A Multi-Processor NoC Platform Applied on the 802.11i TKIP Cryptosystem

January 24, 2008
Jung-Ho Lee

Bit Engineering Lab.
Contents

- Motivation
- Introduction to TKIP
- TKIP overall architecture
- Output Pattern
- Work breakdown and Mapping
- Pipelined Collaboration
- Network Interface
- Conclusion
Motivation

- Since 2001, there have been a myriad of papers on systematic analysis of Multi-Processor System on Chip (MPSoC) and Network on Chip (NoC).

- Nevertheless, we only have a few of their practical application. Till now, main interest of researchers has been to adapt NoC to the communication intensive multimedia system like H.263.

- This paper attempts to expand the domain of NoC platform to one of the wireless security algorithms (TKIP)
  - because its inter-component transaction pattern shows considerable characteristic for NoC.
TKIP Introduction

- In August, 2001, WEP of 802.11a was cracked by S. Fluhrer, I. Mantin, A. Shamir.
- Less Secure than AES, HMAC-SHA-1 based CCMP algorithm
- TKIP: Temporal Key Integrity Protocol
- New features, when compared to the WEP
  - Message Integrity Code (MIC)
  - IV (Initialization Vector) Sequencing
  - New per-packet key construction
  - Key distribution
Overall TKIP Architecture

- TA: Transmit Address
- TK: Temporal Key
- TSC: TKIP Sequence Counter

KeyMix: De-correlate IV and per-packet key

TTAK: TKIP-Mixed transmit address and key

RC4: A block encryption algorithm

MIC Key: Message Integrity Code Key

MIC: Message Integrity Code = A Cryptographic checksum

MSDU: MAC Service Data Unit
Output Pattern

MIC: Message Integrity Code using ‘Michael’ algorithm
ICV: Integrity Check Value from CRC-32

Length of IV (Initialization Vector) is expanded by 16bits (2octets)

WEP Seed [1]: (TSC1 | 0x20) & 0x7f

TSC: TKIP Sequence Counter
TSC5: Most Significant Octet ~
TSC0: Least Significant Octet
Work breakdown and mapping

- Execution cycles for processing a MSDU (2304 Byte) on a single 100 MHz ARM processor ISS (SimItARM)
  - RC4 + KeyGen: 75693 + 5203 = 80896
  - Michael: 63040
  - CRC-32: 32969

\[ Max. cy. = \frac{2304 \text{Bytes} \times 8 \text{bits/Byte}}{10 \times 10^{-9} \text{sec} \times 76 \times 10^{6} \text{bps}} = 24253 \text{cy.} \]
As a result of efficient communication architecture, which is equipped with multicast operation, three components can read a MSDU in 10.15us, as opposed to 30.29us in the case without multicast operation (@ 100 MHz bus and core).
Network Interface

- Network Interface decouples computation and communication workload by translating the language CPU can understand (AMBA/AXI signals), into the language of router (header flit followed by packet body flits), and vice versa.
Network Interface

- Message is forwarded from FIFO to request generator, on its arrival. Then the header builder generates routing path, by referring to the deterministic routing table, and forms header flit. Then the header is forwarded into the NoC domain.
Network Interface

- Following waveform shows a waveform of Verilog simulation of our NI. Simulation parameters are as follows-ARM core clock: 100MHz, NoC clock: 75MHz, Burst length: 4, flit width: 32bits, BUS width: 32bits, Burst mode: INCREMENT. This waveform says that the system delays of packet header and body flit generation processes are 40ns and 70ns, respectively.
Conclusion

- TKIP cryptosystem is mapped onto the 9x9 torus NoC.

- A compact NI, equipped with AXI ports, is proposed.

- A SystemC simulation model is written to verify the proposed system.
Thanks!

*Bit Engineering Lab.*
Appendix

Bit Engineering Lab.
Message Integrity Code (MIC)

- Cryptographic checksum used in the handshaking process.
- The MAC value protects both a message's integrity as well as its authenticity, by allowing verifiers (who also possess the secret key) to detect any changes to the message content.
- This effort is against an attack called 'message forgery'.
- In TKIP 'Michael' is used, satisfying the bleeding edge of implementability of old WEP hardware.
  - For better confidentiality, 802.11i standard recommends to use new CCMP (AES with SHA-1).

MIC generation algorithm

```
Input: Key (K0, K1) and padded MSU (represented as 32-bit words) M0...Mn-1
Output: MIC value (V0, V1)

MICHAEL((K0, K1), (M0, M1, ..., Mn))

(i, r) ← (K0, K1)
for i = 0 to n-1 do
    i ← i ⊕ Mi
    (i, r) ← t(i, r)
return (i, r)
```

'Michael' pseudocode

```
Input: (i, r)
Output: (i, r)

t(L, R)

r ← r ⊕ (j <<< 17)
l ← (j + r) mod 232
r ← r ⊕ XSWAP(l)
l ← (j + r) mod 232
r ← r ⊕ (j <<< 3)
l ← (j + r) mod 232
r ← r ⊕ (j >>> 2)
l ← (j + r) mod 232
return (i, r)
```
Initialization Vector (IV)

- **A replay attack**: A form of network attack in which a valid data transmission is maliciously or fraudulently repeated or delayed.
- To prevent a replay attack we should use a session token which is dedicated to only one transaction.
- In TKIP, the session token is IV (Initialization Vector)
  - TKIP constitutes a session token from Transmit Sequence Counter (TSC)
  - The TSC is implemented as a 48-bit monotonically incrementing counter, initialized to 1 when the corresponding TKIP temporal key is initialized or refreshed.

![Extended IV Diagram]
Integrity Check Value (ICV)

- Integrity Check Value is constructed from CRC-32 value of the decrypted payload and MIC.
- Drop packet if the ICV of \{payload, MIC\} isn’t bitwise accurate.
- MIC is applied to each MSDU, while ICV is applied to each MPDU, which is fragmented \{MSDU, MIC\}
- WEP ICV helps to prevent false detection of MIC failures that would cause countermeasures to be invoked.
Compromising Weak Key

- In WEP, Per-packet key is constructed from concatenation of IV and base-key.
- If the encrypted plain-text includes known sequence, the confidentiality of the key is easily broken.
- Therefore, TKIP adopted a multi-level key management scheme.
- Compute the per-packet key by cryptographically mixing TA (Transmit Address), IV, base key
  - Encrypt (\{Data, MIC, ICV\})
  - With Key_Generator (\{TA, IV, base_key\})
- Mixing function is a Feistel cipher designed by Doug Whiting and Ron Rivest
  - De-correlate IV and per-packet key
  - Feistel cipher:
    ▶ A sort of block cipher that combines multiple rounds of repeated operations, such as Bit-shuffling, Simple non-linear functions, Linear mixing using XOR