

A. Analýza Blue Midnight Wish – útok na vzor,
Vlastimil Klíma, kryptolog konzultant, Praha
<http://cryptography.hyperlink.cz>, v.klima@volny.cz)
Prof. Danilo Gligoroski, Norwegian University of Science and Technology, Norway (danilog@item.ntnu.no ,
<http://www.item.ntnu.no/people/personalpages/fac/danilog/start>)

Článek navazuje na předchozí příspěvek Crypto-Worldu 2009 v čísle 12, ale i v číslech 3 a 7-8. Je první z malé série, kterou chceme věnovat úvahám o bezpečnosti BMW a stimulovat útoky na něj nebo analýzy a prezentovat i otevřené problémy. Ty by se mohly stát předmětem studentských prací. Proč? Velkou výhodou oproti jiným tématům je, že tyto rozbory jsou nyní velmi žádané, ať s negativním nebo pozitivním výsledkem, takže když problém bude vyřešen, nebo naopak nebude vyřešen a ukáže se, že je složitý, je to potřebný, žádaný a velmi dobře publikovatelný výsledek.

Malá aktualizace – urychlení BMW

Reference platform: Intel Core 2 Duo, 2.4 GHz, Windows Vista Ultimate 64-bit edition

Compiler: Intel C++ 11.1.46

Mode: 32-bit (x86)		Performance in cycles/byte for different message lengths (in bytes)								
	1	8	64	576	1024	1536	4096	100000	eBash version	
BMW224/256	1129	142.63	25.33	9.42	8.52	8.14	7.69	7.45	bmw256/optc02	
BMW384/512	1321	165.13	22.33	6.29	5.73	5.28	4.73	4.4	bmw512/optx86sse2	

Mode: 64-bit (x64)		Performance in cycles/byte for different message lengths (in bytes)								
	1	8	64	576	1024	1536	4096	100000	eBash version	
BMW224/256	1081	135.13	24.77	9.02	8.18	7.76	7.29	7.02	bmw256/optc04	
BMW384/512	1105	138.13	17.27	5.09	4.58	4.22	4.05	3.48	bmw512/optc04	

Obr. 1: Nejnovější urychlení BMW v SW, viz [1]

Na obrázku 1 jsou uvedeny nejnovější vynikající časy pro BMW, ještě lepší, než ty, uvedené v minulém čísle Crypto-Worldu. Tím se BMW stala ještě viditelněji nejrychlejším kandidátem v SW ve všech ukazatelích. Je to zásluha týmu BMW (vyjma mě).

Analýza celku a částí

BMW lze zkoumat jako celek, avšak to se velice brzo zastavíme, neboť dostáváme soustavu rovnic, jejíž řešení neumíme nalézt ani v té nej-nej-nejzjednodušenější podobě. Proto budeme chtít řešit alespoň dílčí úlohy, které by mohly k řešení vést. Ukážeme však, že ani dílčí úlohy, týkající se nejjednodušších (atomárních) stavebních bloků BMW, neumíme řešit. A teď doufám, že se najde někdo, kdo bude v opozici, kdo najde nějaké řešení, chybu, nedostatek, odchylku od očekávaného chování atomárních bloků nebo vyšších celků nebo BMW nebo její architektury. Jakákoliv opozice je vítaná, zejména zde, na stránkách Crypto-Worldu, e-mailem, v diskusní skupině sci.crypt, v poštovní konferenci NISTu k hašovacím funkcím apod.

Označení

Článek bude využívat označení zavedené v Crypto-Worldu 12/2009. Připomeňme jen šířku slova $w = 32$ nebo 64 bitů, délku bloku zprávy a průběžné haše $n = 16 \cdot w$ (mají 16 slov) a výpočet haše:

1. Předzpracování

- (a) Doplní zprávu M jednoznačným definovaným způsobem o délku zprávy v bitech a doplněk
- (b) Rozděl zprávu na celistvý násobek (N) m -bitových bloků $M^{(1)}, \dots, M^{(N)}$.
- (c) Nastav počáteční hodnotu průběžné haše $H^{(0)}$ na konstantu (CONST^0).

2. Výpočet haše

For $i = 1$ to N : $H^{(i)} = f(M^{(i)}, H^{(i-1)})$.

3. Finalizace

$H^{\text{final}} = f(H^{(N)}, \text{CONST}^{\text{final}})$, kde $\text{CONST}^{\text{final}}$ je konstanta.

4. Závěr

$H(M) =$ dolních n bitů z hodnoty H^{final} .

BMW vždy projde minimálně dvě iterace

Jak ukazuje následující schéma, BMW vždy projde minimálně dvě iterace - a to první a poslední. Kromě toho projde volitelně podle délky zpracovávané zprávy ještě určité množství tzv. vnitřních bloků mezi prvním a posledním, viz obr. 3 a 4. První a poslední blok mají pevně nastavenou hodnotu H . U prvního bloku je to konstanta CONST^0 , u posledního bloku je to konstanta $\text{CONST}^{\text{final}}$. V první iteraci zpracovává kompresní funkce $f(M^{(1)}, \text{CONST}^0)$ první blok zprávy $M^{(1)}$ a konstantu CONST^0 . Pokud tento blok zprávy není zároveň blokem posledním, následují ještě vnitřní iterace. Výsledkem je poslední průběžná haš H , která vstupuje v roli bloku zprávy do finalizace $f(H, \text{CONST}^{\text{final}})$.

Rychlost a složitost

U BMW jsme použili malý úskok nebo trik, chcete-li, stejně jako někteří další návrháři SHA-3. U funkcí SHA-2 postačí zkoumat jeden blok, který může být také blokem posledním. Tento blok lze zapsat rovnicemi, které udávají složitost problému, pokud se nenajde rychlejší řešení. Soustava rovnic u BMW je dvakrát větší, neboť k zápisu libovolného problému potřebujeme zapsat rovnice pro dva bloky. Trik je v tom, že na rychlost hašování toto má vliv jen u krátkých zpráv, zatímco u delších zpráv se zpracování přídavného posledního bloku „rozpustí“ v čase, potřebném na zpracování velkého počtu bloků předchozích. Rozdíl ve složitosti je ale ohromný, jak vidíme na obrázku 3. Na obrázku 4 vidíme realizaci BMW pro eventuelní libovolný počet vnitřních bloků a na obrázku 2 je jedna iterace zvětšená.

Úloha první - hledání vzoru

Pokud útočník bude znát vzor k dané haši (ať na něj přijde jak chce), bude zcela jistě znát vzor této haše v posledním bloku. K důkazu složitosti nalezení vzoru hašovací funkce postačí proto ukázat, že je příliš složité nalezení vzoru pro poslední iteraci. Situaci znázorňuje obrázek 5. Útočník zná výstupní haš o 8 slovech a hledá vstupní blok M o 16 slovech. Hodnota H , která vstupuje do posledního bloku je neměnná konstanta H^{final} . Útočník zná hnědou hodnotu a hledá žluté proměnné.

Hledání vzoru poslední iterace je ekvivalentní hledání řešení (M) soustavy rovnic na obrázku 6. Je to soustava, kterou můžeme zapsat atomárními operacemi jednoduše následovně:

$$Q_a = A_2(A_1(M \oplus H^{\text{final}})) + \text{ROTL}^1(H^{\text{final}}), \quad (\text{S1})$$

$$Q_b = T^L(T^U(Q_a) + ((B(\text{rot}M) + K) \oplus \text{ROTL}^7(H^{\text{final}}))), \quad (\text{S2})$$

$$G = (M \oplus L_a(Q_b)) + (Q_a \oplus L_b(Q_b)), \quad (\text{S3})$$

$$\text{hash} = 8_lwords_of(f_6(G)). \quad (\text{S4})$$

Povšimněme si, že pokud bychom chtěli zkoušet vstup do posledního (kompresního) bloku f_2 , hrubou silou nebo jakkoliv jinak, jedním ze vstupů je i M. Zpětný chod v bloku f_2 je tedy už přímo zpětným chodem celé kompresní funkce f .

Povšimněme si dále, že druhým vstupem bloku f_2 je Q_a , což je bijektivní obraz M, neboť $Q_a = A_2(A_1(M \oplus \text{const}_1)) + \text{const}_2$. Jakékoliv předvídání části nebo celé hodnoty Q_a je tedy ekvivalentní předvídání části nebo celé hodnoty M.

Třetí vstup do bloku f_2 je Q_b , což je pseudonáhodná funkce M. Zde můžeme provádět určité manipulace, protože jedné hodnotě Q_b může odpovídat žádný, jeden nebo mnoho vzorů M. Pokud bychom například pro více zpráv M docílili toho, že

$$Q_b = \text{const}, \quad (\mathbf{R0})$$

máme vztahy:

$$\begin{aligned} Q_a &= A_2(A_1(M \oplus \text{const})) + \text{const}, \\ \text{const} &= T^L(T^U(Q_a) + ((B(\text{rot}M) + \text{const}) \oplus \text{const})), \\ G &= (M \oplus \text{const}) + (Q_a \oplus \text{const}), \\ \text{hash} &= 8_lwords_of(f_6(G)), \end{aligned}$$

kde **const** jsou nějaké konstanty (obecně různé). Druhou rovnici pro jednoduchost neuvažujeme, tj. předpokládáme, že umíme nalézt několik (třeba velmi mnoho) zpráv M, které vedou na stejnou hodnotu Q_b a pracujeme pouze s těmito zprávami. Zbývají rovnice

$$Q_a = A_2(A_1(M \oplus \text{const})) + \text{const}, \quad (\mathbf{R1})$$

$$G = (M \oplus \text{const}) + (Q_a \oplus \text{const}), \quad (\mathbf{R2})$$

$$\text{hash} = 8_lwords_of(f_6(G)), \quad (\mathbf{R3})$$

tj.

$$G = (M \oplus \text{const}) + ((A_2(A_1(M \oplus \text{const})) + \text{const}) \oplus \text{const}), \quad (\mathbf{R4})$$

$$\text{hash} = 8_lwords_of(f_6(G)), \quad (\mathbf{R5})$$

neboli

$$\text{hash} = 8_lwords_of(f_6((M \oplus \text{const}) + ((A_2(A_1(M \oplus \text{const})) + \text{const}) \oplus \text{const}))). \quad (\mathbf{R6})$$

Udělejme malou odbočku. Pokud uvažujeme nulové konstanty v (R6), máme tvar

$$\text{hash} = 8_lwords_of(f_6(M + A_2A_1(M))). \quad (\mathbf{R6a})$$

Ukazuje se, že by bylo dobré prozkoumat **vlastnosti funkce**

$$M + A_2A_1(M), \quad (\mathbf{R7})$$

což nevypadá vůbec složitě (avšak složitě je !!! a doufáme, že se najdou oponenti) nebo obecněji vlastnosti funkce

$$A_2(A_1(M \oplus \text{const}_1)) + \text{const}_2, \quad (\mathbf{R8})$$

neboli vlastnosti Q_a jakožto funkce M. No ale teď se vraťme k hlavní rovnici.

Zdá se, že v soustavě (R6) máme velmi mnoho stupňů volnosti, neboť pevně je určeno pouze 8 slov (hash) bijektivního obrazu $f_6(G)$ hodnoty G , zatímco zpráva M má 16 volných slov. Bohužel, do soustavy (R6) nevstupuje M libovolně, ale pouze ty M , pro něž je $Q_b = \text{const}$ pro zvolenou konstantu. Proto je velmi důležité zkoumat jak moc je Q_b náhodná a zda by nešlo najít hodně zpráv M se stejnou $Q_b(M)$. **Výzkum Q_b jako funkce M je klíčový.**

Zjednodušíme úlohu (R0, R6):

- Předpokládejme, že výzkum Q_b přinesl velké ovoce a že každé řešení (R0) je už automaticky řešením (R6). Potom, pokud umíme řešit soustavu (R0), umíme najít vzor finální iterace.
- Předpokládejme, že výzkum (R6) přinesl velké ovoce a že každé řešení (R6) je už automaticky řešením (R0). Potom, pokud umíme řešit soustavu (R6), umíme najít vzor finální iterace.

Obě dvě zjednodušené úlohy mohou přispět k posouzení bezpečnosti BMW. Zejména úloha (R6) se zdá jednoduchá.

Pojďme ke složitější úloze. Pokud bychom neuměli nalézat zprávy M pro něž je $Q_b = \text{const}$, hodnota Q_b by byla obecně proměnná a řešili bychom soustavu (S1 - S4). Bez meziproměnných to dává sice jen jednu rovnici, ale nepřiliš vábnou:

$$\text{hash} = \mathbf{8_lswords_of}(f_6((M \oplus L_a(T^L(T^U(A_2(A_1(M \oplus H^{\text{final}})) + \text{ROTL}^1(H^{\text{final}})) + ((B(\text{rot}M) + K) \oplus \text{ROTL}^7(H^{\text{final}})))))) + ((A_2(A_1(M \oplus H^{\text{final}})) + \text{ROTL}^1(H^{\text{final}})) \oplus L_b(T^L(T^U(A_2(A_1(M \oplus H^{\text{final}})) + \text{ROTL}^1(H^{\text{final}})) + ((B(\text{rot}M) + K) \oplus \text{ROTL}^7(H^{\text{final}})))))))). \quad (\text{S5})$$

Soustava (S5) je jednou rovnicí, ale pro osm známých slov na levé straně (hash) a 16 neznámých slov (M), čili je to soustava osmi rovnic o 16 neznámých. To je pro útočníka nadějně - je tu velmi mnoho stupňů volnosti pro nalezení řešení.

Jen pro zajímavost, pokud konstanty nahradíme symbolicky slovem const , máme

$$\text{hash} = \mathbf{8_lswords_of}(f_6((M \oplus L_a(T^L(T^U(A_2(A_1(M \oplus \text{const})) + \text{const}) + ((B(\text{rot}M) + \text{const}) \oplus \text{const})))))) + ((A_2(A_1(M \oplus \text{const})) + \text{const}) \oplus L_b(T^L(T^U(A_2(A_1(M \oplus \text{const})) + \text{const}) + ((B(\text{rot}M) + \text{const}) \oplus \text{const})))))), \quad (\text{S5a})$$

a pokud všechny konstanty uvažujeme nulové, máme soustavu

$$Q_a = A_2 A_1(M), \quad (\text{S1b})$$

$$Q_b = T^L(T^U(Q_a) + B(\text{rot}M)), \quad (\text{S2b})$$

$$G = (M \oplus L_a(Q_b)) + (Q_a \oplus L_b(Q_b)), \quad (\text{S3b})$$

$$\text{hash} = \mathbf{8_lswords_of}(f_6(G)). \quad (\text{S4b})$$

neboli

$$\text{hash} = \mathbf{8_lswords_of}(f_6((M \oplus L_a(T^L(T^U(A_2 A_1(M) + B(\text{rot}M)))))) + (A_2 A_1(M) \oplus L_b(T^L(T^U(A_2 A_1(M) + B(\text{rot}M)))))), \quad (\text{S5b})$$

což je také hodné zkoumání. Nalezení vzoru pro finální iteraci je ekvivalentní úloze (S1-S4), resp. (S5), která je značně složitá. Nalezení jakékoliv zkratky v řešení by pomohlo lépe pochopit bezpečnost BMW.

Pojďme dále ke složitější úloze.

Předpokládejme, že útočník umí řešit předchozí úlohu nalezení vzoru pro finální iteraci. Umí tedy k zadané finální hodnotě haše H o n bitech nalézt blok zprávy X o $2n$ bitech, vedoucí společně s H^{final} k výsledné haši $H = 8_lwords_of(f(X, H^{\text{final}}))$.

Tuto schopnost však útočník nemůže využít k dokončení útoku, neboť nyní má k dispozici výstup X a musí k němu nalézt skutečný blok zprávy M . Teď má ovšem zadanou výstupní hodnotu X o $2n$ bitech, zatímco v předchozí úloze měl jenom n bitů. Jinými slovy je teď neznámých stejně jako rovnic a nadbytečné stupně volnosti z minulé úlohy jsou pryč. Jedná se o úlohu

$$X = f(M, \text{CONST}^{(0)}).$$

Můžeme ji zapsat jako

$$\begin{aligned} Q_a &= A_2(A_1(M \oplus \text{CONST}^{(0)})) + \text{ROTL}^1(\text{CONST}^{(0)}), \\ Q_b &= T^L(T^U(Q_a) + ((B(\text{rot}M) + K) \oplus \text{ROTL}^7(\text{CONST}^{(0)}))) \\ G &= (M \oplus L_a(Q_b)) + (Q_a \oplus L_b(Q_b)), \\ X &= (f_6(G)). \end{aligned}$$

Protože hodnotu G můžeme dopočítat z X , je to pro nás známá hodnota. Takže úloha hledání vnitřního vzoru je ekvivalentní řešení soustavy (T1 - T3) pro zadané konstanty:

$$\begin{aligned} Q_a &= A_2(A_1(M \oplus \text{const}_1)) + \text{const}_1, & \text{(T1)} \\ Q_b &= T^L(T^U(Q_a) + ((B(\text{rot}M) + \text{const}_2) \oplus \text{const}_3)), & \text{(T2)} \\ \text{const}_4 &= (M \oplus L_a(Q_b)) + (Q_a \oplus L_b(Q_b)). & \text{(T3)} \end{aligned}$$

Poznámka. Dále je tu podmínka, že zpracovávaná zpráva by měla mít platný padding (tj. minimálně 65 bitů by mělo rezervovanou hodnotu), ale to bychom útočníkovi mohli prominout. K útoku na BMW by stačilo pouze najít nějakou negativní vlastnost, nemusí být zničena celá, protože v současné době je ve hře stále ještě 14 kandidátů. Pravda, BMW je k odstřelu na prvním místě, protože je nejrychlejší.

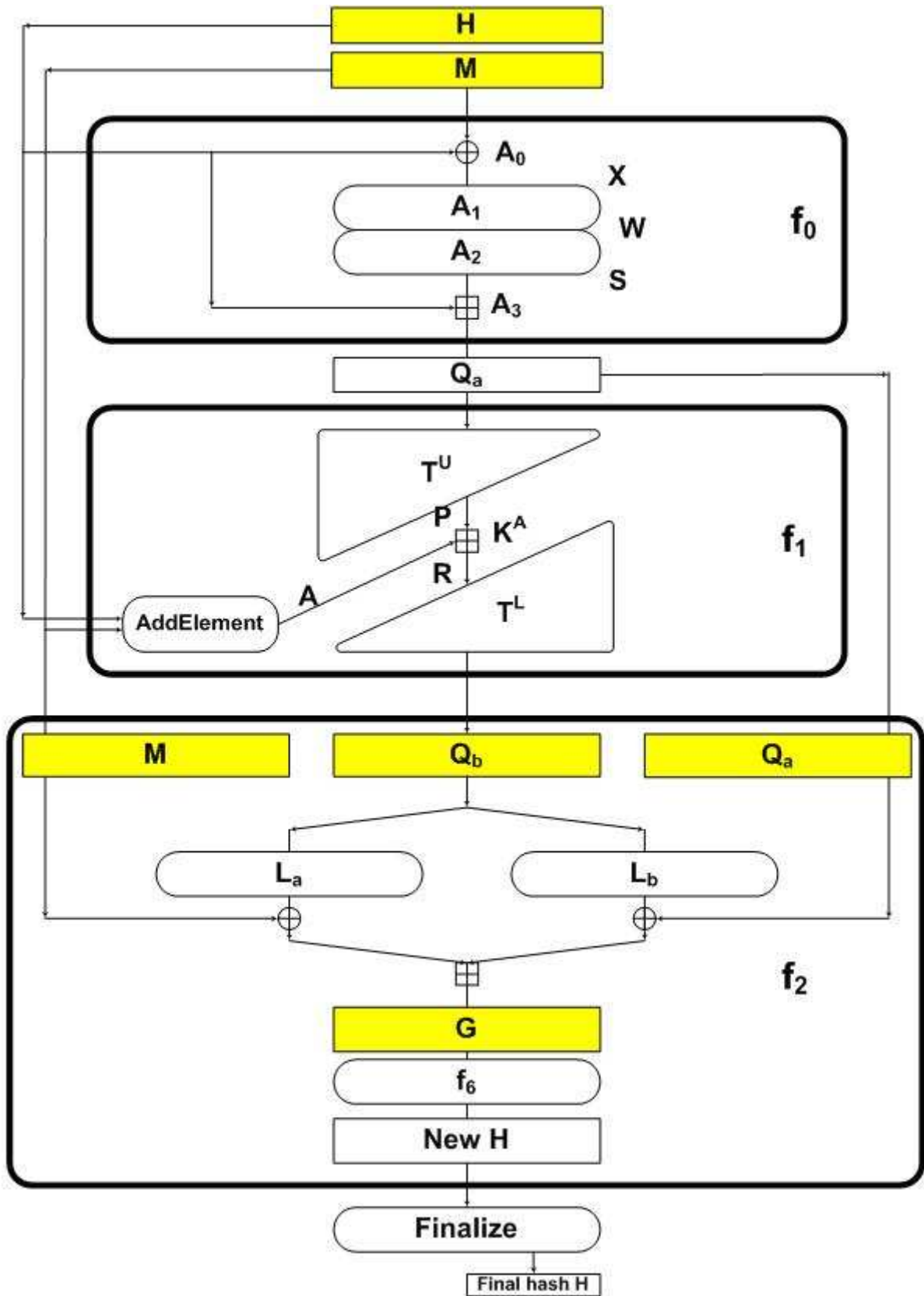
Závěr

V tomto článku jsme uvedli několik dílčích úloh, které se objevují při hledání vzoru k hašovací funkci BMW. Řešení všech těchto úloh je otevřené. Pro přehlednost jsou zvýrazněny tučně. Velice doporučujeme začít s analýzou těch nejjednodušších (R7) nebo (R8) a dále (R6a), (R0), (R6) a dále.

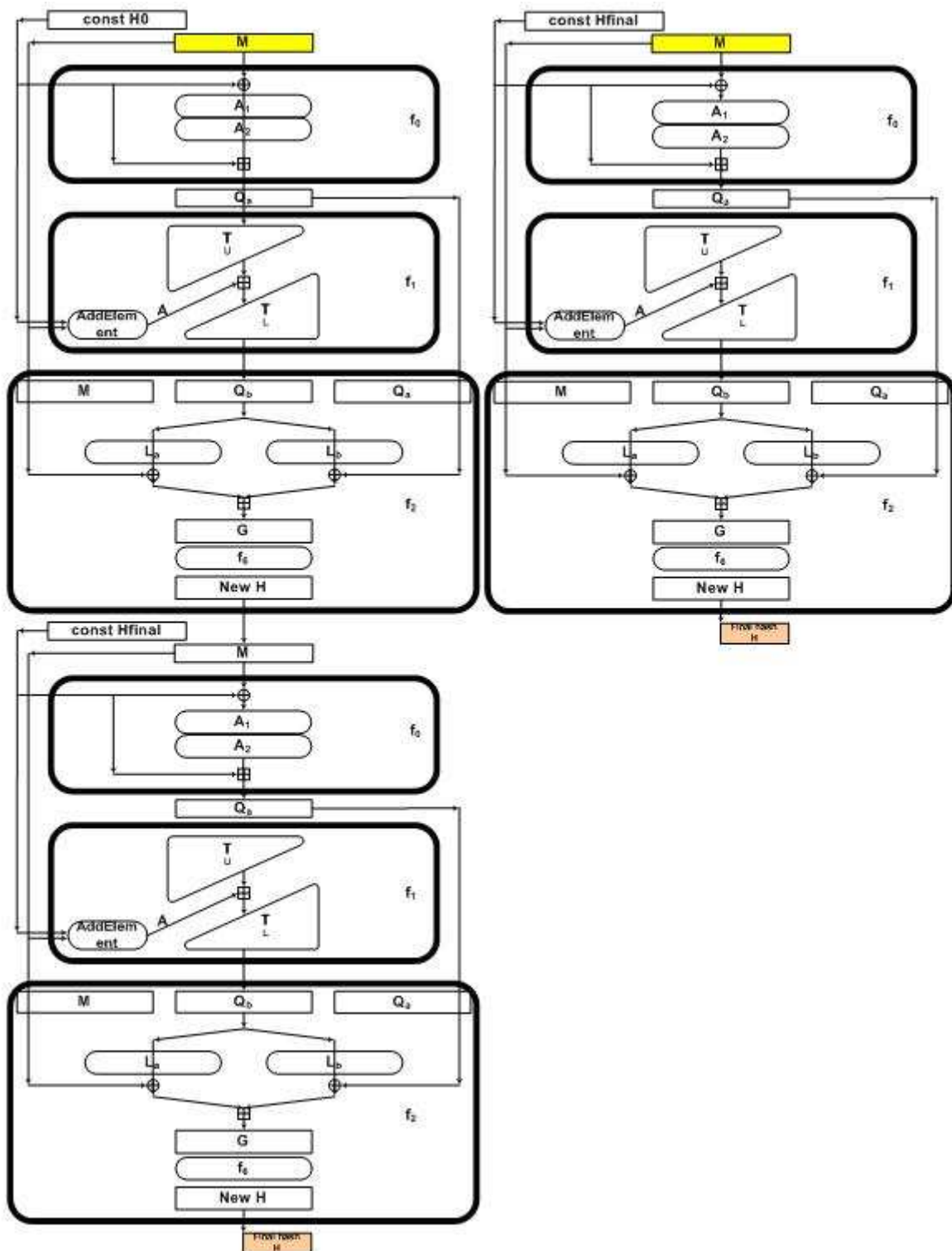
Literatura

- [1] domácí stránka týmu BMW: http://www.q2s.ntnu.no/sha3_nist_competition/start
- [2] stránka NIST k projektu SHA-3: <http://csrc.nist.gov/groups/ST/hash/index.html>
- [3] dokumenty a analýzy BMW a průběžné novinky k projektu SHA-3: http://cryptography.hyperlink.cz/BMW/BMW_CZ.html
- [4] Danilo Gligoroski, Vlastimil Klima, [On Blue Midnight Wish Decomposition](#), SantaCrypt 2009, Dec. 3-4, 2009, Prague, Czech Republic, Proceedings of SantaCrypt 2009, ISBN 978-80-904257-0-5, pp. 41-51

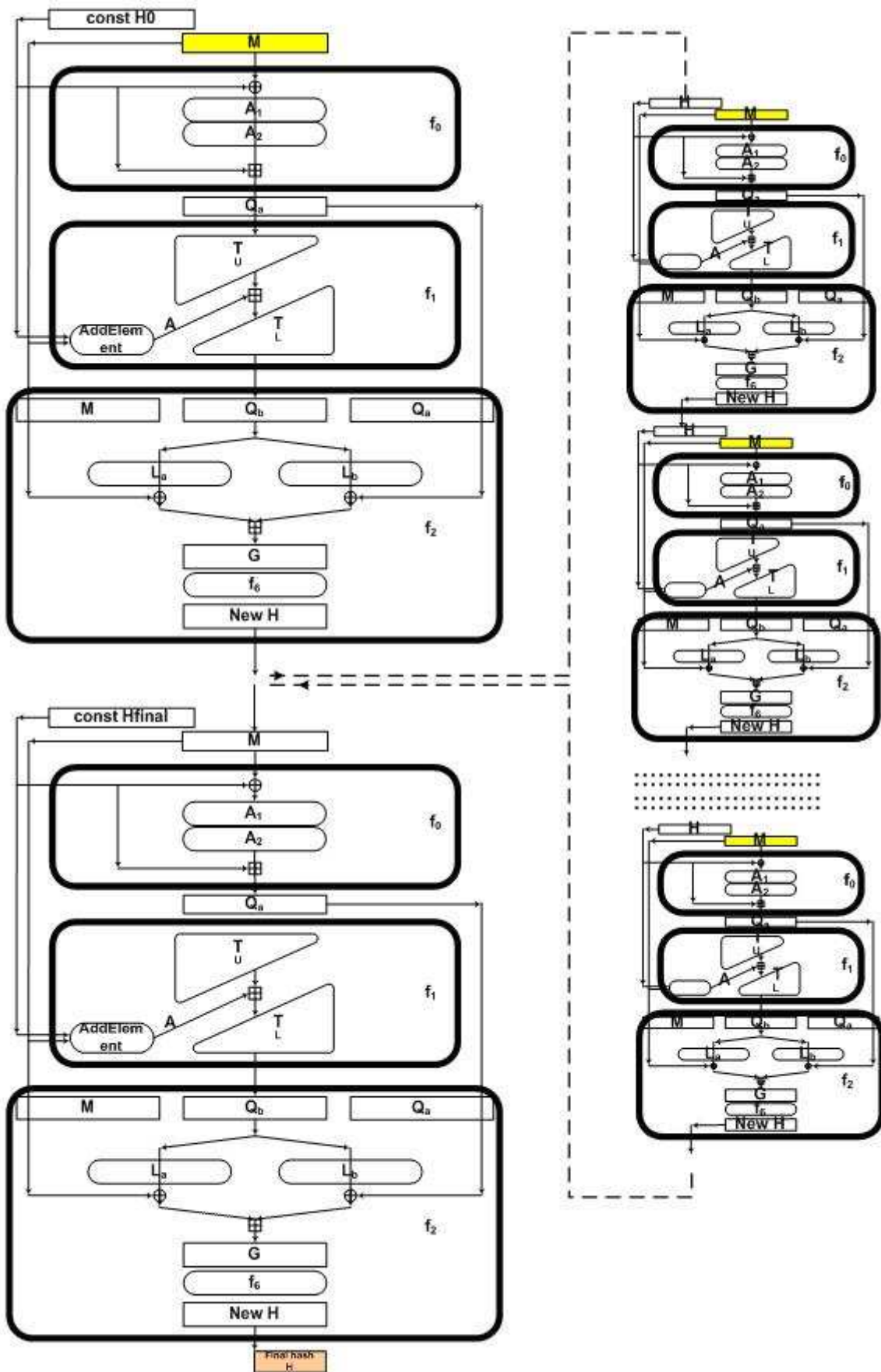
Příloha – Obrázky



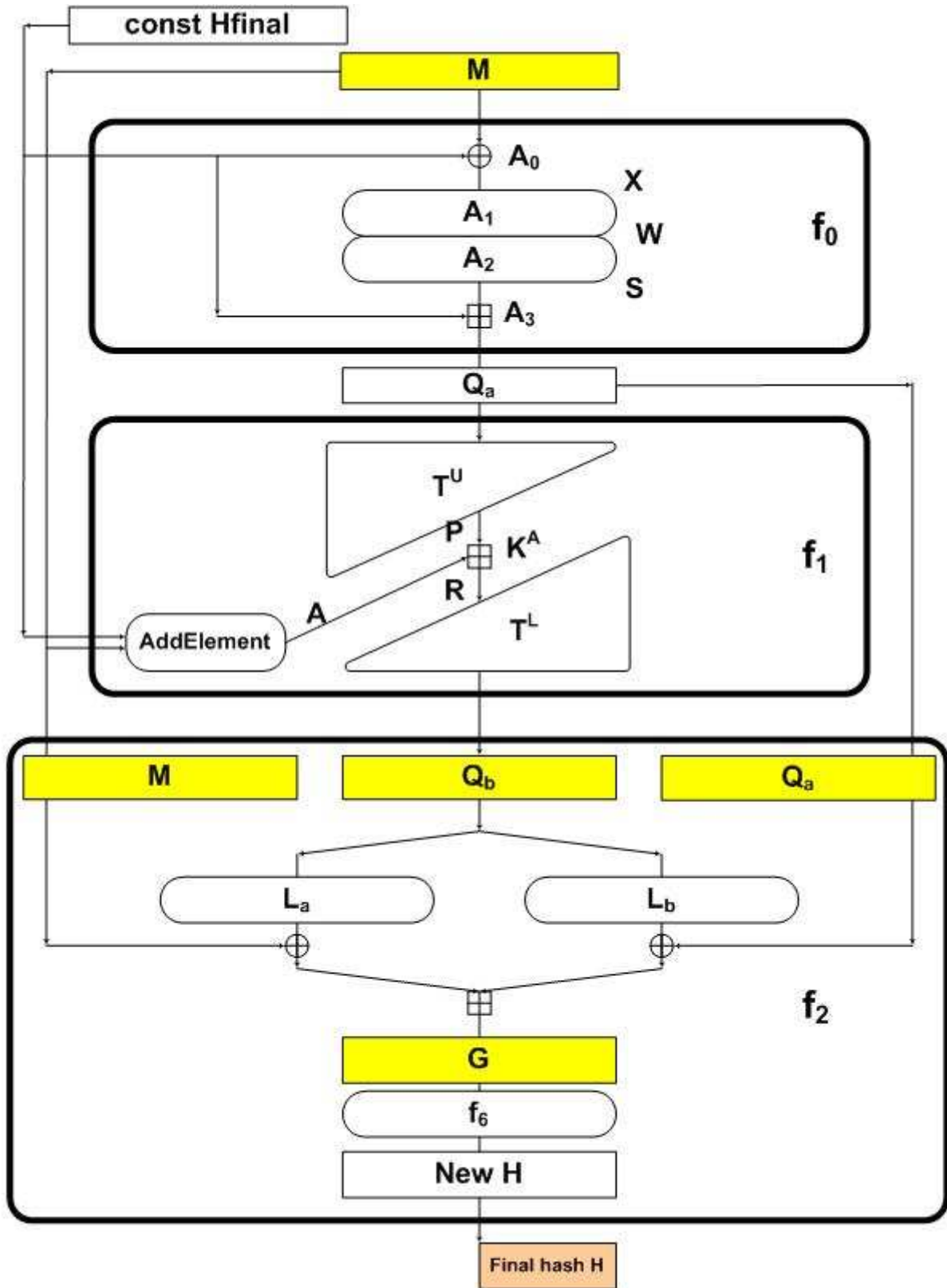
Obr.2: Kompresní funkce BMW



Obr.3: Rozdíl složitosti BMW bez a s přidáním finalizací (jediný vstup je jeden blok na počátku, výstup je na konci, vše ostatní je funkce zpracování jednoho bloku)



Obr.4: BMW s vnitřními bloky



Obr.5: Hledání vzoru u poslední iterace

$$\begin{array}{l}
 Q_0 = H_1 + *6 ((M_5 \oplus H_5) - (M_7 \oplus H_7) + (M_{10} \oplus H_{10}) + (M_{13} \oplus H_{13}) + (M_{14} \oplus H_{14})) \\
 Q_1 = H_2 + *1 ((M_6 \oplus H_6) - (M_8 \oplus H_8) + (M_{11} \oplus H_{11}) + (M_{14} \oplus H_{14}) - (M_{15} \oplus H_{15})) \\
 Q_2 = H_3 + *2 ((M_0 \oplus H_0) + (M_7 \oplus H_7) + (M_9 \oplus H_9) - (M_{12} \oplus H_{12}) + (M_{15} \oplus H_{15})) \\
 Q_3 = H_4 + *3 ((M_0 \oplus H_0) - (M_1 \oplus H_1) + (M_8 \oplus H_8) - (M_{10} \oplus H_{10}) + (M_{13} \oplus H_{13})) \\
 Q_4 = H_5 + *4 ((M_1 \oplus H_1) + (M_2 \oplus H_2) + (M_9 \oplus H_9) - (M_{11} \oplus H_{11}) - (M_{14} \oplus H_{14})) \\
 Q_5 = H_6 + *6 ((M_2 \oplus H_2) - (M_0 \oplus H_0) + (M_{10} \oplus H_{10}) - (M_{12} \oplus H_{12}) + (M_{15} \oplus H_{15})) \\
 Q_6 = H_7 + *1 ((M_4 \oplus H_4) - (M_0 \oplus H_0) - (M_3 \oplus H_3) - (M_{11} \oplus H_{11}) + (M_{13} \oplus H_{13})) \\
 Q_7 = H_8 + *2 ((M_1 \oplus H_1) - (M_4 \oplus H_4) - (M_5 \oplus H_5) - (M_{12} \oplus H_{12}) - (M_{14} \oplus H_{14})) \\
 Q_8 = H_9 + *3 ((M_2 \oplus H_2) - (M_5 \oplus H_5) - (M_6 \oplus H_6) + (M_{13} \oplus H_{13}) - (M_{15} \oplus H_{15})) \\
 Q_9 = H_{10} + *4 ((M_0 \oplus H_0) - (M_3 \oplus H_3) + (M_6 \oplus H_6) - (M_7 \oplus H_7) + (M_{14} \oplus H_{14})) \\
 Q_{10} = H_{11} + *6 ((M_6 \oplus H_6) - (M_1 \oplus H_1) - (M_4 \oplus H_4) - (M_7 \oplus H_7) + (M_{15} \oplus H_{15})) \\
 Q_{11} = H_{12} + *1 ((M_8 \oplus H_8) - (M_0 \oplus H_0) - (M_2 \oplus H_2) - (M_5 \oplus H_5) + (M_9 \oplus H_9)) \\
 Q_{12} = H_{13} + *2 ((M_1 \oplus H_1) + (M_3 \oplus H_3) - (M_6 \oplus H_6) - (M_9 \oplus H_9) + (M_{10} \oplus H_{10})) \\
 Q_{13} = H_{14} + *3 ((M_2 \oplus H_2) + (M_4 \oplus H_4) + (M_7 \oplus H_7) + (M_{10} \oplus H_{10}) + (M_{11} \oplus H_{11})) \\
 Q_{14} = H_{15} + *4 ((M_3 \oplus H_3) - (M_5 \oplus H_5) + (M_8 \oplus H_8) - (M_{11} \oplus H_{11}) - (M_{12} \oplus H_{12})) \\
 Q_{15} = H_0 + *6 ((M_{12} \oplus H_{12}) - (M_4 \oplus H_4) - (M_6 \oplus H_6) - (M_9 \oplus H_9) + (M_{13} \oplus H_{13}))
 \end{array}$$

$$\begin{array}{l}
 Q_{16} = (H_6 \oplus (ROTL^1(M_9) + ROTL^2(M_3) - ROTL^{13}(M_{10}) + K_0)) + *1(Q_0) + *2(Q_1) + *3(Q_2) + *0(Q_3) + \\
 *1(Q_4) + *2(Q_5) + *3(Q_6) + *0(Q_7) + *1(Q_8) + *2(Q_9) + *3(Q_{10}) + *0(Q_{11}) + *1(Q_{12}) + *2(Q_{13}) + *3(Q_{14}) + *0(Q_{15}) \\
 Q_{17} = (H_7 \oplus (ROTL^2(M_1) + ROTL^3(M_4) - ROTL^{12}(M_{11}) + K_1)) + *1(Q_1) + *2(Q_2) + *3(Q_3) + *0(Q_4) + \\
 *1(Q_5) + *2(Q_6) + *3(Q_7) + *0(Q_8) + *1(Q_9) + *2(Q_{10}) + *3(Q_{11}) + *0(Q_{12}) + *1(Q_{13}) + *2(Q_{14}) + *3(Q_{15}) + *0(Q_{16}) \\
 Q_{18} = (H_8 \oplus (ROTL^3(M_2) + ROTL^4(M_5) - ROTL^{11}(M_{12}) + K_2)) + Q_2 + *1(Q_3) + \\
 *4 + *2(Q_5) + Q_6 + *3(Q_7) + Q_8 + *4(Q_9) + Q_{10} + *5(Q_{11}) + Q_{12} + *6(Q_{13}) + Q_{14} + *7(Q_{15}) + *4(Q_{16}) + *5(Q_{17}) \\
 Q_{19} = (H_9 \oplus (ROTL^4(M_3) + ROTL^5(M_6) - ROTL^{10}(M_{13}) + K_3)) + Q_3 + *1(Q_4) + \\
 Q_5 + *2(Q_6) + Q_7 + *3(Q_8) + Q_9 + *4(Q_{10}) + Q_{11} + *5(Q_{12}) + Q_{13} + *6(Q_{14}) + Q_{15} + *7(Q_{16}) + *4(Q_{17}) + *5(Q_{18}) \\
 Q_{20} = (H_{10} \oplus (ROTL^5(M_4) + ROTL^6(M_7) - ROTL^9(M_{14}) + K_4)) + Q_4 + *1(Q_5) + \\
 Q_6 + *2(Q_7) + Q_8 + *3(Q_9) + Q_{10} + *4(Q_{11}) + Q_{12} + *5(Q_{13}) + Q_{14} + *6(Q_{15}) + Q_{16} + *7(Q_{17}) + *4(Q_{18}) + *5(Q_{19}) \\
 Q_{21} = (H_{11} \oplus (ROTL^6(M_5) + ROTL^7(M_8) - ROTL^8(M_{15}) + K_5)) + Q_5 + *1(Q_6) + \\
 Q_7 + *2(Q_8) + Q_9 + *3(Q_{10}) + Q_{11} + *4(Q_{12}) + Q_{13} + *5(Q_{14}) + Q_{15} + *6(Q_{16}) + Q_{17} + *7(Q_{18}) + *4(Q_{19}) + *5(Q_{20}) \\
 Q_{22} = (H_{12} \oplus (ROTL^7(M_0) + ROTL^{10}(M_9) - ROTL^1(M_6) + K_6)) + Q_6 + *1(Q_7) + \\
 Q_8 + *2(Q_9) + Q_{10} + *3(Q_{11}) + Q_{12} + *4(Q_{13}) + Q_{14} + *5(Q_{15}) + Q_{16} + *6(Q_{17}) + Q_{18} + *7(Q_{19}) + *4(Q_{20}) + *5(Q_{21}) \\
 Q_{23} = (H_{13} \oplus (ROTL^8(M_7) + ROTL^{11}(M_{10}) - ROTL^2(M_1) + K_7)) + Q_7 + *1(Q_8) + \\
 Q_9 + *2(Q_{10}) + Q_{11} + *3(Q_{12}) + Q_{13} + *4(Q_{14}) + Q_{15} + *5(Q_{16}) + Q_{17} + *6(Q_{18}) + Q_{19} + *7(Q_{20}) + *4(Q_{21}) + *5(Q_{22}) \\
 Q_{24} = (H_{14} \oplus (ROTL^9(M_8) + ROTL^{12}(M_{11}) - ROTL^3(M_2) + K_8)) + Q_8 + *1(Q_9) + \\
 Q_{10} + *2(Q_{11}) + Q_{12} + *3(Q_{13}) + Q_{14} + *4(Q_{15}) + Q_{16} + *5(Q_{17}) + Q_{18} + *6(Q_{19}) + Q_{20} + *7(Q_{21}) + *4(Q_{22}) + *5(Q_{23}) \\
 Q_{25} = (H_{15} \oplus (ROTL^{10}(M_9) + ROTL^{13}(M_{12}) - ROTL^4(M_3) + K_9)) + Q_9 + *1(Q_{10}) + \\
 Q_{11} + *2(Q_{12}) + Q_{13} + *3(Q_{14}) + Q_{15} + *4(Q_{16}) + Q_{17} + *5(Q_{18}) + Q_{19} + *6(Q_{20}) + Q_{21} + *7(Q_{22}) + *4(Q_{23}) + *5(Q_{24}) \\
 Q_{26} = (H_0 \oplus (ROTL^{11}(M_{10}) + ROTL^{14}(M_{13}) - ROTL^5(M_4) + K_{10})) + Q_{10} + *1(Q_{11}) + \\
 Q_{12} + *2(Q_{13}) + Q_{14} + *3(Q_{15}) + Q_{16} + *4(Q_{17}) + Q_{18} + *5(Q_{19}) + Q_{20} + *6(Q_{21}) + Q_{22} + *7(Q_{23}) + *4(Q_{24}) + *5(Q_{25}) \\
 Q_{27} = (H_1 \oplus (ROTL^{12}(M_{11}) + ROTL^{15}(M_{14}) - ROTL^6(M_5) + K_{11})) + Q_{11} + *1(Q_{12}) + \\
 Q_{13} + *2(Q_{14}) + Q_{15} + *3(Q_{16}) + Q_{17} + *4(Q_{18}) + Q_{19} + *5(Q_{20}) + Q_{21} + *6(Q_{22}) + Q_{23} + *7(Q_{24}) + *4(Q_{25}) + *5(Q_{26}) \\
 Q_{28} = (H_2 \oplus (ROTL^{13}(M_{12}) + ROTL^{16}(M_{15}) - ROTL^7(M_6) + K_{12})) + Q_{12} + *1(Q_{13}) + \\
 Q_{14} + *2(Q_{15}) + Q_{16} + *3(Q_{17}) + Q_{18} + *4(Q_{19}) + Q_{20} + *5(Q_{21}) + Q_{22} + *6(Q_{23}) + Q_{24} + *7(Q_{25}) + *4(Q_{26}) + *5(Q_{27}) \\
 Q_{29} = (H_3 \oplus (ROTL^{14}(M_{13}) + ROTL^{17}(M_0) - ROTL^8(M_7) + K_{13})) + Q_{13} + *1(Q_{14}) + \\
 Q_{15} + *2(Q_{16}) + Q_{17} + *3(Q_{18}) + Q_{19} + *4(Q_{20}) + Q_{21} + *5(Q_{22}) + Q_{23} + *6(Q_{24}) + Q_{25} + *7(Q_{26}) + *4(Q_{27}) + *5(Q_{28}) \\
 Q_{30} = (H_4 \oplus (ROTL^{15}(M_{14}) + ROTL^{18}(M_1) - ROTL^9(M_8) + K_{14})) + Q_{14} + *1(Q_{15}) + \\
 Q_{16} + *2(Q_{17}) + Q_{18} + *3(Q_{19}) + Q_{20} + *4(Q_{21}) + Q_{22} + *5(Q_{23}) + Q_{24} + *6(Q_{25}) + Q_{26} + *7(Q_{27}) + *4(Q_{28}) + *5(Q_{29}) \\
 Q_{31} = (H_5 \oplus (ROTL^{16}(M_{15}) + ROTL^{19}(M_2) - ROTL^{10}(M_9) + K_{15})) + Q_{15} + *1(Q_{16}) + \\
 Q_{17} + *2(Q_{18}) + Q_{19} + *3(Q_{20}) + Q_{21} + *4(Q_{22}) + Q_{23} + *5(Q_{24}) + Q_{25} + *6(Q_{26}) + Q_{27} + *7(Q_{28}) + *4(Q_{29}) + *5(Q_{30})
 \end{array}$$

$$\begin{array}{l}
 XL = Q_{16} \oplus Q_{17} \oplus Q_{18} \oplus Q_{19} \oplus Q_{20} \oplus Q_{21} \oplus Q_{22} \oplus Q_{23} \\
 XR = Q_{16} \oplus Q_{17} \oplus Q_{18} \oplus Q_{19} \oplus Q_{20} \oplus Q_{21} \oplus Q_{22} \oplus Q_{23} \oplus Q_{24} \oplus Q_{25} \oplus Q_{26} \oplus Q_{27} \oplus Q_{28} \oplus Q_{29} \oplus Q_{30} \oplus Q_{31}
 \end{array}$$

$$\begin{array}{l}
 H_8 = ROTL^9((SHR^5(XH) \oplus Q_{20} \oplus M_4) + (XL \oplus Q_{28} \oplus Q_4)) + (XH \oplus Q_{24} \oplus M_8) + (SHL^8(XL) \oplus Q_{23} \oplus Q_8) \\
 H_9 = ROTL^{10}((SHL^6(XH) \oplus SHR^6(Q_{21}) \oplus M_5) + (XL \oplus Q_{29} \oplus Q_5)) + (XH \oplus Q_{25} \oplus M_9) + (SHR^6(XL) \oplus Q_{16} \oplus Q_9) \\
 H_{10} = ROTL^{11}((SHR^4(XH) \oplus SHL^6(Q_{22}) \oplus M_6) + (XL \oplus Q_{30} \oplus Q_6)) + (XH \oplus Q_{26} \oplus M_{10}) + (SHL^6(XL) \oplus Q_{17} \oplus Q_{30}) \\
 H_{11} = ROTL^{12}((SHR^3(XH) \oplus SHL^2(Q_{23}) \oplus M_7) + (XL \oplus Q_{31} \oplus Q_7)) + (XH \oplus Q_{27} \oplus M_{11}) + (SHL^4(XL) \oplus Q_{18} \oplus Q_{11}) \\
 H_{12} = ROTL^{13}((SHL^5(XH) \oplus SHR^5(Q_{16}) \oplus M_0) + (XL \oplus Q_{24} \oplus Q_0)) + (XH \oplus Q_{28} \oplus M_{12}) + (SHR^5(XL) \oplus Q_{19} \oplus Q_{12}) \\
 H_{13} = ROTL^{14}((SHR^7(XH) \oplus SHL^8(Q_{17}) \oplus M_1) + (XL \oplus Q_{25} \oplus Q_1)) + (XH \oplus Q_{29} \oplus M_{13}) + (SHR^4(XL) \oplus Q_{20} \oplus Q_{13}) \\
 H_{14} = ROTL^{15}((SHR^5(XH) \oplus SHL^5(Q_{18}) \oplus M_2) + (XL \oplus Q_{26} \oplus Q_2)) + (XH \oplus Q_{30} \oplus M_{14}) + (SHR^7(XL) \oplus Q_{21} \oplus Q_{14}) \\
 H_{15} = ROTL^{16}((SHR^1(XH) \oplus SHL^5(Q_{19}) \oplus M_3) + (XL \oplus Q_{27} \oplus Q_3)) + (XH \oplus Q_{31} \oplus M_{15}) + (SHR^2(XL) \oplus Q_{22} \oplus Q_{15})
 \end{array}$$

Final hash H

Obr.6: Hledání vzoru poslední iterace je ekvivalentní řešení této soustavy rovnic, kde H je konstanta H^{final} a M je hledaný blok zprávy