Original article
DOI: 10.21683/1729-2646-2017-17-2-36-40

Functional dependability of the display unit software of the BLOK system

Efim N. Rozenberg, *JSC NIIAS, Moscow, Russia*
Natalia G. Penkova, *JSC NIIAS, Moscow, Russia*
Alexander S. Korovin, *JSC NIIAS, Moscow, Russia*


Efim N. Rozenberg


Natalia G. Penkova


Alexander S. Korovin

Abstract. Aim. *The article is dedicated to the challenges of evaluating the functional dependability of the display unit software (SW) that is part of the BLOK vital integrated onboard system as attributed to program errors within a 24-hour target time. One of the key tasks is the calculation of the values of such SW functional dependability characteristics as accuracy, correctness, security, controllability, reliability, fault tolerance and availability, which are the primary indicators for evaluating the health of safety devices. With all this taken into account, it is to be evaluated whether the checking of the software of the display unit before each trip with a departure test is required.* **Method.** *The reference conditions do not contain statistical data of program executions over the course of its maintenance. There is also no information on the structural characteristics of the program (number of operators, operands, cycles, etc.) which prevents the use of statistical models of dependability, such as the Halstead metrics, IBM model or similar ones. That is why the Schumann model was chosen as the initial data definition apparatus. The method of evaluation of the display unit's functional dependability is based on the findings of [1].* **Results.** *At the first stage, the following initial data values were defined: initial number of defects in the program, program failure rate and probability of correct run. At the subsequent stage, the identified values were used to define such dependability parameters as probability of no-error as the result of program run within a given time, probability of no-failure of display unit as the result of program run within a given time and mean time to program failure. After the probability $P_{SW}(t)$ of no-error as the result of program run within a given time was calculated, such SW dependability attributes as accuracy, correctness, security and controllability were evaluated. After the probability of no-failure of the display unit $P_R(t)$ as the result of program run within a given time was calculated, an evaluation was given to such attributes as SW reliability and fault tolerance, while after the mean time to program failure T_{avSW} was calculated, knowing the mean downtime due to elimination of the program error τ_{pdt}, the display unit availability for faultless execution of an information process at an arbitrary point in time C_{fa} was defined. The calculated partial functional availability coefficients for the display unit have shown that pre-trip checking of the unit and immediate elimination of errors, should such be identified, will enable a significant improvement of user performance of the onboard display unit (BIL) in terms of timely notification of the driver on the current operational situation to enable timely train control decision-making.*

Keywords: *BLOK Vital Integrated Onboard System, display unit, functional dependability.*

For citation: *Rozenberg EN, Penkova NG, Korovin AS. Functional dependability of the display unit software of the BLOK system. Dependability 2017; 2: 36-40. DOI: 10.21683/1729-2646-2017-17-2-36-40*

The Vital Integrated Onboard System (BLOK) is an advanced onboard train protection solution that is widely deployed on the Russian railway network. BLOK replaces and integrates the functionalities of such onboard safety devices as KLUB (Integrated Onboard Safety Device), SAUT (Automatic Brake Control System) and TSKBM (Remote Driver Vigilance Supervision System).

BLOK is designed to ensure train protection on lines with autonomous and electric traction equipped with trackside devices of the ALSN (continuous automatic cab signalling with digital coding), ALS-EN (multiaspect continuous automatic cab signalling with phase difference modulation of the carrier frequency) and SAUT systems, digital radio, discrete communication devices, coordinate-based train separation systems, as well as lines equipped with semi-automatic block devices.

A significant role in ensuring traffic safety is given to man-machine interaction that is provided by information display and control devices. The BLOK system includes a display unit. The unit displays all the required information on the operational situation along the line and operation of the onboard equipment, which enable the driver to successfully solve problems, should such arise.

Let us consider an example of evaluation of functional dependability of the display unit software that is part of the BLOK vital integrated onboard system.

Results of display unit software testing

The display unit software is designed to display operational situation to the driver in real time.

The software was tested at the debugging stage. The connection of the display unit is shown in the structural diagram in Figure 1.

The structural chart includes:

- power supply unit;
- onboard display unit (BIL);
- personal computer with simulation software;
- Kvaser, USB to CAN (Controller Area Network) interface converter.

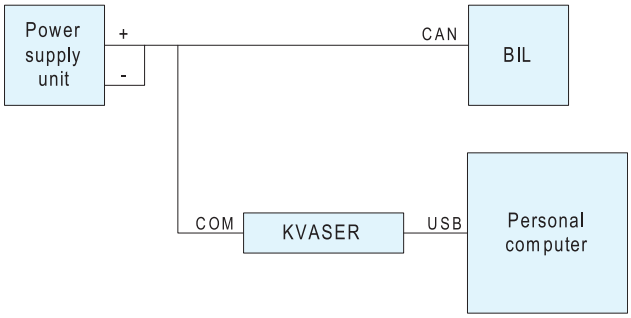

Fig. 1

Program execution requests arrive with the rate of $\gamma = 1800$ 1/h from the simulation software installed on the personal computer.

14 stages of testing were performed:

1. Testing of signal aspect display based on ALSN data along with testing of correct ALSN frequency display.

For 7 hours the simulation program was sending to the display unit CAN messages of alternating signal aspects per ALSN (the signal aspects are white, red, red and yellow, yellow, green) and alternating frequencies per ALSN (the frequencies are 25 Hz, 50 Hz, 75 Hz, 25 Hz). The results of this stage of testing did not indicate any program error.

2. Testing of signal aspect display based on ALS-EN data along with testing of correct display of movement ahead or with deviation.

For 7 hours the simulation program was sending to the display unit CAN messages of alternating signal aspects per ALS-EN (the signal aspects are white, flashing, red, red and yellow, 1TC, 2TC, 3TC, 4TC, 5TC) and alternating movement directions ahead and with deviation (the values are none, ahead, with deviation). At this stage of testing, two errors were identified: when signal aspect per ALS-EN is red and yellow or red, the display unit displays movement ahead or with deviation incorrectly.

3. Verification of correct display of actual, target and allowed speeds.

For 8 hours the simulation program was sending to the display unit messages of actual, target and allowed speed values. Assigned values of actual speed: 0, 20, 40, 60, 80, 100, 120, 140, 160, 180, 200, 220, 240, 260, 120, 0. Assigned values of target speed: 0, 20, 40, 60, 80, 100, 120, 140, 160, 180, 200, 220, 240, 260, 200. Assigned values of allowed speed: 0, 20, 40, 60, 80, 100, 120, 140, 160, 180, 200, 220, 240, 260, 0). The results of this stage of testing did not indicate any program error.

4. Verification of correct display of the coordinate, track number, movement direction forward or backward.

For 9 hours the simulation program was sending to the display unit CAN messages of the coordinates (0 km 0 kp 1 m, 4 km 5 kp 99 m, 9999 km 9 kp 99 m, 0 km 0 kp 0 m), track number values (0, 8, 15, 1), movement direction values (forward, backward). The results of this stage of testing did not indicate any program error.

5. Verification of correct display of the target name, type and distance, as well as station name.

For 9 hours the simulation program was sending to the display unit CAN messages of target names (Iksha, Nakhabino, none), target types (signal, station, hazardous place, bridge, level crossing, platform, tunnel, switch, track circuit, SAUT discrete channel transceiver, siding, tail of train, stop location, work area, conditionally clear signal, station), distances to targets (1 m, 1000 m, 8191 m, 0 m) and station names (Moscow, Tushino, Bolevoue, none). The results of this stage of testing did not indicate any program error.

6. Verification of correct display of pressure in the brake cylinder, brake line and control reservoir.

For 7 hours the simulation program was sending to the display unit CAN messages of the brake cylinder pressure (0.1 MPa, 0.5 MPa, 1.0 MPa, 0 MPa), brake line pressure (0.1 MPa, 0.5 MPa, 1.0 MPa, 0 MPa) and control reservoir pressure (0.1 MPa, 0.5 MPa, 1.0 MPa, 0 MPa). The results of this stage of testing did not indicate any program error.

7. Generation by the display unit of CAN messages on the program operability.

For 10 hours the simulation program verified the generation (every 500 ms) by the display unit of CAN messages on the program operability. At this testing stage two errors were identified and the display unit restarted the software twice and resumed the generation every 500 ms of CAN messages on the operability.

8. Verification of correct display of automatic train operation (ATO) mode, ATO target speed value, ATO schedule time.

For 9 hours the simulation program was sending to the display unit CAN messages of ATO mode values (ATO off, ATO in advisory mode, ATO in automatic mode), ATO target speed values (0, 20, 40, 60, 80, 100, 120, 140, 160, 180, 200, 220, 240, 260, 120, 0), ATO schedule time values (0 hours 10 minutes 15 seconds; 3 hours 12 minutes 19 seconds; 5 hours 19 minutes 22 seconds; 11 hours 23 minutes 27 seconds; 15 hours 27 minutes 34 seconds; 19 hours 34 minutes 38 seconds; 22 hours 44 minutes 47 seconds; 23 hours 57 minutes 59 seconds). The results of this stage of testing did not indicate any program error.

9. Verification of correct display of preliminary signalling, preliminary TSKBM signalling, driver vigilance confirmation request and stability of radio communication.

For 7 hours the simulation program was sending to the display unit CAN messages of the preliminary signalling value (on, off), TSKBM preliminary signalling value (on, off), driver vigilance confirmation request value (request, no request) and radio communication stability value (availability of communication, non-availability of communication). The results of this stage of testing did not indicate any program error.

10. Verification of correct display of acceleration, presence of neutral/dead section and distance to neutral/dead section.

For 9 hours the simulation program was sending to the display unit CAN messages of the acceleration values (-1,0, -0,92, -0,80, -0,73, -0,69, -0,53, -0,44, -0,30, -0,22, -0,10, -0,02, 0,04, 0,15, 0,23, 0,37, 0,49, 0,56, 0,66, 0,79, 0,83, 0,94, 1,0), presence of neutral/dead section values (no neutral or dead section, neutral section present, dead section present), distance to neutral/dead section values (0 m, 140 m, 631 m, 1202 m, 3044 m, 9999 m, 42142 m, 65535 m). The results of this stage of testing did not indicate any program error.

11. Verification of correct command input in the display unit data line.

For 8 hours the display unit keyboard was used to input data line commands (C0, C5, C6, C7, C71, C70, C91, C92, C261, C517, C522, C773, C799, C809, C800, C1029, C2565). The results of this stage of testing did not indicate any program error.

12. Verification of correct display of diagnostics messages in the display unit's data line.

For 7 hours the simulation program was sending to the display unit CAN messages of the following diagnostics messages: trip. KON, trip. EPV TSKBM, trip. EPV SAUT, slippage, electronic map number, display of the presence of BLOK modules in the configuration, display of the version, subversion and BLOK modules checksums). The results of this stage of testing did not indicate any program error.

Table 1.

Number of testing stage	Duration of testing stage *t*, h	Number of identified program errors *m*
1	7	0
2	7	2
3	8	0
4	9	0
5	8	0
6	7	0
7	10	2
8	9	0
9	7	0
10	7	0
11	8	0
12	7	0
13	7	0
14	7	0

13. Verification of correct display of current time.

For 7 hours the simulation program was sending to the display unit CAN messages of the current time values (0 hours 10 minutes 15 seconds; 3 hours 12 minutes 19 seconds; 5 hours 19 minutes 22 seconds; 11 hours 23 minutes 27 seconds; 15 hours 27 minutes 34 seconds; 19 hours 34 minutes 38 seconds; 22 hours 44 minutes 47 seconds; 23 hours 57 minutes 59 seconds). The results of this stage of testing did not indicate any program error.

14. Verification of correct display of recorder unit status and operating mode.

For 7 hours the simulation program was sending to the display unit messages of recorder unit status value (present, absent) and operating mode values (main-line, shunting, double heading). The results of this stage of testing did not indicate any program error.

The results of test stages are given in Table 1.

Reference conditions for calculation of display unit software dependability

Beside the test results described above the BIL software dependability indicators calculation involves the following initial data. The probability of SW error causing unit failure, $g_{ft} = 0,047$ [1]. The failure rate of the display unit's hardware components is $\lambda_{hw} = 3,01 \cdot 10^{-6}$ 1/h [4]. A self-test subprogram is assumed to be available with a failure detection probability at around $\alpha = 0,5$ [3]. Same goes for the failure response mechanisms with the probability of successful mitigation of an identified functional failure of $\beta = 0,99$.

$$\beta = 1 - \lambda_{hw} * t,$$

where λ_{hw} is the failure rate of the display unit's hardware components;

t is the target time of system operation.

The average downtime of the display unit caused by the required elimination of software error is $\tau_{dt} = 24$ h.

It is required to calculate the unit's dependability indicators as regards software errors under the target system operation time $t = 24$ h on the assumption of the absence of fault-inducing errors, a well as on the assumption that with the correction of identified errors no new defects are introduced in the software.

Calculation of display unit software dependability indicators

The reference conditions do not contain statistical data of program executions over the course of its maintenance. There is also no information on the structural characteristics of the program (number of operators, operands, cycles, etc.) which prevents the use of statistical models of dependability, such as the Halstead metrics, IBM model or similar ones. Therefore, let us choose the solution based on the Schumann model. Meaning, let us find the initial number of software defects, then the error rate and values

of unknown probability values that describe the SW dependability indicators.

The initial number of software defects N is calculated using the following equation:

$$\sum_{j=1}^{k} m_j \cdot \frac{\sum_{j=1}^{k} t_j}{\sum_{j=1}^{k} \frac{m_j}{N - n_{j-1}}} = \sum_{j=1}^{k} (N - n_{j-1}) t_j,$$

where $k = 14$ (number of testing stages);

m_j is the number of identified software errors at the j^{th} testing stage;

$n_j = m_1 + m_2 + \ldots + m_j$ is the total number of identified software errors at the 1-j^{th} testing stage;

t_j is the duration of the j^{th} testing stage.

Values m_j, n_j and t_j are given in Table 1. In accordance with those values, by means of the trail and error method, $N = 7$ was deduced.

The software error rate λ_{sw} is calculated using the Schumann formula

$$\lambda_{sw} = \frac{\sum_{j=1}^{k} \frac{m_j}{N - n_{j-1}}}{\sum_{j=1}^{k} t_j} (N - n_k)$$

Given the values of variables calculated at the previous step, λ_{sw} is

The probability of correct program run $P_{sw\,run}$ after troubleshooting is calculated using the formula

$$P_{sw\,run} = \frac{1 - \frac{\lambda_{sw}}{\gamma}}{P_n} = \frac{1 - \frac{0,014}{1800}}{1} = 1 - 7,77\ 10^{-6} = 0,99999,$$

where P_c is the probability of absence of fault-inducing errors that, assuming there are no such errors (see section Initial data), equals to 1.

Now let us define the unit's dependability indicators as attributed to software errors within a 24-hour target time.

The probability $P_{sw}(t)$ of no-error as the result of program run within the given time $t = 24$ h.

$$P_{sw}(t) = \exp(-\lambda_{sw} t) = \exp(-24 \cdot 0,014) \approx 0.7.$$

Mean time to software error $T_{av\,sw}$

$$T_{avsw} = \frac{1}{\lambda_{sw}} = \frac{1}{0.014} = 71,43\,h.$$

The probability of no-failure of display unit P_R(t) as the result of program run within the given time $t = 24$ h.

$$P_R(t) = \exp(-\lambda_{hw} t)[1 - (1 - \alpha\beta) g_{ft} (1 - \exp(-\lambda_{sw} t))] =$$
$$= \exp(-24 \cdot 3,01 \cdot 10^{-6}) \cdot [1 - (1 - 0,5 \cdot 0,99) \cdot 0,047 \cdot$$
$$\cdot (1 - \exp(-24 \cdot 0,014))] = 0,99.$$

Mean time to partial functional failure of display unit $T_{av\,un}$

$$T_{av\,un} = \frac{1 - g_{ft}(1 + \alpha\beta)}{\lambda_{hw}} + \frac{g_{ft}(1 + \alpha\beta)}{\lambda_{sw} + \lambda_{hw}} =$$
$$= \frac{1 - 0,047 \cdot (1 + 0,5 \cdot 0,99)}{3,01 \cdot 10^{-6}} +$$
$$+ \frac{1 - 0,047 \cdot (1 + 0,5 \cdot 0,99)}{0,014 + 3,01 \cdot 10^{-6}} = 3,08 \cdot 10^5\ h.$$

Partial functional availability factor of display unit

$$C_{fa} = \frac{T_{av\,hw}}{T_{av\,sw} + \tau_{pdt}} = \frac{71,43}{71,43 + 24} = 0,749.$$

As the result of benchmark tests of the display unit software at the troubleshooting stage several errors were identified. However, it cannot be ruled out that the operation of the software in actual use environment will not uncover other errors. Therefore, it is advisable to check the display unit software before each trip and immediately eliminate identified errors should the departure test indicate the presence of such. If systematic, this procedure will significantly improve the availability of BLOK. For instance, reducing the error elimination time from 24 to 1 hour through timely elimination of identified failures ($\tau_{pdt} = 1$ h) enables a display unit partial functional availability indicator equal to

$$C_{fa} = \frac{T_{av\,sw}}{T_{av\,sw} + \tau_{pdt}} = \frac{71,43}{71,43 + 1} = 0,986.$$

Thus, if the software error elimination time is $\tau_{pdt} = 24$ h, the display unit partial functional availability equals to $C_{fa} = 0,749$, while the reduction of the error elimination time to $\tau_{pdt} = 1$ h enables the display unit partial functional availability value equal to $C_{fa} = 0,986$, which practically improves the BIL SW operating characteristics by 25 percent.

Conclusion

The article considered an example of evaluation of functional dependability of the display unit software that is part of the BLOK vital integrated onboard system. The example of this isolated case shows that the chosen method is effective and can be used in practice.

The resultant partial functional availability indicators of the display unit show that pre-trip checking of the unit and immediate elimination of errors, should such be identified, will enable a significant improvement of user performance of BIL in terms of timely notification of the driver on the current operational situation to enable timely train control decision-making.

Acknowledgement

The authors express their gratitude to Prof. Igor B. Shubinsky, Doctor of Engineering, for his assistance, valuable advice and observations that contributed to this paper.

References

1. Shubinsky IB. Funktsionalnaia nadiozhnost informatsionnykh system. Metody analiza [Functional reliability of information systems. Analysis methods]. Dependability Journal LLC 2012 [in Russian].

2. Shukhina EE, Astrakhan VI. Bezopasni lokomotivni obiedinenni kompleks BLOK [BLOK Vital Integrated Onboard System]. Moscow; 2013 [in Russian].

3. GOST R IEC 61508–7–2012. Functional safety of electrical, electronic, programmable electronic safety-related systems. Part 7: Overview of techniques and measures.

4. Explanatory note with the display unit dependability calculation.

About the authors

Efim N. Rozenberg, Professor, Doctor of Engineering, First Deputy Director General, JSC NIIAS. 27, bldg 1 Nizhegorodskaya St., 109029 Moscow, Russia, phone: +7 (499) 262 62 17, e-mail: info@vniias.ru

Natalia G. Penkova, Deputy Head of Safety and Algorithmic Support, JSC NIIAS. 27, bldg 1 Nizhegorodskaya St., 109029 Moscow, Russia, phone: +7 (499) 260 77 52, e-mail: N.Penkova@vniias.ru

Alexander S. Korovin, Chief Specialist of Computer-Based Devices Development, JSC NIIAS. 27, bldg 1 Nizhegorodskaya St., 109029 Moscow, Russia, phone: +7 (499) 262 82 53, e-mail: A.Korovin@vniias.ru

Received on 06.12.2016