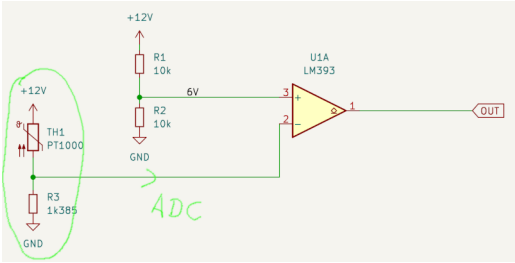
1 Digital SCADA (OSI model, layer 1)

In the previous chapter „Back In the Days" I have described quite briefly solutions based on analog signal processing, which was used amongst the first SCADA systems. As an example, we used analog comparator to create a very simple analog signal processing circuit, which had one major drawback: every change of the signal processing algorithm (e.g. alarm triggering value) required us to change the electronic circuit as well. Moreover, adding another physical quantity required adding another analog circuit, etc., and that - in the case of processing many physical quantities at the same time - significantly influenced the complexity and scaling problems. Both of these problems are solved by SCADA systems based on digital signal processing, and this is mainly possible thanks to the enormous progress that has been made in the computing power of a modern computers.

1.1 Signal processing algorithm

First of all, the concept of analog signal processing was dropped in favor of digital one. As you remember from the previous example, our analog processing signal design was composed of a comparator that compared two quantities and when the alarm value was reached, it output an appropriate signal:



If we look at this solution over again, we see that our only processed variable is the voltage coming out of the voltage divider constituted on the pair of TH_1 and R_3 (marked green). The remaining elements of the circuit are the algorithm. Due to the fact that today we are dealing with computers having much greater computational capabilities, instead of processing this variable in an analog algorithm as we initially did, we can code that into software, and the only thing we really need is the U_{R3} voltage value (which can be fed into software by using analog-to-digital converter). Our code most probably will read the temperature value corresponding to a recorded voltage and compare it with the alarm treshold value and act accordingly. If we want to change something - i.e. triggering value - or even present a signal trace to the operator - we just update the program itself. Much less of an effort, right?

1.2 Cabling reduction and data transfer

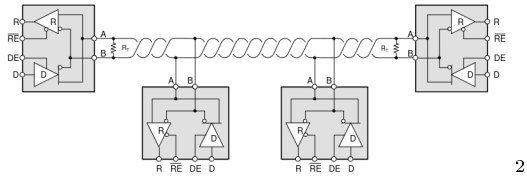
While in the above example (1.1) we have gotten rid of the first problem: analog signal processing, we still have to solve the second problem: the amount of cabling needed to connect all the system elements together with the SCADA computer. This problem was solved by introducing communication buses between the SCADA computer and system nodes. As our discussion dives into details now, to keep everything in order, we should stick to the OSI model from here.

2 OSI model, layer 1: physical layer

Modern solutions are based on serial data transmission between distributed system elements. The most common solution is to use twisted pair cable for differential signal transmission. There are also solutions based on fiber optics or radio links. Since it would be impossible to discuss all possible variants, I decided to discuss the two most common ones.

2.1 RS422/ RS485 bus

The electrical signal is transmitted differentially over a twisted pair, which makes it immune to interference with the electromagnetic field. The maximum range is 1200 m at a speed of 100Kbit/s, while using higher speeds the range shortens to 10m at 35Mbit/s[1]. There are plenty of protocols are based on EIA/TIA RS485 standard: Modbus, Profibus, DeviceNet, HVAC and many, many others.



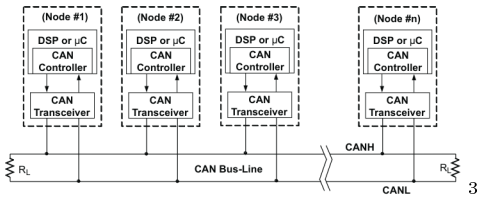
[2]

2.2 CAN bus

In the CAN bus, the signal is also transmitted differentially via twisted pair, which ensures immunity to interference. The maximum range is 1000 m at a speed of 50Kbit/s, while using higher speeds shortens the range to 40 m at a speed of 1Mbps. This bus is widely used standard by automotive industry.

[1] Rule of thumb: the speed in bits/s multiplied by the length in meters should not exceed 10^8

[2] https://www.ti.com/lit/wp/slla545/slla545.pdf?ts=1705742417876



[3]

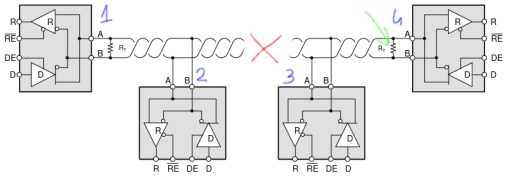
3 Vulnerabilities and potential vectors of exploitation of OSI 1 layer

In the case of linear signal processing designs, each physical quantity entering into the system had a set of separate remote element (e.g. a temperature sensor), its own signal cables and its own input card with its algorithm (see „Back in the days").

In digital solutions, we use a communication bus (CAN, RS485) that connects distributed system nodes into one whole system. Looking at it, the question immediately arises: what if the bus is disabled? Let us consider individual cases.

3.1 Bus continuity

I think that's the first case that comes in mind.

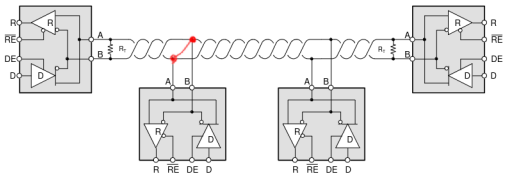


Let's assume that node 1 is the interface of our SCADA computer that collects data from nodes 2, 3, 4. In the event of a line break in the SCADA bus, we will lose communication with nodes 3 and 4. And will connectivity 1-2 be maintained? Not necessarily. Note that the discontinuity also results in the loss of the line terminal (marked with a green arrow) - I will discuss the importance of this fact in more detail in section 3.3.

3.2 Short circuit of bus

A bus short circuit most often occurs as a result of wiring problems, but this is not the rule. It may happen that one of the system nodes has a damaged transceiver circuit, which can cause a signal distorsion or a short circuit. The latter case is not such uncommon as You'd think, this type of faults are often the result of lightning discharges during heavy storms.

[3] https://www.ti.com/lit/an/slla270/slla270.pdf?ts=1705755301250



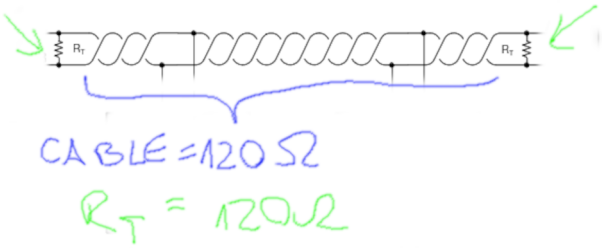
If a short circuit occurs on the bus, we not only lose communication, but also overload the differential amplifier circuits of the transceivers that send data to the bus. Does this damage them? Looking through the documentation for the MAX485 system from Maxim Integrated, we find the following information:

„Drivers are short-circuit current limited and are protected against excessive power dissipation by thermal shutdown circuitry that places the driver outputs into a high impedance state"[4]

It is worth making sure that we use ICs from a trusted source, which will be resistant to short circuits. If we use some IC clones of unknown origin, it may turn out that in the event of a bus short circuit, the transceivers will be overloaded and irreversibly damaged, and the process of restoring traffic in our bus will take a lot of time to search for system nodes with (occasionally) damaged transceivers.

3.3 Line impedance and termination problem

Line impedance is a characteristic factor of every data transmission system based on electrical cables (twisted pairs, coaxial cables). In the case of the RS485 standard, described in TIA/EIA-485, impedance[5] line is 120Ω and only for this value the optimal quality of the transmitted signal is maintained:



This requires us to use a cable that strictly meets these criteria, and terminating it with 120Ω resistors on both ends of the bus. If these conditions were not met, the maximum signal value would not reach our bus transceivers, but part of it would be reflected and create a standing wave in the cable, which would further

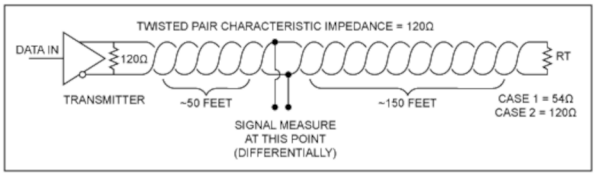
[4] https://www.analog.com/media/en/technical-documentation/data-sheets/MAX1487-MAX491.pdf

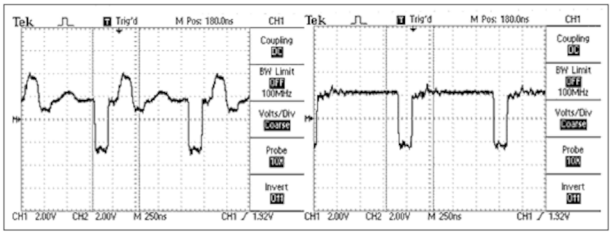
[5] Impedance is not equal to resistance, although both of them are represented in Ω. Impedance, apart of R includes also capacitive and inductive components: $Z = \sqrt{R^2 + (X_L - X_C)^2}$, where $X_L = \omega L$ and $X_C = \frac{1}{\omega C}$

interfere with the actual signal. The signal reflection coefficient Γ is represented by the formula below:

$$\Gamma = \frac{Z_L - Z_S}{Z_L + Z_s}, \text{ where } Z_L\text{- input impedance, } Z_S\text{- source impedance}^6$$

To illustrate how much the signal changes, let's look at the example published by Analog Devices[7], showing what the signal on the bus looks like if only one of the terminating resistors has the wrong value (it is twice too small):



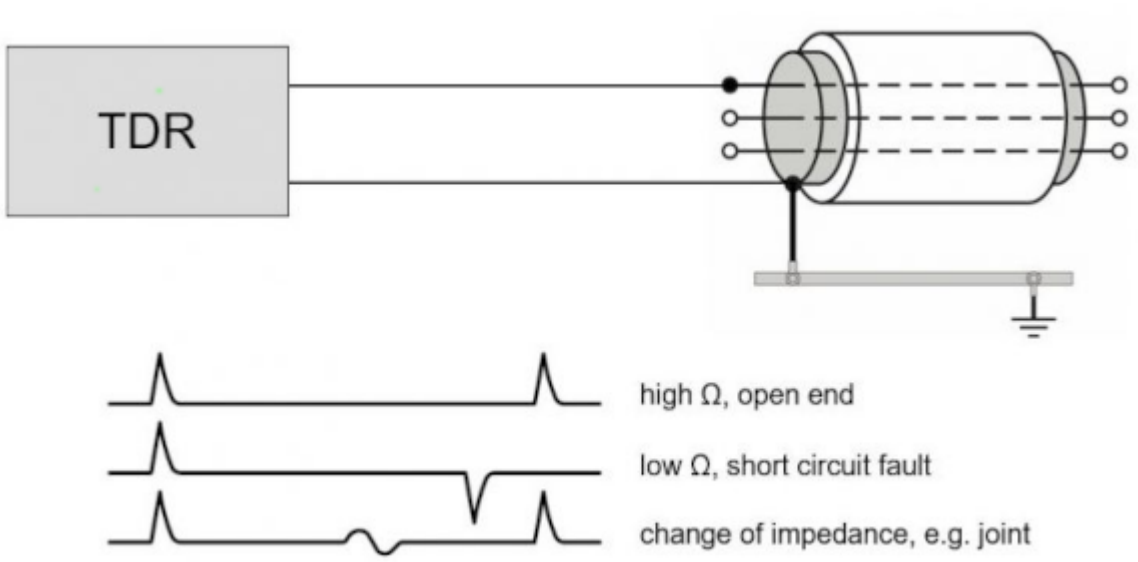


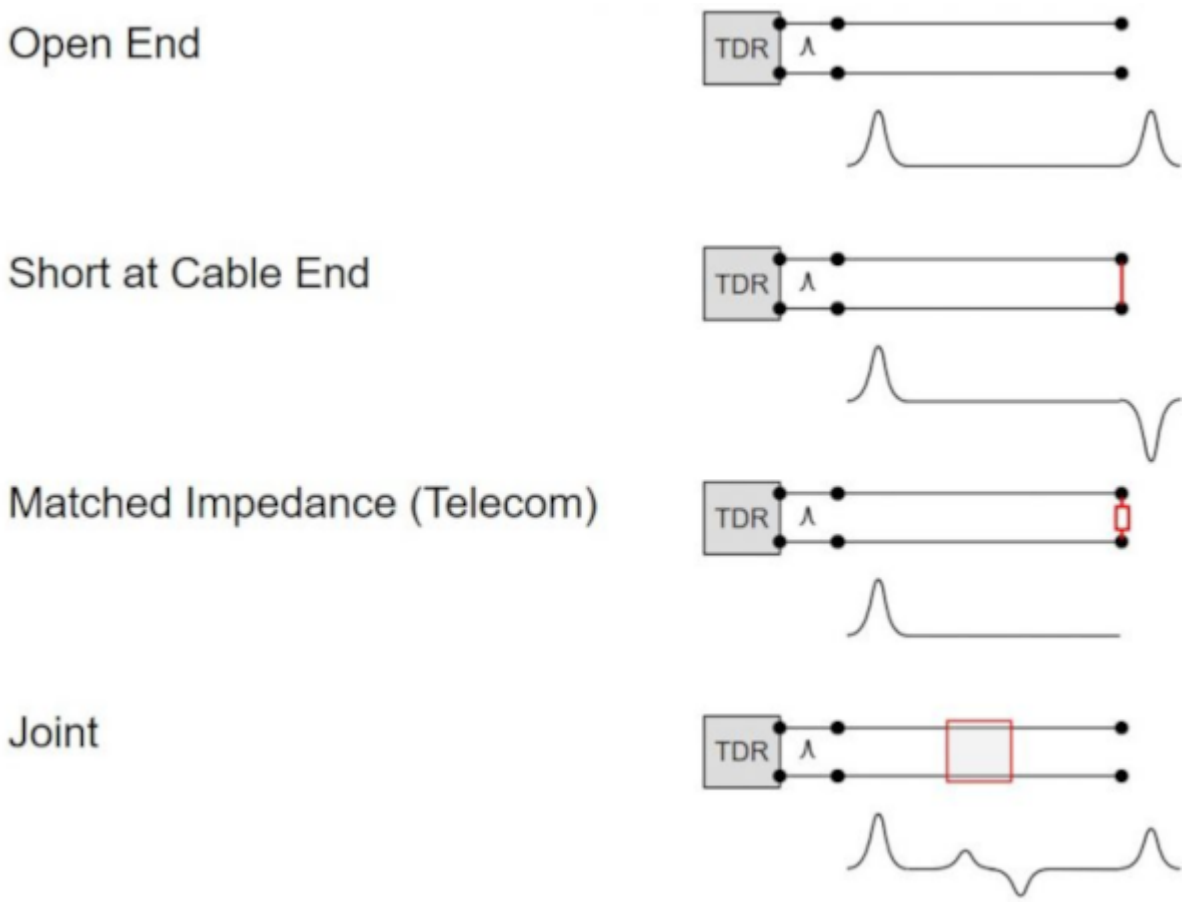
Testing the bus using a TDR reflectometer can tell us a lot about the technical condition of the physical layer. Usually, already at the stage of completed electrical installation works, such a test is carried out to verify and certify the solution in terms of meeting the telecommunications requirements of the bus. TDR reflectometers can be used to examine wired, but also fiber optic networks. The test involves sending a short pulse to the cabling and recording its response behavior. The Γ coefficient is measured in the time domain, which allows us to indirectly determine the type of phenomenon we are dealing with and the distance of the event from the measurement point. For example:[8]

[6] More theory at https://www.ti.com/document-viewer/lit/html/SSZTB23

[7] https://www.analog.com/en/technical-articles/rs485-cable-specification-guide--maxim-integrated.html

[8] https://www.hvtechnologies.com/the-basics-of-time-domain-reflectometry-tdr/


TDR
high Ω, open end
low Ω, short circuit fault
change of impedance, e.g. joint



3.4 Power supply

In sections 3.1-3.3 we talked over issues related to the physical layer used to exchange data between system nodes. However, it should be noted that all nodes require a power source. There are two variants, that are possible:

1. Each node has its own power module and from the perspective of system security, this variant is the best of choice. In the event of a failure of a power module, at a certain node, we only lose communication to this exact node only.

2. There is one power supply module for the bus system, from which power is distributed sequentially between individual nodes (usually together with the communication bus wiring). In this case, you should be aware that any power system failure will disable the entire bus. The failure may occur spontaneously (caused by aging and wear of the power supply module),

may also be caused by a short circuit in the electrical installation or by damaged electronic circuits of a node. This situation usually activates the overload protection (if there is any) and cutting off power to all system nodes.

4 Prevention practice

In practice, counteracting problems related to the physical layer is quite a complex problem. If we get back for a moment to our examples of the RS485 or CAN standard - we will notice that such a network can extend over a distance of up to 1 km! Finding a fault in reasonable time at such a distance without a TDR device is basically impossible, because it would require us to carefully review the installation centimeter by centimeter to locate the problem. Therefore, I believe that having a TDR reflectometer is a must, and it is good practice to periodically test the bus and respond to any problems that arise.

4.1 Defining source of problems

When it comes to the source of problems, we should divide them into:

- faults that occurred during normal operation of the SCADA system and
- faults caused intentionally (acts of sabotage).

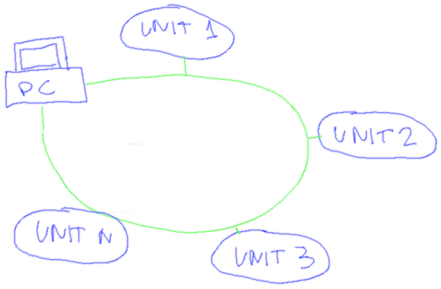
Basically, already at the design stage of the SCADA system, we should consider the possible threat actors to which our installation is a subject, what their consequences are and set the required level of protection resulting from these two analysis.

4.2 Bus malfunction

For now, let's put aside intentional actions aimed at damaging the SCADA and consider what options we have in relation to the issues discussed in 3.1. - 3.3.?

4.2.1 Ring topology

The simplest solution is to build a network in a ring topology, as in the sketch below.



In this case, there is a network interface of a SCADA computer at each physical end of the bus . If the bus continuity is interrupted, the nodes will split into two sets, and each of them will be connected to its respective SCADA network interface. Of course, this assumes that the amount of bus signal interference can be compensated by re-transmission. Typically, in normal operation, each nodes is pooled alternately by each SCADA interfaces, for example:

- 1st interface scans in a straight order 1->2->3->n, while
- 2nd interface scans in reversive order n->3->2->1

If communication with any of the node is lost, that triggers a network failure alarm (usually forwarded to the system operator's panel), and the network interfaces stop working in alternative mode, but communicate with those system nodes with whichever they can.

4.2.2 Bus redundancy

In the example described in 4.2.1, the protection of the physical layer relies on too many assumptions, in particular that when the bus is broken, there will be no unfavorable signal reflections distorting the transmitted information to such an extent that it will be unreadable, or that - even if such phenomena occur - at some point, after repeated attempts, we will manage to eventually pass the information correctly. Therefore, a much better option is to build a redundant bus network.

In the event of a network failure, the SCADA computer signals the network failure to the operator and automatically changes over to an alternative network through which it maintains data acquisition. Once the alarm condition is raised, we can start maintenance work on the damaged bus: test it with a TDR reflectometer and perform the necessary repair work, having the comfort that our SCADA system still functions properly.

4.2.3 Power supply redundancy

If we use a design based on one power module that powers all system nodes, it is worth consideration of having two alternative power sources with change over possibility that allows maintenance work, similarly to the one described in 4.2.2.

4.3 Sabotage

Preventing these types of faults is very complex, especially in the case of large, critical infrastructure networks.

Note that even if we provide backup solutions, i.e. in form of network redundancy - in a process of electrical installation, both buses (normal/backup) would be probably placed on the same cable route, and - even if not - both buses would certainly have to reach each and every of the nodes, so in fact gaining access to any SCADA element become a good place to launch an attack.

Attacks can be carried out using the brute-force technique, where, for example, the bus is a subject of mechanical damage (i.e. cut). However, using diagnostic devices, such as the mentioned above reflectometers, the bus problems can be quickly located. Another attacking technique may be irreversible damage to bus transceivers, analogous to a lightning strike discharge(described in section 3.2).

A more sophisticated attack technique - and much more difficult to combat - is modifying a system node (or nodes), that will periodically (or when commanded to do so) cause signal disruption on the communication line or overload the power systems.

Generally speaking: compromising the physical layer of SCADA systems is usually not difficult, the only difficulty lays in gaining physical access to the system. Therefore, it is necessary - in a manner appropriate to the level of security - introducing monitoring and access control measures to the space in which the system operates, so as to prevent access by unauthorized persons.

A very good practice is to use network segmentation, i.e. instead of using one long bus, splitting it into several smaller ones. In this case, when one of the buses is being attacked, the others work uninterrupted, preventing the spread of cyber threats. In this case, devices are usually grouped taking into account their physical location in the system and their importance for infrastructure.

5 Conclusion

In this chapter, I wanted to discuss SCADA solutions at their most basic layer - the physical layer, therefore it was necessary to refer to some specific solution, and I have choosen the popular EIA/TIA RS485 standard. Currently, there are many variants - from those based on twisted pair cables, through radio links, to fiber optic systems. It would be impossible to discuss them all here, nor would it make much of a sense. However, I think that the issues I've mentioned can serve as a good starting point for analyzing other physical layer standards. In the next chapter that I plan to prepare, we will deal with the link layer, and I promise there would be much more fun involved!