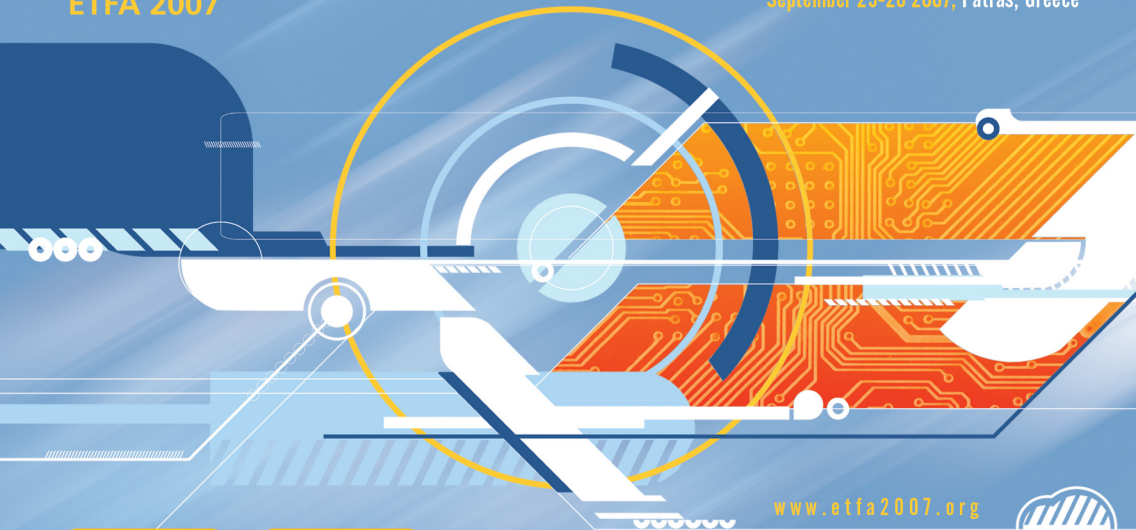




12th IEEE International Conference on Emerging Technologies and Factory Automation

September 25-28 2007, Patras, Greece



www.etfa2007.org



Welcome

Committees

Table of Contents

Keynotes

Technical Program

Author Index

Help

Search

© 2007 IEEE. Personal use of this material is permitted. However, permission to reprint/republish this material for advertising or promotional purposes or for creating new collective works for resale or redistribution to servers or lists, or to reuse any copyrighted component of this work in other works must be obtained from the IEEE.



TECHNICAL PROGRAM

Session: T1.1	Room: I4	Wednesday, Sep. 26, 11:30 - 13:00
Communication in Automation Systems: Possibilities and Limitations		
Chairing: Juergen Jasperneite, Alexander Fay		

<i>Life-cycle Oriented Data Access for a Maintenance Framework</i> Andreas Gössling, Martin Wollschlaeger	1
<i>OWL Based Information Agent Services for Process Monitoring</i> Antti Pakonen, Teppo Pirttioja, Ilkka Seilonen, Teemu Tømmila	9
<i>Limits of Increasing the Performance of Industrial Ethernet Protocols</i> Juergen Jasperneite, Markus Schumacher, Karl Weber	17
<i>Prediction of End-to-End Deadline Missing in Distributed Threads Systems</i> Patricia Della M��a Plentz, Carlos Montez, R��mulo Silva de Oliveira	25
<i>Performance Evaluation and Prediction of a Bluetooth Based Real-Time Sensor Actuator System in Harsh Industrial Environments</i> Uwe Meier, Stefan Witte, Kai Helmig, Michael Hoeing, Markus Schnueckel, Hermann Krause	33
<i>Formalised specification of a test tool for safety related communication</i> Mathias M��hlhause, Christian Diedrich, Matthias Riedl, Daniel Schmidt	38

Session: T6.1	Room: I 10	Wednesday, Sep. 26, 11:30 - 13:00
Embedded Model Control and Emerging Ttechnologies		
Chairing: E. Canuto, J. Ospina		

<i>Embedded Model Control: principles and applications. Part I</i> Enrico Canuto, Luis David Prieto	45
<i>Embedded Model Control: principles and applications. Part II</i> Enrico Canuto, Luis David Prieto	53
<i>Embedded Model Control: sub-microradian horizontality of the Nanobalance thrust-stand</i> Enrico Canuto, Fabio Musso, Luca Massotti	61
<i>Emerging technologies in the ESA Science and Earth Observation Programme</i> Luca Massotti, Enrico Canuto	69
<i>Multilayer control of an optical reference cavity for space applications</i> Enrico Canuto, Jos�� Ospina, Angelo Ripa, Fabio Musso, Franco Alasia, Fabrizio Bertinetto, Marco Bisi, Paolo Cordiale	77

Session: T3.1	Room: I 11	Wednesday, Sep. 26, 11:30 - 13:00
Scheduling and Resource Management		
Chairing: Bjorn Anderson, Luigi Sassoli		

<i>Sensitization of Symbolic Runs in Real-Time Testing Using the ORIS Tool</i> Enrico Vicario, Luigi Sassoli, Laura Carnevali	85
<i>Virtual Execution Environment for Real-Time TDL Components</i> Claudiu Farcas, Wolfgang Pree	93
<i>Deriving Exact Stochastic Response Times of Periodic Tasks in Hybrid Priority-driven Soft Real-time Systems</i> Giordano Kaczynski, Lucia Lo Bello, Thomas Nolte	101
<i>Uniprocessor Scheduling Under Time-Interval Constraints</i> F��bio Rodrigues de la Rocha, R��mulo Silva de Oliveira, Carlos Montez	111
<i>Resource Management for Dynamically-Challenged Reconfigurable Systems</i> Muhammad Hasan, Sotirios Ziavras	119
<i>Reliable Scheduling of a Distributed Real-time Embedded Application Considering Common Cause Failures</i> Thanikesavan Sivanthi	127

Session: SS2.1 **Room: I 12** **Wednesday, Sep. 26, 11:30 - 13:00**

Emerging Technologies to Enable Integrated Manufacturing and Service Systems (IMSS)

Chairing: Goh Kiah Mok, Cristian Vasar

- A Rapid Configurable Embedded Development Framework***
Kiah Mok Goh, Benny Tjahjono, Anton J.R. Aendenroomer 135
- The Wireless Sensor Networks for Factory Automation: Issues and Challenges***
L.Q. Zhuang, K.M. Goh, J.B. Zhang 141
- A Service Based Multi-Agent System Design Tool for Modeling Integrated Manufacturing and Service Systems***
Han Yu, Zhiqi Shen, Chunyan Miao, Jing Wen, Qizhen Yang 149
- Model-based Monitoring and Failure Detection Methodology for Ball-nose End Milling***
Sheng Huang, Kiah Mok Goh, Yoke San Wong, Geok Soon Hong, Kah Chuan Shaw 155
- Fault Detection Methods for Frequency Converters Fed Induction Machines***
Lucian Mihet, Octavian Prosteian, Ioan Filip, Iosif Szeidert, Cristian Vasar 161
- Semantic Enhancement and Ontology for Interoperability in Integration of Design Information Systems***
Qizhen Yang, Chunyan Miao 169

Session: T2.1 **Room: I4** **Wednesday, Sep. 26, 14:30 - 16:00**

Wireless Industrial Communications

Chairing: Christos Koulamas, Julian Proenza

- Development and Performance Evaluation of an Antenna Diversity Module for Industrial Communication based on IEEE 802.15.4***
Andreas Vedral, Thomas Kruse, Jörg Wollert 177
- Reasoning about communication latencies in real WLANs***
Claudio Zunino, Gianluca Cena, Ivan Cibrario Bertolotti, Adriano Valenzano 187
- Monitoring, Controlling and Configuring of ZigBee Wireless Household-Electric Network through Remote Virtual Interface***
Antonio Heronaldo de Sousa, Ana Teruko Yokomizo Watanabe, Luiz Ricardo Lima, Bruno Henrique Kikumoto de Paula 195
- Fast Hand Off for Mobile Wireless Process Control***
Orazio Mirabella, Lucia Lo Bello, Antonino Raucea, Michele Brischetto 202
- The Use of Clustered Wireless Mesh Networks in Industrial Settings***
Urban Bilstrup, Katrin Bilstrup, Bertil Svensson, Per-Arne Wiberg 211
- Requirements, Drivers and Analysis of Wireless Sensor Network Solutions for the Oil & Gas Industry***
Stig Petersen, Paula Doyle, Christian Salbu Aasland, Svein Vatland, Trond Michael Andersen, Dag Sjong 219

Session: T8 **Room: I 10** **Wednesday, Sep. 26, 14:30 - 16:00**

Computational Intelligence in Automation

Chairing: E. Man, J. Tar

- Texture Recognition for Frog Identification***
Flavio Cannavo', Boray Tek, Izzet Kale, Giuseppe Nunnari 227
- Modeling Supply Chain's Reconfigurability using Fuzzy Logic***
Bin Ma, Laura Xu, Roland Lim 234
- On the Application of Recurrent Neural Network Techniques for Detecting Instability Trends in an Industrial Process***
Eva Portillo, Itziar Cabanes, Marga Marcos, Asier Zubizarreta 242
- Intelligent Control in Automation Based on Wireless Traffic Analysis***
Kurt W. Derr, Milos Manic 249

<i>On-line Identification of Hybrid Systems Using an Adaptive Growing and Pruning RBF Neural Network</i>	257
Tohid Alizadeh, Karim Salahshoor, Mohammad Reza Jafari, Abdollah Alizadeh, Mehdi Gholami	
<i>Fault diagnosis and fuzzy logic decision for stochastic timed automata</i>	265
Ghada Beydoun, Zemouri Ryad	

Session: T5.1	Room: I 11	Wednesday, Sep. 26, 14:30 - 16:00
Architectures, Methods and Technologies for Enterprise Integration		
Chairing: Rei Itsuki, Jose Lastra		

<i>Impact of the Delay of Subcontracting in a Context of Integrated Maintenance: Analytical Approach</i>	273
Mohammed Dahane, Christian Clementz, Nidhal Rezg	
<i>Development of Communications Protocols between Manufacturing Execution System and Production Equipment</i>	280
Satoshi Iwatsu, Yuji Watanabe, Kiminobu Kodama	
<i>On Ontology Mapping in Factory Automation Domain</i>	288
Corina Popescu, Jose L. Martinez Lastra	
<i>Integration of SOA-ready Networked Embedded Devices in Enterprise Systems via a Cross-Layered Web Service Infrastructure</i>	293
Stamatis Karnouskos, Oliver Baecker, Luciana Moreira Sa de Souza, Patrik Spiess	
<i>An Information Management System in Inter-organization Supply Chain by Secure RFID Tag</i>	301
Kazuhiro Kawashima, Norihisa Komoda	
<i>An Approach for Integrating Real and Virtual Production Automation Devices Applying the Service-oriented Architecture Paradigm</i>	309
Daniel Cachapa, Armando Colombo, Martin Feike, Axel Bepperling	

Session: SS2.2	Room: I 12	Wednesday, Sep. 26, 14:30 - 16:00
Planning and Integration Technologies for Manufacturing and Service Systems		
Chairing: Angle Goh, He Wei		

<i>Web 2.0 Concepts and Technologies for Dynamic B2B Integration</i>	315
Chong Minsk Goh, Siew Poh Lee, Wei He, Puay Siew Tan	
<i>Composing OWL-S Web Services</i>	322
B.D. Tran, P.S. Tan, A. Goh	
<i>An Investigative Approach on Improving B2B Interactions and Communication Capabilities for Enterprise Integration using Web 2.0 Technologies</i>	330
Wei He, Puay Siew Tan, Chong Minsk Goh, Siew Poh Lee, Eng Wah Lee	
<i>Common Capacity Modelling for Multi-site Planning: Case Studies</i>	336
F.Y. Wang, T.J. Chua, T.X. Cai, L.S Chai	

Session: T1.2	Room: I4	Wednesday, Sep. 26, 16:30 - 18:00
IT in the Design Process of Automation Systems		
Chairing: Alexander Fay, Juergen Jasperneite		

<i>Introducing the Modeling and Verification process in SysML</i>	344
Marcos Vinicius Linhares, Rômulo Silva de Oliveira, Jean-Marie Farines, François Vernadat	
<i>Automated PLC Software Generation Based on Standardized Digital Process Elements</i>	352
Martin Bergert, Jens Kiefer, Christian Diedrich, Thomas Bär	
<i>A rule format for industrial plant information reasoning</i>	360
Till Schmidberger, Alexander Fay	
<i>Software Quality Measures to determine the Diagnosability of PLC Applications</i>	368
Mohammed Bani Younis, Georg Frey	

Engineering Components for Flexible and Interoperable Real-Time Distributed Supervision and Control Systems	
Sandro Andrade, Raimundo Macêdo	376
Interactively Configurable Framework for Industrial Agents	
Sebastian Theiss, Joern Ploennigs, Volodymyr Vasyutynskyy, Jens Naake, Klaus Kabitzsch	384
A Linear Programming Based Heuristic for Solving a Two-Stage Flowshop Scheduling Problem	
Ewa Figielska	392

Session: T6.2	Room: I 10	Wednesday, Sep. 26, 16:30 - 18:00
----------------------	-------------------	--

Industry/Bank Automation

Chairing: R. Vilanova, F. Koumboulis

A Survey on Automata-based Methods for Modelling and Simulation of Industrial Systems	
Vasilis Deligiannis, Stamatis Manesis	398
On the automatic generation of timed automata models from ISA 5.2 diagrams	
Luiz Paulo Barbosa, Kyller Gorgônio, Antonio Marcus Lima, Angelo Perkusich, Leandro Silva	406
Bank Note Classification Using Neural Networks	
Sigeru Omatu, Michifumi Yoshioka, Toshihisa Kosaka	413
Feedforward Control for uncertain systems. Internal Model Control approach	
Ramon Vilanova	418

Session: T4	Room: I 11	Wednesday, Sep. 26, 16:30 - 18:00
--------------------	-------------------	--

Intelligent Sensors and Sensor Networks

Chairing: Pedro M. Ruiz Martinez, Paul M. Havinga

Utilising Noise Effects on Infrared Pattern Reception for Position Estimation on a Grid Plane	
Nikos Petrellis, Nikos Konofaos, George Alexiou	426
A Tight Lower Bound for Art Gallery Sensor Location Algorithms	
Andrea Bottino, Aldo Laurentini, Luisa Rosano	434
Implementation and Evaluation of a Hybrid Network utilizing TinyOS-based systems and Ethernet	
Angelos Anastasopoulos, Dimitrios Tsitsipis, Spilios Giannoulis, Stavros Koubias	441
SensorScheme: Supply Chain Management Automation using Wireless Sensor Networks	
Leon Evers, Paul Havinga, Jan Kuper, Maria Lijding, Nirvana Meratnia	448
METATRO: A Real Time RFID enabled haulage monitoring system for perishable comestibles	
George Asimakopoulos, Spiros Loubros, Vassilis Triantafillou	456
Performance Evaluation of Instrumentation Sensor Network Design Using a Data Reconciliation Technique Based on the Unscented Kalman Filter	
Karim Salahshoor, Mohammad Reza Bayat, Mohsen Mosallaei	460
50 Ways to Build your Application: A Survey of Middleware and Systems for Wireless Sensor Networks	
Ioannis Chatzigiannakis, Georgios Mylonas, Sotiris Nikolettseas	466

Session: SS1.1	Room: I 12	Wednesday, Sep. 26, 16:30 - 18:00
-----------------------	-------------------	--

The IEC 61499 Function Block Model in Control and Automation

Chairing: Kleanthis Thramboulidis, Georg Frey

Benchmarking of IEC 61499 runtime environments	
Christoph Sünder, Alois Zoitl, Hermann Rofner, Thomas Strasser, Jeroen Brunnenkreef	474
Educational Approaches for the Industrial Acceptance of IEC 61499	
Seppo A Sierla, James H Christensen, Kari O Koskinen, Jukka P Peltola	482
Incorporating Industrial Experience to IEC 61499 Based Development Methodologies and Toolsets	
Mika P. Strömman, Kleanthis C. Thramboulidis, Seppo A. Sierla, Nikolaos Papakonstantinou, Kari O. Koskinen	490

<i>Implementing IEC 61499 Communication with the CIP Protocol</i> Frans Weehuizen, Aidan Brown, Christoph Sünder, Oliver Hummer	498
<i>Deployment of IEC 61499 Compliant Distributed Control Applications</i> Tanvir Hussain, Georg Frey	502
<i>Integrating CNet and IEC 61499 function blocks</i> Nils Hagge	506

Session: SS3	Room: I4	Thursday, Sep. 27, 11:00 - 12:30
Methods and Instrumentation for Performance Measurement in Real-time Networks		
Chairing: Alessandra Flammini, José A. Fonseca		

<i>Precision of Ethernet Measurements based on Software Tools</i> Iwan Schafer, Max Felser	510
<i>Delay Measurement System for Real-Time Serial Data Streams</i> Paulo Bartolomeu, Valter Silva, José Fonseca	516
<i>A new distributed instrument for Real Time Ethernet networks: experimental tests and characterization</i> Paolo Ferrari, Alessandra Flammini, Daniele Marioli, Andrea Taroni	524
<i>Measuring the impact of vertical integration on response times in Ethernet fieldbuses</i> Bruno Denis, Silvain Ruel, Jean-Marc Faure, Gaëlle Marsal, Georg Frey	532
<i>Measuring Real Time Performances of PC-based Industrial Control Systems</i> Micaela Caserza Magro, Paolo Pinceti	540

Session: T6.3	Room: I 10	Thursday, Sep. 27, 11:00 - 12:30
Control Theory and Applications		
Chairing: A. Tzes, L. Massotti		

<i>Stability margins characterization of a combined Servo/Regulation tuning for PID controllers</i> Orlando Arrieta, Ramón Vilanova	548
<i>I/O Decoupling And Disturbance Rejection For General Linear Time Delay Systems Via Measurement Output Feedback</i> Fotis N. Koumboulis, George E. Panagiotakis	555
<i>An Adaptive Input Shaping Technique for the Suppression of Payload Swing in Three-Dimensional Overhead Cranes with Hoisting Mechanism</i> John Stergiopoulos, Anthony Tzes	565
<i>Dynamic modeling of Proton Exchange Membrane Fuel Cell: The effect of temperature and membrane humidity</i> Ahmad Haddad, Rachid Bouyekhf, Abdellah El Moudni, Maxime Wack	569
<i>Automation of diagnosis of electric power transformers in Itaipu Hydroelectric Plant with a Fuzzy Expert System</i> Jovelino Falqueto, Matheus Telles	577

Session: T5.2	Room: I 11	Thursday, Sep. 27, 11:00 - 12:30
Emerging Issues and Solutions		
Chairing: Masanori Akiyoshi, Jose Lastra		

<i>Towards Biologically Inspired Control of Distributed Manufacturing Systems</i> Dania A. El Kebbe, Nils Kretzschmar	585
<i>An Alert Management System for Concrete Batching Plant</i> Jason C.S. Chung, Dickson K.W. Chiu, Eleanna Kafenza	591
<i>Electric power service selection considered flexibility of contract</i> Shigeyuki Tani, Masaharu Akatsu, Norihisa Komoda	599
<i>Enforcing Transition Deadlines in Time Petri Nets</i> Haisheng Wang, Liviu Grigore, Ugo Buy, Houshang Darabi	604

<i>Controlling Residential Co-Generation System Based on Hierarchical Decentralized Model</i> Takuya Matsumoto, Hisashi Tamaki, Hajime Murao	612
<i>Construction of Traceability System by using Simple and Handy type RFID reader</i> Rei Itsuki	619

Session: SS5	Room: I 12	Thursday, Sep. 27, 11:00 - 12:30
Embedded Systems Security		
Chairing: D.N. Serpanos, W.H. Wolf		

<i>Implementation of HSSec: a High-Speed Cryptographic Co-processor</i> Athanasios Kakarountas, Haralambos Michail, Costas Goutis, Constantinos Efstathiou	625
<i>Using Value Locality to Reduce Memory Encryption Overhead in Embedded Processors</i> George Keramidas, Pavlos Petoumenos, Alexandros Antonopoulos, Stefanos Kaxiras, Dimitrios Serpanos	632
<i>An Integrated Security Model for Component-Based Systems</i> Nimal Nissanke	638
<i>Security - Lifetime Tradeoffs for Wireless Sensor Networks</i> Zdravko Karakehayov	646
<i>Security and DRM in Indoor/Outdoor Heterogeneous Networking Applications for User – Centric Frameworks</i> Tasos Fragopoulos, Antonios Athanasopoulos, Artemios Vogiatzis, Evangelos Topalis, John Gialelis, Stavros Koubias	651

Session: T7	Room: I 13	Thursday, Sep. 27, 11:00 - 12:30
Distributed Intelligent Control for Flexible Manufacturing		
Chairing: George Chryssolouris, Nidhal Rezg		

<i>Hierarchical Distributed Controllers - Design and Verification</i> Dirk Missal, Martin Hirsch, Hans-Michael Hanisch	657
<i>Dynamic Workflow Prioritization Based on Block Finite Position Machines</i> Jesus Trujillo, Zbigniew Pasek, Enrique Baeyens	665
<i>Analytical Method for Generating Feasible Control Sequences in Controller Development</i> Jesus Trujillo, Zbigniew Pasek, Enrique Baeyens	673
<i>Structural Reasoning in Proving System Correctness</i> Andrei Lobov, Jose Luis Martinez Lastra	681
<i>Application Of The Supervisory Control Theory To Automated Systems Of Multi-Product Manufacturing</i> Daniel Balieiro, Eduardo Portela, Agnelo Vieira, Marco Buseti	689
<i>Management and manipulation of products using RFID-IMS in chain of production and distribution</i> Antonio Abarca, Julio Encinas, Andres Garcia	697

Session: WIP 1	Room: I4	Thursday, Sep. 27, 16:00 - 17:00
Industrial Networks and Factory Automation		
Chairing: Thilo Sauter		

<i>Topology Discovery in PROFINET</i> Iwan Schafer, Max Felser	704
<i>Retrieval of Diagnostic Information from PROFINET Networks</i> Tim Keane, Hassan Kaghazchi	708
<i>Uniform Engineering of Distributed Control Systems – The VAN Approach</i> Martin Hoffmann, Mathias Muehlhause, Marco Chiari, Christian Schwab	712
<i>Context-aware infrastructure to support distributed industrial services</i> Loubna Ali, Mayyad Jaber, Sodki Chaari, Frederique Biennier	716

<i>Development of Web-Based Software for a Multi-Fieldbus Diagnosis Tool</i>	
Scott Warner, Hassan Kaghazchi	720
<i>Assessment of the Ontological Approach in Factory Automation from the Perspectives of Connectionism</i>	
Aleksandra Dvorynachikova, Jose Lastra	724
<i>OPC server implementation with MMS over Ethernet</i>	
Hubert Kirmann, Sébastien Chatelanat, Michael Obrist	728
<i>Modeling Logical and Temporal Conditions to Formally Validate Factory Automation Web Services</i>	
Corina Popescu, Jose L. Martinez Lastra	732
<i>Message-Oriented Middleware for Automated Piezomotor Manufacturing</i>	
Patrick Otto, Bernd Lindner, Martin Wollschlaeger	736
<i>A Simulation Study of Ethernet Powerlink Networks</i>	
Stefano Vitturi, Lucia Seno	740
<i>Using a Packet Manipulation Tool for Security Analysis of Industrial Network Protocols</i>	
Tiago H. Kobayashi, Aginaldo B. Batista Jr., Agostinho M. Brito Jr., Paulo S. Motta Pires	744

Session: WIP 2	Room: I 10	Thursday, Sep. 27, 16:00 - 17:00
Embedded Systems and Tools		
Chairing: Thilo Sauter		

<i>A Power Manager for Deeply Embedded Systems</i>	
Geovani R. Wiedenhoft, Arliones Hoeller Jr., Antônio A. Fröhlich	748
<i>Coprime factorization based strong stabilizing controller design</i>	
Salva Alcántara, Carles Pedret, Ramon Vilanova, Romualdo Moreno	752
<i>A Constraint Logic Programming Framework for the Synthesis of Fault-Tolerant Schedules for Distributed Embedded Systems</i>	
Kåre Harbo Poulsen, Paul Pop, Viacheslav Izosimov	756
<i>Embedded linux scheduler monitoring</i>	
Zdenek Slanina, Vilem Srovnal	760
<i>Enhanced Engineering of Downtimeless System Evolution by use of Hardware Capability Descriptions within the eCEDAC Approach</i>	
Christoph Sünder, Oliver Hummer, Bernard Favre-Bulle	764
<i>Security in Agent-based Automation Systems</i>	
Basit Ahmed Khan, Jörgen Mad, Albert Treytl	768
<i>Automating Security Tests For Industrial Automation Devices Using Neural Networks</i>	
Joao Paulo S. Medeiros, Allison C. da Cunha, Agostinho M. Brito Jr., Paulo S. Motta Pires	772
<i>New Developments in EPOS Tools for Configuring and Generating Embedded Systems</i>	
Rafael Luiz Cancian, Marcelo Ricardo Stemmer, Antônio Augusto Fröhlich	776
<i>Genetic Algorithms Multiobjective Optimization of a 2 DOF Micro Parallel Robot</i>	
Dan Stan, Vistrian Maties, Radu Balan	780
<i>A Hidden Markov Models Tool for Estimating the Deterioration Level of a Power Transformer</i>	
Fotios Sotiropoulos, Panayiotis Alefragis, Efthymios Housos	784
<i>A Graphical Editor for the Input-Output Place-Transition Petri Net Class</i>	
Ricardo Nunes, Luis Gomes, Joao Paulo Barros	788
<i>Feasibility Conditions with Kernel Overheads for Mixed Preemptive FP/FIFO Scheduling with Priority Ceiling Protocol on an Event Driven OSEK System</i>	
Franck Bimbard, Laurent George	792

Session: SS9	Room: I 11	Thursday, Sep. 27, 16:00 - 17:30
Business Intelligence and its Applications in Industrial Ecosystems		
Chairing: Elizabeth Chang, Tharam Dillon		

<i>An FCA-based mapping generator</i>	796
Paolo Ceravolo, Zhan Cui, Alex Gusmini, Marcello Leida	
<i>Addressing The Challenges Of Enetwork Cyberengineering</i>	804
Miheala Ulieru, Mohsin Sohail	
<i>Trust based Decision Making Approach for Protein Ontology</i>	810
Amandeep Sidhu, Farookh Hussain, Tharam Dillon, Elizabeth Chang	
<i>Application of SPARQL in Semantic Search</i>	816
Hai Dong, Farookh Hussain, Elizabeth Chang	
<i>Quantifying the Level of Failure in a Digital Business Ecosystem Interactions</i>	820
Omar Hussain, Elizabeth Chang, Farookh Hussain, Tharam Dillon	
<i>An Overview of the interpretations of trust and reputation</i>	826
Omar Hussain	
<i>Ontology Engineering and (Digital) Business Ecosystems: a case for a Pragmatic Web</i>	831
Peter Spyns, Robert Meersman	

Session: SS6.1	Room: I 12	Thursday, Sep. 27, 16:00 - 17:30
Interoperability Issues and Research Work in Progress		
Chairing: Vincent Chapurlat, Athanasios Kalogeras		

<i>Enterprise modelling and verification approach for characterizing and checking organizational interoperability</i>	839
Vallespir Bruno, Chapurlat Vincent	
<i>Enterprise Semantic Modelling for Interoperability</i>	847
Nacer Boudjlida, Hervé Panetto	
<i>An Ontology-based Interoperability Framework for Distributed Manufacturing Control</i>	855
Daniel Diep, Christos Alexakos, Thomas Wagner	
<i>Interoperable Language Family for Agent Communication in Industrial Applications</i>	863
Thomas Wagner, Albert Treytl, Basit Ahmed Khan	
<i>Multilevel Order Decomposition in Distributed Production</i>	872
Daniela Wuensch, Aleksey Bratukhin	

Session: WIP 3	Room: I 13	Thursday, Sep. 27, 16:00 - 17:00
Sensors and Actuators		
Chairing: Thilo Sauter		

<i>Accuracy analysis of a 3D measurement system based on a laser profile scanner mounted on an industrial robot with a turntable</i>	880
Mohamed Rahayem, J.A.P Kjellander, Sören Larsson	
<i>Group Management System of RFID Passwords for Item Life Cycle</i>	884
Yuichi Kobayashi, Toshiyuki Kuwana, Yoji Taniguchi, Norihisa Komoda	
<i>Service Oriented Architecture for Mobile Robot Localization</i>	888
Camilo Christo, Carlos Cardeira	
<i>Active Beacon System with the fast processing architecture for Indoor Localization</i>	892
Byoung-hoon Kim, Jong-suk Choi	
<i>Line based robot localization using a rotary sonar</i>	896
Danilo Navarro, Ginés Benet, Milagros Martínez	
<i>Surveillance of Mobile Objects using Coordinated Wireless Sensor Nodes</i>	900
Tony Larsson	

<i>RAVEN: A Maritime Surveillance Project Using Small UAV</i>	
<i>Siu O'Young, Paul Hubbard</i>	904
<i>Intelligent Multisensorsystem for In-line Process- and Quality Monitoring of Welding Seams using Methods of Pattern Recognition</i>	
<i>Michael Kuhl, Reimund Neugebauer, Paul-Michael Mickel</i>	908
<i>Sensor Enabled Rule Based Alarm System for the Agricultural Industry</i>	
<i>Christos Gogos, Panayiotis Alefragis, Efthymios Housos</i>	912
<i>Matching Images of Imprinted Tablets</i>	
<i>Ziga Spiclin, Marko Bukovec, Franjo Pernus, Bostjan Likar</i>	916
<i>Wireless Vibrating Monitoring (WiVib) An industrial case study</i>	
<i>Jonas Neander, Stefan Svensson, Tomas Lennvall, Mats Björkman, Mikael Nolin</i>	920

Session: T3.2	Room: I 10	Thursday, Sep. 27, 17:00 - 18:00
Real-Time and Control		
Chairing: José A. Fonseca		

<i>Optimal Flow Routing in Multi-hop Sensor Networks with Real-Time Constraints through Linear Programming</i>	
<i>Jiri Trdlicka, Zdenek Hanzalek, Mikael Johansson</i>	924
<i>Simple PID Control Algorithm adapted to Deadband Sampling</i>	
<i>Volodymyr Vasyutynskyy, Klaus Kabitzsch</i>	932
<i>Second order sliding mode real-time network control of a robotic manipulator</i>	
<i>Luca Capisani, Tullio Facchinetti, Antonella Ferrara</i>	941
<i>On the practical issues of implementing the VTPE-hBEB protocol in small processing power controllers</i>	
<i>Jose Fonseca, Paulo Bartolomeu, Valter Silva, Francisco Carreiro</i>	949

Session: T9.1	Room: I4	Friday, Sep. 28, 10:00 - 11:30
Intelligent Robots I		
Chairing: Josep M. Mirats, Yolanda Bolea		

<i>A New Time-Independent Image Path Tracker to Guide Robots Using Visual Servoing</i>	
<i>Gabriel J. García, Jorge Pomares, Fernando Torres</i>	957
<i>Real-Time Architecture for Mobile Assistant Robots</i>	
<i>Pedro Sousa, Rui Araújo, Urbano Nunes, Luis Alves, Ana Lopes</i>	965
<i>Hierarchical Distributed Architectures for Autonomous Mobile Robots: a Case Study</i>	
<i>Jose Azevedo, Bernardo Cunha, Luis Almeida</i>	973
<i>Camera Localization and Mapping using Delayed Feature Initialization and Inverse Depth Parametrization</i>	
<i>Rodrigo Munguia, Antoni Grau</i>	981
<i>An Outdoor Guidepath Navigation System for AMRs Based on Robust Detection of Magnetic Markers</i>	
<i>Ana Lopes, Fernando Moita, Urbano Nunes, Razvan Solea</i>	989
<i>Decision Making among Alternative Routes for UAVs in Dynamic Environments</i>	
<i>Jose J. Ruz, Orlando J. Arevalo, Gonzalo Pajares, Jesus M. de la Cruz</i>	997

Session: T5.3	Room: I 11	Friday, Sep. 28, 10:00 - 11:30
Automated Manufacturing Systems		
Chairing: Toshiya Kaihara, Jose Lastra		

<i>Development of a holistic Guidance System for the NC Process Chain for benchmarking Machining Operations</i>	
<i>Ulrich Berger, Ralf Kretschmann, Matthias Aner</i>	1005
<i>Design and Realization of a STEP-NC Compliant CNC Embedded Controller</i>	
<i>Francesco Calabrese, Giovanni Celentano</i>	1010

<i>A Study on Automated Scheduling Methodology for Machining Job Shop</i>	1018
Yoshihiro Yao, Toshiya Kaihara, Kentaro Sashio, Susumu Fujii	
<i>A Heuristic Approach For Scheduling Multi-Chip Packages For Semiconductor Backend Assembly</i>	1024
Tay Jin Chua, Tian Xiang Cai, Xiao Feng Yin	
<i>A Formal Approach for the Specification, Verification and Control of Flexible Manufacturing Systems</i>	1031
Sajeh Zairi, Belhassen Zouari, Laurent Piétrac	
<i>Design and Implementation of Petrinet Based Distributed Control Architecture for Robotic Manufacturing Systems</i>	1039
G. Yasuda	

Session: SS8	Room: I 12	Friday, Sep. 28, 10:00 - 11:30
Design and Analysis of Distributed Automation Systems		
Chairing: Georg Frey, Alexander Fay		

<i>Formal verification of redundant media extension of Ethernet PowerLink</i>	1045
Steve Limal, Bruno Denis, Jean-Jacques Lesage, Stéphane Potier	
<i>DesLaNAS – a language for describing Networked Automation Systems</i>	1053
Jürgen Greifeneder, Georg Frey	
<i>Simulation Approach for Evaluating Response Times in Networked Automation Systems</i>	1061
Liu Liu, Georg Frey	
<i>UML-based safety analysis of distributed automation systems</i>	1069
Sebastian Schreiber, Till Schmidberger, Alexander Fay, Jörg May, Jörn Drewes, Eckehard Schnieder	
<i>Incremental design of distributed control systems using GAIA-UML</i>	1076
Arndt Lüder, Jörn Peschke	
<i>Distributed control programming in Java - The JAKOBI system</i>	1084
Michael Heinze, Joern Peschke	

Session: T3.3	Room: I 13	Friday, Sep. 28, 10:00 - 11:30
Distributed Real-time Systems		
Chairing: Thomas Nolte, Orazio Mirabella		

<i>Simulation for end-to-end delays distribution on a switched Ethernet</i>	1092
Jean-Luc Scharbarg, Christian Fraboul	
<i>Exploiting a Prioritized MAC Protocol to Efficiently Compute Interpolations</i>	1100
Björn Andersson, Nuno Pereira, Eduardo Tovar	
<i>Master Replication and Bus Error Detection in FTT-CAN with Multiple Buses</i>	1107
Valter Silva, Joaquim Ferreira, José Fonseca	
<i>Embedded Web Services for Industrial TCP/IP Services Monitoring</i>	1115
Francisco Maciá-Pérez, Diego Marcos-Jorquera, Virgilio Gilart-Iglesias	

Session: T10	Room: I4	Friday, Sep. 28, 12:00 - 13:30
Emerging Issues		
Chairing: Gianluca Cena, Dacfe Dzong		

<i>The Effect of Quartz Drift on Convergence-Average based Clock Synchronization</i>	1123
Eric Armengaud, Andreas Steininger, Alexander Hanzlik	
<i>Supply Chain Performance Evaluation from Structural and Operational Levels</i>	1131
Zhengping Li, Arun Kumar, Xiaoxia Xu	
<i>Common Approach to Functional Safety and System Security in Building Automation and Control Systems</i>	1141
Thomas Novak, Albert Treytl, Peter Palensky	

<i>A Development Process for Mechatronic Products: Integrating Software Engineering and Product Engineering</i>	
<i>Ana Patrícia Magalhães, Aline Andrade, Leila Silva, Herman Lepikson</i>	1149
<i>A Novel Class of Multi-Agent Algorithms for Highly Dynamic Transport Planning Inspired by Honey Bee Behavior</i>	
<i>Horst F. Wedde, Sebastian Lehnhoff, Bernhard van Bonn</i>	1157
<i>Introducing and Evaluating a Relaying Concept for the IEEE 802.16 Wireless Metropolitan Networks</i>	
<i>Christos Antonopoulos, Kostas Stamatis</i>	1165

Session: SS4	Room: I 10	Friday, Sep. 28, 12:00 - 13:00
Innovative E-Learning Experiences		
Chairing: Luis Gomes, Orazio Mirabella		

<i>Synchronous Multipoint E-Learning Realized on an Intelligent Software-Router Platform over Unicast Networks: Design and Performance Issues</i>	
<i>Giuseppe Maraviglia, Marina Masi, Vincenzo Merlo, Francesco Licandro, Alessandra Russo, Giovanni Schembra</i>	1172
<i>A Proposal of Remote Laboratory for Distance Training in Robotic Applications</i>	
<i>Marc Murtra, Gloria Jansa, Herminio Martinez, Joan Domingo, Juan Gamiz, Antoni Grau</i>	1180
<i>Remote Laboratory for Control Engineering Degree</i>	
<i>Yolanda Bolea, Antoni Grau</i>	1188

Session: SS7	Room: I 10	Friday, Sep. 28, 13:00 - 13:30
Grouping and Cooperating of Services		
Chairing: Carsten Buschmann, Reinhardt Karnapke		

<i>In-network Processing and Collective Operations using the Cocos-Framework</i>	
<i>Maik Krüger, Reinhardt Karnapke, Jörg Nolte</i>	1194
<i>Lean and Robust Phenomenon Boundary Approximation</i>	
<i>Carsten Buschmann, Daniela Krueger, Stefan Fischer</i>	1202

Session: T5.4	Room: I 11	Friday, Sep. 28, 12:00 - 13:30
Multi-agent Systems for Manufacturing Control		
Chairing: Masanori Akiyoshi, Jose Lastra		

<i>Methodology for the efficient distribution a manufacturing ontology to a multi-agent system utilizing a relevant Meta-Ontology</i>	
<i>Manos Georgoudakis, Christos Alexakos, Athanasios Kalogeras, John Gialelis, Stavros Koubias</i>	1210
<i>Agent-based Control of Rapidly Reconfigurable Material Handling System</i>	
<i>Jani Jokinen, Jose L. Martinez Lastra</i>	1217
<i>Agent Based Prototype for Interoperation of Production Planning and Control and Manufacturing Automation</i>	
<i>Rui M. Lima, Rui M. Sousa</i>	1225
<i>Agent-Based Control Model for Reconfigurable Manufacturing Systems</i>	
<i>Omar López, Jose Lastra</i>	1233
<i>A holonic approach for manufacturing execution system design: an industrial application</i>	
<i>Blanc Pascal, Demongodin Isabel, Castagna Pierre, Hennet Jean-Claude</i>	1239

Session: T2.2	Room: I 12	Friday, Sep. 28, 12:00 - 13:30
Scheduling, Safety and Response Times of Industrial Communication Networks		
Chairing: Julian Proenza, Thomas Nolte		

<i>Network Recovery Time Measurements of RSTP in an Ethernet Ring Topology</i>	
<i>Gunnar Prytz</i>	1247

<i>Evaluation of timing characteristics of a prototype system based on PROFINET IO RT_Class 3</i>	
Paolo Ferrari, Alessandra Flammini, Daniele Marioli, Andrea Taroni, Francesco Venturini	1254
<i>Hyperperiod Bus Scheduling and Optimizations for TDL Components</i>	
Emilia Farcas, Wolfgang Pree	1262
<i>Testing Approach for Online Hardware Self Tests in Embedded Safety Related Systems</i>	
Thomas Tamandl, Peter Preininger, Thomas Novak, Peter Palensky	1270
<i>BuST: Budget Sharing Token Protocol for Hard Real-Time Communication</i>	
Gianluca Franchino, Giorgio C. Buttazzo, Tullio Facchinetti	1278

Session: SS6.2	Room: I 13	Friday, Sep. 28, 12:00 - 13:30
Interoperability Applications		
Chairing: Athanasios Kalogeras, Ioannis Gialelis		

<i>Semantically-Enabled Inter-Enterprise Integration in the Tourist Sector</i>	
Christos Alexakos, Panagiotis Konstantinopoulos, Kostas Charatsis, Stavros Koubias	1286
<i>Interoperability Issues in Virtual Organization – How to Proceed?</i>	
Taivo Kangilaski	1293
<i>Towards an ontology-based system for intelligent prediction of firms with fraudulent financial statements</i>	
Dimitris Kanellopoulos, Sotiris Kotsiantis, Vasilis Tampakas	1300
<i>AHP Based Supply Chain Performance Measurement System</i>	
Laura Xiao Xia Xu	1308

Session: T9.2	Room: I4	Friday, Sep. 28, 13:30 - 15:00
Intelligent Robots II		
Chairing: Antoni Grau, Gabriel J. Garcia		

<i>Accurate Range Image Registration: Eliminating or Modelling Outliers</i>	
Yonghuai Liu, Honghai Liu, Longzhuang Li, Baogang Wei	1316
<i>A Two Stage Robot Control for Liquid Transfer</i>	
Maria P. Tzamtzi, Fotis N. Koumboulis, Nicholas D. Kouvakas	1324
<i>Dynamic equations of motion for a 3-bar tensegrity based mobile robot</i>	
Josep M. Mirats Tur, Sergi Hernandez Juan, Albert Graells Rovira	1334
<i>Onto computing the Uncertainty for the Odometry Pose Estimate of a mobile robot</i>	
Josep M. Mirats Tur	1340
<i>Solving the Inverse Kinematics Problem Symbolically by Means of Knowledge-Based and Linear Algebra-Based Methods</i>	
Michael Wenz, Heinz Wörn	1346
<i>Multivariable Iterative Feedback Tuning – A Step-Wise Safe Switching Approach</i>	
Fotis N. Koumboulis, Maria P. Tzamtzi, George E. Chamilothis	1354
<i>Fuzzy Cooperative Control of Automated Ground Passenger Vehicles</i>	
Francesco M. Raimondi, Maurizio Melluso	1364

Session: WIP 4	Room: I 10	Friday, Sep. 28, 13:30 - 14:30
Wireless and Dependable Networks		
Chairing: Thilo Sauter		

<i>Implementation of Power Aware Features in AODV for Ad Hoc Sensor Networks. A Simulation Study</i>	
Konstantina Pappa, Antonis Athanasopoulos, Evangelos Topalis, Stavros Koubias	1372
<i>Integrating Building Automation Systems and Wireless Sensor Networks</i>	
Erik Pramsten, Daniel Roberthson, Fredrik Österlind, Joakim Eriksson, Niclas Finne, Thiemo Voigt	1376

<i>Multicast Communication in Wireless Home and Building Automation: ZigBee and DCMF</i>	
Christian Reinisch, Wolfgang Kastner, Georg Neugschwandtner	1380
<i>Using Time-Triggered Communications over IEEE 802.15.4</i>	
Nuno Ferreira, José A. Fonseca	1384
<i>On a IEEE 802.15.4/ZigBee to IEEE 802.11 Gateway for the ART-WiSe Architecture</i>	
João Leal, André Cunha, Mário Alves, Anis Koubâa	1388
<i>Performance measurements of 802.11 WLANs with burst background traffic</i>	
Claudio Zunino	1392
<i>IEC 62439 PRP: Bumpless Recovery for Highly Available, Hard Real-Time Industrial Networks</i>	
Hubert Kirmann, Mats Hansson, Peter Müri	1396
<i>A Two-Competitive Approximate Schedulability Analysis of CAN</i>	
Björn Andersson, Nuno Pereira, Eduardo Tovar	1400
<i>Modelling MajorCAN with UPPAAL</i>	
Matias Bonet, Gabriel Donaire, Julian Proenza	1404
<i>A Decentralized Intrusion Detection System for Increasing Security of Wireless Sensor Networks</i>	
Ioannis Chatzigiannakis, Andreas Strikos	1408
<i>Energy Efficient Authentication in Wireless Sensor Networks - An industrial case</i>	
Rickard Soderlund, Stefan Svensson, Tomas Lennvall	1412

Session: WIP 5	Room: I 11	Friday, Sep. 28, 13:30 - 14:30
Control		
Chairing: Thilo Sauter		

<i>A Visual-Servoing System for a Humanlike Shape Memory Alloy Actuated Finger</i>	
Konstantinos Andrianesis, Anthony Tzes, Efthymios Kolyvas, Yannis Koveos	1417
<i>Optimization rules for mill cutter and cutting parameters selection incorporating a control algorithm</i>	
Luis Rubio, Manuel De la Sen	1421
<i>Ontology-driven Control Application Design Methodology</i>	
Athanasios Kalogeras, Luca Ferrarini, Arndt Lueder, John Gialelis, Christos Alexakos, Joern Peschke, Carlo Veber	1425
<i>A Metaheuristic Approach for Controller Design of Multivariable Processes</i>	
Fotis N. Koumboulis, Maria P. Tzamtzi	1429
<i>Robust Controller Design for Active Hydraulic Suspension</i>	
Michael G. Skarpetis, Fotis N. Koumboulis, Achilleas S. Ntellis, Apostolis Sarris	1433
<i>Robust Lane Keeping for a Tractor-Trailer</i>	
Michael G. Skarpetis, Fotis N. Koumboulis, Achilleas S. Ntellis, Thomas E. Tsimos	1437
<i>Fuzzy Control of Sparing in Disk Arrays</i>	
Guillermo Navarro, Milos Manic	1441
<i>CARUSO - Towards a Context-Sensitive Architecture for Unified Supervision and Control</i>	
Rolf Kistler, Stefan Knauth, Daniel Käslin, Alexander Klapproth	1445
<i>Smoothing out the Adaptive Variable Structure Control Law for Induction Motors</i>	
Oscar Barambones, Patxi Alkorta, Izaskun Garrido, Aitor Garrido	1449
<i>Industrial robot motion control with real-time Java and EtherCAT</i>	
Sven Robertz, Klas Nilsson, Roger Henriksson, Anders Blomdell	1453

Session: T2.3	Room: I 12	Friday, Sep. 28, 13:30 - 14:30
Clock Synchronization and Multimedia Real-time Communications		
Chairing: Thomas Nolte, Christos Koulamas		

<i>A PLL-Based Approach to Clock Synchronization for Trajectory Rebuilding in Event-Triggered Communication Systems</i>	
Carlo Rossi, Manuel Spera	1457

A Simulation Framework for Fault-Tolerant Clock Synchronization in Industrial Automation Networks

Fritz Praus, Wolfgang Granzer, Georg Gaderer, Thilo Sauter 1465

Dynamic QoS Management for Multimedia Real-Time Transmission in Industrial Environments

Javier Silvestre, Luis Almeida, Ricardo Marau, Paulo Pedreiras 1473

Integration of a flexible time triggered network in the FRESCOR resource contracting framework

Ricardo Marau, Paulo Pedreiras, Luis Almeida, Michael Harbour, Daniel Sangorrin, Julio Medina 1481

Session: SS1.2	Room: I 13	Friday, Sep. 28, 13:30 - 15:00
IEC61499 Implementations		
Chairing: Kleanthis Thramboulidis, Georg Frey		

RTAI-based Execution Environments for Function Block Based Control Applications

George Doukas, Alessandro Brusafferri, Marco Colla, Kleanthis Thramboulidis 1489

Testing Approach for Online Hardware Self Tests in Embedded Safety Related Systems

Thomas Tamandl, Peter Preininger, Thomas Novak, Peter Palensky

Vienna University of Technology,
Institute of Computer Technology
Gusshausstrasse 27-29
1040 Vienna, Austria

{tamandl, preininger, novakt, palensky}@ict.tuwien.ac.at

Abstract

In safety related systems online hardware self tests are integrated to reach a defined level of hardware integrity. These tests comprise testing of the static and volatile memory and the CPU internals. The higher the level of integrity should be, the more efficient tests must be used. Additionally, the effort to verify the correctness of the tests is rising. Hence, designing the test, and the validation and verification of the tests, is a critical part in the development process of safety related systems.

The verification of the correct behavior requires sophisticated methods for stimulating errors that must be detected by the tests. The document describes an environment based on boundary scan technology for testing the online hardware self tests automatically.

1. Introduction

Today's functional safety related systems (short safety related systems) are mostly accomplished with microcontrollers. They are created to reduce the inherent risk of a device or system, for example a node in a fieldbus system, to a tolerable level.

The international standard IEC 61508 [1] defines requirements for designing safety related systems. It specifies a life cycle model including all activities required to avoid systematic failures and to handle stochastic failures.

IEC 61508 defines functional safety as "part of the overall safety that depends on a system or equipment operating correctly in response to its inputs" [1]. A safety related system is a system that executes safety functions and cares for the required safety integrity of the safety functions. A safety function is responsible for reaching or keeping a safe state of a device. The safety integrity of a safety related system includes the hardware integrity and the systematic integrity.

Systematic integrity means applying measures to reduce the risk coming from systematic failure during

the design or operation phase of the system. Typical measures are on the one hand management tools and on the other hand functions to monitor the program flow or the supply voltage. To grant a designated level of hardware integrity a defined amount of stochastic hardware failures must be detected.

A method to detect stochastic hardware failures is to test the hardware components with online hardware self tests (short hardware tests) implemented in software. Normally hardware tests are periodically inspecting the volatile and non volatile memory as well as the central processing unit (CPU) including its registers, flags and arithmetic logic unit (ALU) for hardware failures.

There are different hardware test algorithms available capable of checking the volatile memory and registers of the CPU. Examples for RAM test methods are the walking pattern test [2], Abraham test [3] or galloping pattern test [2]. Classically static or invariant memory is tested for failures by means of parity bits, checksums or cyclic redundancy checks (CRC). Flags and the ALU are tested by inserting bit patterns and comparing the output with the specification [2]. The algorithms and methods to ensure hardware integrity differ in performance and diagnostic coverage (DC) (= ratio of detected failures to the total amount of failures).

The safety integrity of hardware is categorized by four safety integrity levels (SIL) [1]. Each SIL specifies an error probability per hour, i.e. the tolerable number of dangerous errors per hour. SIL 1 defines the highest value of error probability, whereas SIL 4 specifies the lowest level. The higher the SIL the more rigorous are the safety integrity requirements. A way to meet these requirements is to implement hardware tests with a high diagnostic coverage.

Unfortunately, complete avoidance of human mistakes during implementation of hardware tests is not realistic. For example, even a well-trained and experienced programmer makes an error in every 100 statements [4]. Thus the standard IEC 61508 specifies a great amount of requirements for the testing process too. These requirements shall ensure a high level of software

quality and therefore reduce the probability of malfunctions.

2. Software Testing

Generally, software can be tested code-based or requirements-based. Test cases derived from the source code are white box test cases, others derived from the software requirements are black box test cases. Black box test cases are specified using methods such as equivalence class partitioning or boundary value analysis [5]. On the contrary, white box test cases are defined depending on the code coverage, e.g. statement, branch or path coverage. The coverage is always the fraction of total number of statements, branches or paths having been executed by different test cases.

Test cases are used to perform different types of testing such as function testing (verification of functionality) or performance testing. These test types are carried out at various steps of the software testing process.

The first step of software testing is unit testing. The software unit, e.g. a function, is examined if it can perform its specified functionality properly. This type of testing is described here. The second step is integration testing where the units are integrated to form a subsystem. At this point attention is paid to the interfaces between the units. The final step is system testing where the software has to run on the target platform and interacts with external software.

3. Overview of Test Approach

As already mentioned in section 1 safety integrity is specified by 4 safety integrity levels (SIL). A higher SIL not only results in more rigorous safety integrity requirements, but also in a higher testing effort. The higher the SIL the more comprehensive test cases with white and black box test cases have to be performed on the software. Thus automatic tests tools are desirable to keep the test effort moderate.

The test approach for hardware tests presented in the following sections allows checking the functionality of the hardware tests automatically. It enables the tester to execute white box and black box test cases during the unit, integration and system testing.

The test approach was elaborated during the development process of a safety related system in the project SafetyLon. The European collective research project SafetyLon supported by the European Union within the sixth framework program has the goal to make the LON technology [6] safe. It is foreseen to meet safety requirements as CANOpen-safety [7] or PROFISafe [8] does.

Within the project standard LON nodes are enhanced with additional safety related hardware and special

embedded safe software. Parts of the safe software comprise the hardware tests for the central processing unit, the volatile- and non volatile memory. The correct behavior of these must be verified. As a result testing techniques to verify the hardware test were developed that are presented in the following.

4. Technical Overview

Hardware tests must be performed to guarantee the correct functionality of the system. The implementation of these tests is described in [9]. In the following the methodology how to verify the behavior of these tests is introduced.

For hardware test verification fault injection technologies are necessary to insert faults that must be detected by the hardware tests. Standard fault injection technologies use direct pin level injection, additional software modules, and external electromagnetic fields or similar to provoke a fault. [10] provides a good overview on some technologies already used.

The microcontrollers used to run the hardware tests are based on a standard ARM7TDMI core with internal RAM and FLASH memory. As all components are located within one package, there are no external data- or address-busses. Due to the lack of accessibility of the internal data busses direct pin level fault injection is impossible. As there was the requirement not to change the software running on the target hardware, software fault injection, regardless of runtime or compile time injection could not be chosen. Hence, approaches chosen from [11], [12] or [13] are not suitable.

The problem, how to manipulate data without any access to internal signals, can be addressed by means of debugging features offered by the ARM core. [14] discusses pin level fault injection in opposite to using fault injection boundary scan. This technology, enhanced with the internal scan chains offered by the ARM7 controller, allows the simulation of internal faults without changing the software running on the target hardware.

For debugging purposes the ARM7 core provides a Joint Test Action Group (JTAG) conform debugging interface. The international standard IEEE 1149.1 [15] Standard Test Access Port and Boundary-Scan Architecture describes this interface that was initially intended for testing printed circuits. In case of the ARM7 core it grants the access to the core internals via the test access port (TAP) controller. Hence mostly all processor internals are accessible from the outside. The JTAG interface allows controlling the ARM7 core from outside, Figure 1 shows a typical debugging environment.

A debug host computer using a high level debugging language is communicating with the debug target. Communication is handled a protocol converter which

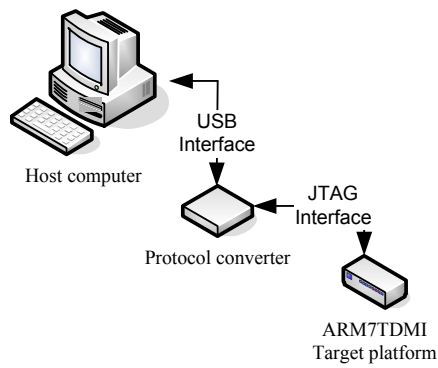


Figure 1. Typical debug environment

converts the high level language into the JTAG compliant commands.

As already mentioned the debugger on the host computer is communicating with the TAP controller. This device is controlling three internal scan chains connected to the main processor logic and the EmbeddedICE-RT logic described later, see Figure 2 for details. Scan chain 0 and 1 interfaces the processor core and the data bus and scan chain 2 grants the access to the EmbeddedICE-RT registers. Using this scan chains all processor internals can be accessed, examined and modified.

The ARM7 core itself offers the following debug features. The program execution can be halted, the state of the system can be examined, as described before, and the program execution can be resumed.

The EmbeddedICE-RT logic provides additional features. It can be accessed and configured by means of scan chain 2.

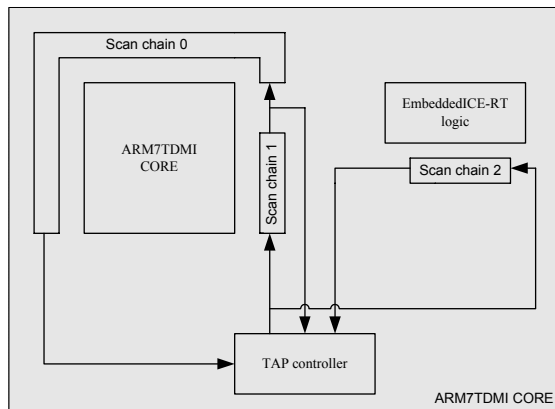


Figure 2. ARM7TDMI scan chains [16]

The EmbeddedICE-RT logic monitors processor internal signals such as data- and address-bus. Beside others this logic provides two configurable break or watchpoints. The breakpoints can be configured to force a halt of the processor if the appropriate address is reached. Watchpoints can be configured to halt the processor when specified memory cells are addressed. Refer to [16] for details on the ARM7TDMI internals.

The aforementioned features permit to halt the processor without having direct access to the signals of the core logic when accessing specified memory addresses. In halt mode modifications can be done and the program can be resumed. These features allow automatic testing of the hardware tests presented in [9].

For setting up an automatic test environment a high level language is useful for the communication with the TAP controller. Therefore a programming interface was developed by the company Segger Microcontroller Systems [17]. Its purpose is to provide third party applications the whole functionality of the J-Link interface.

Segger Microcontroller Systeme GmbH offers a set of tools for developing applications running on ARM7 cores. A protocol converter named J-Link – see Figure 1 – is used to connect the host PC via USB to the JTAG port of the processor. Normally a debugger is communicating with the J-Link. For automatic fault injection a debugger is not suitable so an adequate application was developed. Therefore direct control – without using a debugger – over the J-Link interface is required.

The J-Link Software Development Kit, including the J-Link application program interface, is satisfying these requirements. The communication itself is performed like a standard communication resource. After opening and configuring the JTAG connection the ARM7 core can be controlled by running the appropriate commands. The automatic test tool described in the next chapter is based on the J-Link application program interface.

5. Test Tool Design Basics

Based on the information provided in the previous chapter the J-Link application program interface can be used to set up an automated test tool. In the following the principles the automated test tool is based on are explained.

5.1. Testing Principles

A test tool for safety related systems has to be able to perform a huge number of different test cases. The best solution is a tool which can perform the tests automatically. Such a tool needs information on how to perform a test and how a test is considered as successful or failed. Therefore it must be able to accept defined test cases which are stored in an input file. Details on the input file are presented in section 6.3. The necessary functions of a software test tool are on the whole similar to debugging functions. The most important representatives are:

- Reset ARM7 core
- Start and halt ARM7 core

- Set and reset breakpoint
- Single stepping
- Manipulation of memory contents
- Read and store memory contents
- Manipulation of registers
- Read registers contents
- Manipulation of register flags

In order to set up an automatic test environment additionally the following functions are necessary:

- Read and compare memory contents to specifiable values
- Read and compare register contents to specifiable values
- Read and compare flag values to specifiable contents
- Specification of test case fail/success depending on the result of the comparisons

Beside the functionality of an automated test tool, the ability to generate reports must be taken into account. In order to detect erroneous memory contents, functions for comparing memory cells or registers to specific values are available. The result of the comparison can be used to define the failing or passing of a test case.

5.2. White Box Testing

The presented functions allow performing white box tests for hardware tests. Obviously performing white box test cases is only possible on assembler level. If C-code has to be tested, this can only be done with the underlying compiler generated assembler code.

Figure 3 shows the standard program flow of the test tool. The flow does not include the more complicated black box testing implementations. Line per line of the input file is interpreted by the JLink interface. According to the interpretation the appropriate JLink operation is executed and logged to a report file. If the operation sets the ARM7 core into a run mode, the program waits until the ARM7 core is halted due to reaching a breakpoint or watchpoint. As soon as the ARM7 core is halted a log line is written and the results of the operation are evaluated and logged (if applicable). If the operation is a test operation, the tool evaluates whether the test is considered to be successful. If the test is successful, the complete process is executed again until either a failed test occurred or a test is considered as failed. If a test is considered as failed the program is stopped.

White box tests require the user to alter register and memory contents in order to guide the program into the different program paths. For doing that, the user can run the program to a certain point specified by a breakpoint and modify the memory or register contents (and therefore change the path of program execution). A second breakpoint can be defined for testing if the path

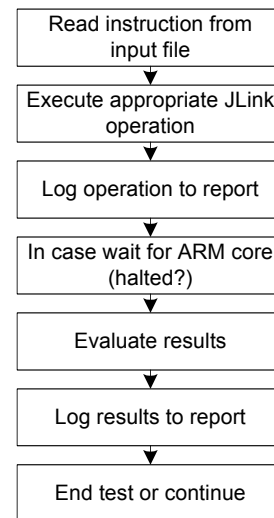


Figure 3. Standard program flow of test tool

had been executed properly. E.g., the user can query if a memory cell or a register contains the expected results. A wrong value can cause the test to fail.

All these actions are logged to the report in order to keep track of the program proceedings. In case of a failure, the logged data can be used to check if the test itself had been set up wrongly or the program produced a failure.

It is the main task of the tester to define tests which cover all possible program paths in order to detect possible errors in the program flow.

5.3. Black Box Testing

For testing the hardware tests of volatile or non volatile memory black box testing methods are required. In opposite to white box testing these kinds of tests are independent from the programming language used for designing the hardware test.

The test of the hardware tests are based on simulating stuck at faults and coupling faults. A stuck-at fault means that a memory cell delivers only a 1 or 0 when accessed. The value is independent of the content that the cell should have. A coupling fault on the other hand concerns two different memory cells. One memory cell, which is accessed, causes the other memory cell to be influenced. This behavior is called a coupling fault. It can depend on the access mode (read or write) of the influencing cell as well as on the contents of the memory cells.

For the simulation it is necessary to specify which bits of a memory cell are stuck-at 1 and which ones are stuck-at 0 (or coupled to 1 or 0).

The simulation of both faults uses the watchpoint functionality offered by the EmbeddedICE-RT unit of the ARM7 core. This feature halts the ARM7 core as soon as the specified memory address is accessed. The possibilities for the watchpoint configuration are numerous. Read/write access can be specified as well as

the halt on one specific memory address or a masked memory address. In order to keep the tool user-friendly only one specific memory address can be specified for the watchpoints.

The difficulty of using watchpoints is that the ARM7 core is halted after an instruction occurred. This means that in case of a write access the ARM7 core is halted after the memory cell has been written – in case of a read access the registers already contain the appropriate values.

In order to be able to simulate a stuck-at or coupling failure the previously executed instruction has to be identified and evaluated. For doing this the program counter is read via the JLink and the instruction at the appropriate address is read.

In case of a write to memory instruction the simulation is straight forward. The appropriate memory cell is modified. After performing a read instruction the register which contains the memory data has to be identified. This can be more complicated in the case of a multiple read access.

In fact the difficulty is that the executed instruction which led to the halt has to be identified, translated and the effects of the instruction on the memory cells and CPU registers has to be made undone.

As an example Figure 4 shows the implementation of the stuck-at simulation. The implementation is only shown for a single run, after the execution of the run, the ARM7 core is restarted. As shown in the diagram the OP_Code of the instruction which led to the watchpoint is evaluated using a case instruction. The appropriate concerned register is extracted out of the opcode.

The simulation of the coupling failures is very similar to the implementation presented in Figure 4. The only difference is that the watchpoint is set to the original

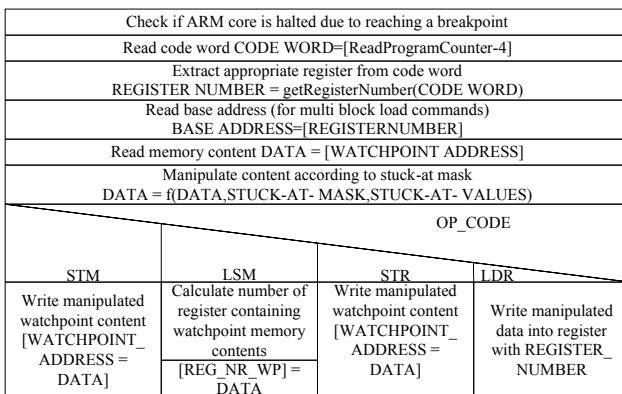


Figure 4. Nassi-Schneidermann diagram of simulation of a stuck at failure

memory cell (which changes the coupled cell if accessed). The main difficulty is the behavior of the multiple load and store commands. If both – the original and the coupled cell – are accessed within a multiple load or store command this issue has to be taken into account.

Based on the two failure models, the stuck-at and the coupling fault model, the correct behavior of the hardware tests for volatile and non volatile memory can be verified. Also the performance of the tests, regarding their capability of detecting different failures can be tested.

6. Performing Tests

As presented in section 5 there are a lot of possible configurations for the automated test tool. For performing tests on hardware tests only a few instructions are necessary to configure the test tool according to the specified test. In the following the test tool is presented on a more detailed level. Later, on behalf of a white box test case for a test of the arithmetic logic unit, it is shown exemplarily how to set up a concrete test. Following this model all test mentioned above can be implemented.

6.1. Tool Configuration

At the beginning an input file, designed according to the comma-separated value (csv) file format, is defined. The csv-format is used for storing tabular data, where the columns are separated by a delimiter. It allows editing the file with a simple text editor or a spreadsheet calculator. Hence, simple test cases can be designed very quickly.

In order to gain the required breakpoints the map files generated by the linker have to be examined. These map files contain the information on the addresses which are used for setting the breakpoints. Every time the source code or the placement of the object file changes, it is likely that the addresses of the breakpoints are changed too. Depending on the changes done, a spreadsheet calculator may help to address the problem of changing addresses. E.g. the addresses of the breakpoints then can be defined as the sum of the module address and an offset. Hence, as long as the module internals do not change, only the module addresses must be updated. This does not help if the tested module itself is changed. Of course the specification of the test procedure and its breakpoints has to be done very carefully, as the number of tests usually is rather numerous (for example, there are specified more than 270 test cases to test the online RAM test).

To avoid failures during the creation process of the input file a very clearly arranged format is used. First a command is given specifying the appropriate action. The other values are depending on the specified command. The test tool simply interprets the command together with the specified data and performs the appropriate action of the Segger interface. Also comments can be specified in the input file in order to make it maintainable more easily. Lines within the input file are considered as comments which are logged directly to the

report file. Additionally a line number referring to the input line number is written into the output file.

Table 1 shows the instructions which can be used within the csv input file.

Table 1. Instruction set of the test interface

Instruction	Description
RESET	Reset ARM7 core
HALT	Stop ARM7 core
RUN	Start program execution
STEP	Singlestep ARM7 core
STEPN	Execute n instructions
RUNS	RUN Specific - Run ARM7core to a specific address using single stepping, no breakpoints are used in this case.
BPS/BPD	Breakpoint Set/Delete - Set/delete a breakpoint
WPS/YPD	Watchpoint Set/Delete - Set/delete a watchpoint
SIMSTUCKAT	SIMulate STUCKAT - Simulate stuck-at fault at a specified memory address. Both possible stuck-at faults can be simulated.
SIMCROSSTALK	SIMulate CROSSTALK Simulate a coupling fault for two specified addresses (uni-directional). The crosstalk simulation is similar to the stuck-at simulation. The only difference is that the stuck-at is injected into the influenced memory cell.
RM	Read Memory cell - 1, 2 or 4 byte can be read
CM	Compare Memory cell to given data. It can be specified that the test has to be aborted, if the data do or do not match.
WM	Write Memory content - 1, 2 or 4 bytes can be written.
READMEMIMG	READ MEMory IMaGe Read the complete memory and write it to a file.
RR	Read Register
CR	Compare Register content to a given value, if the data do or do not match, the test can be aborted.
WR	Write Register content - a value is written to the specified register

RAR	Read All processor Registers
RCF	Read Condition Flags
WCF	Write Condition Flags
OCF/ACF/XCF	Or Condition Flags And Condition Flags Xor Condition Flags
Other phrases	Use mask to modify the flags, using OR/AND/XOR Other phrases are interpreted as comments which are also written into the test report.

As soon as the simulation is executed it receives its tasks from the input file, a message is logged to the report file and the core is set into run mode until either a watchpoint or a breakpoint is reached. In case of a watchpoint, they are used to simulate a coupling- or stuck-at-fault, the processor performs the necessary data manipulation and is set to run mode again.

After stopping the program execution at a breakpoint the test tool refers to the next line of the input file in order to get new instructions.

6.2. Standard Workflow

The standard workflow for the execution of a test case is shown in Figure 5. First the program has to be compiled, linked and downloaded into the microcontrollers FLASH storage using the development tools of the project (e.g. an integrated development environment). The second important step is to specify the possible and necessary breakpoints for the test respectively. In order to recognize misbehaving parts of a program, one breakpoint is set where the memory contents can be manipulated. A second breakpoint is set to the return instruction of a function which returns an OK value or a failure value. This return value (which is usually stored in register R0) is compared to the expected value. The position of the breakpoints within the program flow can be found in the MAP file, which is generated by the linker.

After downloading the hardware test to the microcontroller, the test tool is started gaining its commands from the file. The first instructions within the csv-file must contain the specification of the breakpoints, followed by a reset and a run instruction. At some point within the program execution the configured break- or watchpoint is hit. Once the processor stops, the actions defined in the input file are performed. Afterwards a breakpoint is set to an instruction, e.g. to a return command, where the effects of the performed actions can be evaluated. Based on this evaluation the test of the hardware test can be considered as failed or passed.

Using this methodology allows to perform various tests. Within one input file several test cases can be placed consecutively. For starting a new test case the former must be performed successfully.

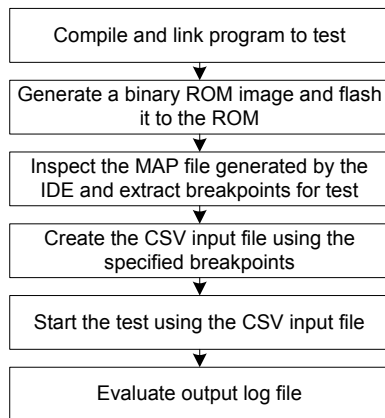


Figure 5. Work flow of a test case execution

6.3. Sample Test Input File

The following lines show an example of an input file written in csv-format:

```

TEST30-1;;;
HALT;;;
BPD;0;;;
BPS;0;14476;;;
BPS;1;15252;;;
RESET;;;
RUN;;;
WR;2;0xFFFFFFFF;;
RUN;;;
CR;0;0;NORES;PONM;Error -> test failed

```

First the test is named. The name of the test case is written to the output log file. At the beginning the ARM7 core is halted and the breakpoints 0 and 1 of previous test cases are deleted. Afterwards the breakpoints which are necessary for the test are set. Breakpoint 0 is set to a position within the code where a calculation result is compared to a special value. The second breakpoint – breakpoint 1 is set to the return of the function in order to be able to check the return value of the function. The ARM7 core is then reset and started. After the RUN statement the test tool waits until the ARM7 core is halted. This happens at breakpoint 0. At this position the register R3 which contains the calculation result, is manipulated in order to contain a wrong value. The ARM7 core is set into run state again and is halted at breakpoint 1. At this position in the source code, register R0 should contain a value (return value) different to 0. No reset has to be done after the comparison – the test is passed, if the value of R0 is not equal to 0 (PONM – pass on no match). If failed, the message “Error -> test failed” is written to the log file.

```

0 Automatic Test log using JLINK
  Start of logging at Mon Jun 26 21:28:51 2006
  DLL Version: 3.20a
  JLINK Compilation Date: Apr 27 2006 23:07:04
  Firmware: J-Link compiled Apr 27 2006
  12:55:19 ARM Rev.5

```

```

J-Link speed = 30
ARM core ID: 0x3F0F0F0F
*****
TEST30-1
*****
81 ARM Core halted
82 Breakpoint number 0 deleted
83 Breakpoint number 0 set at address 14476
84 ARM Core reset
85 ARM core started
   ARM core halted
   Current position: 0x0000388C
86 Value 0xFFFFFFFF (-2) written to ARM_REG_R2
87 ARM core started
   ARM core halted
   Current position: 0x00003B94
88 ARM_REG_R0 = FFFFFFFF
   Register      content      0xFFFFFFFF      of
   ARM REG R0    does not match to value
   0x00000000 (0)
   (null)
***** T E S T   P A S S E D *****

```

The sample above shows the output of the automated test presented at the beginning of the section. The header of the output file presents information on the hardware and software versions used and the test start time. Afterwards the test identifier is shown and the information on the test results is logged. In the end of the test case the test condition is verified and the test case is logged as passed.

7. Conclusion

Using boundary scan technology paired with the debugging features offered from the ARM7 core allows the simulation of faults without having direct access to the internal components. For the presented method no expensive hardware or complex software is required.

It was shown that a relatively limited tool is able to perform the necessary fault injection in order to verify the correct behavior of hardware tests. Due to the high degree of adaptation the test tool may not be suitable for performing tests within a comprehensive project. The advantage of the presented method is that no adapted software running on the device under test is required. Instead of adapted software extensive study of the linker generated map files is necessary to set the appropriate breakpoints and parameters. An additional drawback is the changed timing behavior. Every time a hardware fault is simulated the target processor is halted and the appropriate tasks to simulate a hardware fault take place.

By means of the J-Link application program it is possible to control the ARM7 core internals via a high level programming language. Hence the test tool can easily be adapted to new requirements without knowing details about the underlying communication from the development host and the target hardware.

Results after performing tests show that white box tests as well as black box tests can be performed very efficiently. To enhance the usability of the test tool a more convenient method to extract and set the required break points might be useful.

References

- [1] "IEC 61508 – Functional safety of electric/electronic/programmable electronic safety-related systems", 1999.
- [2] H. Hölscher, J. Rader, *Microcomputers in Safety Technique*, TÜV Rheinland, 1986.
- [3] Ravindra Nair, Satish M. Thatte, Jacob A. Abraham, "Efficient Algorithms for Testing Semiconductor Random-Access Memories", *IEEE Transactions on Computers*, Volume 27, pp. 572-576, 1978.
- [4] L. Wang, K. Ch. Tan, "Software Testing for Safety-Critical Applications", *IEEE Instrumentation & Measurement Magazine*, pp.38-47, 2005.
- [5] W. Zuser, S. Biffl, T. Grechnig, M. Köhle, *Software-Engineering mit UML und dem Unified Process*, Pearson Studium, München, Germany, 2001.
- [6] "EN 14908 – Open data communication in building automation, controls and building management – control network protocol", 2006.
- [7] "CANopen Framework for Safety-Related Communication", *CiA work Draft 304*, CAN in Automation e.V., 2000.
- [8] "PROFISafe, Profile for Failsafe Technology," *PROFIBUS-Nutzorganisation Karlsruhe*, v1.0, 1999.
- [9] P. Preininger, "Hardware Selftests For Safety Critical Fieldbus Nodes", *M.S.thesis*, Institute of Computer Technology, Vienna University of Technology, Austria, 2006.
- [10] M.C. Hsueh, T.K. Tsai, R.K. Iyer, "Fault Injection Techniques and Tools", *Computer*, Volume 30, pp. 75-82, 1997.
- [11] G.A. Kanawati, N. A. Kanawati. J.A. Abraham, "FERRARI: A Tool for The Validation of System Dependability Properties", *Twenty-Second International Symposium on fault-Tolerant Computing, FTCS-22. Digest of Papers*, pp.336-344, 1992.
- [12] J. Carreira, H. Madeira, J.G. Silva, "Xception: A Technique for the Experimental Evaluation of Dependability in Modern Computers", *IEEE Transactions on Software Engineering*, Volume 24, pp. 125-136, 1998.
- [13] Z. Segall, D. Vrsalovic, D. Siewiorek, D. Yaskin, J. Kownacki, J. Barton, R.Dancey, A. Robinson, T. Lin, "FIAT-Fault Injection Based Automated Testing Environment", *Eighteenth International Symposium on fault-Tolerant Computing, FTCS-18. Digest of Papers*, pp. 102-107, 1998.
- [14] S. Chau, "Fault Injection Boundary Scan Design for Verification of Fault Tolerant Systems", *Proceedings of International Test Conference*, pp. 677-682, 1994.
- [15] "IEEE Standard Test Access Port and Boundary-Scan Architecture", IEEE Std 1149.1-2001.
- [16] ARM7TDMI (Rev 4) Technical Reference Manual, ARM.
- [17] Segger, J-Link ARM API, "Users guide of the J-Link application program interface (API)", Version 3.20, Manual Rev. 1.