392	arm-elf-gcc -Os	4.3.2	R8
groestl256	armcryptolib	13532139	1212	416	arm-elf-gcc -Os	4.3.2	R8
blake64	armcryptolib	304626	1820	516	arm-elf-gcc -Os	4.3.2	R8
bmw256	armcryptolib-speed	47798	4216	524	arm-elf-gcc -Os	4.3.2	R8
skein512	armcryptolib	1020578	1472	540	arm-elf-gcc -Os	4.3.2	R8
echo256	generic_opt64	460881	40000	564	arm-elf-gcc -O3	4.3.2	R8
echosp256	generic_opt64	371223	39880	564	arm-elf-gcc -O3	4.3.2	R8
echo512	generic_opt64	822161	40000	596	arm-elf-gcc -O3	4.3.2	R8
echosp512	generic_opt64	591926	39880	596	arm-elf-gcc -O3	4.3.2	R8
hamsi	sphlib	329495	39680	640	arm-elf-gcc -O3	4.3.2	R8
keccak512	armcryptolib	2243751	1520	644	arm-elf-gcc -Os	4.3.2	R8
keccak256	armcryptolib	1233459	1520	676	arm-elf-gcc -Os	4.3.2	R8
groestl512	armcryptolib	20089518	1252	708	arm-elf-gcc -Os	4.3.2	R8
fugue256	sphlib	241837	12940	780	arm-elf-gcc -Os	4.3.2	R8
luffa384	opt32	253048	9820	828	arm-elf-gcc -O2	4.3.2	R8
luffa512	sphlib	282564	8444	856	arm-elf-gcc -Os	4.3.2	R8
simd256	sphlib-small	382183	8276	884	arm-elf-gcc -Os	4.3.2	R8
bmw512	sphlib-small	166867	8488	914	arm-elf-gcc -O2	4.3.2	R7
fugue512	sphlib	544565	15228	916	arm-elf-gcc -Os	4.3.2	R8

Table 18. Smallest Hashes Considering ROM Usage on lm3s811 Platform

Hash	Implementation	Try Cycles (1536)	ROM Usage	RAM Usage	Compiler	Version	Source
cubehash1632	armcryptolib	1245857	672	272	arm-elf-gcc -Os	4.3.2	R8
shabal256	armcryptolib	146185	952	336	arm-elf-gcc -Os	4.3.2	R8
shabal512	armcryptolib	146221	952	368	arm-elf-gcc -Os	4.3.2	R8
groestl256	armcryptolib	13532139	1212	416	arm-elf-gcc -Os	4.3.2	R8
groestl512	armcryptolib	20089518	1252	708	arm-elf-gcc -Os	4.3.2	R8
blake32	armcryptolib	231876	1372	284	arm-elf-gcc -Os	4.3.2	R8
skein256	armcryptolib	998777	1372	356	arm-elf-gcc -Os	4.3.2	R8
skein512	armcryptolib	1020578	1472	540	arm-elf-gcc -Os	4.3.2	R8
keccak512	armcryptolib	2243751	1520	644	arm-elf-gcc -Os	4.3.2	R8
keccak256	armcryptolib	1233459	1520	676	arm-elf-gcc -Os	4.3.2	R8
echo512	armcryptolib	6710545	1644	800	arm-elf-gcc -Os	4.3.2	R8
echo256	armcryptolib	3733830	1724	768	arm-elf-gcc -Os	4.3.2	R8
blake64	armcryptolib	304626	1820	516	arm-elf-gcc -Os	4.3.2	R8
bmw256	armcryptolib	98701	2480	564	arm-elf-gcc -O2	4.3.2	R8
jh256	bitslice_ref32	669629	3324	352	arm-elf-gcc -Os	4.3.2	R8
jh512	bitslice_ref32	669683	3324	384	arm-elf-gcc -Os	4.3.2	R8
bmw512	armcryptolib	260318	3340	916	arm-elf-gcc -Os	4.3.2	R8
luffa256	sphlib	157161	4228	392	arm-elf-gcc -Os	4.3.2	R8
jh224	bitslice_ref32	669651	5016	348	arm-elf-gcc -Os	4.3.2	R8
jh384	bitslice_ref32	669664	5016	368	arm-elf-gcc -Os	4.3.2	R8
luffa512	opt32	426422	7836	884	arm-elf-gcc -Os	4.3.2	R8
simd256	sphlib-small	382183	8276	884	arm-elf-gcc -Os	4.3.2	R8
luffa384	opt32	293940	9528	832	arm-elf-gcc -Os	4.3.2	R8
hamsi	sphlib-small	409208	11556	864	arm-elf-gcc -O2	4.3.2	R8
fugue256	sphlib	215094	12632	884	arm-elf-gcc -O2	4.3.2	R8
fugue512	sphlib	479769	14728	916	arm-elf-gcc -O2	4.3.2	R8
echosp256	powerpc_pp32cv1	521731	18988	872	arm-elf-gcc -Os	4.3.2	R8
echosp512	powerpc_pp32cv1	815826	18988	904	arm-elf-gcc -Os	4.3.2	R8

3 Appendix

3.1 Compiler Options Used

Table 19 gives the exact compiler options used.

3.2 RIndex

Table 20 lists the measurement batches we ran to obtain the benchmarking data.

References

1. Christian Wenzel-Benner and Jens Gräf: "XBX: eXternal Benchmarking eXtension for the SUPERCOP Crypto Benchmarking Framework" in: S. Mangard and F.-X. Standaert (Eds.): CHES 2010, LNCS 6225, pp. 294305, 2010.

Table 19. Compiler Options Used

Shortname	Compiler
gcc 1281 -O0	gcc -pipe -funsigned-char -funsigned-bitfields -fpack-struct -fshort-enums -mmcu=atmega1281 -DF CPU=16000000L -ffunction-sections -fdata-sections -Wl,-gc-sections -O0
gcc 1281 -O1	gcc -pipe -funsigned-char -funsigned-bitfields -fpack-struct -fshort-enums -mmcu=atmega1281 -DF CPU=16000000L -ffunction-sections -fdata-sections -Wl,-gc-sections -O1
gcc 1281 -O2	gcc -pipe -funsigned-char -funsigned-bitfields -fpack-struct -fshort-enums -mmcu=atmega1281 -DF CPU=16000000L -ffunction-sections -fdata-sections -Wl,-gc-sections -O2
gcc 1281 -O3	gcc -pipe -funsigned-char -funsigned-bitfields -fpack-struct -fshort-enums -mmcu=atmega1281 -DF CPU=16000000L -ffunction-sections -fdata-sections -Wl,-gc-sections -O3
gcc 1281 -Os	gcc -pipe -funsigned-char -funsigned-bitfields -fpack-struct -fshort-enums -mmcu=atmega1281 -DF CPU=16000000L -ffunction-sections -fdata-sections -Wl,-gc-sections -Os
arm-elf-gcc -O0	arm-elf-gcc -pipe -Wl,-entry,ResetISR -Wl,-T,XBD AF ld -mthumb -mcpu=cortex-m3 -MD -std=c99 -DPART LM3S811 -Dgcc -fomit-frame-pointer -ffunction-sections -fdata-sections -Wl,-gc-sections -lc -lgcc -O0
arm-elf-gcc -O1	arm-elf-gcc -pipe -Wl,-entry,ResetISR -Wl,-T,XBD AF ld -mthumb -mcpu=cortex-m3 -MD -std=c99 -DPART LM3S811 -Dgcc -fomit-frame-pointer -ffunction-sections -fdata-sections -Wl,-gc-sections -lc -lgcc -O1
arm-elf-gcc -O2	arm-elf-gcc -pipe -Wl,-entry,ResetISR -Wl,-T,XBD AF ld -mthumb -mcpu=cortex-m3 -MD -std=c99 -DPART LM3S811 -Dgcc -fomit-frame-pointer -ffunction-sections -fdata-sections -Wl,-gc-sections -lc -lgcc -O2
arm-elf-gcc -O3	arm-elf-gcc -pipe -Wl,-entry,ResetISR -Wl,-T,XBD AF ld -mthumb -mcpu=cortex-m3 -MD -std=c99 -DPART LM3S811 -Dgcc -fomit-frame-pointer -ffunction-sections -fdata-sections -Wl,-gc-sections -lc -lgcc -O3
arm-elf-gcc -Os	arm-elf-gcc -pipe -Wl,-entry,ResetISR -Wl,-T,XBD AF ld -mthumb -mcpu=cortex-m3 -MD -std=c99 -DPART LM3S811 -Dgcc -fomit-frame-pointer -ffunction-sections -fdata-sections -Wl,-gc-sections -lc -lgcc -Os
mips-gcc -O0	mipsel-linux-uclibc-gcc -pipe -O0 -fomit-frame-pointer -ffunction-sections -fdata-sections -Wl,-gc-sections
mips-gcc -O1	mipsel-linux-uclibc-gcc -pipe -O1 -fomit-frame-pointer -ffunction-sections -fdata-sections -Wl,-gc-sections
mips-gcc -O2	mipsel-linux-uclibc-gcc -pipe -O2 -fomit-frame-pointer -ffunction-sections -fdata-sections -Wl,-gc-sections
mips-gcc -O3	mipsel-linux-uclibc-gcc -pipe -O3 -fomit-frame-pointer -ffunction-sections -fdata-sections -Wl,-gc-sections
mips-gcc -Os	mipsel-linux-uclibc-gcc -pipe -Os -fomit-frame-pointer -ffunction-sections -fdata-sections -Wl,-gc-sections
arm-gcc -O0	arm-linux-gcc -O0 -fomit-frame-pointer -ffunction-sections -fdata-sections -Wl,-gc-sections
arm-gcc -O1	arm-linux-gcc -O1 -fomit-frame-pointer -ffunction-sections -fdata-sections -Wl,-gc-sections
arm-gcc -O2	arm-linux-gcc -O2 -fomit-frame-pointer -ffunction-sections -fdata-sections -Wl,-gc-sections
arm-gcc -O3	arm-linux-gcc -O3 -fomit-frame-pointer -ffunction-sections -fdata-sections -Wl,-gc-sections
arm-gcc -Os	arm-linux-gcc -Os -fomit-frame-pointer -ffunction-sections -fdata-sections -Wl,-gc-sections
armeb-gcc -O0	armeb-linux-uclibc-gcc -pipe -O0 -fomit-frame-pointer -ffunction-sections -fdata-sections -Wl,-gc-sections
armeb-gcc -O1	armeb-linux-uclibc-gcc -pipe -O1 -fomit-frame-pointer -ffunction-sections -fdata-sections -Wl,-gc-sections
armeb-gcc -O2	armeb-linux-uclibc-gcc -pipe -O2 -fomit-frame-pointer -ffunction-sections -fdata-sections -Wl,-gc-sections
armeb-gcc -O3	armeb-linux-uclibc-gcc -pipe -O3 -fomit-frame-pointer -ffunction-sections -fdata-sections -Wl,-gc-sections
armeb-gcc -Os	armeb-linux-uclibc-gcc -pipe -Os -fomit-frame-pointer -ffunction-sections -fdata-sections -Wl,-gc-sections
gcc 1284p -O0	gcc -pipe -funsigned-char -funsigned-bitfields -fpack-struct -fshort-enums -mmcu=atmega1284p -DF CPU=16000000L -ffunction-sections -fdata-sections -Wl,-gc-sections -O0
gcc 1284p -O1	gcc -pipe -funsigned-char -funsigned-bitfields -fpack-struct -fshort-enums -mmcu=atmega1284p -DF CPU=16000000L -ffunction-sections -fdata-sections -Wl,-gc-sections -O1
gcc 1284p -O2	gcc -pipe -funsigned-char -funsigned-bitfields -fpack-struct -fshort-enums -mmcu=atmega1284p -DF CPU=16000000L -ffunction-sections -fdata-sections -Wl,-gc-sections -O2
gcc 1284p -O3	gcc -pipe -funsigned-char -funsigned-bitfields -fpack-struct -fshort-enums -mmcu=atmega1284p -DF CPU=16000000L -ffunction-sections -fdata-sections -Wl,-gc-sections -O3
gcc 1284p -Os	gcc -pipe -funsigned-char -funsigned-bitfields -fpack-struct -fshort-enums -mmcu=atmega1284p -DF CPU=16000000L -ffunction-sections -fdata-sections -Wl,-gc-sections -Os

Table 20. Measurement Batches

ID	Date	Platform	Hostname	Version	XBH Revision	Bootloader Revision
22	20100802	atmega1284p_16mhz	blix	2.0.0	01120	01039
21	20100801	atmega1284p_16mhz	blix	2.0.0	01120	01039
20	20100730	nslu2-openwrt	knulp	2.0.0	01112	01138
19	20100729	nslu2-openwrt	knulp	2.0.0	01112	01138
18	20100729	nslu2-openwrt	knulp	2.0.0	01112	01138
17	20100728	nslu2-openwrt	knulp	2.0.0	01112	01138
16	20100728	artila_m501	hal	2.0.0	01110	01092
15	20100727	artila_m501	hal	2.0.0	01110	01092
14	20100726	artila_m501	hal	2.0.0	01110	01092
13	20100725	artila_m501	hal	2.0.0	01110	01092
12	20100728	fritzbox-7170	blix	2.0.0	01112	01119
11	20100727	fritzbox-7170	blix	2.0.0	01112	01119
10	20100727	fritzbox-7170	blix	2.0.0	01112	01119
9	20100727	fritzbox-7170	blix	2.0.0	01112	01119
8	20100728	lm3s811-evb	blix	2.0.0	01085	01099
7	20100728	lm3s811-evb	blix	2.0.0	01085	01099
6	20100727	lm3s811-evb	blix	2.0.0	01085	01099
5	20100721	lm3s811-evb	blix	2.0.0	01085	01099
4	20100728	atmega1281_16mhz	blix	2.0.0	01076	01099
3	20100727	atmega1281_16mhz	blix	2.0.0	01076	01099
2	20100722	atmega1281_16mhz	blix	2.0.0	01076	01099
1	20100721	atmega1281_16mhz	blix	2.0.0	01076	01099