Crocs: Cross-Technology Clock Synchronization for WiFi and ZigBee

Zihao Yu, Chengkun Jiang, Yuan He, Xiaolong Zheng, Xiuzhen Guo
Tsinghua University
A scenario of Industrial IoT

Connect to network

Deployed in the factory

Controller

Gateway

Collector

Mobile device

Server

PC

Internet

Curtain  Clean  Feed  Fan  Drencher  NH₃  Light Temperature  Humidity  Water
Clock synchronization in different networks

- Wireless network
  - NTP
  - PTP
  - ...

- Wireless sensor networks
  - TPSN
  - FTSP
  - Glossy
  - ...

Clock calibration

- Time alignment
- Timestamp transmission
- Timestamp record
- Timestamp exchange
Clock synchronization between heterogeneous devices

- WiFi device
- ZigBee device

Timestamp exchange

The direct communication between WiFi and ZigBee is unavailable
Cross-technology communication

- **Standard**: WiFi - IEEE 802.11, ZigBee - IEEE 802.15.4

- **Channel**: Overlapping on the 2.4GHz ISM band

- **Idea**: Physical layer encoding

- **Protocol**: WeBee

- **Free side channel**: FreeBee, WiZig
Cross-technology communication

Time modulation (FreeBee)

Energy modulation (Wizig)

The timing of packets

Side channel

The energy of packets
Timestamp transmission

Timestamp: XXXX...0101 0011 0111 1011

Divided by group of bits

X X ... 0101 0011 0111 1011

Encode the digits sequentially

CTC Symbol

Temporal Modulation

0101 0011 0111

15 ms 13 ms 17 ms

40 ms

TX
Timestamp transmission

The example of energy modulation

```
1011
```

- **CTC Symbol**
- **Encoding**
- **Sampling**
- **Decoding**

```
1 0 1 1
```

‘1’: packet presence   ‘0’: packet absence
Observation of cross-technology communication

➢ Low throughput
➢ Limited robustness

Three packets with a fixed time interval pattern
Three packets with a fixed energy pattern

The simple pattern of packets may be destroyed by noise
Time alignment

Record the time $T_1$
Embed $T_1$ into the packet

Record the time $T_2$
Read $T_1$ in the packet

Sender

Receiver

Hard to decide when to record $T_2$

Clock calibration

Encode $T_1$ by intervals of the packets
Decode $T_1$ by intervals of the packets
Decoupled synchronization

- **Sender**
  - Record the time of alignment event (T1)
  - Encode T1 by intervals or energy of the packet

- **Receiver**
  - Record the time of alignment event (T2)
  - Decode T1 by intervals or energy of the packet

- **Time of receiver T2**
- **Time of sender T1**
- **Clock calibration**
Crocs beacon

Record the time of alignment event (T1)

Record the time of alignment event (T2)

Sender

Crocs beacon: A special sequence of packets

Receiver

Vulnerable to the noise

RSS of the packet

Unique to distinguish

Robust to noise

RSS sample of only one packet

RSS sample of Crocs beacon
Barker code

A finite sequence of $N$ values of +1 and -1: $a_j$ for $j = 1, 2, \ldots, N$

Property:

$$c_v = \sum_{j=1}^{N-v} a_j a_{j+v}$$

$$|c_v| \leq 1$$

for all $1 \leq v < N$.

| No of panels | Barker Code       | Barker code with different length |
|--------------|-------------------|-----------------------------------|
| 2            | +1 -1             | +1 +1                             |
| 3            | +1 +1 -1          |                                   |
| 4            | +1 +1 -1 +1       | +1 +1 +1 -1                       |
| 5            | +1 +1 +1 -1 +1    |                                   |
| 7            | +1 +1 +1 -1 -1 +1 | +1 +1 +1 -1 +1 -1 +1 -1 +1 -1     |

Autocorrelation function of Barker-7 code
Time alignment design

How to encode the Barker code:

Energy  ✗ Not robust to noise
Interval  ✔ Use two unit intervals, $t_1$ and $t_2$ to create the Barker code

Realization:
Evaluation

➢ One USRP acts as WiFi sender
➢ Another USRP generates noise
➢ TelosB mote is used as ZigBee device

USRP/N210 ➔ Crocs beacon & timestamp ➔ TelosB ➔ Noise ➔ USRP/N210

WiFi sender ➔ ZigBee receiver ➔ WiFi noise source
Beacon matching rate

The beacon matching rate with relatively low noise

The beacon matching rate with relatively high noise
Time error

Time error without clock calibration

Time error with clock calibration
Summary

◆ We design Crocs, the first cross-technology clock synchronization protocol that works for WiFi and ZigBee.

◆ We design a Barker code based beacon to trigger the event of synchronization.