Formal Analysis of Secure and Unsecured Pairing in ZigBee RF4CE

San Choosang and Steven Gordon
Sirindhorn International Institute of Technology
Thammasat University
Bangkadi, Thailand 12000
Email: san_choosang@hotmail.com, steve@siit.tu.ac.th

***Abstract*—ZigBee RF4CE is a new standard for radio frequency remote control for customer electronics. It provides more benefits than traditional infrared remote control. This paper presents a formal Coloured Petri nets (CPNs) model of pairing mechanism of ZigBee RF4CE protocol. After the model is created, state space analysis of the model is used to investigate the unexpected behaviors of pairing phase of the protocol. The CPN model not only serves as a mean to investigate the unexpected behaviors, but can be adapted to do the performance and security analysis of the protocol as well.**

***Index Terms*—formal verification, communication protocols, ZigBee RF4CE, Coloured Petri nets**

I. Introduction

The ZigBee Alliance has developed a new radio frequency standard for customer electronics (CE) called ZigBee RF4CE[1]. This standard defines a remote control network that defines a simple, robust, low-cost communication network that allows wireless connectivity in applications in CE domain. In comparison with traditional infrared remote control, ZigBee RF4CE brings more advantages such as more reliable communication, longer distance, two ways communication, and non line-of-sight. Solutions adopting the ZigBee RF4CE standard will be embedded in customer electronic device remote controls; TV, DVD, home theater, and input devices (i.e. mouse and keyboard).

One of the important service in network layer of ZigBee RF4CE is pairing. This service allows nodes, originator and recipient, in the remote control network to set up a pairing link in order to begin communication. Nodes within the network may only communicate directly with other nodes on the network if a pairing link exist between the originator and the recipient nodes.

This paper presents a formal model of pairing in ZigBee RF4CE protocol, as well as analysis results. Formal modelling and analysis of communication protocols, or protocol verification [2], [3], is important in system design stage as functional errors discovered during testing and usage are expensive to fix. The aim of our research is to verify the functional correctness of pairing in ZigBee RF4CE. The results show that under normal condition ZigBee RF4CE pairing works as required. However in the presence of packet loss unexpected states arised in that the originator and recipient are not synchronize. This is because the last acknowledgement frame has been lost. The key contributions of this paper are the development of Coloured Petri nets (CPNs) [4] model of pairing in ZigBee RF4CE, as well investigate, using state space analysis, the unexpected behaviors of the pairing. Formal modelling of pairing is the first step towards full verification of the ZigBee RF4CE.

The paper is structured as follows. Section II provides background material on ZigBee RF4CE, pairing and CPNs. Section III presents our CPNs model of pairing. Analysis results are given in Section IV, followed by conclusion in Section V.

II. Background and Related Work

In this research, CPNs is selected to model and analyse the pairing in ZigBee RF4CE because of their ability to express concurrency, nondeterminism and system concepts at different levels of abstractions. They have an underlying mathematical definition, therefore allowing for proof of static and dynamic properties of the system modelled, as well as a graphical notation with computer tool support. Section II-A and Section II-B gives some background in ZigBee RF4CE and pairing, respectively. Section II-C provides an informal definition of CPNs, as well as related work in this area.

A. ZigBee RF4CE

The Radio Frequency for Customer Electronics (RF4CE) of the ZigBee, ZigBee RF4CE[1], is a standard defined by the ZigBee Alliance which allows wireless connectivity in applications in the Customer Electronics (CE) domain. Target products are remote controls, input devices, and 3D glasses. Instead of using infrared as a medium, ZigBee RF4CE uses radio frequency that provides more benefits such as more reliable, longer distance, two ways communication and non line-of-sight. Physical and MAC layer of ZigBee RF4CE are based on IEEE 802.15.4 standard. Network layer is designed to be simple together with standard application profiles which can interface to the end user application. The ZigBee RF4CE stack architecture is shown in Figure 1.

In network layer, two services are provided. The first one is network layer data service, interfacing to the network layer data entity (NLDE) and the second one is network layer management service, interfacing to the network layer management entity (NLME). The transmission and reception of network protocol data units (NPDUs) is enabled by the network layer

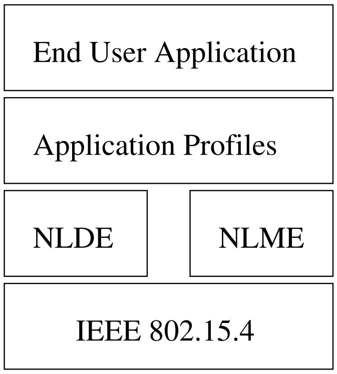

Fig. 1. ZigBee RF4CE Stack Architecture

data service, while the network layer management service permits;

- Service discovery: A procedure to find other suitable nodes that can be paired to
- Pairing: A procedure to create a pairing link between nodes to begin communication
- Unpairing: A procedure for removing a pairing link
- Node initialization: A procedure that allows node to configure the stack of itself, as a controller node or target node, and start a network
- NIB attribute manipulation: A procedure to manage (i.e. get or set values) the NIB attribute from the NLME

This research, and subsequent description, focuses on the pairing mechanism in the network layer management service. Only features relevant to the modelling/analysis tasks are described; for a full treatment of ZigBee RF4CE see [1].

B. Pairing

Before nodes in a network can communicate to each other a pairing link must exists between two nodes, originator and recipient. A pairing request is one of the services permitted by NLME to create a pairing link between nodes. The target node can choose weather to accept or reject the pair and confirms the pairing request back to the originator node.

If the target node accepts the pair and the pairing request was successful, both nodes store a pairing link in their respective pairing tables. This table stores all the necessary information that used to transmit a frame to the target node by network layer. This allows the originator node to communicate with the recipient node and the recipient node can communicate back to the originator node (format of pairing table is given in Table 49 of [1]).

Frame can be sent by a number of transmission options that can be used by an application and combined as appropriate:

- Acknowledged: Originator data is confirmed by the recipient
- Unacknowledged: Originator data is not confirmed by the recipient
- Unicast: Originator data is sent to a specific recipient
- Broadcast: Originator data is sent to all recipients
- Multiple channel: Originator attempts transmission using frequency re-acquisition mechanism
- Single channel: Originator attempts transmission on the expected channel

This paper model the pairing by using the transmission options in combination of acknowledged unicast with single channel.

To communicate between layers of an entity, ZigBee RF4CE use the concept of service primitives which have four types; Request, Indication, Response, and Confirm. Please refer to IEEE Standard 802.2 1998 edition for more detailed information.

Different command frame types are used in ZigBee RF4CE. A pair request command frame allows a device to request to pair with another device, while a pair response command frame allows a device to respond to a pair request and pass information relevant to the pairing link back to the originator. If the security is required, the key seed command frame is used to exchange security key seed values with a remote device in order to generate a security link key (the key generation procedure is described in Section 3.5.11.1 of [1]). A ping request command frame allows a device to send a ping command frame to another device and get a response. In the other hand, a ping response command frame allows a device to respond to a ping request command frame from another device.

A successful pairing attempt with security support is illustrated in Figure 2. In this message sequence chart, instances are labeled with the layer (APL for the application and NWK for the network) followed by the node type (ORG for the originator and REC for the recipient). Primitives are shown in normal style while over the air command frames are labeled in italic text.

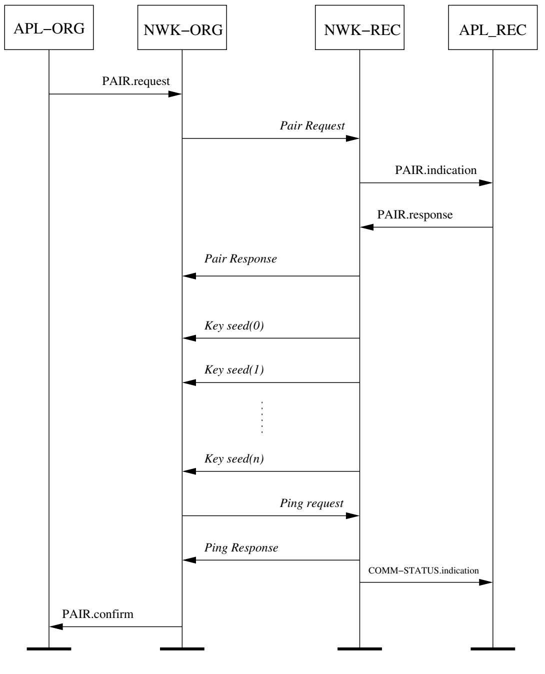

Fig. 2. Message Sequence Chart for Pairing

From Figure 2, pair request command frame is triggered by PAIR.request primitive and upon receipt the network layer of recipient indicates its application layer by PAIR.indication primitive. The recipient decides whether to accept the pair or not and respond back to the originator via Pair.response primitive and pair response command frame. If security is required, the recipient will send a number of key seed command frames. Once the originator received all of the key seed command frames, it will generate the security link key and transmit the ping request command frame encrypted with that key. On receipt of the ping request command frame, the recipient verify that frame and send ping response command frame back to the originator. The originator also verify the ping response command frame. Pair.confirm and COMM.indication primitives are sent to application layer of each side to indicate the status of pairing.

If a frame is unsuccessfully sent to another side (i.e. timeout occurs or frame errors), the pairing process is stopped and the entry in the pairing table is removed.

C. Coloured Petri nets

CPNs are directed graphs with two types of nodes: a set of *places*, P, and a set of *transitions*, T. These nodes are normally illustrated as ellipses and rectangles, respectively. Directed arcs can only be from place to transition (*input arcs*) or transition to place (*output arcs*). Directed arcs can only be between nodes of different types. Places are typed by a *colour set*. The colour set determines the type of values that can mark a place. These values are called *tokens*. A multiset of tokens in a place p is called its marking ($M(p)$) and the marking of the CPN comprises the marking of all places (M). Transitions and arcs can also have inscriptions (expressions).

The execution of a CPN consists of occurrence of transitions. A transition can occur if it is *enabled*, and it is enabled if: for all input places, sufficient tokens exist that satisfy the input arc inscriptions; and the transition inscription (or *guard*) evaluates to true.

Variables, which are local to a transition, may be used in inscriptions. The values they are bound to on occurrence of a transition give, together with the transition name, a *binding element*. When a transition occurs, tokens required by the input arcs are removed from the input places, and the evaluation of the expression on the output arcs give the tokens to be added to the output places.

CPNs have been used successfully for the modelling and analysis of a wide range of concurrent and distributed systems [5] including communication systems and protocols [6], [7], [8], [9]. [10] presents a key agreement protocol for RF4CE that the key seed information is shared between the customer device and manufacturer, but no formal model and analysis is included. As far as we are awared this paper is the first to analyse pairing of RF4CE.

III. CPNs Model of Pairing in ZigBee RF4CE Protocol

In this section, CPNs model of pairing is introduced. Modelling is performed in CPN Tools, the most popular tool for creating and analysing CPNs. Overall structure of the model and the model description are presented in Section III-A and Section III-B, respectively.

A. Model Structure

ZigBee RF4CE is modelled as a hierarchical CPN. The top-level page contains substitution transitions, which in fact represent a CPN on a sub-page. The hierarchy is shown in Figure 3. Two entities of the protocol, ORG and REC, are separated in the second level. Four sub-pages of each entity model detail of: passing the primitives between application layer and network layer; managing the network layer process such as generating frames and handling timeouts; transmit frames to another entity; and receive frames from another entity. In total there are 16 places and 56 transitions. Important declarations used in the model are in Figure 4 and due to page limited, details of *colset* starts with ► sign are not shown. colset NWK_State and Enumerations are enumerate colour set used to keep network's state name and all enumerations (list in Table 45 of [1]), respectively. Colour sets in group SERVICE PRIMITIVES are typed *record* used to record the semantics of primitives listing in Section 3 of [1]. Field is also typed *record* that used to record field's values of the command frame showing in Section 3.2.2.2 of [1]. The details of key pages are described in Section III-B—the remaining pages are presented in Appendix.

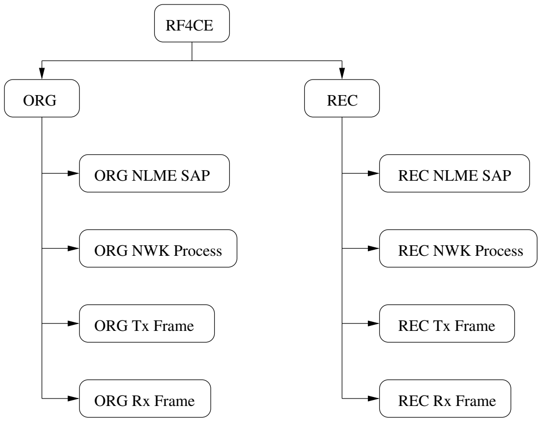

Fig. 3. ZigBee RF4CE CPN Model: Heirarchy

B. Model Description

Figure 5 shows the top-level page of the model, which illustrates the frames flow between protocol entities. The model comprises three main parts: the ORG (originator), the REC (recipient) and a bidirectional communication medium, Ch1 and Ch2, in the middle.

a) ORG: The ORG page, shown in Figure 6, consists of five places, four substitution transitions and their interconnecting arcs. The places, APL ORG and NWK ORG Primitive, are typed by the colour set PRIMITIVE and used to store the service primitives in the application layer and the network layer, respectively. Place APL ORG has an initial marking of one PAIR_REQ token indicating that the Originator is initially to request pairing service. The NWK ORG Frame place stores command frames, which is either the frames to be sent to the Recipient through place A or the frames to be received from the Recipient through place C. These places are typed by the

▼ PROTOCOL STATE
 ► colset NWK_State
▼ ENUMERATIONS
 ► colset Enumerations
▼ SERVICE PRIMITIVES
 ► colset PAIR_REQ
 ► colset PAIR_IND
 ► colset PAIR_RES
 ► colset PAIR_CON
 ► colset COMM_IND
▼ NWK CMD FIELD
 ► colset Field
▼ FRAMES
 ▼ colset Entity = with A $|B$;
 ▼ colset SignKey = STRING;
 ▼ colset EncryptKey = STRING;
 ▼ colset Frame = record
  sender : Entity $*$
  receiver : Entity $*$
  field : Field $*$
  sk : SignKey $*$
  ek : EncryptKey;
 ▼ colset Frames = list Frame;

Fig. 4. Selected ZigBee RF4CE Declarations

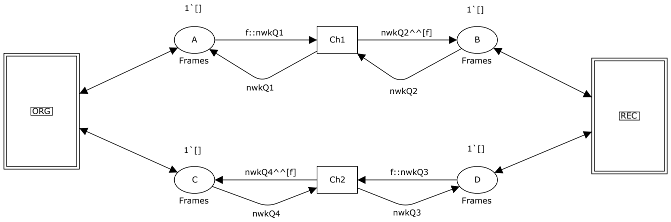

Fig. 5. ZigBee RF4CE CPN Model: Top-level page

colour set Frames and has an initial marking of an empty list(1‘[])

Substitution transition ORG NLME SAP models the transmission and receipt of the service primitives between application layer and network layer. Substitution transition ORG NWK Process models the internal mechanism of the network layer, i.e. checking the capacity of pairing table, generating network command frames, and handling timeout. Substitution transition ORG Tx Frame and ORG Rx Frame model the transmission and receipt of the network command frames to/from the Recipient entity, respectively. In the ORG NWK Process, ORG Tx Frame, and ORG Rx Frame substitution transition, there are additional two places: NWK ORG PT represents the state of pairing table; and NWK ORG STATE represents the current state of the protocol.

b) REC: For the recipient side, the mechanism of places and substitution transitions are similar to the originator side but changes the label from ORG to REC.

c) Communication Medium: The underlying communication medium is modelled as a bidirectional channel with FIFO queue consisting of four places: A; B; C; and D, and two transition: Ch1; and Ch2. The communication channels allow frames to be lost. The lost behavior can corresponds to either loss in the network (due to the congestion in the network), or discard the frames due to the checksum failure.

IV. Analysis of Pairing in ZigBee RF4CE

The pairing in ZigBee RF4CE protocol model is analysed by using state space analysis in CPN Tools. The analysis aims to investigate the unexpected behaviors of pairing.

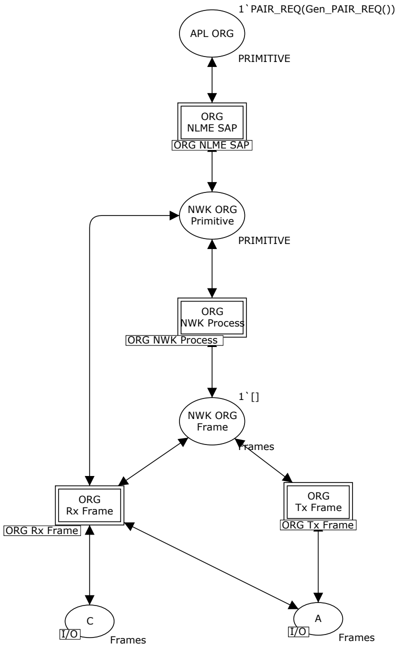

Fig. 6. ZigBee RF4CE CPN Model: ORG Page

A. Expected and Unexpected Behaviors

The goal of pairing in ZigBee RF4CE is to create a pairing link and store the informations in pairing table between nodes. So at the end of this procedure there must be an entry in pairing table that contains the informations of a paired node, an expected behavior. Conversely, if the pairing procedure ends up with the originator has an entry of the recipient in a pairing table but the recipient does not have, or vice versa. The unexpected behavior arised.

B. State Space Analysis

In the state space analysis of CPNs model, there may be certain states in which no transitions are enabled. These states are known as *dead marking*. After the state space analysis is performed in CPN Tools, we can check all dead markings whether it is an unexpected dead marking or not, described in Section IV-A.

The state space analysis of pairing model takes into account both secure mode and unsecured mode. Size of state space (number of nodes and arcs) is manageable. From the analysis results, two unexpected dead markings are discovered: one from secure mode and one from unsecured mode. The summary of the analysis result is shown in Figure 7.

	Nodes	Arcs	All Dead Markings	Unexpected Dead Markings
Unsecure Mode	452	1041	9	1
Secure Mode	826	1872	21	1

Fig. 7. Analysis Result

All of the unexpected behaviors are arised from the protocol itself due to the last acknowledgement lost problem which is a common problem of many protocols. Figure 8 shows the scenarios of unexpected behaviors of pairing.

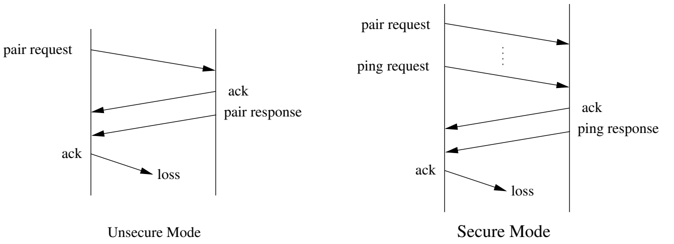

Fig. 8. Unexpected Behaviors of Pairing

V. Conclusions

ZigBee RF4CE is a new standard for remote control devices that use radio frequency. It provides many benefits compare to traditional infrared remote control. One of many services of ZigBee RF4CE is pairing, a procedure that allows two devices to communicate to each others. Key contributions of the paper are:

1) A formal CPNs model of pairing mechanism of ZigBee RF4CE protocol.
2) From state space analysis, it shows some unexpected behaviors of the pairing phase.

Future work includes verifying other services of ZigBee RF4CE, and adapted the model to support performance or security analysis of the ZigBee RF4CE.

VI. Acknowledgement

This work was supported by the National Research University Project of Thailand Office of High Education Commission.

References

[1] *ZigBee RF4CE Specification Version 1.00*, ZigBee Alliance, March 2009.

[2] G. J. Holzmann, "Design and validation of computer protocols." Englewood Cliffs, NJ: Prentice Hall, 1991.

[3] J. Billington, G. E. Gallasch, and B. Han, "A Coloured Petri net approach to protocol verification," in *Lectures on Concurrency and Petri Nets, Advances in Petri Nets*. Springer-Verlag, 2004, pp. 210–290.

[4] K. Jensen and L. M. Kristensen, *Coloured Petri Nets: Modelling and Validation of Concurrent Systems*. Springer, 2009.

[5] K. Jensen, "Coloured petri nets. basic concepts, analysis methods and practical use," in *Practical Use*. Springer-Verlag, 1997.

[6] S. Vanit-Anunchai, "Towards formal modelling and analysis of sctp connection management," in *Proceedings of the ninth Workshop and Tutorial on Practical Use of Coloured Petri Nets and the CPN Tools*, Aarhus, Denmark, October 2008.

[7] L. Liu and J. Billington, "Verification of the capability exchange signalling protocol," *International Journal on Software Tools for Technology Transfer*, vol. 9, no. 3-4, pp. 305–326, June 2007.

[8] S. Gordon, L. Kristensen, and J. Billington, "Verification of a revised wap wireless transaction protocol," in *Proceedings of the 23rd International Conference on Applications and Theory of Petri Nets*, Adelaide, Australia, 2002, pp. 182–202.

[9] S. Gordon and S. Choosang, "Verification of the flexray transport protocol for autosar in-vehicle communications," *International Journal of Vehicular Technology*, 2010.

[10] H. M. K. Han and T. Shon, "Design of secure key agreement protocol for pairing in rf4ce," in *Proceedings of International Conference for Internet Technology and Secure Transactions (ICITST)*, December 2010, pp. 1–5.

Appendix

The following figures illustrate the remaining of originator side pages (the recipient side pages are quite similar to the originator side, different in some details) of the pairing in ZigBee RF4CE protocol CPN model.

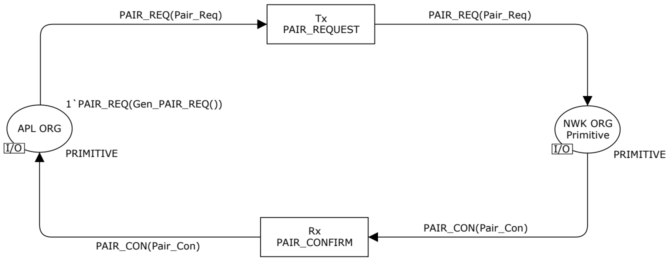

Fig. 9. Pairing CPN Model: ORG-NLME-SAP

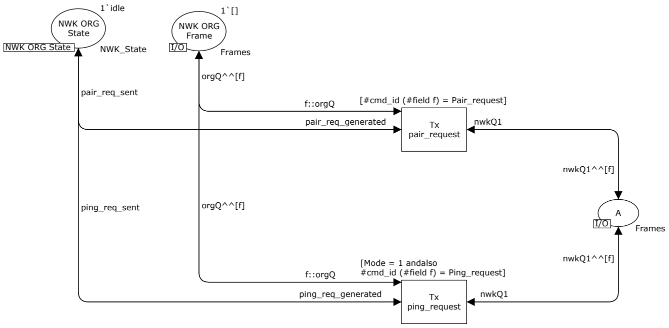

Fig. 10. Pairing CPN Model: ORG-Tx-Frame

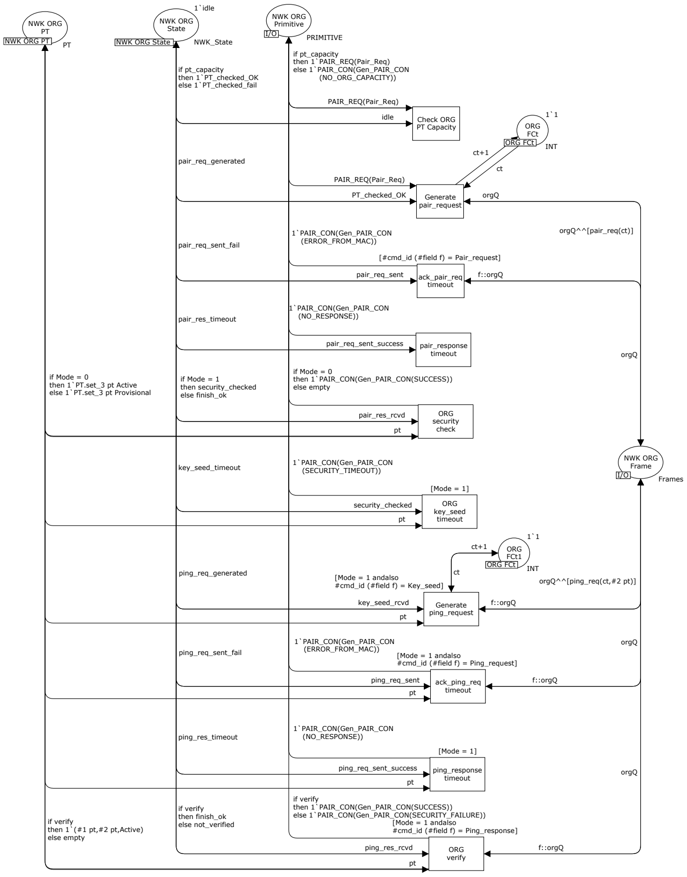
NWK ORG PT
NWK ORG PT
PT
NWK ORG State
NWK ORG State
NWK_State
1`idle
NWK ORG Primitive
I/O
PRIMITIVE
if pt_capacity then 1`PAIR_REQ(Pair_Req) else 1`PAIR_CON(Gen_PAIR_CON (NO_ORG_CAPACITY))
if pt_capacity then 1`PT_checked_OK else 1`PT_checked_fail
PAIR_REQ(Pair_Req)
idle
Check ORG PT Capacity
ORG FCt
1`1
ORG FCt
INT
ct+1
ct
pair_req_generated
PAIR_REQ(Pair_Req)
PT_checked_OK
Generate pair_request
orgQ
orgQ^^[pair_req(ct)]
pair_req_sent_fail
1`PAIR_CON(Gen_PAIR_CON (ERROR_FROM_MAC))
[#cmd_id (#field f) = Pair_request]
pair_req_sent
ack_pair_req timeout
f::orgQ
pair_res_timeout
1`PAIR_CON(Gen_PAIR_CON (NO_RESPONSE))
pair_req_sent_success
pair_response timeout
orgQ
if Mode = 0 then 1`PT.set_3 pt Active else 1`PT.set_3 pt Provisional
if Mode = 1 then security_checked else finish_ok
if Mode = 0 then 1`PAIR_CON(Gen_PAIR_CON(SUCCESS)) else empty
pair_res_rcvd
pt
ORG security check
NWK ORG Frame
I/O
Frames
key_seed_timeout
1`PAIR_CON(Gen_PAIR_CON (SECURITY_TIMEOUT))
[Mode = 1]
security_checked
pt
ORG key_seed timeout
ORG FCt1
1`1
ORG FCt
INT
ct+1
ct
ping_req_generated
[Mode = 1 andalso #cmd_id (#field f) = Key_seed]
orgQ^^[ping_req(ct,#2 pt)]
key_seed_rcvd
pt
Generate ping_request
f::orgQ
ping_req_sent_fail
1`PAIR_CON(Gen_PAIR_CON (ERROR_FROM_MAC))
[Mode = 1 andalso #cmd_id (#field f) = Ping_request]
orgQ
ping_req_sent
pt
ack_ping_req timeout
f::orgQ
ping_res_timeout
1`PAIR_CON(Gen_PAIR_CON (NO_RESPONSE))
[Mode = 1]
ping_req_sent_success
pt
ping_response timeout
orgQ
if verify then 1`(#1 pt,#2 pt,Active) else empty
if verify then finish_ok else not_verified
if verify then 1`PAIR_CON(Gen_PAIR_CON(SUCCESS)) else 1`PAIR_CON(Gen_PAIR_CON(SECURITY_FAILURE))
[Mode = 1 andalso #cmd_id (#field f) = Ping_response]
ping_res_rcvd
pt
ORG verify
f::orgQ

Fig. 11. Pairing CPN Model: ORG-NWK-Process

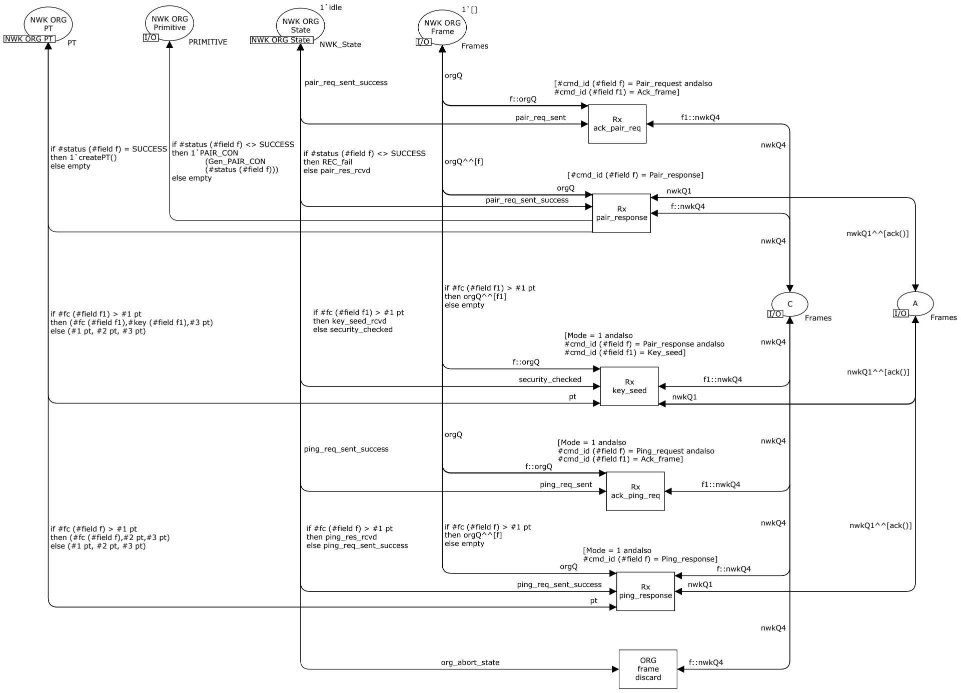
NWK ORG PT
NWK ORG PT
PT
NWK ORG Primitive
I/O
PRIMITIVE
NWK ORG State
1`idle
NWK ORG State
NWK_State
NWK ORG Frame
1`[]
I/O
Frames
pair_req_sent_success
orgQ
[#cmd_id (#field f) = Pair_request andalso #cmd_id (#field f1) = Ack_frame]
f::orgQ
pair_req_sent
Rx ack_pair_req
f1::nwkQ4
nwkQ4
if #status (#field f) = SUCCESS then 1`createPT() else empty
if #status (#field f) <> SUCCESS then 1`PAIR_CON (Gen_PAIR_CON (#status (#field f))) else empty
if #status (#field f) <> SUCCESS then REC_fail else pair_res_rcvd
orgQ^^[f]
[#cmd_id (#field f) = Pair_response]
orgQ
nwkQ1
pair_req_sent_success
f::nwkQ4
Rx pair_response
nwkQ1^^[ack()]
nwkQ4
if #fc (#field f1) > #1 pt then orgQ^^[f1] else empty
C
I/O
Frames
A
I/O
Frames
if #fc (#field f1) > #1 pt then (#fc (#field f1),#key (#field f1),#3 pt) else (#1 pt, #2 pt, #3 pt)
if #fc (#field f1) > #1 pt then key_seed_rcvd else security_checked
[Mode = 1 andalso #cmd_id (#field f) = Pair_response andalso #cmd_id (#field f1) = Key_seed]
nwkQ4
f::orgQ
security_checked
Rx key_seed
f1::nwkQ4
nwkQ1^^[ack()]
pt
nwkQ1
orgQ
[Mode = 1 andalso #cmd_id (#field f) = Ping_request andalso #cmd_id (#field f1) = Ack_frame]
nwkQ4
ping_req_sent_success
f::orgQ
ping_req_sent
Rx ack_ping_req
f1::nwkQ4
if #fc (#field f) > #1 pt then (#fc (#field f),#2 pt,#3 pt) else (#1 pt, #2 pt, #3 pt)
if #fc (#field f) > #1 pt then ping_res_rcvd else ping_req_sent_success
if #fc (#field f) > #1 pt then orgQ^^[f] else empty
nwkQ4
nwkQ1^^[ack()]
[Mode = 1 andalso #cmd_id (#field f) = Ping_response]
orgQ
f::nwkQ4
ping_req_sent_success
nwkQ1
Rx ping_response
pt
nwkQ4
org_abort_state
ORG frame discard
f::nwkQ4

Fig. 12. Pairing CPN Model: ORG-Rx-Frame