A Review on Formal Verification of Basic Algorithms in Time Triggered Architecture

Sheena N[1]., Shelbi Joseph[2]
Division of IT
School of Engineering, CUSAT
Kerala, India
sheena.cemp@gmail.com[1], shelbi@cusat.ac.in[2]

Suresh Kumar N.
Department of Computer Science & Engineering
College of Engineering & Management Punnapra
Kerala, India
cnsuresh2000@gmail.com

***Abstract*—In recent days time triggered architecture play a vital role in safety-critical real time applications such as avionics, automotive industry and many other industrial cyber physical systems. Start-up, group membership, clock synchronization and clique avoidance are some of the basic algorithms of time triggered architecture. Now-a-days the design of time triggered systems is much more difficult because they depend on the temporal properties. So formal verification methods are used for real time safety critical applications. In this paper demonstrates the different modeling and verification methods adopted for assuring the properties of time triggered algorithms in time triggered communication protocols like TTA, TTCAN and FlexRay.**

Keywords—Time-triggered architectures; Safety-critical systems, TTA; TTCAN; FlexRay; Verification

I. INTRODUCTION

Advancement in technology leads to the communication between the cyber world and the physical world. Cyber Physical Systems (CPS)[2] refers to the integration of information-processing systems and physical world through some communication network. Examples of the cyber physical systems include the autonomous vehicular systems, space dragon systems, quad copters, medical monitoring systems etc [16]. As it is cross domain architecture, design of CPS faces many challenges in its temporal behavior. Another challenge is the lack of adequate models in computing, which makes reliable real time performance difficult. Developing a fault tolerant safety critical are the major task of cyber physical system (CPS) applications that require the integration of both physical time dynamics and discrete time dynamics, the time-triggered architecture effectively meets this purpose.

Design of real time safety critical time triggered systems are very difficult due to the limitation in manual testing and verification processes. This leads to the development of formal verification methods where models are built and the correctness properties are applied on it to verify the properties. Different approaches to model continuous and discrete time dynamics are timed automata, calendar based model, time out based model and synchronous calendar models.

978-1-5386-0796-1/18/$31.00 ©2018 IEEE

This paper is organized as follows. Section 2 describes the time triggered architecture. The basic algorithms TTA and its properties are explained in section 3. Section 4 analyses the different verification methods adopted in various time triggered algorithms. Section 5 concludes with future challenges.

II. TIME TRIGGERED ARCHITECTURE

Safety critical Cyber Physical Systems are embedded with large number of computing systems that share common characteristics. The time-triggered architecture (TTA) [7] provides such a framework for the domain of large distributed embedded real-time systems in high-dependability environments. TTA treat the real or physical time as the first order quantity. In the design of TTA the basic building block is a node which consists of a processor with memory, an input output subsystem, time triggered communication controller, operating system, and the relevant application software[8]. Nodes are connected to form a cluster via communication channels. The nodes access the communication channel by time division multiple access scheme (TDMA).

TTA is implemented based on three different network topologies viz, bus, star and hybrid topology. In bus topology all the nodes are connected to one or more than one buses whereas in star topology the nodes are connected through star coupler. Hybrid topology is the combination of star and bus topology.

III. ALGORITHMS OF TTA

Time triggered architecture depends on a number of basic algorithms like Start-up, group membership, clock synchronization and clique avoidance for its proper operational purposes:

A. Start-up algorithms

Start-up algorithms [5][17] setup values for the local clocks in such a way that the nodes get synchronized quickly when the

power is on. The four different states of the state machine diagram of node are initial, listen, sync and coldstart. i-frame and cs-frame are the different messages in startup algorithm. The nodes enters into the initial state when system is powered on and moves to listen state at different point of time. Depending upon the timing information extracted from the messages sent by nodes in sync or coldstart state, nodes in listen state change their state to sync or coldstart. From the coldstart state, the node broadcast messages based on the global and local timings and moved to listen or coldstart itself. Different protocols use different strategies to implement the startup algorithms. The main properties of startup algorithms are:

Safety:-The active state of two nodes will always agree on slot time.

Liveness :- All nonfaulty nodes will reach the sync state.

Timeliness: - All nonfaulty nodes will reach the sync state within a bounded time. Timeliness is the bounded liveness property.

B. Group membership algorithms

Instead of clock synchronization which tolerates only single fault, group membership algorithms keep track of the record of nonfaulty nodes. Each node in the time triggered architecture keep a record of nonfaulty nodes currently available. Due to asynchronous failures, the membership view of the nodes may change, group membership algorithm delete the faulty processors from the membership view. The properties [11][6] of group membership algorithms are:

Agreement: - All the nonfaulty nodes must be in the same membership view of the algorithm

Validity: - Ensures the nonmembership of the faulty processors from the consistent view of the algorithm.

Self-diagnosis:- A faulty node should remove itself from the membership when it identifies its fault.

Termination: - If the output of the processes should not be changed if the input of the process are same.

Safety: - The committed views of the processes should not be changed.

C. Clock synchronization

Clock synchronization is very important in distributed systems because different nodes sent messages. Due to the real time safety critical application, clock synchronization has significant role in proper functioning of the time triggered architecture. To minimize the faults the clocks of all processes to be kept close together.

D. Clique avoidance

Cliques are groups nodes communicating with each other. Clique avoidance [10] deals with communication between cliques. In time triggered architecture it's very difficult to create a subset of nodes that communicate each other in such a way that communication between cliques must be eliminated.

Since most of the applications of the time-triggered architectures are safety critical, verification is of great importance. While formal verification techniques are known to suffer from lack of scalability, they have been successfully applied to small scale systems to get useful insight about the design of distributed algorithms. Though the comprehensive verification of a large-scale system like a time-triggered architecture is beyond our reach with the current formal methods technologies, the efforts on the verification of algorithms demonstrate that formal method-based verification techniques can be very useful to acquire confidence on small but critical components of safety-critical systems.

IV. VERIFICATION OF TIME TRIGGERED ALGORITHMS

Here we discuss the modeling and verification of startup, group membership, bus guardian, click avoidance and clock synchronization algorithms of time triggered architecture performed by different researchers in the past. Properties of each basic algorithm are identified. in the survey we depict which model checkers are used, what properties are verified and what is the performance of each algorithm in TTA, TTCAN and FlexRay time triggered protocols. Here we discuss the performance, verification and modeling of each time triggered algorithm.

A. Start-up algorithm

Modeling and verification of startup protocols for time-triggered architectures is a very hot research area in the recent years witnessed by works in [14][5][17] for TTA, [9][4] for FlexRay protocol and [15][13]for TTCAN protocols as shown in table 1 . There are three works on verification of TTA startup algorithm in the literature survey. In paper [17] verification is performed using SAL model checker and for the propagation of time discrete time modeling is used. The startup algorithm of TTA described in [17] verified safety, liveness and timeliness properties and also simulated exhaustive fault. This ensures a safe and timely system startup in the presence of a faulty node or a faulty hub. The limitation of this work is that properties are verified only for discrete time and dynamic and dense time modeling is not demonstrated. In paper [5] verified only safety property of startup protocol of TTA in dense time using SAL model checker. In this work, continuous time dynamics have been modeled by using Calendar based model. They have modeled the protocol with a reliable active hub, but a single node may be Byzantine faulty, and can attempt to broadcast arbitrary frames at any time. This work can be extended to verify timeliness and liveliness properties. Safety and liveness properties of startup algorithm of TTA was verified in paper [14] using spin model checker timeout and calendar based discrete time models are built for the verification process. Here a canonical finitary reduction technique was used to reduce the infinite state space of timeout and calendar based transition systems to a finite state space.

In this survey we compared the works in papers [4] and [9] for FlexRay startup algorithms. Instead of common properties of startup algorithms these two works has its own properties and they are verified using some modeling techniques. Paper [4] describes the discrete time modeling of startup phase of FlexRay whereas in [9] timed automata models are verified

Table 1 : Comparison of modeling & verification of startup algorithms in TTA,TTCAN & FlexRay protocols

TT Protocol	*Reference Paper*	*Properties Verified*	*Model checker used*	*Time Model*	*Performance*	*Scope for future works*
TTA	Steiner et.al[17]	Safety, Liveness, Timeliness	SAL	Discrete time modeling	3, 4 or 5 nodes	Verification of safety and correctness properties of protocols used in aerospace and automotive domain, safety critical in-vehicle applicatons to develop a fault tolerant systems
TTA	Dutertre et.al[5]	Safety	SAL	Timeout and calendar based dense time modeling	Upto 10 nodes	
TTA	Saha et.al.[14]	Safety, Liveness	SPIN	Timeout and calendar based discrete time modeling	Upto 9 nodes	
FlexRay	Malinsk et.al.[5]	Verified some queries	UPPAL	Discrete time modeling using Timed automata & CTL		
FlexRay	Cranen [4]	Verified some scenarios like 2 nodes, silent node, deaf node, restting node, noisy channel	mCRL2 language	Discrete time modeling		
TTCAN	Saha et.al.[15]	Safety, Liveness	SAL	Synchronous calendar based continuous time modeling	Upto 9 nodes	
TTCAN	Saha et.al.[13]	Recurrence, Invariance, Response, Guarantee	SPIN	5 modes of discrete time models are built		

Table 2 : Comparison of modeling & verification of Group membership algorithms in TTA,TTCAN & FlexRay protocols

TT Protocol	*Reference Paper*	*Properties Verified*	*Model checker used*	*Time Model*	*Performance*	*Scope forfuture work*
TTP	Pfeifer [11]	Validity, Agreement, Self-diagnosis	PVS theorem prover	PVS theorem prover		Interaction between the group membership and clique avoidance algorithms and their joint behavior under various fault hypotheses, The mutual interdependence of clock synchronization and group membership algorithms
FlexRay	Rossett et.al. [18]	Agreement, Validity	UPPAL	Timed automata	3, 4 or 5 nodes	
FlexRay, TTCAN	Barbosa et.al. [3]	Agreement, Integrity, Accuracy, Self-exclusion, Liveness	SPIN	PROMEELA	4 to 7 nodes	
SLMP	Mikael et.al. [1]	Soundness, Completeness, Freshness	PRISM	Discrete time Markov chain model	3 nodes	

using mCRL2 language. Former one verified the system consisting of two coldstart nodes and one non-coldstart node, using a few different settings for a number of FlexRay parameters. The system is checked for deadlock and start up normally under different conditions

TTCAN protocol analysis in [13] built a discrete time modeling using Spin model checker. TTCAN ensure the transmission of all safety related messages because the communication is based on a time master. Five models of TTCAN is modeled using DT-Spin basic model, basic model with remote request, basic model with error handling, basic model with fault confinement and basic model with fault tolerance. Specification of properties like Progression of Time (POT), Starvation Freedom Properties, Data Consistency (DC), Bus Off (BO), Bus Access Method (BAM), Automatic Retransmission (AR), Remote Data Request (RDR), Error Signaling (ES1), Error Signaling (ES2), Fault Tolerance (FT1) and Fault Tolerance (FT2) that are gleaned from the properties of the original CAN protocol from ISO 11898 Progression are verified. In [15] time is modeled as synchronous calender verification is performed using SAL model checker. Verification of safety and liveness properties are successfully completed in that.

B. Group membership

Here we surveyed 4 works under group membership algorithm as shown in table 2. Among the properties of group membership algorithms only agreement property is satisfied in all the works. But the termination, self-diagnosis and safety properties are considered only in some of them. The performance of the algorithm is restricted to maximum 5 nodes and state explosion is major a issue. So some finite state reduction techniques may be applied to improve the performance of the algorithm.

C. Clock synchronization algorithms

Two works related to clock synchronization algorithm are described in table 3. Properties like correct relay and integrity are verified in [19] and abstract properties are modeled in [12] using theorem proving approach. Rigorous verification of clock synchronization has a significant importance in real time distributed systems.

Table 3 : Comparison of modelling & verification of clock syncronization & bus guardian in TTA & FlexRay protocols

TT Protocol	*Algorithm*	*Reference Paper*	*Properties Verified*	*Model checker used*	*Time Model*
FlexRay	Bus guardian, Clock synchronization	BoZhang[19]	Correct delay, Integrity	Isbelle/HOL	Theorem prover
TTP	Clock synchronization	Pfeifer et. l.[12]	Abstract properties	PVS system	PVS theorem prover

V. CONCLUSION

Time triggered architecture and its basic algorithms bring up a significant role in cyber physical systems such as avionics, automobile industry, and industrial process control systems etc. As these systems are safety critical an exhaustive verification of the algorithms are great importance. In this paper we surveyed how the basic algorithms like startup, group membership, click avoidance and clock synchronization are verified under different conditions in TTA, TTCAN and FlexRay protocols. The correctness properties of these algorithms are modeled and verified under different parametric conditions and time domain. The interaction between the basic algorithms viz, interaction between the group membership and clique avoidance algorithms and their joint behavior under various fault hypotheses, the mutual interdependence of clock synchronization and group membership algorithms can be considered in future.

REFERENCES

[1] Mikael Asplund and L Jakob. Specification , Implementation and Verification of Dynamic Group Membership for Vehicle Coordination. pages 321–328, 2017.

[2] Radhakisan Baheti and Helen Gill. Cyber-physical Systems. (1), 2011.

[3] Raul Barbosa and Johan Karlsson. Formal specification and verification of a protocol for consistent diagnosis in real-time embedded systems. SIES'2008 - 3rd International Symposium on Industrial Embedded Systems, pages 192–199, 2008.

[4] Sjoerd Cranen. Model checking the FlexRay startup phase. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 7437 LNCS:131–145, 2012.

[5] Bruno Dutertre and Maria Sorea. Modeling and Verification of a Fault-Tolerant Real-Time Startup Protocol Using Calendar Automata. Formats/Ftrtft, 3253:199–214, 2004.

[6] Massimo Franceschetti, Student Member, and Jehoshua Bruck. A Group Membership Algorithm with a Practical Specification. 12(11):1190–1200, 2001.

[7] Hermann Kopetz and Günther Bauer. The Time-Triggered Architecture. 91(1), 2003.

[8] Reinhard Maier, Günther Bauer, Georg Stöger, and Stefan Poledna. Time-Triggered Architecture: a Consistentcomputing Platform. page 10, 2002.G.

[9] Jan Malinský and Jií Novák. Verification of flexray start-up mechanism by timed automata. Metrology and Measurement Systems, 17(3):13, 2010.

[10] Paul Milbredt, Audi Ag, Martin Horauer, and Andreas Steininger. An Investigation of the Clique Problem in FlexRay.

[11] Holger Pfeifer. FORMAL VERIFICATION OF THE TTP GROUP. 2000.

[12] Holger Pfeifer and Detlef Schwier. Formal Verification for Time-Triggered Clock Synchronization. 12(January):207–226, 1999.

[13] Indranil Saha Suman Roy. A Finite State Analysis of Time-Triggered CAN (TTCAN) Protocol Using Spin. 2007 International Conference on Computing: Theory and Applications (ICCTA'07), 2007.

[14] Indranil Saha, Janardan Misra, and Suman Roy. Timeout and Calendar based Finite State Modeling and Verification of Real-Time Systems. pages 284–299, 2007.

[15] Indranil Saha and Suman Roy. Modeling and Verification of TTCAN Startup Protocol Using Synchronous Calendar.

[16] Teodora Sanislav and Liviu Miclea. Cyber-Physical Systems - Concept , Challenges and Research Areas . 14(2):28–33, 2013.

[17] Wilfried Steiner, John Rushby, and K Abteilung. Model Checking a Fault-Tolerant Startup Algorithm : From Design Exploration To Exhaustive Fault Simulation. pages 1–20.

[18] Valério RossetPedro F SoutoFrancisco Vasques. Formal Verification of a Group Membership Protocol Using Model Checking. On the Move to Meaningful Internet Systems 2007: CoopIS, DOA, ODBASE, GADA, and IS, 0(0), 2007.

[19] Bo Zhang. On the Formal Verification of the FlexRay Communication Protocol. 6th International Workshop on Automated Verification of Critical Systems, (AVoCS):184–189, 2006