Empowered by Innovation NEC

メッセージ認証コード PC-MAC-AES

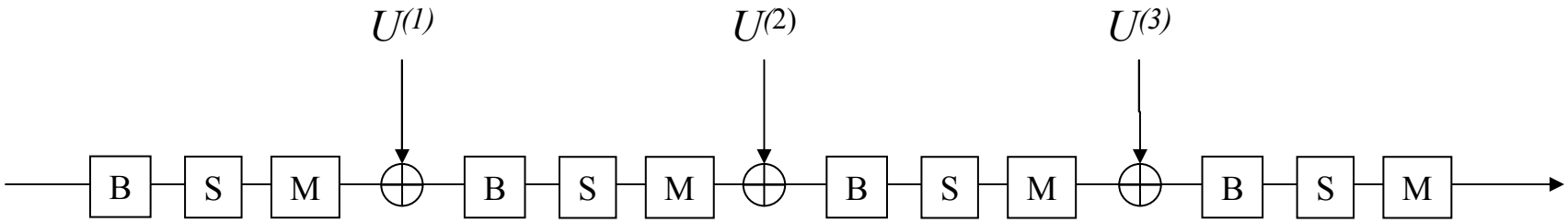
峯松 一彦（日本電気株式会社）

概要

- **ブロック暗号AESをベースとしたメッセージ認証コード**(Message Authentication Code, MAC)
  - **カウンターなどの初期値は不要**(deterministic MAC)
- CBC-MAC**同様の反復処理，ただし**AES**とその短縮段**(4**段**)による処理を組み合わせることで高速化**
  - **推奨パラメータにおいて，**CBC-MAC-AES**から**1.4**から**2**倍ほどの高速化が可能**
- **証明可能安全：**AES**が安全ならメッセージの偽造は困難**
  - CMAC-AES**などと同じ仮定，同じ安全性基準**

技術仕様

- **秘密鍵**：AES 128**ビット鍵**K, 128**ビット副鍵**L (**計**256**ビット**)
- **パラメータ**：**オーダー** d, **タグ長** π **ビット**
  - d**は正整数(**5**以下を推奨**)
  - π **は**128**以下の正整数(**64**以上を推奨**)
- **構成要素**：AES**暗号化関数** E_K, 4**段**AES**関数** G_U
  - U**は後半**3**段分の段鍵(※)：計**384**ビット**
  - U**は** E_K **を用いた鍵スケジュールにより**d**個生成する(後述)**



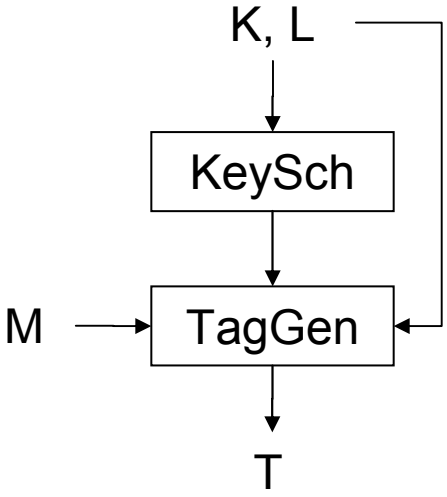
B : SubBytes
S : ShiftRows
M : MixColumns

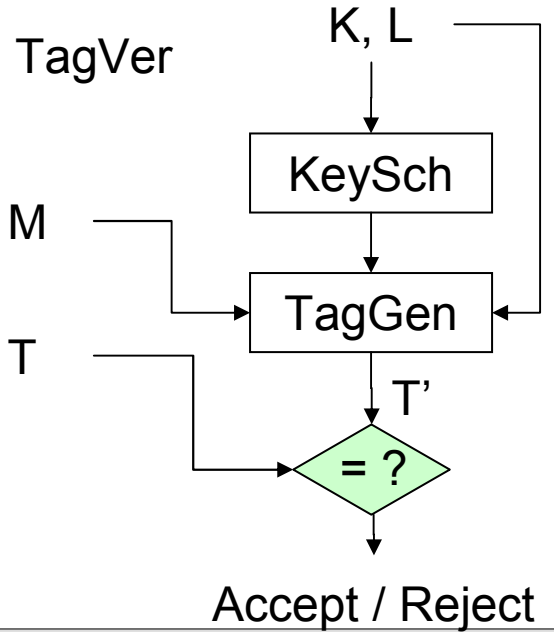
4**段**AES**関数** G_U
(U=(U(1)||U(2)||U(3)))

※AES**の正式な仕様に従えば"前半**
3段分の鍵"とも言える

全体構成

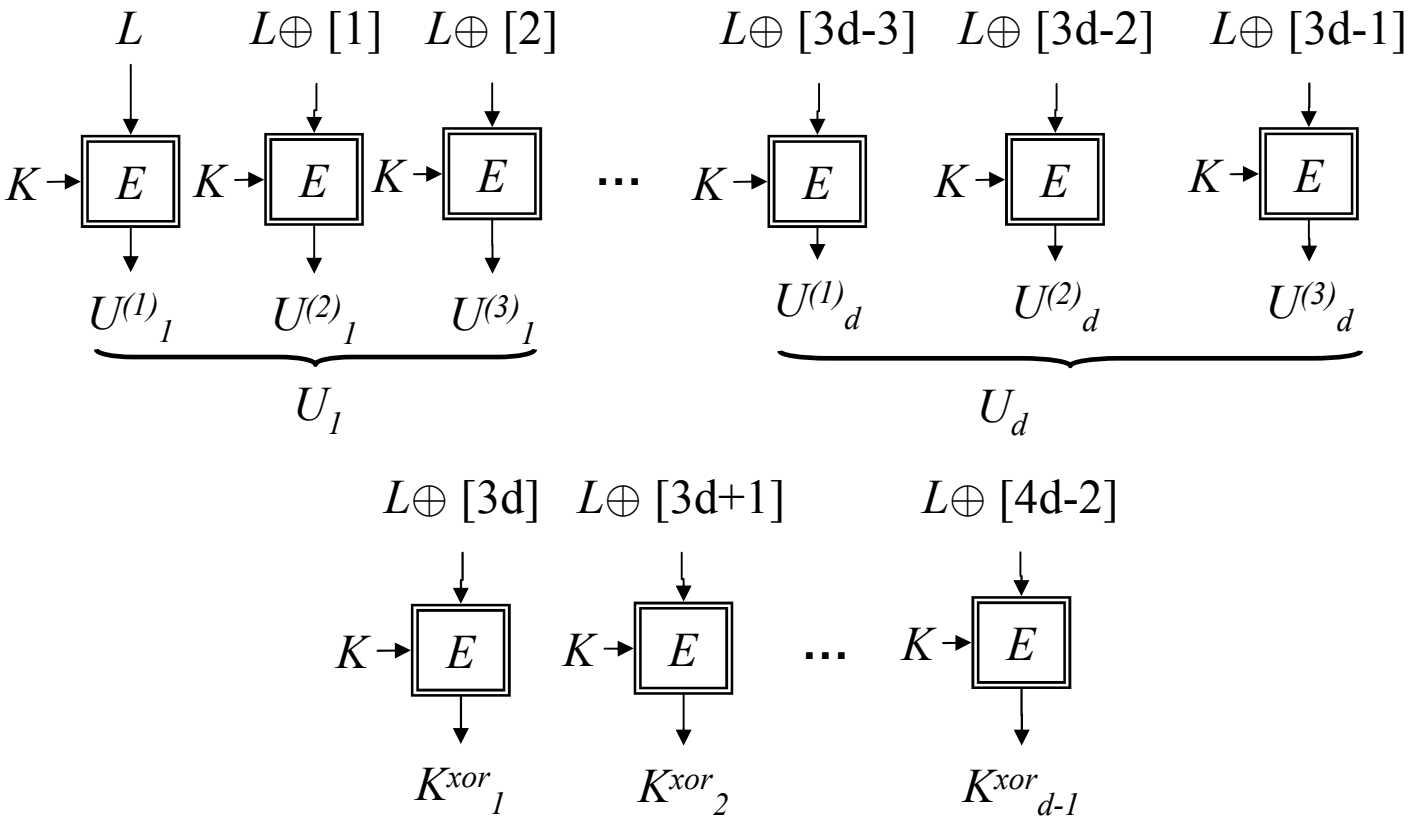
- **鍵スケジュール**KeySch
  - **タグ生成の前に行う事前処理**
  - d**個の**384**ビット鍵**U**と**d-1**個の**128**ビット鍵**K^{xor}**を生成(計**384d + 128(d-1) **ビット)**
- **タグ生成**TagGen
  - **任意長メッセージ**M**に対して**π **ビットのタグ**T**を生成**
- **タグ検証**TagVer
  - **受信したメッセージとタグのペアが正当かどうか検証**
  - TagGen**で求めたタグと受信したタグの比較**





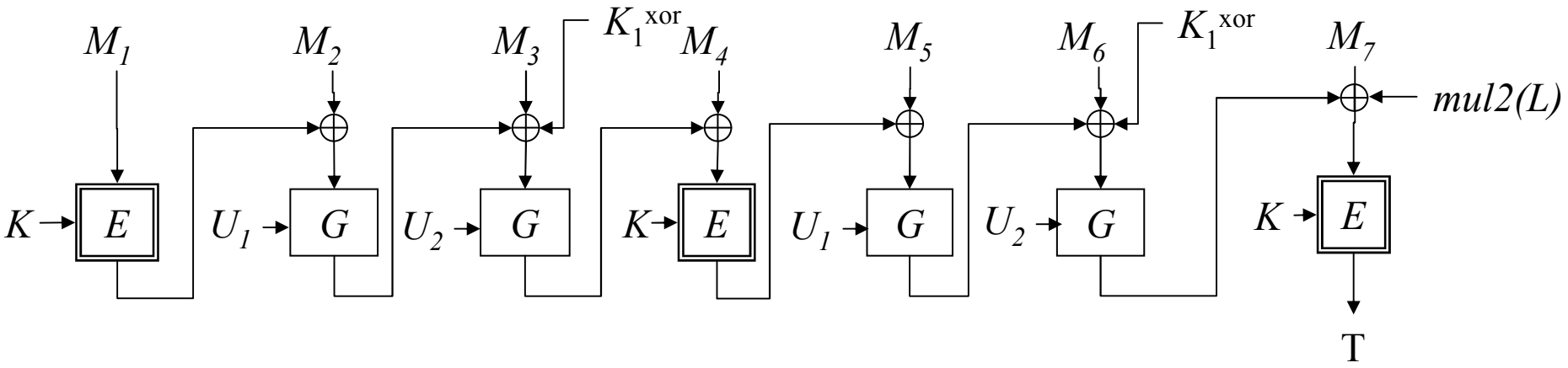
鍵スケジュール KeySch

- E_Kによるカウンターモード（Lを初期値とする）
- d個の384ビット鍵Uとd-1個の128ビット鍵K^{xor}を生成
  - 計384d + 128(d-1) ビット, d=1のときはUのみ生成



タグ生成 TagGen

- **任意長のメッセージMについてπビットタグTを計算**
- **CBC-MACと類似した反復処理，E_K, $G_{U_1},\ldots,G_{U_d}$が周期的にコールされる**
- **最終ブロックはLによるマスキングの後E_Kで暗号化**
  - **マスク形式はCMACと同様（有限体上の2倍算）**



7ブロックメッセージの処理の例
（d=2, π=128, $|M_7|$=128）

※検証手続きTagVerは通常処理なので省略

安全性評価

- AESの擬似ランダム性のみを仮定とした証明可能安全性（選択平文攻撃による偽造困難性）を有する
  - 通常のブロック暗号モードによるMACと同じ仮定，同じ安全性基準
    - IACR主催の国際学会（Fast Software Encryption 2006）にて発表済み
  - 4段AESを用いた類似のMAC（alpha-MAC, Pelican）では不可能
    - これらは証明可能安全性を有さないため，AESそのものの安全性とは独立に攻撃が発見される可能性がある
  - 安全性証明において既知の4段AESの差分確率上界を利用
- 安全性上界：おおよそ$q<<2^{56}$回の選択平文攻撃と不正な検証に対する安全性を保証
  - CMACなどでは$q<<2^{64}$
  - 攻撃報告があるが，上界を越えた質問回数で証明と矛盾しない
    - $2^{85.5}$の選択平文，2^{128}の計算量で鍵回復（Yuan et al. CRYPTO '09）

実装性評価

- 基本的にはAESの暗号モード
  - AES段関数にアクセスできれば実装可能(例:AES-NI(インテルCPU専用AES命令))
- 効率:CBC-MAC-AES(など)より1.4から最大2.5倍
  - 推奨パラメータの範囲では1.4から2.0倍の高速化
  - 少ないプラットフォーム依存性
  - 10段処理1ブロック+4段処理dブロック → 平均して4+6/(d+1)段処理/ブロック
  - 事前計算量とメモリはdについて線形に増加(事前計算量と速度のトレードオフ)
- CMAC同様のパディング処理により,短いメッセージでも良い実行効率

ソフトウェア実装

パブリックドメインのAESソース利用の実装例

- CPU : Intel Core Duo 1.66GHz
- OS : Microsoft Windows XP Professional
- **記述言語** : ANSI C
- **コンパイラ** : Microsoft Visual C++ 2008 Express Edition
- **最適化** : **速度最適化**(/O2**オプション指定**)

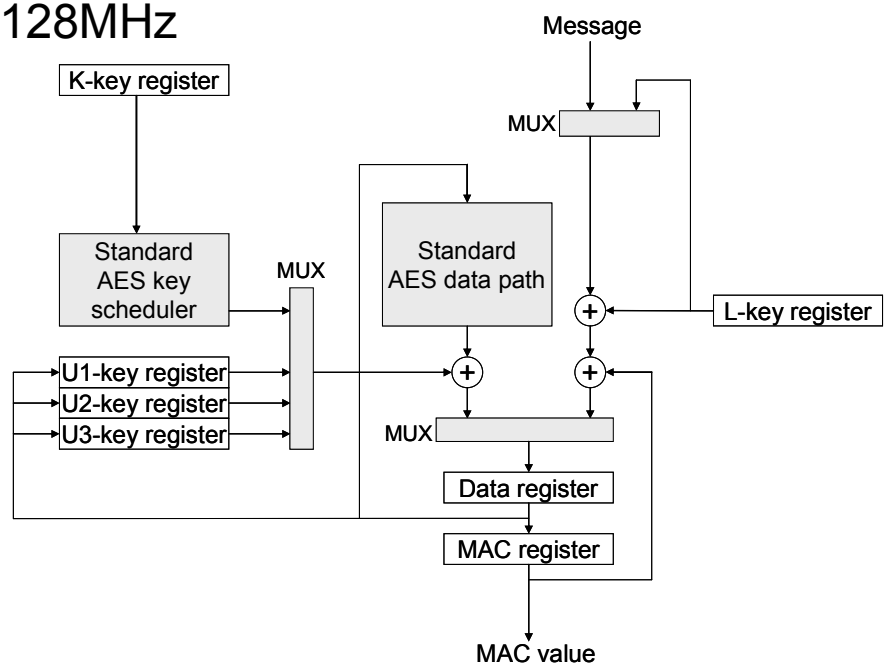
平均サイクル数(32 **ブロックメッセージ**)

MAC	KeySch	TagGen
PC-MAC-AES (d=1)	1532	10158
PC-MAC-AES (d=2)	3101	9204
PC-MAC-AES (d=3)	4601	8803
PC-MAC-AES (d=4)	6327	8614
PC-MAC-AES (d=5)	7711	8396
CMAC-AES	759	12991

※可読性と移植性重視のため, FSE'06**での実装より**
CMAC-AES**との性能比で若干劣る**

ハードウェア実装

- AESのハードウェア記述言語(HDL)を利用して効率よく実装することが可能
- 実装例(Verilog-HDL)
  - ターゲットデバイス : Virtex5 xc5vlx50-3
  - 設計ツール : Xilinx ISE ver9.2i
  - 合成オプション: 速度優先，effort high指定，そのほかはデフォルト
- 合成結果
  - スライス数 : 4356
  - 最高動作周波数 : 128MHz



ライセンス情報

民間企業等による営利目的の使用の場合を除き，無償とする予定

- 有償の場合は妥当かつ非差別的条件で実施を許諾する

開発チーム

- 技術統括：角尾 幸保（NEC）
- 設計者：峯松 一彦（NEC）
- SW実装：久保 博靖，洲崎 智保，茂 真紀，齊藤 照夫，川幡 剛嗣，中川 弘勝（NECソフトウェア北陸）、辻原 悦子(ワイ・デー・ケー）
- HW実装：森岡 澄夫（NEC）

Empowered by Innovation