

IoT challenge: older test machines modernization in an automotive plant

Juraj Pančík¹, Vladimír Beneš²

Abstract. In presented paper we describe engineering design and implementation of an information system based on IoT approach for remote collection, storing and processing information on the number of carried out mechanical cycles for group of older existing fatigue test machines in an automotive components production. The modernization of existing test machines infrastructure we built on the electronic counter modules with developed Ethernet interface. We carried out three ways to verify of accuracy of this manner counting and collecting of mechanical cycles and we reached results comparable with manual collecting of data. Thanks to automatic detection of ends of fatigue tests we reached more effective utilization of test machines (more than 25%).

Keywords: fatigue tests; mechanical cycles counting; process optimization; automotive testing

1 Introduction

What is modernizing in our case? Simply stated it is the addition of existing machines designed for fatigue tests of components in an automotive plant. With IoT approach we developed an information system for collection, storage and analysis of data. Goal was not just to get the data but also to transform data to information. Task seems simple but transforming data into valuable information is no small task. As a starting point, we asked ourselves what value of the obtained data and derived information for procedural role before and after implementation of information system. The most important thing is to maximize utilization of testing machines. Fatigue tests have not clearly defined end time of the tests. So the know when to stop testing machines is a key information for efficient test machine utilization. Another objective was to facilitate the automatic processing of data in the VBA language (Visual Basic for Application is script language for Microsoft Excel spreadsheet in order monitor and visualize data.

¹ Bankovní institut vysoká škola, a.s., Prague, Department of Informatics and Quantitative Methods, Nárožní

2600/9, 158 00 Praha 5, Czech Republic, email: jpancik@bivs.cz

² Bankovní institut vysoká škola, a.s., Prague, Department of Informatics and Quantitative Methods, Nárožní

2600/9, 158 00 Praha 5, Czech Republic, email: vbenes@bivs.cz

1.1 Existing fatigue testing machines

A fatigue testing machine may be classified from different viewpoints such as purpose of the test, type of stressing, means of producing the load, operation characteristics, type of load etc. [1]. Base on purpose of test, fatigue testing machine can be divided into following [2]: 1. General purpose fatigue testing machine, 2. Special purpose fatigue testing machine, and 3. Equipment for testing parts and assemblies (it was our case)

A. Cycle counting

Cycle counting is used to summarize (often lengthy) irregular load-versus-time histories by providing number of times cycles of various sizes occur. The definition of a cycle varies with the method of cycle counting. These practices cover the procedures used to obtain cycle counts by various methods, including level-crossing counting, peak counting, simple-range counting, range-pair counting, and rain flow counting. Cycle counts can be made for time histories of force, stress, strain, torque, acceleration, deflection, or other loading parameters of interest [3].

B. Specifications of the mechanical cycle

According [3] we applied simple-range counting method. For this method, a range is defined as the difference between two successive reversals, the range being positive when a valley is followed by a peak and negative when a peak is followed by a valley. Positive ranges, negative ranges, or both, may be counted with this method. If only positive or only negative ranges are counted, then each is counted as one cycle. In this text we mean term mechanical cycle as synonymous for one cycle.

C. Electromechanical digital counter

Electromechanical counters are digital counters that serve as a part device of test machines. They give the equivalent value of number of mechanical cycles. In this case the electromechanical counter is synonym of the totalizing counter. Totalizing counter displays a total of events, parts, products, strokes, revolutions, etc. Electronic and electromechanical counters accept a variety of inputs, from sensors, switches, encoders and relays that increment the counter each time a pulse is received. In our case original digital totalizing counter accept inputs 24 V pulses from test machine at end of each mechanical cycle.

2 IoT modernization challenge: information system design objectives

Our goal was creating the information system for collecting, storing, processing and visualization of data from a set of test machines with digital electromechanical totalizing counters. The design of this information system has to take into account existing information system with electromechanical counter of mechanical cycles and with manual data collection. The new information system must enable remote data collection and must take advantage of existing plant network infrastructure at plant (LAN network). As secondary objectives are: price of solutions using modern technology based on the IoT (Internet of Things), ease of implementation, scalability, Web-based user interface and use of modern tools for data processing and their analyzing. Manager's objective is to obtain an overview about of utilization of the testing machines, reducing of downtime of test machines and the developing of visualization management dashboard based on web technologies.

2.1 System requirements

- The need to maintain existing electromechanical totalizing counters
- Increasing of the utilization efficiency of testing machines by reducing downtime (note : tested parts will be suddenly destroyed and test engineers should be informed about it in the shortest time)
- Automated data collection will replace the manual collection of data from electro - mechanic counters whose is in connection with manual reading and writing data into the paper sheets
- As legacy there is the need of ensuring reading data from electromechanical counters (manual reading) and simultaneously from electronic modules (automatic reading)
- The utilization of existing network infrastructure so that existing counter of mechanical cycles will be resolved with IP-based electronic counter module (IP - Internet Protocol) which will have LAN network connectivity
- The need to centralize data on accumulated mechanical cycles to the database
- The need for data evaluation and visualization in one place using the application server
- Take advantage of client - server architecture, where the client is IP counter module and server will consist of web server, application server and database
- To ensure long-term reliability and accuracy of mechanical cycles counting
- To build application servers on the basis of free available open source tools/components
- Good price of a programming and hardware solutions

2.2 Electronic Counter IP modules Requirements

- The counting inputs are electrical pulses from test machine with positive amplitude 0 - 24V (this is also parallel input to the existing electromechanical counter (totalizing counter)) with 2 Hz input frequency (therefore 2 mechanical cycles per second)
- In the COUNT mode the electronic counter divides of input pulses from testing machine in any ratio (the grouping of mechanical cycles). It will generate output information for application server based on defined number of input pulses
- Electronic counter IP module has possibility of operating in a simulated input pulses counting mode (PHANTOM mode). In this mode it automatically generates output information the application server
- IP module supports common internet protocol

2.3 Application server requirements

- Possibility of parallel information process entry from at least 8 IP modules (at frequency of 2 Hz input pulses for each IP module, such way the overall frequency can reach 16 Hz)
- Free of charge LINUX operating system
- Graphical visualizing web pages (document) generation with application server databases as sources of data
- Common software framework will serve as base for application server building

3 Design and implementation

3.1 Design scope

Proposal of our information system is based on the future use of technology IoT (Internet of Things). We decided at this moment for SNMP – it is well known Internet protocol [6]. Next we decided to take on the market available IP hardware module with SNMP and its known hardware and software solution. Later we can change this decision and we can select more modern IoT protocol as MQTT, DDS etc. Another basis for system design is using of cheaply operating system and database server. We are using Linux operating system and MySQL database

server. Very important for subsequent processing of collected data is ability to export data (CSV file extension) and ability to direct connection of appropriate analytical server to the MySQL database.

3.2 Architecture of the information system

The system architecture is described at Fig.1. We consider that this figure clearly illustrates all components of the information system. Explanation is needed for embedded web server, which serves to collect data on temperature and humidity from sensor. Like IP module for electronic counter this is another type of manufactured IP module [4]. This IP module communicates with its surrounding with HTTP protocol. Our application server respects this and managed provided data as the client. The client parses receive messages via HTTP.

3.3 Electronic counter IP module

The core of electronic counter is frequency divider with Atmel ATtiny85 and IP module TCW112-CM [5] (Fig.2). Optically isolated input signals ensure fault tolerance. Frequency divider mode switch ensures COUNT and PHANTOM mode mentioned in requirement part. Frequency divider board is powered from the IP module. The frequency divider divides the frequency of the input pulses with split ratio 1: 100. Output of the frequency divider ensures a signal for digital input of the module TCW112-CM. On this base the IP module creates one SNMP event per each 100 input pulses (mechanical cycles) which is processed by application server. To avoid of generating SNMP events that belongs to analog input we joined analog input with analog power (Fig.2).

3.4 Application server side description

On the server side we developed the application with name 'Monitoring Dashboard'. This web application was written on the framework Ruby on Rails, or simply Rails. The Rails is a Model–View–Controller (MVC) architecture framework, providing default structures for a database, a web service, and web pages. It encourages and facilitates the use of web standards such as JSON or XML for data transfer, and HTML, CSS and JavaScript for display and user interfacing [7]. The application server can be divided into three parts:

- Web server (front-end). It consists from a web-based user interface, which provides the user output (collected data from database), as well as provid-

ing user input (system configuration or data export). The main advantage of web applications is that the user requires no installation and they are independent of platforms (OS) and device (PC, tablet, smart phone) which users access the system. The only requirement is a modern web browser. The web server ensures the user interface for front-end. The application 'Monitoring Dashboard' uses the web server Puma.

Database systems. Relational database management system MySQL records mechanical cycles and also stores system configuration. It consists from several tables with one-to-many relations. Non-relational database system Redis is used by task queues system with name Net-SNMP. The Net-SNMP is packet of tools for work with SNMP protocol. It ensures the receiving of SNMP traps, the notification coming from the electronic counter IP module, processing ones and it ensures its subsequent forwarding to the application server (backend).

Application server (Backend). It consists from a set of components that are invisible to ordinary users. Backend application is designed for the Linux platform (ideal for Ubuntu LTS). Backend ensures the data stream designed for processing each received SNMP messages (each of them represents 100 input mechanical cycles):

1. Electronic counter IP module: sends SNMP trap notification
2. Net-SNMP : SNMP trap notification acceptance (snmpd Linux process), processing notifications (snmptt Linux process), transmission of relevant data to the web application
3. Backend : receiving data, building job, insertion of the job to the Task Queue (task-queue system is used to prevent the system bottlenecks)
4. Backend : job worker processes tasks in the task queue and writes data to the relational database

3.5 Verification of accuracy of counting of mechanical cycles

Counting of mechanical cycles is a long process. During this time, the number of mechanical cycles lies in orders of hundreds of thousands or millions. It is very important to be sure that electronic counter IP module counts mechanical cycles correctly and without errors. We checked three ways to verify the accuracy of counting of the mechanical cycles:

- The verification of the operation of IP module electronics and verification of frequency divider solution on the AMEL AVR base. We used Nanoline PLC (Programmable Logic Controller) from Phoenix Contact, USA, which generates the exact number of 24V pulses as input for electronic counter IP module. The number of pulses generated by the PLC Nanoline is compared with the number of received SNMP traps. Here, we have gained knowledge of the electrical interferences at unconnected inputs (they generate false SNMP traps) that led to the

grounding of the analog input of TCW112CM (see Fig.2, the shortcut between AIN and +VA inputs).

The verification of the application server data throughput. The aim was to verify the application server data throughput so that we have verified the ability to capture SNMP traps in real time. Real-time verification is done by eight IP modules worked in PHANTOM mode. Each frequency divider generated in this mode one pulse per minute, which means that an application server received eight SNMP traps per minute. We compared the number of application servers recorded SNMP traps with the calculated number of pulses generated by a calculation of time multiplied by input pulses frequency.

The last method uses comparison of recorded values of mechanical cycles in the application server data-base with the values recorded by totalizing counter.

3.6 Developed tools for process control, data visualization a data processing

- Information system administrator's role

The administrator can set necessary settings like IP address of the IP counter module, his name and another via dedicated web interface generated by web server. As an example, the Figure 3 shows information about performed cycles for all IP counters.

- Test engineer's role

The test engineer is checking the end of fatigue test with two independent interfaces: with EXCEL sheet and with web interface. The graphical interface program module "Semaphores" developed in EXCEL VBA serves for visualizing of running and stopped end of tests (Fig.4). The basic display of measured mechanical cycles is reached through the website. The Fig.5 shows the hourly number of reported SNMP traps for one IP module (1 SNMP trap = 100 mechanical cycles). Next necessary information are temperatures and humidity in the test laboratories. One example of graphical representation of measured temperature and humidity in the test laboratory is at Fig.6. The email is sent to the test engineer automatically in case of test end. This is ensured with small VBA macro and installed Microsoft Outlook application.

- Manager's role EXCEL VBA based dashboard

The manager can reach information about utilization of the test machines during one day or one month with EXCEL sheet powered by VBA program. His goal is also to ensure the maximum utilization of test machines also during weekends and nights. The graphical interface program module "History - Month" and "His-

tory - Day” developed in EXCEL VBA serves for visualizing history of tests. These manager’s dashboards are showed at Figure 7 and at Figure 8. Test manager can choose arbitrarily value of date (year, month and day) and he can observe KPI of utilization of tests machines very well.

- Future data analytics role

Processing capabilities and data analysis we are planning to extend with the R language server based on free of charge Microsoft (previous Revolution) R Server. Revolution R open–source software allows to R users to process, visualize, and model terabyte- class data sets at a fraction of the time of legacy products without requiring expensive or specialized hardware. Data analyst works in R language and he can reach data from exported CSV files or he can read data from MySQL server direct with R MySQL package [8].

4 Test processes optimization

More test’s processes were influenced with modernized architecture. Thanks to automatic detection of ends of fatigue tests we reached more effective utilization of test machines (more than 25%). Test machine automatic stops of test after destroying of component. The software application recognize mechanical cycle delay and send email to the responsible persons (technician and test engineers) with information about test finish. Further optimization of processes results from use of databases and data processing using software tools such as EXCEL and R language.

5 Conclusions

In presented paper we describe engineering design and implementation of an information system for re-remote collection, storing and processing information on number of carried out mechanical cycles for group of older existing fatigue test machines in an automotive components production. The modernization of existing test machines infrastructure we built on the electronic counter modules with Ethernet interface developed by us. Electronic counter modules in the role of clients replaced existing electromechanical totalizing counters. Web and application servers were built on Linux operating system. The MVC architecture (Model-View-Controller) serves as the base for web application and it was written together with backend application server around the Ruby on Rails web framework. The database is built on MySQL server. We carried out three ways to verify of accuracy of this manner counting and collecting of mechanical cycles and we reached results comparable with manual collecting of data. Thanks to automatic detection of

ends of fatigue tests we reached more effective utilization of test machines (increasing of utilization is more than 25%). Our possible contribution is in low cost upgrade older test machines with embedded platform, free of charge LINUX server and web application framework without need of buying expensive industrial grade communication systems. The data analytical services we are planning to built around free of charge Microsoft (Revolution) R open - source software. There is possibility to prepare reports and managers dashboards with common spreadsheet also (Microsoft EXCEL). Our solution of upgrading of existing industrial infrastructure for fatigue tests is very reasonably priced and is suitable for older industrial facilities.

References

- [1] WALODDI, W. Fatigue Testing And Analysis Of Results. Pergamon Press, 1961. ISBN 1483154165.
- [2] GBASOUZOR, A.I., OKEKE,O.C ., CHIMA,L.O.: Design and Characterization of a Fatigue Testing Machine. San Francisco, USA , 2013. Proceedings of the World Congress on Engineering and Computer Science 2013 Vol I.
- [3] ASTM International. Standard Practices for Cycle Counting in Fatigue Analysis, Designation: E 1049 – 85 (Reapproved 2005). ASTM International, 100 Barr Harbor Drive, PO Box C700, West Conshohocken, PA 19428-2959, United States., 2005.
- [4] Teracom Ltd. TCW122B-CM - Remote environmental monitoring. [Online] Teracom Ltd. [30.9.2016] <http://www.teracom.cc/products/tcw122b-cm-remote-environmental-monitoring/>.
- [5] —. TCW112-CM - Environmental IP monitoring board. [Online][30.9.2016.] <http://www.teracom.cc/products/tcw112-cm-environmental-ip-monitoring-board/>.
- [6] MAURO, D.R., SCHMIDT, K.J. Essential SNMP. Sebastopol: O'Reilly Media, Inc., 2005. ISBN: 0-596-00840-6.
- [7] HARTL, M. Ruby on Rails Tutorial: Learn Web Development with Rails. Addison-Wesley, 2015. ISBN: 0134077709.
- [8] Revolution R Open The world's most popular data analysis software, enhanced. [Online] Revolution Analytics. [Access: 30. 9. 2016] <http://www.revolutionanalytics.com/revolution-r-open>.

Figure 1. Architecture of the system for collection data from fatigue test machines in automotive industry plant

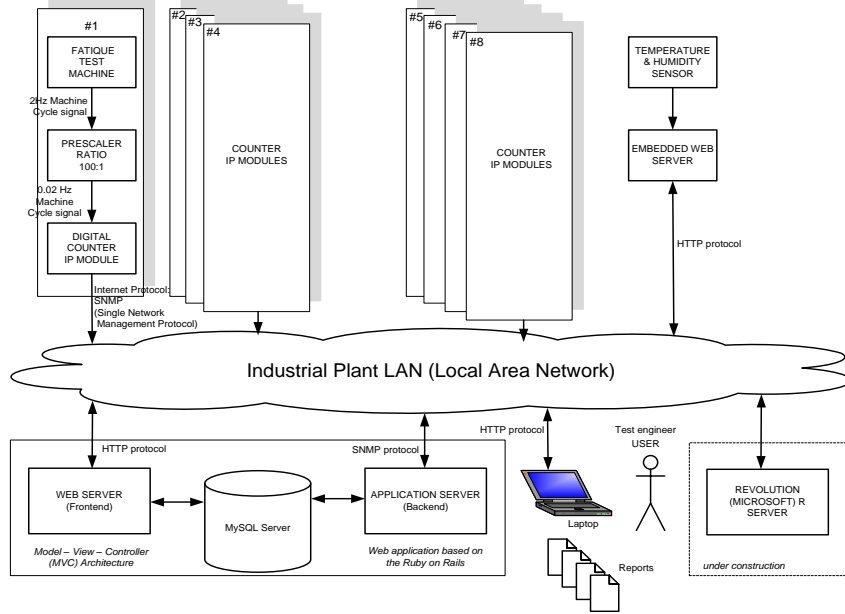


Figure 2. Electronic counter IP module – signal processing diagram

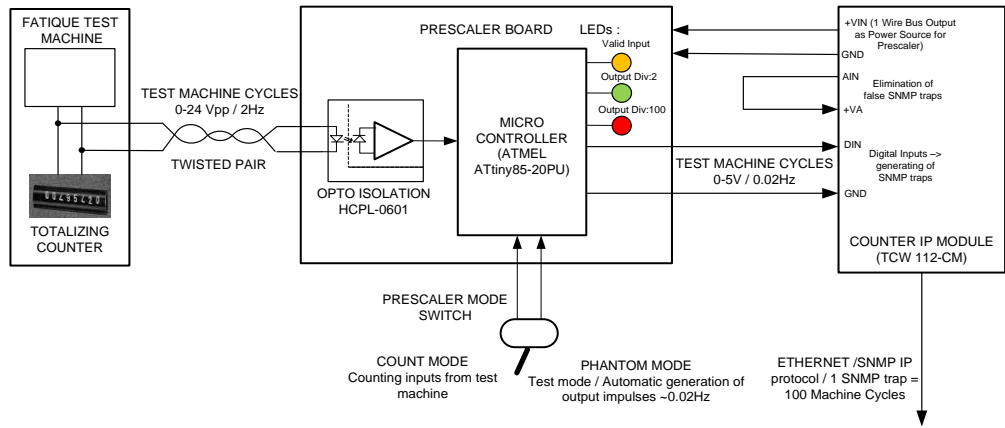


Figure 3. The example of administrator’s web interface - information about performed mechanical cycles for all test machines

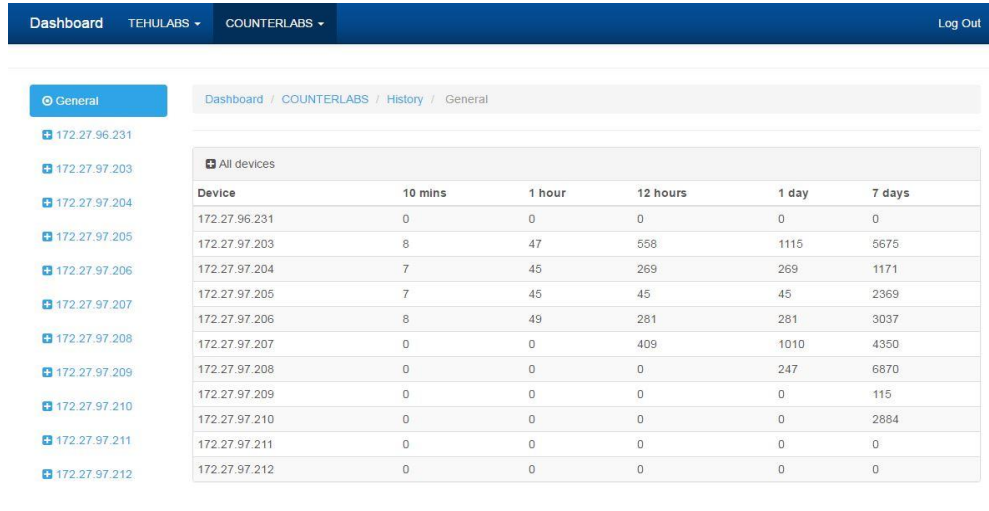


Figure 4. The graphical interface program module “Semaphores” serves for visualizing of running and stopped tests

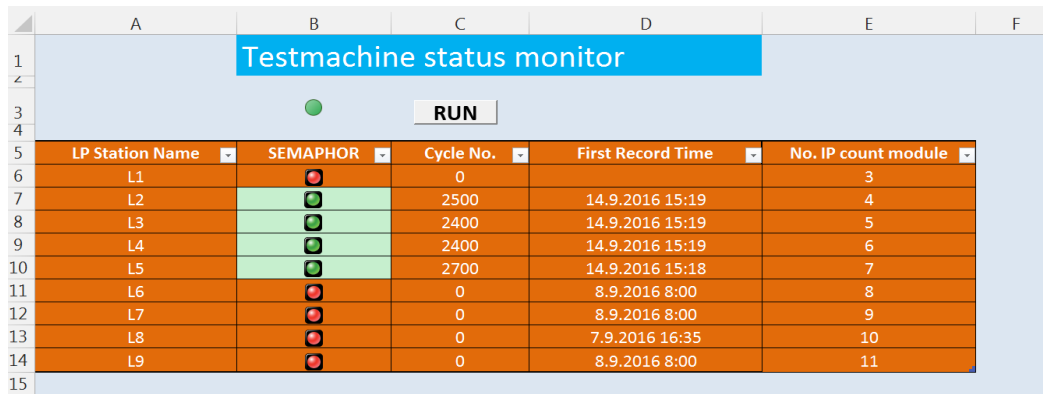


Figure 5. Web graph visualization of performed mechanical cycles (x100) for 12 hour – displayed for one electronic counter IP module

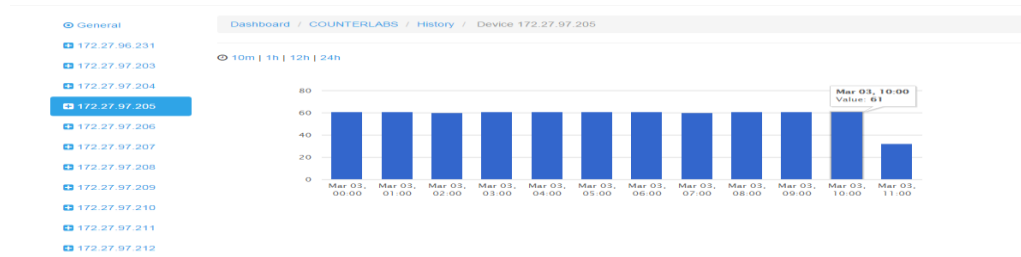


Figure 6. Web graph visualization of measured temperature and humidity in the test laboratory

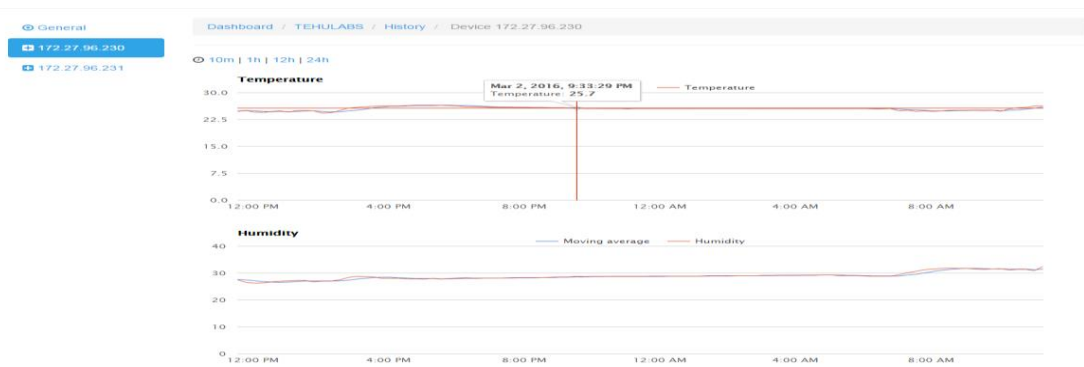


Figure 7. EXCEL and VBA dashboard visualization of one day mechanical cycles history and test machines utilization

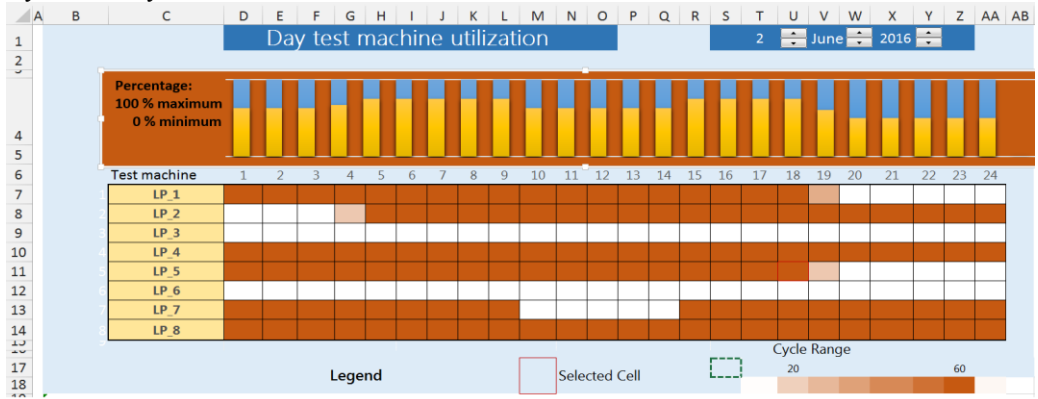


Figure 8. EXCEL and VBA dashboard - visualization of one month mechanical cycles history and test machines utilization

