

# Automated Testing System for a CAN Communication Circuit

Florin PRUTIANU, Viorel POPESCU, Dan LASCU

Politehnica University Timisoara, Bd. Vasile Parvan 2, 300223, Timisoara, Romania

Electronics and Telecommunications Faculty, Applied Electronics Department

florin.prutianu@etc.upt.ro, viorel.popescu@etc.upt.ro, dan.lascu@etc.upt.ro

**Abstract**—The paper presents a method for validation/testing a control area network (CAN) communication circuit used in all electronic control units (ECUs) developed in automotive industry after 2000. Using a specific hardware configuration and remotely controlled by LabVIEW. The author's presents their own vision regarding operational software algorithm implementation and integration / execution of some test cases in order to validate a CAN circuit. Using this method, it is possible to validate/test CAN hardware circuits in a short time and with the possibility of saving the test results. Human operator is interfering with the system only through the graphical user interface. The error sources for this system are reduced to minimum.

**Index Terms**—automated measuring/validation/testing system, LabVIEW programming, testing algorithm, remote control.

## I. INTRODUCTION

Three important steps must be fulfilled when a product is developed: design, test and results analysis. In this paper an automated testing system that improves the testing/measurement/ integrating process is presented. As the trend in the last years is to reduce the time spent for product development, the process for product development was decreased from 3 years to 1.5 years. Consequently it became of crucial importance to develop new fast methods for testing/ measuring/validating in small execution times.

In actual automotive industry a great impact is represented by the testing/ measuring/ validating process of products. The future of these processes is based on automation and execution time reduction. With the growth of the modules that are automatically tested the time allocated for product design can be decreased. In the beginning of the automotive industry in a city car there were just a few ECUs responsible for implementing important functions in the vehicle like: board computer, start engine, management for fuel injection or comfort functions. Nowadays, in automotive industry one ECU is responsible for one or more functions. This was the reason why the answer to the needs of a standard communication protocol between these ECUs was first given by the Robert Bosch GmbH in 1983, while the official standard was launched in 1986. In 1992 the standard for control area network (CAN) communication protocol 2.0 [24][25] was first published.

The CAN bus circuit is used in all vehicles to connect the engine control unit to the transmission control unit, door

lock control unit, airbag units, seat control and/or other ECUs that exist in vehicle architecture. First the CAN was designed for automotive industry only, but many other fields have adopted this standard, such as industrial automation and medical equipment.

Considering all these industries that adopted this type of standard communication and corroborated with the need for having this circuit tested, more standards have appeared to clarify how this circuit could be tested [9],[10]. This paper is presenting a hardware concept for the automated testing system (ATS)[1]-[8],[11], a software (SW) algorithm containing the main program and the graphical user interface (GUI), both developed in LabVIEW, and test cases comprising the results obtained according to international standard ISOCD11898-2. Similar solutions are developed by Teradyne Co, but with equipment fixed in a rack or in a bloc and for one project. Basically for the ATS proposed by the authors in this paper, the test engineer can adapt the system to the product needs.

## II. ATS HARDWARE CONCEPT

Starting from the requirements of the CAN circuit [9],[10], it can be concluded that the equipment needed is that in Table I:

TABLE I. ATS EQUIPMENT

Name	Model	Communication Protocol	Quantity
LabVIEW (SW)	8.5	IEEE488, VISA, RS232	1 Licence
Computer	Dell Optiplex	USB, RS232, LAN	1
Power Supply	Agilent E3465A	IEEE488, RSR232	1
DMM	Agilent 34401A	IEEE488, RS232	1
Converter USB→GPIB	Agilent 82357A	USB→GPIB	1
Function Generator	Agilent 33220A	IEEE488, RS232	1
Oscilloscope (OSC)	LeCroy 424	LAN, RS232	1
Switch Unit	Agilent 34970A	IEEE488, RS232	1
Vector	CANcaseXL	CAN	1
Nails Bed	For each circuit it is own developed		1
Measuring Interface	For each circuit it is own developed		1

System design involves the hardware equipment presented in Table 1 and a programming language intended to remotely control all devices used in these test cases, with the facility to build a graphical user interface (GUI) [23]-[25]. The selected program for SW algorithm implementation is LabVIEW - version 8.5 [15]-[20]. More details related to this program are presented in the next paragraphs.

This work was partially supported by the strategic grant POSDRU/88/1.5/S/50783, Project ID50783 (2009), co-financed by the European Social Fund – Investing in People, within the Sectoral Operational Programme Human Resource Development 2007-2013.

Digital Object Identifier 10.4316/AECE.2012.02012

The devices are remotely controlled through an universal serial bus (USB)→general purpose interface bus (GPIB) converter, provided by Agilent. This converter is able to control up to 14 devices using the IEEE488 (GPIB) protocol [21],[22]. These devices have a unique address, with the possibility to control all devices in parallel.

The input voltage of ECU is provided by a power supply that in this case provides an input voltage in the range of 0V up to 40V and a 1.3A input current [14]. This corresponds to the automotive range where an ECU should be active.

A second requirement for the ATS is to have at least one measuring device where the required signals to be measured will be connected to. In this application a digital multimeter (DMM) or an oscilloscope were used. This chain architecture is possible only if the following hardware devices are present:

- A nails bed, with the needles connected to ECU test points for signal acquisition
- An interface to be built, namely connections between the nails bed and the measuring devices.
- Switch Unit – by closing a relay signals are routed inside the interface to a measuring device. In this switch unit more modules can be chosen. In this paper Agilent 34907A was selected.

The Vector CanCaseXL device it is used to start the ECU and to support the communication between computer (PC) and ECU.

In figure 1 the block diagram of the entire circuit of the ATS proposed by the author's is depicted.

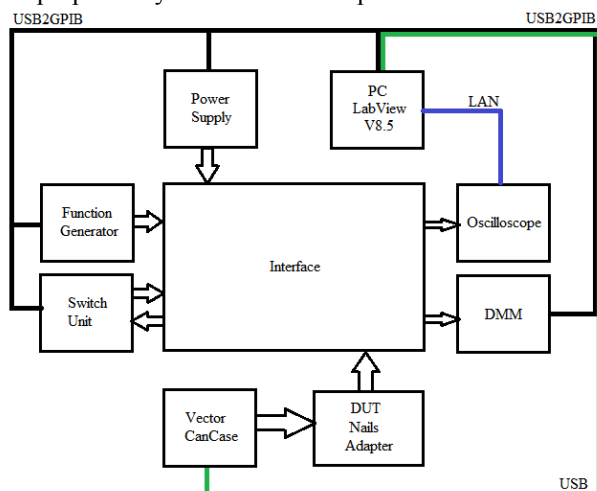


Figure 1. ATS block diagram.

It can be observed that three communication buses are managed. With these three protocols these devices are remote controlled. GPIB and LAN protocols are bidirectional and used for devices. USB is used to have the CAN communication active with the ECU, through Vector CanCase. Due to application software you may store pictures from the oscilloscope on a local hard drive, in a desired path on the local hard drive. This feature is an advantage when test results are presented or analyzed and when a test report is generated.

### III. SOFTWARE: GRAPHICAL USER INTERFACE AND ALGORITHM

LabVIEW is a graphical programming language that allows users to build graphical user interfaces, to create

algorithms for a function, to generate files on a local computer and/or to remote control all devices connected to the GPIB bus. LabVIEW is also able to control devices connected to Ethernet or to RS232 port [12],[13].

GUI is created through this visual programming language and exhibits the following proprieties [26]:

- It is intuitive/user friendly.
- Provides enabling/ disabling control of a device.
- Display of user & product information.
- Selects the test cases from a pop-up window
- Display of the results
- Enables test report generation

In the GUI the human operator must have the possibility to both set all important parameters for one test and be able to observe in every moment what is the status of a selected test in the results window.

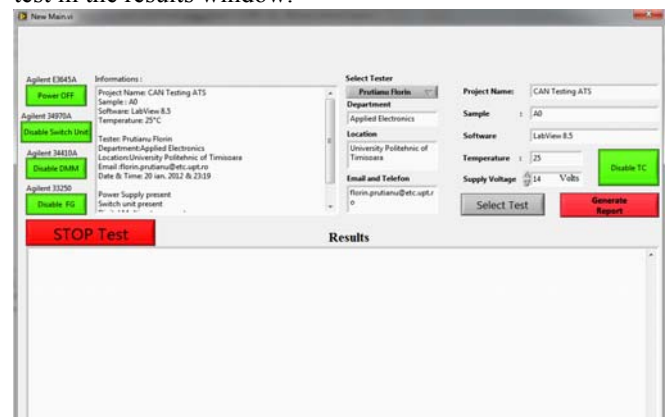


Figure 2. The GUI for the ATS.

Even if the graphical user interface is very intuitive and there is the possibility to prepare almost everything to perform a test, one important part consists in the main function when application software is executed. The main steps of the execution algorithm can be observed in figure 3.

The start in figure 3 is associated with a button that is present in the main Select Test window. This pop up window, presented in figure 4, shows up when the Select Test button from GUI is pressed.

An important step in each testing software algorithm is device initialization. In this sequence it can be for the first time observed whether a device responds correctly or it has some flaws. If all the selected/ enabled devices needed for performing the test case are not responding OK, then this would lead to a premature end of the SW routine and consequently the test cannot be performed.

When this step is ended with a positive response, the test cases should be selected. In the pop-up window shown in figure 4 all test cases that can be selected and performed are presented. The OK button in this window is assigned with Start Testing. When no test case (TC) is selected, the OK button is not working because nothing can start. Pop up window cannot be closed by pressing ok, only by pressing X button and that means the end of SW routine and for the main program.

When some test cases are selected, started and performed, the following step is to save all results. Normally the GUI provides a possibility to activate/deactivate test report generation. The reason of such an option is the following: when the TCs are created for the users for the first time it is not important to save all the results/reports, but it is more

important to display them in results window. Here, the test engineer can take decisions about what to change in the TC or to check the test specification (TS) once again and what equipment to use. After the test results are saved, the system is brought into the default state. The main program must be run once again to repeat or to select other TCs.

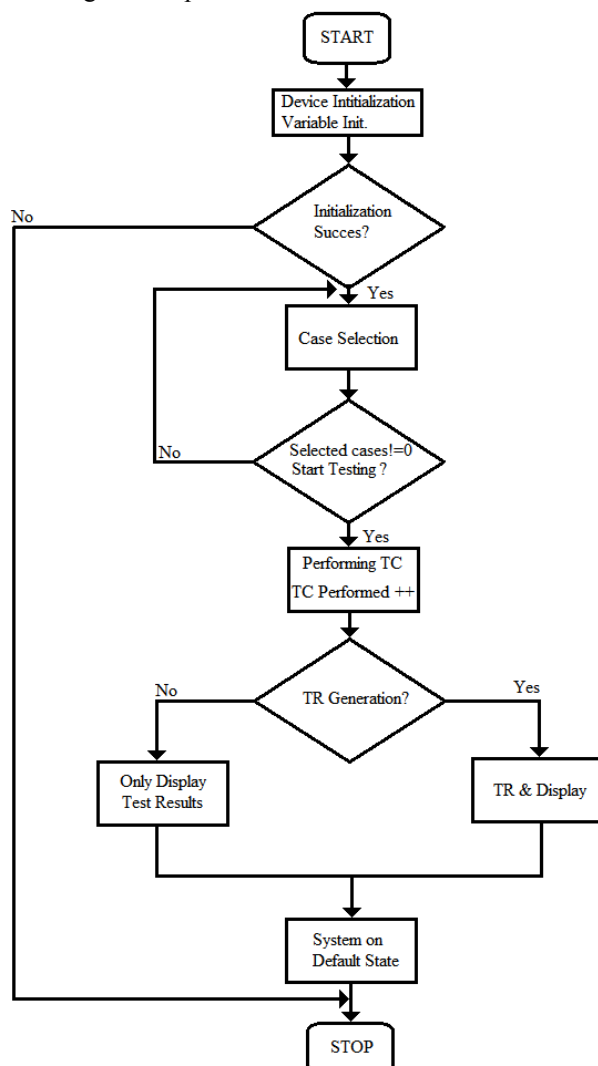


Figure 3. Main software algorithm.

For the test engineers the test execution algorithm shown in figure 5 is important, too. When this subroutine is called and performed, it can be observed that if no test case is selected then this subroutine doesn't do anything. If there some TCs are selected then all variables needed for performing the TCs are initialized. Due to the fact that most of the time also other users have access and rights to modify this program, a routine for checking if the correct variables are initialized and to verify/check if any errors occur is inserted into the algorithm.

If the variables cannot be initialized then it is concluded that there is a problem with the software, the subroutine is ended and the TCs cannot be performed.

If this sequence is successfully passed, it comes the part where all the devices are set according to the TS. For example, the voltage is set to 14V, relays no. 14, 15, 16, 19 are closed, the oscilloscope is set to be active with all 4 channels and its trigger is related to CH1 at 2.5V on a rising edge. All these can be done with the help of the GPIB bus and the LAN bus and are all managed by LabVIEW.

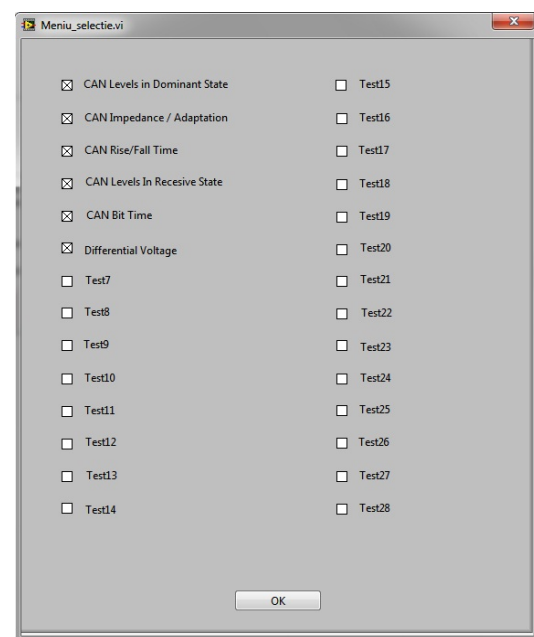


Figure 4. Test Case selection pop up window.

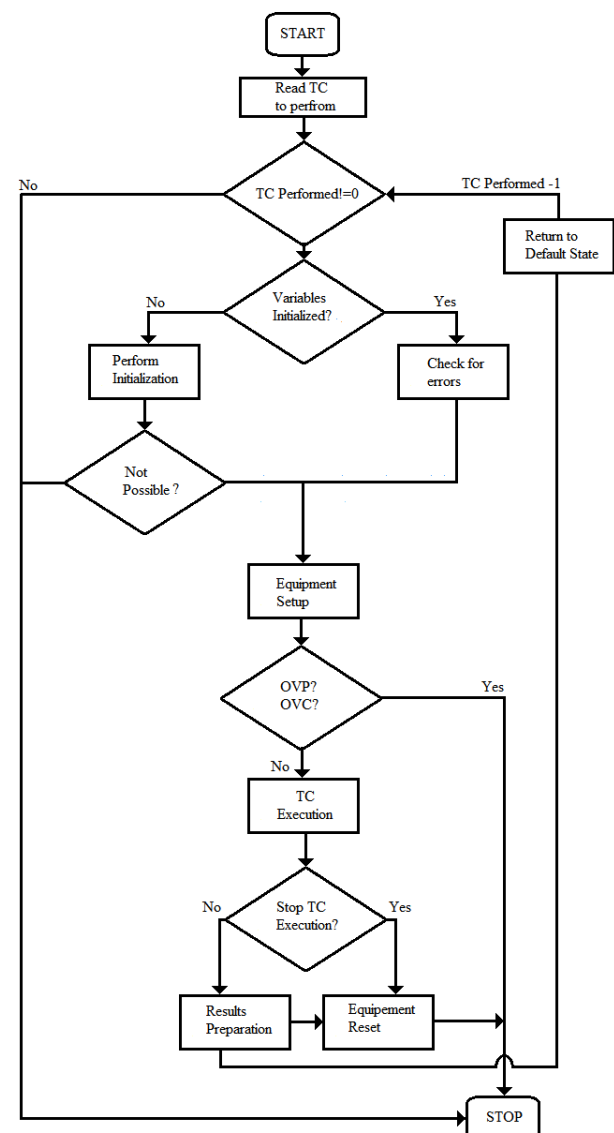


Figure 5. Software Test Case Execution algorithm.

After configuring all devices, a protection routine is implemented to check if any short circuit has occurred or if the input voltage is correctly set. When this protection routine test is not passed then no input voltage for the ECU is available. The responsible switch for powering up the circuit is closed only after it receives a positive response from the power supply.

After this routine the TCs are executed. TCs can also be stopped at this time if abnormal results are noticed. After test execution, some mathematical operations can be done with the obtained results, such as addition, subtraction, signal processing or comparison with other values. When the main scope of the TC validates one precise value by using the boundary method for testing, a quick result is obtained and displayed. Depending on the test engineer who develops the program, an option with “failed” and “passed” can also be added. When one test case is finished and the obtained results are prepared to be reported, the performed TC is deleted from the vector, all devices are brought to their initial state and the next test case is ready to be performed. If no other test cases are to be performed, this subroutine is ended and the program returns into the main program.

Of course, new options can be added and the application software can be further developed, but for the scope of this paper only these two subroutines were successfully performed and good results were provided.

#### IV. CAN CIRCUIT, TEST CASES & TEST SPECIFICATIONS

Even in this paper only one type of CAN circuit is presented, all other circuits are similar and should respect the standard developed by Robert Bosch in 1992. With the ATS presented in this paper all types of CAN circuits can be validated in less than 5 minutes.

In figure 6 the validated circuit, proposed by the author's, is enclosed.

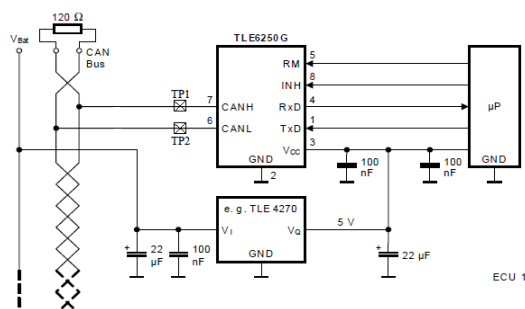


Figure 6. CAN circuit structure for automotive ECUs.

This circuit is a typical application. Each original equipment manufacturer (OEM) is developing its own circuit according to the international standards. In the sequel some typical CAN terms definitions and validation test cases are presented. Although various other test cases can be developed, for a basic validation of CAN circuit these TCs are sufficient.

$V_{CAN\_H}$  &  $V_{CAN\_L}$  are the bus voltages, CAN\_L (L=low) and CAN\_H (H=high) with respect to ground [25]. A first TC is to determine their level in the two logical states of the CAN circuit, that is in the recessive and in the dominant states.

In the recessive state,  $V_{CAN\_H}$  &  $V_{CAN\_L}$  are fixed to the average voltage level, depending on termination of the CAN

bus.  $V_{diff}$  represents the differential voltage between the two CAN lines.

$$V_{diff} = V_{CAN\_H} - V_{CAN\_L} \quad (1)$$

Dominant state means that value for  $V_{diff}$  is higher than a minimum threshold measured during a dominant bit. When another ECU is connected to the CAN bus and the communication is enabled in the transmitting/receiving state, then the condition for a dominant bit is fulfilled. When all ECUs are in idle state, then the recessive state/mode is present and no communication is available.

$R_{diff}$  – represents the differential internal resistance, measured during a recessive state when a CAN node is disconnected from the CAN bus.

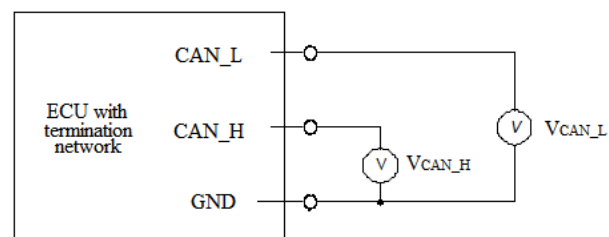


Figure 7.  $V_{diff}$  measurement setup.

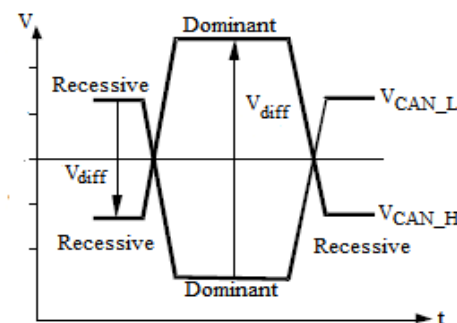


Figure 8. Voltages & physical bit representation.

Other test cases include  $V_{diff}$  value measurement in the recessive and dominant states. Of course, it is also important to see the front details of the voltage signals, like rise time, fall time and bit time/frequency for a dominant bit.

TABLE II – CAN BUS VOLTAGE PARAMETERS IN RECESSIVE STATE

Parameter	Notation	Unit	Value			Condition
			min	nom	max	
Common mode bus voltage	$V_{CAN\_H}$	V		2.5	7	Respect for GND, in each CAN node
	$V_{CAN\_L}$	V	-2.0	2.5		
Differential bus voltage	$V_{diff}$	mV	-120	0	12	Measured in each CAN node connected to the BUS

TABLE III – CAN BUS VOLTAGE PARAM. IN DOMINANT STATE

Parameter	Notation	Unit	Value			Condition
			min	nom	max	
Common mode bus voltage	$V_{CAN\_H}$	V		3.5	7	Respect for GND, in each CAN node
	$V_{CAN\_L}$	V	-2.0	1.5		
Differential bus voltage	$V_{diff}$	V	1.2	2.0	3.0	CAN nodes connected to the BUS

TABLE IV – MAXIMUM RATINGS OF  $V_{CAN\_H}$  &  $V_{CAN\_L}$

Nominal Battery Voltage	Notation	Unit	Value	
			min	max
12V	$V_{CAN\_H}$	V	-3.0	16
	$V_{CAN\_L}$	V	-3.0	16
24V	$V_{CAN\_H}$	V	-3.0	32
	$V_{CAN\_L}$	V	-3.0	32



TABLE V – DC PARAMETERS DURING RECESSIVE STATE FOR A CAN NODE

Parameter	Notation	Unit	Value			Condition
			min	nom	max	
Output Bus CAN Voltage	$V_{CAN\_H}$	V	2.0	2.5	3.0	No load
	$V_{CAN\_L}$	V	2.0	2.5	3.0	
Differential output CAN Bus Voltage	$V_{diff}$	mV	-500	0	50	No load
Differential Resistor	$R_{diff}$	k $\Omega$	10		100	No load
Internal Resistor	$R_{in}$	k $\Omega$	5		50	

TABLE VI – DC PARAMETERS DURING DOMINANT STATE FOR A CAN NODE

Parameter	Notation	Unit	Value			Condition
			min	nom	max	
Bus CAN Voltage	$V_{CAN\_H}$	V	2.75	3.5	4.5	No load
	$V_{CAN\_L}$	V	0.5	1.5	2.25	
Differential output Bus Voltage	$V_{diff}$	V	1.5	2.0	3.0	No load

## V. EXPERIMENTAL RESULTS

The two devices used for measuring the above parameters for the CAN circuit were the DMM and the OSC.

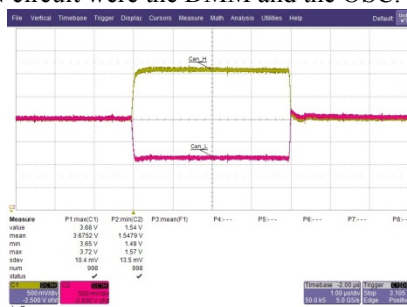
Figure 9. CAN Bus maximum ratings for  $V_{CAN\_H}$  &  $V_{CAN\_L}$ .

Figure 10. CAN voltage levels for the dominant state.



Figure 11. CAN Bus Voltages for the recessive state.



Figure 12. Rise Time for CAN\_H / Fall Time for CAN\_L



Figure 13. Fall Time for CAN\_H / Rise Time for CAN\_L

Table VII – Resistance for a CAN node

Parameter	Notation	Unit	Value
Differential Resistor	CAN_H & GND	k $\Omega$	16.53
	CAN_L & GND	k $\Omega$	16.54
R between terminals	CAN_H & CAN_L	k $\Omega$	2.6

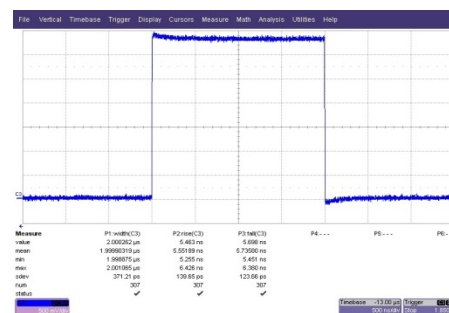


Figure 14. Bit width / time / frequency

## VI. CONCLUSIONS AND DISCUSSIONS

ATS proposed in this paper can be used to perform following measurements: voltage levels, rise times, fall times, pulse widths, frequencies and different parameters for an electrical signal can be measured. It can be observed that this ATS can be used both for circuit validation and for measurement/testing.

The proposed ATS represents a requirement of our day's automotive industry. The ATS proposed by the authors is having the following advantages:

- Short execution times, 3 to 5 time faster as manual testing
- Low cost for in a 5 year time period. Costs are high at the beginning, around 22.000 Euro, but decreasing with every project due to high reusability propriety. In the form presented in this paper one objective was to be built in such a way that all modules to be reused to other ATS. Thus functionality can be increased, and new features could be integrated.
- Measuring error introduced by the system is under

1%, measured on a 3V3 reference voltage. Theoretical value is  $3.300V \pm 1\%$  and measured value for this voltage with this system is 3.318V.

- no possibility that test results to be influenced by the human operator
- high repeatability, good traceability and quick test case execution

Also the measurement setup is easy to be built; full assembly of the system from ground to functional operation can take maximum three hours, including software installation and updates.

The algorithms for the application software can be further improved, if desired. Algorithms presented in figure 3 and 5 are representing the author's vision related to operational software. The purpose of this paper was just to validate the CAN circuit that is found in all automotive ECUs. Routines can be further developed, but even in this simple form this ATS can be a serious candidate when a quick solution for testing/ validating is needed. The ATS can be further developed for other modules from automotive ECUs like: power modules, digital modules, analog modules. Other test cases can be implemented and the number of TCs performed can be increased, this depending only on the ability of the test engineer. The number of devices remotely controlled can be increased to 14 plus two: one on the LAN and one on the RS232 communication protocol. To develop the test specifications and the TCs CAN international standard ISOCD11898-2 has been used. Other OEMs may have their own requirements, but most of the new requirements are based on this standard. This paper presented a minimal test setup/configuration for a CAN circuit validation.

The application software was built on a modular structure, so the modules from it can be used in other applications. There were created LabVIEW subroutines as standalone virtual instruments.

#### REFERENCES

- [1] Goad K.G., Tront J.G., McKeeman J.C., "A PC-based automated guided probe testing system", System Theory 1989 Proceedings, Twenty-First Southeastern Symposium on, pp. 345-348, March 1989 Available: [http://ieeexplore.ieee.org/xpl/freeabs\\_all.jsp?arnumber=72489](http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=72489)
- [2] Hawrylak P.J., Ogirala A., Cain J.T., Mickle M.H., "Automated Test System for ISO 18000-7 Active RFID", RFID, 2008 IEEE International conference, pp. 9-18, April 2008, Available: [http://ieeexplore.ieee.org/xpl/freeabs\\_all.jsp?arnumber=4519355](http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=4519355)
- [3] Zhang Wenfan, Liao Hui, "Design and research of performance of automated test system of electro-hydraulic proportional valve", International Conference on Electronics, Communication and Control (ICECC) 2011, pp. 1989-1991, September 2011, Available: [http://ieeexplore.ieee.org/xpl/freeabs\\_all.jsp?arnumber=6066615](http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=6066615)
- [4] Jiangguang Jia, Jingming Kuang, Zunwen He, Jun Fang, "Design of automated test system based on GPIB", ICEMI '09<sup>th</sup> International Conference on Electronic Measurement & Instruments 2009, pp. 1-943 - 1-948, August 2009, Available: [http://ieeexplore.ieee.org/xpl/freeabs\\_all.jsp?arnumber=5274384](http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=5274384)
- [5] Celanovic N.L., Celanovic I.L., Ivanovic Z.R., "Cyber Physical System: A new approach to power electronics Simulation, Control and Testing", vol.12, pp.33-38, February 2012, Available: <http://www.aece.ro/abstractplus.php?year=2012&number=1&article=6>
- [6] Ono K., Ohmori K., Fujii T., Asakura T., "Automated testing system for switching systems with multiple modules", International IEEE Conference on Communications, 1997 ICC Montreal Towards the Knowledge Millennium, pp. 236-240, Jun. 1997, Available: [http://ieeexplore.ieee.org/xpl/freeabs\\_all.jsp?arnumber=605213](http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=605213)
- [7] Hongbo Yu, Yihua Lan, Haozheng Ren, "The Research about an Automated Software Testing System Run Tool", 2011 3<sup>rd</sup> International Workshop on Intelligent Systems and Applications USA, pp. 1-4, May 2011, Available: [http://ieeexplore.ieee.org/xpl/freeabs\\_all.jsp?arnumber=5873331](http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=5873331)
- [8] Czapski J., "Useful features of automated test systems in the R&D laboratory", 2000 IEEE Autotestcon Proceedings, pp. 601-613, Sep. 2000, Available: [http://ieeexplore.ieee.org/xpl/freeabs\\_all.jsp?arnumber=885647](http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=885647)
- [9] Novak J., "Flexible approach to the Controller Area Networks test and evaluation", 2009 IDAACS IEEE International Workshop on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, pp. 44-48, Sept. 2009
- [10] Novak J., Kocourek P., "Automated Testing of Electronic Control Units Compatibility in Vehicle CAN Networks", 2005 Proceedings of the IEEE International Symposium on Industrial Electronics, pp. 1423-1428, June 2005.
- [11] Lindlar F., Windisch A., "A Search Based Approach to Functional Hardware in the Loop Testing", 2010 Second International Symposium on Search Based Software Engineering (SSBSE), pp. 111-119, Sept. 2010, Available: [http://ieeexplore.ieee.org/xpl/freeabs\\_all.jsp?arnumber=5635149](http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=5635149)
- [12] Bruno L., Andre M., "A Comparison between LAN and GPIB Instrument Interfaces in a Automated Test System", 2007 ICEMI 8<sup>th</sup> International Conference on Electronic measurement and Instruments, pp. 1761-1762, Aug. 2007, Available: [http://ieeexplore.ieee.org/xpl/freeabs\\_all.jsp?arnumber=4350562](http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=4350562)
- [13] Zulkifli M.Z., Harun S.W., Thambiratanam, Ahmad H., "Self Calibrating Automated Characterization Systems for Depressed Cladding EDFA Applications Using LabVIEW Software with GPIB", IEEE transactions on Instrumentation and Measurement, pp. 2677-2681, Nov. 2008, Available: [http://ieeexplore.ieee.org/xpl/freeabs\\_all.jsp?arnumber=4534703](http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=4534703)
- [14] Goulart T.J., Consonni D., "Automated system for measuring electrical three-phase power components", IEEE Transaction on Education, pp. 336-341, Nov. 2001, Available: [http://ieeexplore.ieee.org/xpl/freeabs\\_all.jsp?arnumber=965781](http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=965781)
- [15] Jan G.J., Law S.F., "Implementation and use of general purpose interface bus interface in an automatic testing and measurement system of a management power supply", Review of Scientific Instruments, pp. 2488-2493, Nov. 1988
- [16] Neskovic M.D., Spasic J.S., Celebic V.V., Salom I.M. (2010) "An Automated hardware testing using PXI hardware and LabVIEW software", 2010 5<sup>th</sup> European Conference on Circuits and systems for communication (ECCSC), pp. 232-235, 2010
- [17] Leikai Ma, Liang Xu, Kunpeng Zhang, Wei Wu, "The measurement of series and shunt resistance of the silicon solar cell based on LabVIEW", 2011 International Conference on Electrical and Control Engineering (ICECE), pp. 2711-2714, Sept. 2011.
- [18] Swain N.K., Anderson J.A., Ajit Singh, Swain M., Fulton M., Garrett J., Tucker O., "Remote data acquisition, control and analysis using LabVIEW front panel and real time engine", IEEE Proceedings 2003 SoutheastCon 2003, pp. 1-6, April 2003, Available: [http://ieeexplore.ieee.org/xpl/freeabs\\_all.jsp?arnumber=1268423](http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=1268423)
- [19] Li Pengfei, Nie Luhua, "Remote Control Laboratory Based on LabVIEW", ICICTA '09 Second International Conference on Intelligent Computation Technology and Automation, pp. 84-87, Oct. 2009
- [20] Zhang Xinjun, Tashiro Y., Takata H., Fujii T., Yamamuro M., Kasai D., Shimomura T., "An approach to automated program testing and debugging", 1999 Proceedings Sixth Asia Pacific in Software Engineering Conference, pp. 582-589, Dec. 1999
- [21] www. Agilent.com, "Test system Development Guide ", Available: <http://cp.literature.agilent.com/litweb/pdf/5989-5367EN.pdf>
- [22] www.agilent.com, "Agilent Connectivity Hardware for PC-to-Instrument Connections", Available: <http://cp.literature.agilent.com/litweb/pdf/5989-1889EN.pdf>
- [23] www.ni.com, "Designing Automated Test Systems", Available: [http://download.ni.com/evaluation/ate/designing\\_automated\\_test\\_system\\_guide.pdf](http://download.ni.com/evaluation/ate/designing_automated_test_system_guide.pdf)
- [24] International Standard, Road Vehicles – Controller Area Network Part II: High-speed access unit, ISOCD 11898-2
- [25] International Standard, Road vehicles – Controller Area Network Part V, ISO11898-5
- [26] D. Skvorc, I. Zuzak, S. Srblic, "Formal Model for Data Dependency Analysis between Controls and Actions of a Graphical User Interface," Advances in Electrical and Computer Engineering, vol. 12, no. 1, pp. 63-70, 2012. [Online]. Available: <http://dx.doi.org/10.4316/AECE.2012.01011>