Keywords: MAXQ30, MAXQ1050, MAXQ1850, MAXQ1103, secure computing, modular arithmetic, modulo timing, cryptography, RSA, ECDSA, modular exponentiation, secure micro, MAA

APPLICATION NOTE 5145

Modular-Exponentiation Timing with MAXQ30 Microcontrollers

Oct 10, 2011

Abstract: The 32-bit DeepCover® Secure Microcontrollers (MAXQ1050, MAXQ1850, and MAXQ1103) provide hardware support for performing modular arithmetic. This is done using an engine called the modular arithmetic accelerator (MAA). This application note gives typical execution times for various modulus sizes, key types, and optimization levels.

Introduction

Modular exponentiation is used in several cryptographic algorithms, notably the RSA public key algorithm and the elliptic curve digital signature algorithm (ECDSA). It is also used to discover prime numbers and to find modular inverses. This application note describes what modular exponentiation is, provides an overview of the MAA, and lists typical times to execute various sized exponentiations.

The MAXQ30 architecture uses a reduced instruction set computer (RISC) where all instructions are 16 bits in length and execute in a single cycle. The 32-bit arithmetic and logic unit (ALU) works with 32-bit registers and values while connected to a 32-bit bus.

Modular Exponentiation

Modular exponentiation is described by the equation:

$$result = base^{exponent} \bmod modulus.$$

For example: $9 = 7^2 \bmod 10$.

In this example, 9 is the result, 7 is the base, 2 is the exponent, and 10 is the modulus. In this case, since the modulus 10 is 4 bits long in binary, the size is four.

The MAA performs modular addition, subtraction, multiplication, squaring, squaring followed by a multiplication, and modular exponentiation. All these operations can be done with a modulus size up to 2048 bits in length.

The MAA operates from the cryptographic clock. This clock may be sourced from the system clock, which is determined by the external crystal frequency or run from the cryptographic ring. The internal cryptographic ring for the DeepCover® Secure Microcontrollers (MAXQ1050 and MAXQ1850) runs from 55MHz to 75MHz with a typical speed of 65MHz. The internal cryptographic ring for the DeepCover Secure Microcontroller (MAXQ1103) can run at a speed from 45MHz to 65MHz with a typical speed of 55MHz.

The MAA on the MAXQ1050 and MAXQ1850 are identical, so the timings when running from the cryptographic ring are the same. The MAA on these two parts use a 32 × 16-bit multiplier with a 32-bit data bus. The implementation of the MAA on the MAXQ1103 has a 64 × 32-bit multiplier and has a 64-bit data bus. The MAA on the MAXQ1103 executes faster at the expense of using more silicon area.

Power analysis attacks such as simple power analysis (SPA) and differential power analysis (DPA) might be able to extract exponent information when running with optimization enabled. It is recommended that nonoptimized calculations always be done with private keys.

The data presented in **Tables 1** to **3** are typical run times. Each entry is the average time of 400 calculations using uniform random numbers for the base, modulus, and exponent, with the most significant bit set in the modulus. In the case of public key calculations, the hex value of 0x10001 was used instead of a random number. This is a typical value for the public exponent in RSA. The time calculated is from when the operation starts until it is finished. The time to load values into memory for the calculations is not included.

A significant speed improvement in modular-exponential operations can be realized by employing the Chinese remainder theorem (CRT). Using the CRT requires two smaller modular-exponentiation operations rather than one large one. Instead of performing a modular-exponential calculation on the large modulus, modular-exponential calculations are done on the two factors of the modulus. For example, in RSA, the modulus is the product of two prime numbers, p and q. If p and q are both 1024 bits, doing two modular-exponential operations on these would take approximately 165ms using the MAXQ1103. Without the CRT, a 2048-bit modular-exponential operation is required and will take approximately 557ms. The CRT algorithm requires additional calculations which will increase the total time, but it is expected to be better than twice as fast.

The data in the left side of Table 1 is the most interesting. These are the typical elapsed times to perform a modular exponentiation when running from the cryptographic ring in a nonoptimized mode. Typical elapsed times using optimization and a public key are in the right two columns.

Table 1. Typical Times While Running from the Cryptographic Ring

	MAA Running from Cryptographic Ring (Times in Milliseconds)							
	Private Key				Public Key = 0x10001			
	Nonoptimized		Optimized		Nonoptimized		Optimized	
Size	MAXQ1050/MAXQ1850 at 65MHz	MAXQ1103 at 55MHz	MAXQ1050/MAXQ1850 at 65MHz	MAXQ1103 at 55MHz	MAXQ1050/MAXQ1850 at 65MHz	MAXQ1103 at 55MHz	MAXQ1050/MAXQ1850 at 65MHz	MAXQ1103 at 55MHz
160	1.89	1.07	1.42	0.809	0.21	0.123	0.116	0.0723
192	2.91	1.36	2.19	1.02	0.26	0.130	0.147	0.0768
224	4.22	2.16	3.18	1.62	0.32	0.173	0.182	0.101
256	5.87	2.59	4.41	1.95	0.39	0.183	0.220	0.107
384	16.5	6.72	12.4	5.05	0.73	0.310	0.404	0.178
512	35.2	13.6	26.4	10.2	1.16	0.466	0.642	0.266
640	64.4	24.0	48.3	18.0	1.69	0.650	0.933	0.368
768	106.0	38.5	79.7	28.9	2.32	0.864	1.28	0.487
1024	237.0	82.5	178.0	61.9	3.86	1.38	2.12	0.772
1536	750.0	249.0	563.0	187.0	8.12	2.75	4.46	1.53
2048	1,720.0	557.0	1,290.0	418.0	13.9	4.58	7.64	2.54

Table 2 lists the typical times to perform a modular exponentiation with a private key data in both optimized and nonoptimized modes. Table 3 lists the typical times to perform a modular exponentiation with a public key in optimized and nonoptimized modes for the three parts.

Table 2. Typical Private Key Times While Running from the System Clock

	MAA Running from System Clock (Times in Milliseconds)					
	Private Key/Nonoptimized			Public Key/Optimized		
Size	MAXQ1050 at 25MHz	MAXQ1850 at 16MHz	MAXQ1103 at 25MHz	MAXQ1050 at 25MHz	MAXQ1850 at 16MHz	MAXQ1103 at 25MHz
160	4.93	7.68	2.37	3.71	5.78	1.79
192	7.58	11.8	3.00	5.70	8.88	2.26
224	11.0	17.2	4.75	8.27	12.9	3.58
256	15.3	23.9	5.71	11.5	17.9	4.29
384	42.9	67.0	14.8	32.2	50.3	11.1
512	91.7	143.0	30.0	68.9	107.0	22.5
640	167.0	262.0	52.9	126.0	196.0	39.6
768	276.0	432.0	84.8	208.0	324.0	63.6
1024	617.0	964.0	182.0	463.0	722.0	136.0
1536	1,950.0	3,050.0	549.0	1,460.0	2,290.0	412.0
2048	4,480.0	6,990.0	1,230.0	3,360.0	5,250.0	921.0

Table 3. Typical Public Key Times While Running from the System Clock

	MAA Running from System Clock (Times in Milliseconds)					
	Public Key = 0x10001/Nonoptimized			Public Key = 0x10001/Optimized		
Size	MAXQ1050 at 25MHz	MAXQ1850 at 16MHz	MAXQ1103 at 25MHz	MAXQ1050 at 25MHz	MAXQ1850 at 16MHz	MAXQ1103 at 25MHz
160	0.532	0.831	0.269	0.299	0.468	0.158
192	0.679	1.06	0.285	0.381	0.595	0.168
224	0.840	1.31	0.381	0.470	0.736	0.221
256	1.02	1.59	0.401	0.570	0.889	0.234
384	1.89	2.96	0.681	1.05	1.64	0.392
512	3.02	4.71	1.02	1.67	2.61	0.584
640	4.40	6.87	1.43	2.43	3.79	0.811
768	6.03	9.42	1.90	3.32	5.19	1.07
1024	10.1	15.7	3.03	5.53	8.64	1.70
1536	21.1	33.0	6.05	11.6	18.1	3.37
2048	36.3	56.7	10.1	19.9	31.1	5.59

Related Parts		
MAXQ1050	DeepCover Secure Microcontroller with USB and Hardware Cryptography	
MAXQ1103	DeepCover Secure Microcontroller with Rapid Zeroization Technology and Cryptography	
MAXQ1850	DeepCover Secure Microcontroller with Rapid Zeroization Technology and Cryptography	Free Samples

More Information
For Technical Support: http://www.maximintegrated.com/support
For Samples: http://www.maximintegrated.com/samples
Other Questions and Comments: http://www.maximintegrated.com/contact

Application Note 5145: http://www.maximintegrated.com/an5145
APPLICATION NOTE 5145, AN5145, AN 5145, APP5145, Appnote5145, Appnote 5145