International Journal of Electrical and Computer Engineering (IJECE)
Vol. 4, No. 2, April 2014, pp. 162~168
ISSN: 2088-8708

Modeling and Simulation of NFC Logical Layer Peer-to-Peer Mode using CPN and TA

Saeed Ghasemi*, Leili Mohammad Khanli*, Mina Zolfi*, Ghader Tahmasebpour*
* Departement of Electrical and Computer Engineering, University of Tabriz, Tabriz, Iran.

Article Info

Article history:

Received Nov 17, 2013
Revised Aug 20, 2014
Accepted Aug 26, 2014

Keyword:

Application layer
Colored petri net
Logical layer control protocol
Near field communication
Timed automata

ABSTRACT

Network communication technologies have been growing explosively due to the increasing demand on faster and simpler communication; hence, providing new communication technologies is a challenging task. To make this task easy, many researchers have developed different network modeling and simulation tools with different characteristics. In this paper, simulation of Near Field Communication (NFC) logical layer control protocol is proposed to investigate efficiency of NFC device in peer-to-peer mode. For this purpose, Colored Petri Net (CPN) and Timed Automata (TA) have been used for analyses. According to the results, CPN was better than TA for simulating NFC logical layer control protocol because it could provide more details on complex communication network.

Corresponding Author:

Saeed Ghasemi,
Departement of Electrical and Computer Engineering,
University of Tabriz,
Tabriz, Iran.
Email: ghasemi.saeed@gmail.com

1. INTRODUCTION

RFID (radio frequency identification) is new type of technology that transfers power and data without any contact and stored data in silicon chip. It uses magnetic and electromagnetic technology to transfer data between data-carrying devices. The advantages of this technology in compare with other identification systems make it growing too fast and conquering huge market place [1]. Improvement of RFID resulted in development of some technologies such as Near Field Communication (NFC) that works in the range of 13.56 Mhz. NFC is similar to Infrared and Bluetooth technologies, but there are some differences. In the Infrared technology data transfers from one device to the other device when two interfaces was placed in direct line, but in NFC technology, communication occurred when two devices close to each other not exactly in the straight line. Moreover, Bluetooth was introduced the device that communicates with each other in a further distance. However, ease of use, establishing secure channel for communication, usage in different situation and also high speed that needs very low energy to transfer data and some other advantages made NFC technology growing too fast [2] and make some famous companies such as Sony, Samsung and Apple used this technology in their devices.

The NFC speed transfer rate is 106, 212 or 424 kbps [3]. NFC operates in different modes: reader/writer, peer-to-peer, and card emulation. Communication in the Peer-to-Peer mode occurs between two NFC devices and they should be worked in the active mode. When two devices want to establish connection between each other, one of them considers as an initiator or a master and the other device gets the feature of target or slave mode. The master device always initiates communication and after complete the initialization between two parts they could be able to exchange data. They could be able to transfer virtual business cards, digital photos, and some other kind of data. The architecture of the RF communication protocol in peer-to-peer mode is represented by ISO/IEC 18092 as NFCIP-1 [4]. The NFCIP-1 standardized

Analog and digital protocols that describe lower layer. NFC applications that Communicate in two directions, need LLCP layer. LLCP increases functionality provided by NFCIP-1 [5].

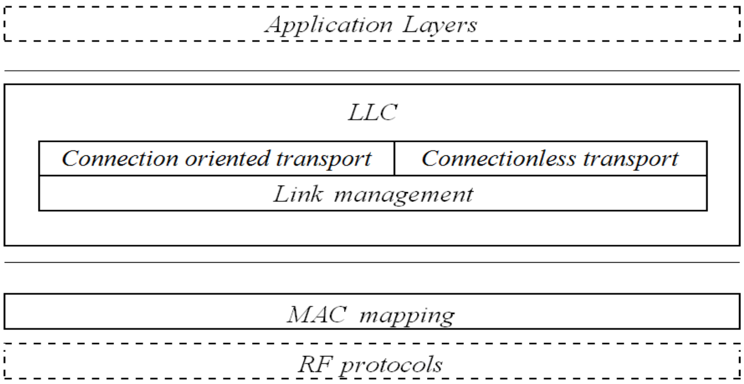

Figure 1. OSI models for NFC peer-to-peer mode [5]

The upper half of the OSI data link layer was named LLC layer and the lower half was named the Medium Access and Control (MAC) layer which act as mapping for specifies binding requirement to LLCP (see Figure 1). LLCP includes connectionless transport, connection oriented transport, link activation, supervision and deactivation and asynchronous balanced communication.

Connectionless transport provides an unacknowledged data transmission service and has minimum overhead. Connection oriented transport provides a data transmission service with sequenced and guaranteed delivery of service data units. LLCP by using link activation knows whether two NFC devices in communication range and recognize compatibility of those devices. Supervision supervises the connection to the remote peer device and deactivation deactivates link when request from one of the devices. LLCP enables Asynchronous Balanced Mode (ABM) between service endpoints in each peer devices by using a symmetry mechanism. Using ABM service endpoints may initialize, supervise, recover from errors and send information at any time [6, 7]. Communication system Implementation must be associated with analyses the Correctness and Checking the event sequence of the system.

In this article, Analysis of NFC protocol in peer-to-peer mode is investigated. LLCP layer is checked with Time Petri Nets (TPN) and Timed Automata (TA). These two methods are widely used for modeling and analysis of communication systems.

2. METHODOLOGY

2.1. Timed Automata

Timed automaton is a finite state machine extended with a set of real-valued clocks. Clocks increase in the same rate and synchronously and construct from integer value that indicate real time. The transition puts some constraints on guard, which indicate whether transition is enabled or disabled, and transitions may include some clock reset. TA models and analyzes the timing behavior and checks the safety and liveliness of the system. TA definition would be simple but powerful and considered as formal language theory. Hence, it has closure properties, decision problems, and subclasses and also it contains nondeterministic and deterministic transition structures.

In Timed Automaton A is a tuple $(L, l_0, X, \Sigma\varepsilon, E, Inv, F, R)$ where: L is a finite set of locations; l_0 is the initial location; X is a finite set of positive real-valued clocks; $\Sigma\varepsilon = \Sigma \cup \{\varepsilon\}$ is a finite set of actions and ε is the silent action; $E \subseteq L \times C(X) \times \Sigma\varepsilon \times 2X \times L$ is a finite set of edges, $e = \langle l, \gamma, a, R, l' \rangle \in E$ represents an edge from the location l to the location l′ with the guard γ, the label a and the reset set $R \subseteq X$; $Inv \in C(X)\ L$ assigns an invariant to any location [8]. They restrict the invariants to conjuncts of terms of the form $x \leq r$ for $x \in X$ and r c N and $\leq \in \{<, \leq\}$. $F \subseteq L$ is the set of final locations and $R \subseteq L$ is the set of repeated locations [8].

The semantics of a timed automaton A = (L, l0, C, Σε, E, Act, Inv, F, R) is a timed transition system $SA=(Q, q_0, \Sigma\varepsilon, \rightarrow, F', R')$ with $Q=L\times(R\leq 0)\ X$, $q_0 =(l_0, 0)$ is the initial state, $F' =\{(l, \nu) | l \in F\}$ and R′ ={(l, v)|l∈R}, and → is defined by: the discrete transitions relation (l, v) → (l, v) iff ∋(l, γ, a, R, l) ∈ E, v′ = v[R → 0] and Inv (l′)(v′) = tt.

2.2. Time Petri Net

Time Petri net is used formal method for simulation, modeling and validation of concurrent and distribution system and has graphical notation. Formal methods are mathematically techniques determined syntax and semantics for the specification and verification of software and hardware systems. Formal specification is used to remove the ambiguities in the informal specification. Formal verification defines syntax and semantics of software and hardware systems and checks the correctness of system. TPN includes transition for present events that could occur and places to show condition of the system.

Time Petri Net N is a tuple (P, T, Σε, I, O, M_0, Λ, I, F, R) where: P is a finite set of places and T is a finite set of transitions and P ∩ T = Ø; Σ is a finite set of actions I ∈ (NP)T is the backward incidence mapping; O ∈ (NP)T is the forward incidence mapping; M_0 ∈ NP is the initial marking; Λ : T → Σε is the labeling function; I : T → I(Q≥0) associates with each transition a firing interval; R ⊆ NP is the set of final markings and F ⊆ NP is the set of repeated markings [8].

A marking M of a TPN is a mapping in NP and M (p_i) is the number of tokens in place p_i. A transition t is enabled in a marking M iff M ≥ •t. We denote En (M) the set of enabled transitions in M. To decide whether a transition t can be fired we need to know for how long it has been enabled: if this amount of time lays into the interval I (t), t can actually be fired, otherwise it cannot. The key point in the semantics is to define when a transition is newly enabled and one has to reset its clock.

In this article, it is worked on intermediate semantics of TPN and used Color Petri Net (CPN) software to implement system according to TPN. CPN tool provides simulation, state space analysis and visualize the behavior of the system [9]. Also, Uppaal is the other toolbox that use in this article and it is build base on TA theory to verified real-time systems. It provides integer variables, structured data types, user defined functions, and channel synchronization [10].

3. EXPREMENT AND RESULTS

The NFC peer-to-peer mode architecture includes an application layer, logical link control protocol, MAC mapping and RF layer. The RF layer has build upon the ISO 18092, which provide NFCIP-1 NFC Interface and Protocol. NFCIP-1 describes active communication mode and passive communication mode. Any application could be used the NFC peer-to-peer protocol to communicate between two devices. The MAC layer present on the ISO/IEC 18092 specification. This standard MAC layer is usable only for LLCP layer of NFC peer-to-peer mode. A set of mappings would be use as interface to transfer data between MAC layers and LLCP. Also, the binding that each mapping made are specifies LLCP to one especially MAC Layer.

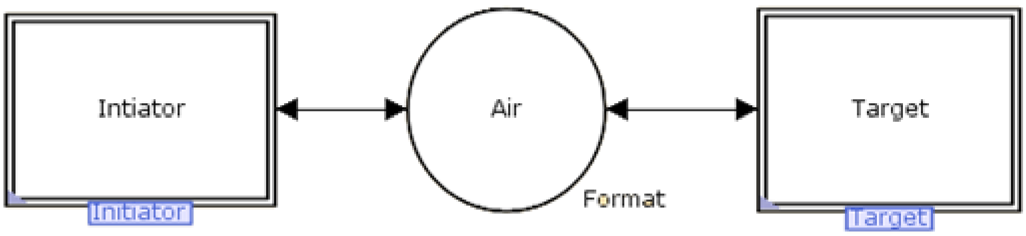

Figure 2. Communication of LLCP Protocol

NFC device transfer data between two devices, initiator starts communication and target responses to initiator request. Figure 2 illustrates the general view of the LLCP protocol connection.

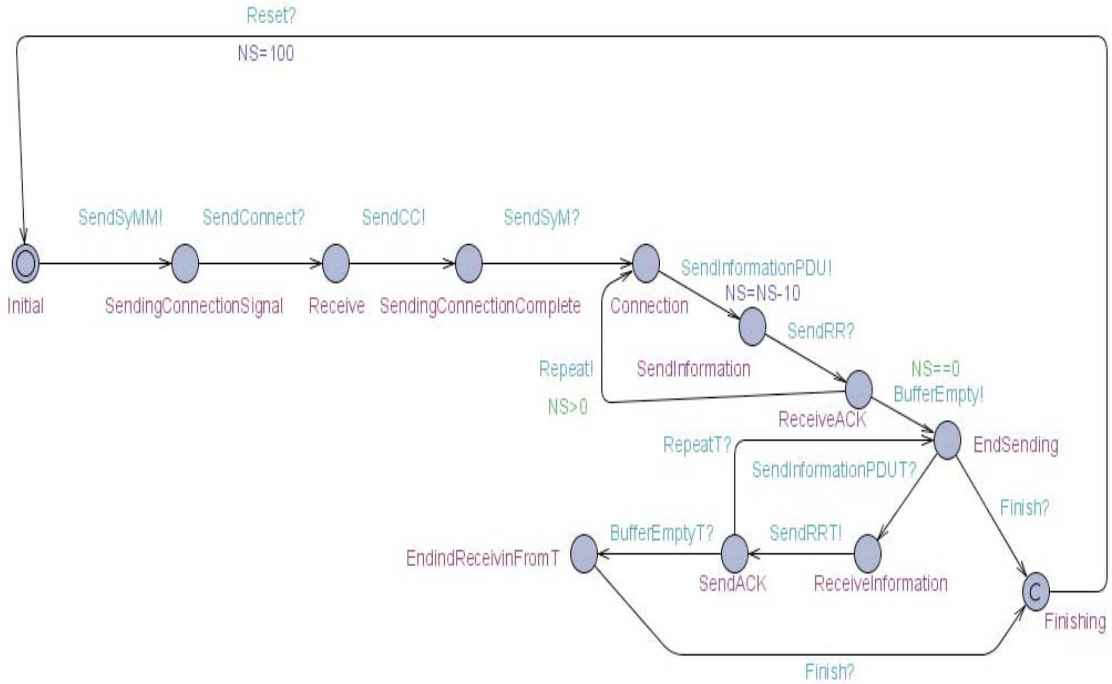

Figure 3. Communication of LLCP layer with TA method

Generally, peer-to-peer mode session split in to 4 phases:

- Initialize phase: The initiator send request packet periodically and if target receive it, send back acknowledge. On the other hand, this phase has a mechanism to avoid conflict when initiator wants to establish communication with single target.
- Activation phase: The initiator begins a new session by detecting target and send specific message to active communication between target and initiator.
- Exchange phase: Data transfers from one side to other side and contain sending and receiving procedure.
- Deactivation phase: After transferring data between devices. Initiator sends deactivate message to target and receive release response.

The communication between initiator and target is automatically started when the distance between the two NFC devices would be small enough, but connection in the MAC layer stop when the distance between them is not sufficient or the operation is end.

Following the initialization and anti-collision procedure, the initiator device sends the Attribute Request ATR_REQ command. ATR_REQ command contains device id, the version that can support, link timeout and some other parameter that are created by MAC component. Target MAC component receives and verifies ATR_REQ if the sequence is equal to the NFC Forum LLCP magic number and other field have appropriate values, it responds the initiator by sending ATR_RES. Then, the initiator and target MAC layer should be able to accept LLC PDUs.

The LLCP packet conations three critical fields, DSAP (Destination Service Access Point), SSAP (Source Service Access Point) and PTYPE (Protocol data unit type field). There are sixteen types of LLCP packets, identified by PTYPE values ranging between 0 and 15. Here, it was used only the part of PDUs such as Symmetry (SYMM), Connect (CONNECT), Disconnect (DISC), Disconnected Mode (DM), Connection Complete (CC), Information (I), Receive Ready (RR), Receive Not Ready (RNR).

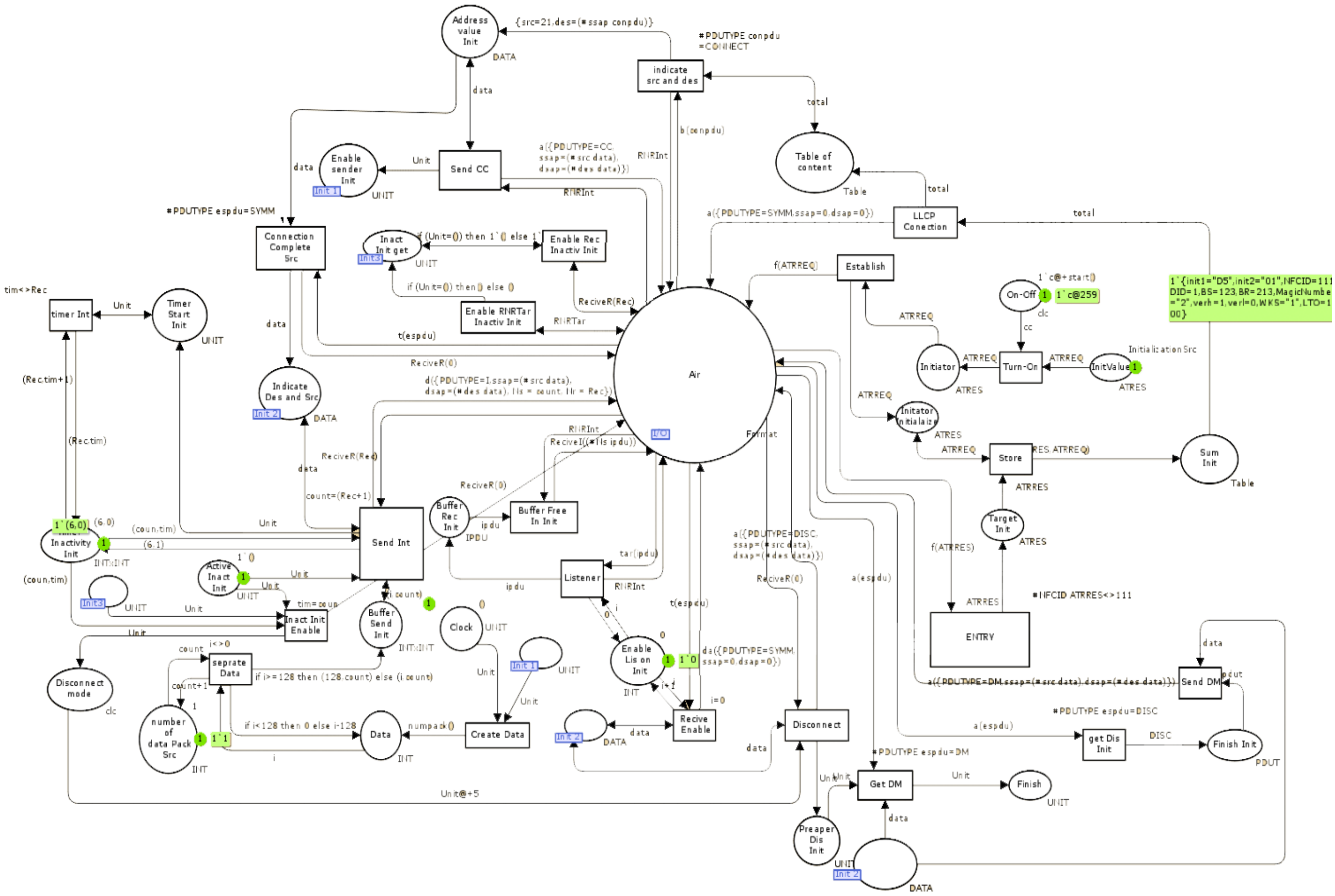

Figure 4. Initiator side of NFC LLCP Protocol with Petri Net method

The LLC link activation procedure starts after the MAC link activation procedure has been successfully completed, then the local LLC component in a peer device able to execute LLCP. LLC PDUs transmits SSAP, DSAP and PTYPE in both sides and sets source and destination address of the device and prepares systems for next phase of communication. Either part could be able to establish a data link connection by sending the CONNECT PDU. Note that, the local LLC might not send more than one CONNECT PDU with the same SSAP address value before getting CC or DM PDU with the DSAP value that the same as the SSAP value in the pending CONNECT PDU. If the LLC receives a CC PDU with a DSAP value equal to the SSAP value of a sent but does not yet acknowledged CONNECT PDU, it sets the state variables send state variable (V (S)), receive state variable (V(R)), send acknowledgement state variable (V (SA)), receive acknowledgement state variable (V (RA)) to zero and sends CC PDU to destination. In the other hand, the DM PDU returns to other side if the local LLC receives a CONNECT PDU but could not process the connection request. In the information transfer phase LLC uses the Send Sequence Number (N (S)), Receive Sequence Number (N (R)), V (S), V(R), V (SA), RW(R) and V (RA) variables to control sending and receiving data between each side, if V (S) is equal to the V (SA) plus the remote receive window size RW(R), the LLC not be able to send an I PDU on that data link connection. When the LLC sends an I PDU, N (S) field should equal to the current value of the V (S) and the N (R) set to the current value of the V (R). After sending the I PDU, the LLC increments the V (S) by one.

In the receiver part, I PDU is received with the N (S) and it would be equal to V (R), the LLC passes the service data unit contained in the information field to the service access point and increments its by one V(R). Moreover, even one I PDU exists for sending on this data link connection, the received I PDU will be acknowledged. If there is no I PDU is available and the data link connection is not presently in the receiver busy condition, the LLC sends an RR PDU, otherwise, if the data link connection is presently in the receiver busy condition, the LLC sends an RNR PDU. In the device that receiving Acknowledgement, the N(R) field value indicates that I PDUs numbered up through N(R)-1 have been received correctly and This N(R) value becomes the new value of the send V (SA).

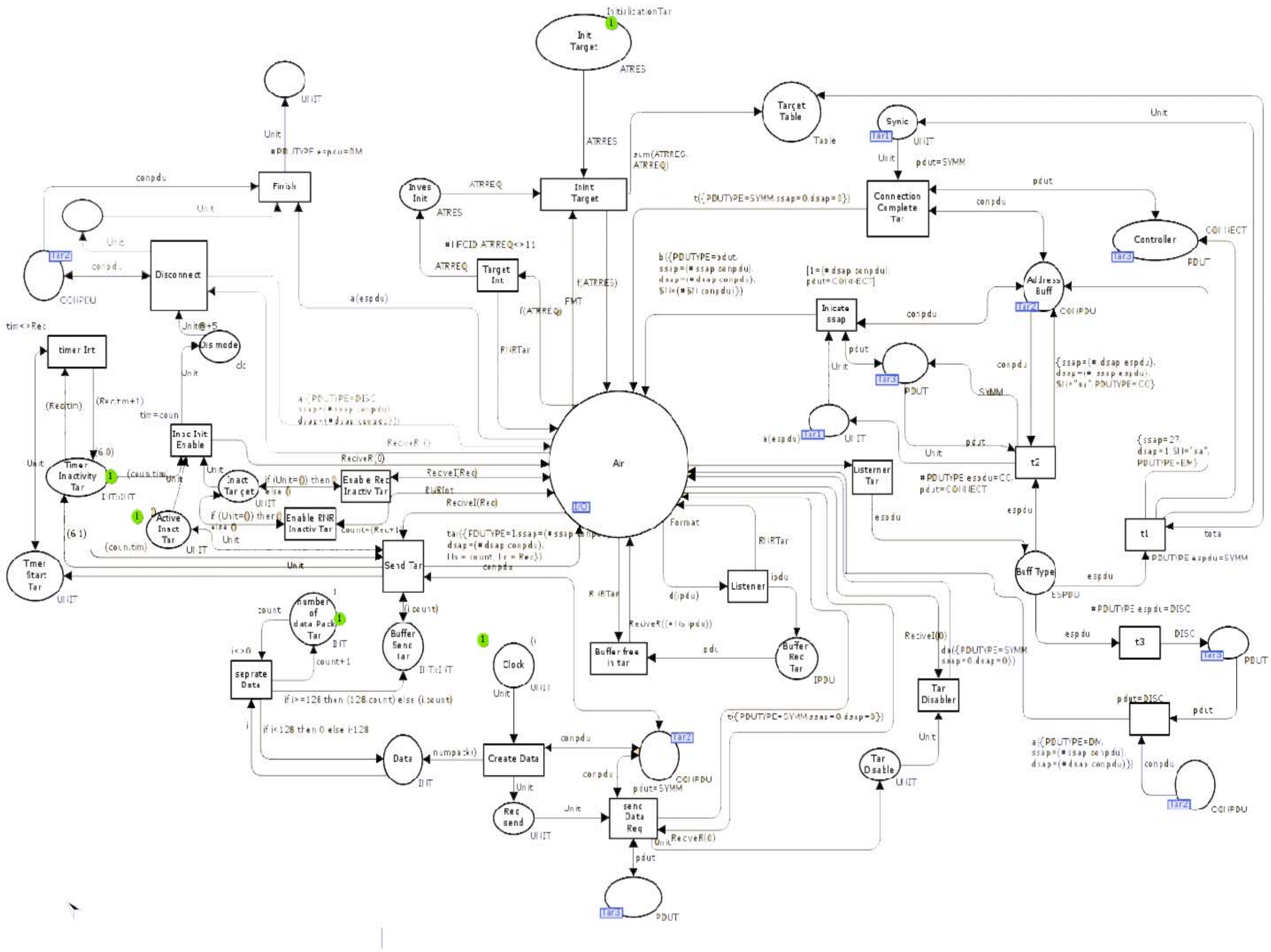

Figure 5. Target side of NFC LLCP Protocol with Petri Net method

The LLC link deactivation phase used to terminate normal operation between the local and the remote LLC. Each side of LLC may initiate link deactivation that occurs base on request from the remote LLC, or as the result of a link timeout. When executing the link deactivation procedure, the local LLC should pass disconnect indications to the service layer and then execute the MAC link deactivation. Deactivation phase initiates by sending a DISC PDU and after sending the DISC PDU, the LLC that initiates the link deactivation, deactivate the local link.

In this section, we illustrate the architecture of LLCP with CPN and UPPAAL tools [11]. LLCP is essential for devices that transfer data in bi-directional communications.

In TA method we talk about the overall view of the system, analyze each side and investigate on the behavior of initiator and target sides (Figure 3). The initiator side sends connection signal and waits for receiving connection signal from target side. Then initiator sends connection complete and receives symmetry command from other side. They continue sending data until no more data exist to send.

Petri Net helps us to study logical layer control protocol (LLCP) with more detail. So, we consider LLCP with 5 major procedures. We illustrate the manner of procedure communication in the figures 4 and 5 in more details.

4. DISCUSSION

This section provides a comparison between two models and talk about capability and ability to verify system.

Table 1. Comparison between Time Petri Net and Time Automata

	Easy to learn	Workflow systems	Dynamic System	Concise for Uncertainly	Support high-level graphical for aggregate system	Deadlock
Time Petri Nets	No	Powerful	Weak	Good	No	Never
Timed Automata	Yes	Weak	Weak	Not Good	No	Sometimes

4.1. Timed Automata

Timed automata are used to model formal system, which has powerful theory and also has a verification tools named UPPAAL. Timed automata are useful formalism method for design industrial system and the UPPAAL software was enough mature for modeling time automat of system. Also, the syntax and semantic of UPPALL is the same as C language and make it proper approach for modeling and verifying the system need time.

This modeling has some weakness on workflow system that need create new condition dynamically. In the workflow system, it would be better to use color petri net (CPN) tools and TPN. Briefly, TPN more concise than TA and the size of TPN model for uncertainly large system might be smaller than TA. Time automata also does not support high-level graphical model for aggregate systems. In addition, TA use constant variable explicitly and it cause sometime deadlock happen in the system and inconsistency emerge. However, inconsistency might not be able to occur in TPN.

4.2. Time Petri Nets

Time Petri Nets models diverse number of real time system specially network and workflow system. It helps to avoid the misbehavior that occurs in construction of the system and bugs reveal before implementation. Unlimited TPN is not proper for automatic verification and not practical to implement in real system, so we used limit TPN for verification. In addition, TPN could not create new state dynamically because the structure of net is limited and it is not flexible. At the end, the syntax of TPN is base on ML language and sometime realizes the concept of the system would be hard.

5. CONCLUSION

In this study, two popular models for simulating Near Field Communication (NFC) logical layer control protocol were introduced and compared the features of them. Although, these two methods have the same features, it seems Timed Petri Net (TPN) is more powerful to simulate the behavior of the system. Here, we face with complex network system that has a number of states; we can use TPN to model system and analyze system according to that design. Moreover, two famous modeling tools for TPN and Timed Automata (TA) were introduced. UPPAAL was a tool for modeling TA system and Color Petri Net (CPN) was developed for design system base on TPN.

REFERENCES

[1] Hardgrave, Bill C, Aloysius, John, Goyal, Sandeep. “Does RFID improve inventory accuracy? A preliminary analysis”. *International Journal of RF Technologies: Research and Applications*. 2009; 1(1): 45–56.

[2] Mario Munoz Organero, Samuel King Opoku. “Using NFC Technology for Fast-Tracking Large-Size Multi-Touch Screens”. *Cyber Journals: Multidisciplinary Journals in Science and Technology, Journal of Selected Areas in Telecommunications (JSAT)*. 2011.

[3] Klaus Finkenzeller. *”RFID HANDBOOK”*. John Wiley & Sons, L, THIRD EDITION. 2010.

[4] Vedat coskun, Kerem Ok and Busra ozdenizci. *“Near Field Communitcation”*. first published 2012, John Wiley & Sons Ltd

[5] *Logical Link Control Protocol, Technical Specification NFC ForumTM LLCP 1.1 ,NFCForum-TS-LLCP_1.1.* 2011-06-20.

[6] Vedat Coskun, Kerem Ok, Busra Ozdenizci. *Near Field Communication (NFC): From Theory to Practice*. Wiley. 2011.

[7] Kevin Curran, Amanda Millar, Conor Mc Garvey. *Near Field Communication.* Institute of Advanced Engineering and Science. Apr 24 , 2012.

[8] Bérard B, F Cassez, et al. “Comparison of the Expressiveness of Timed Automata and Time Petri Nets”. *Formal Modeling and Analysis of Timed Systems Lecture Notes in Computer Science*. 2005; 3829: 211-225.

[9] K Jensen, LM Kristensen. *Coloured Petri Nets, DOI 10.1007/b95112, Springer-Verlag Berlin Heidelberg.* 2009.

[10] Akshay Uttama Nambi SN, Prabhakar TV, Jamadagni HS, Kishan Ganapathi, Pramod BK, Rakesh CM, and Sanjay Naik R. Near Field Communication. *Applications and Performance Studies*. 2012.

[11] Larsen KG, Pettersson P, Yi W. *Uppaal in a Nutshell. International Journal on Software Tools for Technology Transfer*. 1997; 1(1–2): 134–152.