OETCS/WP3/D3.7.0

ITEA2 Project
Call 6 11025
2012 – 2015

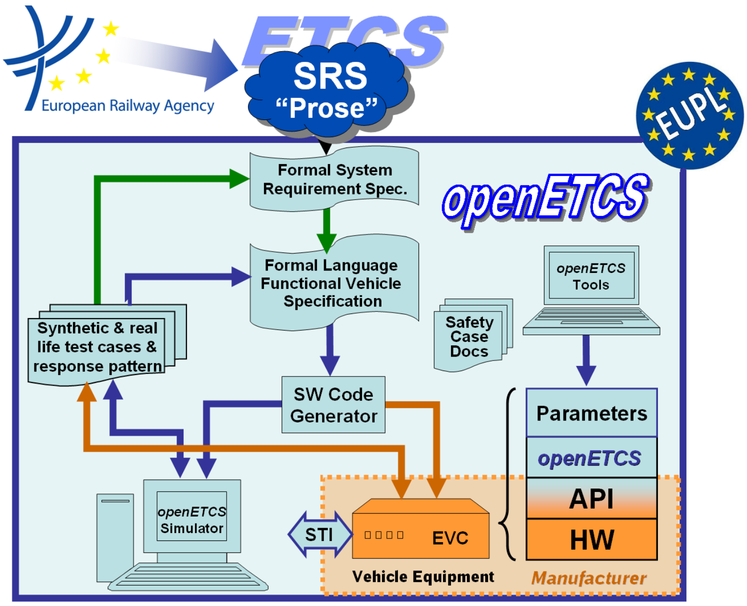
Work Package 3: “Modeling”

openETCS System Architecture requirements specification

Hardware Component requirements specification

Christian Giraud, Fausto Cochetti

November 2015



Funded by:

Federal Ministry of Education and Research

Région de Bruxelles-Capitale

openETCS

This page is intentionally left blank

Work Package 3: "Modeling" **OETCS/WP3/D3.7.0**
November 2015

openETCS System Architecture requirements specification

Hardware Component requirements specification

Document approbation

Lead author:	Technical assessor:	Quality assessor:	Project lead:
location / date	location / date	location / date	location / date
signature Fausto Cochetti (Alstom)	signature Christian Giraud (Alstom)	signature Marc Behrens (DLR)	signature Jacques Pore (Alstom)

Christian Giraud, Fausto Cochetti
Alstom

Architecture and Design Specification

Prepared for openETCS@ITEA2 Project

Abstract: The aim of this document is to introduce a *physical system perspective* matching with the Open ETCS program.

The openETCS modeling work-package does not include the development of a specific target architecture to be integrated with the train on-board system neither does it contemplate the model of such an architecture. The functional model designed in the Scade environment does refer to an application programming interface (API) to reference system resources and interface management. This approach based on an industrial proven specification [1][2][3] provides a level of abstraction from proprietary architectures. The API gives a clear insight on the platform constraints, such as dynamic aspects and timing management, principles of separation of functions and event interrupt concepts, when relevant for the application.

This document describes a generic hardware decomposition aimed to support the designer with a system perspective on the train-borne subsystem applied for the ERTMS/ETCS functionality.

The design paradigms for a suitable physical on-board system architecture remain open and need to be integrated with specific design choices. The principles refer to a proven industrial platform and to the openETCS on-board functional model. The concepts are described at a sufficiently high level so that they are easily customizable for a specific as well as for a generic not relating to any specific proprietary system design.

The hardware abstraction in the software system is obtained by referencing to a generic API. The requirements of this API are described in a separate document delivered by Alstom [4].

The functional scope of the openETCS OBU model is documented in D3.5.x, where x denotes the iteration.

Modification History

Version	Sections	Modification / Description	Author	Date
0.1	all	Initial release of the document	Fausto Cochetti	30.11.2015
0.1.1	appendix	technical compendium		30.11.2015

Table of Contents

List of Figures

1 Purpose of the document

This document is managed as a deliverable of the modeling work package with denomination D3.7.x, and contains advices and recommendation for the design of a physical system architecture.

The development of the functional model is done iteratively increasing the scope in steps, the last digit of the deliverable identifier, i.e. x, denotes the release of the model to which it applies. If the functional model requires to update the system architecture a consistent version number will be applied to this document as required by the Model release version.

This document complements the indications contained in the API requirements specification and the documentation derived from this as the generic openETCS Application Programming Interface (API), available at https://github.com/openETCS/modeling/blob/master/API/description/api-description.pdf. [4]

1.1 Input Documents

The following documents provide a context for the system perspective.

- ERA Subset-026 [5], V3.3.0
- ERA TSI CCS Documents
- openETCS API documentation, available at https://github.com/openETCS/modeling/blob/master/API/description/api-description.pdf [4][2][3]

2 Introduction

Designing a sub system integrable with the train borne system is a complex task. The designer faces a large variety of serious challenges and design complexities.

Before the functions are actually implemented, a system architect will have to select an appropriate hardware-software concept out of the large number of available boards, controllers, network and bus constraints. He will as well include robustness criteria against environmental influences.

Memories, operating systems, drivers, generic and application software segregation as well as selection criteria for sensors and actuators need to be correctly assessed.

The target architecture has to meet a large variety of requirements. Criteria of timing, Bus bandwidth, processor and peripheral performance, memory size, safety principles and possible processing or data transfer bottlenecks. Environmental conditions, timing constraints, robustness against specific interferences shall constantly be tracked.

Power requirement as well as allocation of availability, maintainability figures to enumerate only the most relevant items accompany all the design phases.

On top of this a specific vital architecture has to be selected and the required integrity level has to be granted. The relative safety constraints have to be assured and maybe exported.

Selecting the components matching these is a critical phase. Over-dimensioning the architecture may impact on cost factors relevant for the market access of the system. Under-dimensioning the architecture design could result in not achieving performance constraints, thus compromising system quality and suitability. Early architectural choices have a dominant impact on the success of the new system.

The system architects will commit to efficient design choices according to the target project margins and all this within the frame of a defined project delivery time schedule.

Due to the fact that the design verification phase, requiring to have completed all the integration steps, may be very late in the release process a high precision during the system architecture design is mandatory.

Therefore highly experienced System Designer are considered as the key factor for a reliable achievement of expected design result.

A primary goal of the openETCS ITEA2 project is to provide a formal specification and a model of an ETCS onboard functionality according to the specification defined in Subset-026 [5] by the European Railway Agency (ERA).

The Model-Based Development process is an approach that allows engineers to specify the behavior of a system and to simulate and execute it in a very early development stage.

Once a model-based development process has been established, engineers should be able to apply new technologies and tools to enhance and shorten product development cycles, e.g. by introducing generation of Model Validation test cases and target Code directly from the model. This enables to improve the V based development process to save development time and effort while preserving or improving the dependability of the developed systems.

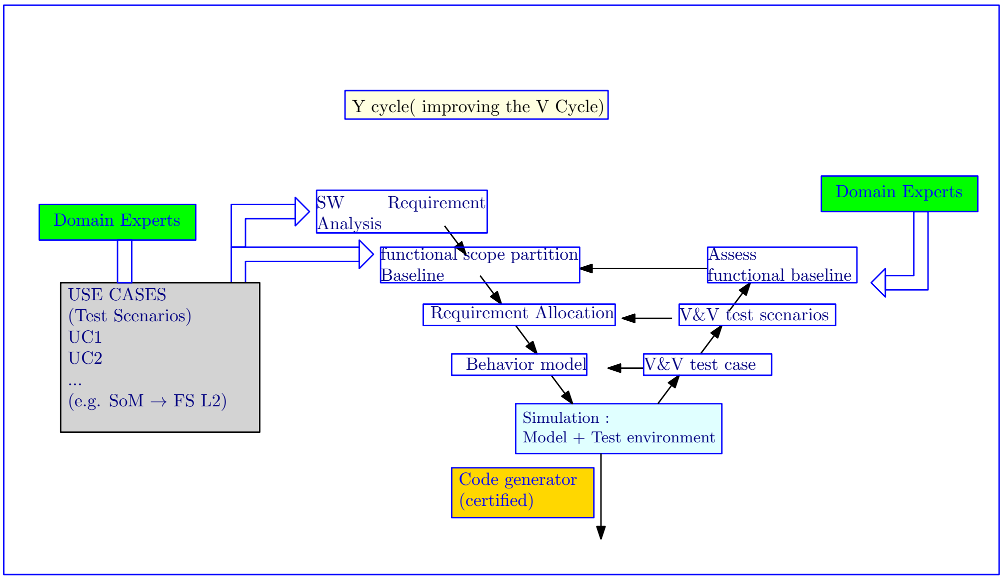


Figure 1. SRS modeling cycle

The methodology makes it easier to understand requirements and increases the correctness of the requirements, the correctness of the design and the code with respect to the requirements. An integration of system-level and design-level modeling tools allows a virtually integrated V-process that is sharpened up to a Y-based process with the required steps at the bottom of the former V being considerably automated (see figure1)

Nevertheless when specifying the overall software architecture, the designer should be aware of the implications of software design decisions on the target end system.

2.1 Safety Integrity and Functional Safety according CENELEC

The Railway Industry currently relies on the international standard group of coordinated standards: EN 50126 Railway applications The specification and demonstration of Reliability, Availability, Maintainability and Safety (RAMS) the EN 50129 Railway applications Safety related electronic systems for signalling and the EN 50128 Railway applications - Communications, signalling and processing systems Software for railway control and protection systems to provide a rational and consistent approach for the development of safety-related systems.

This group of standards owes much of its direction and contents to the IEC 61508 standard that is a generic safety standard for electrical/electronic/programmable electronics safety-related systems.

Both of these IEC and EN standards share the same philosophy in the sense that they:

- consider all relevant product and software safety life-cycle phases, from an initial concept phase to maintenance and decommissioning when these systems are used to perform safety functions;
- intend to shape a safety awareness;
- have been conceived with a rapidly developing technology in mind;
- provide methods and rules for defining safety requirements necessary to achieve defined functional safety.
- use Safety Integrity Levels (SIL) for specifying the target level of safety integrity for the safety functions to be implemented.
- adopt a statistical risk-based approach for the determination of the SIL requirements;
- distinguish between safe and unsafe failure modes and requires precautions against undetected failures.

According the Cenelec norms the product is subject to a certification process. The definition of the equipment under control (EUC) depends on the scope of the certification. It can be, for example the complete ERTMS/ETCS subsystem or a module of it.

The term safety-related is used to describe systems that are required to perform a specific function to ensure that risks are kept at an acceptable level. Such functions are, by definition, safety functions. Two types of requirements are necessary to achieve functional safety:

- Safety function requirements (what the function does),
- Safety integrity requirements (the required likelihood of a safety function being performed satisfactorily).

The safety function requirements are derived from a risk analysis phase, in the scope of EN 50126, where significant risks for equipment and any associated control system in its intended environment have to be identified. This analysis determines whether functional safety is necessary to ensure adequate protection from unacceptable risks. Functional safety is therefore a method of dealing with risks to eliminate them or reduce them to an acceptable level. EN 50128 specifies four levels of safety performance for a safety function. These are called Software Safety Integrity Levels (SwSIL).

2.2 Reference to the openETCS functional Model

The openETCS OBU partial model has been developed according to the specification given in ERA Subset-026 [5], Version 3.3.0. The software release is publicly available on a repository at

https://github.com/openETCS/modeling/tree/v0.3-D3.6.3

3 Design principles

3.1 Use of Scade modeling tool

While the use of the modeling tool provides a strong support for the coherent application of an uniform coding standard, it introduces some limits and constraint on the use of operators and data structures. This issues should be addressed at an early stage of design in order to provide best performing result from the code generator.

A specificity of the European Vital Computer (EVC) system is the coexistence of quite linear complexity as e.g. the one found when managing level and mode transition, together with high demanding algorithms like the ones found when approaching the braking curves in target supervision. The first type of function can be directly derived from requirements and can be coded without any additional interpretation or design criteria. This applies in the same way to other safety related systems like e.g. Interlocking applications, where functional (safety) rules are assessed and fixed and can be applied according to the specific configuration. The second type of functions need to be accurately analyzed in order to provide best performing results and this implies among others to provide suitable operators and data structures that allow fastest results lowest memory allocation and simplest rules.

3.2 System Time and Time Stamps

Basically a Scade model is based on the cycle number as its main timing source. If the model will not be clocked with an equidistant period, a different time source (actual system time) is needed as an input. The following rules shall apply for the model in such case:

- The model must not rely on a fixed period between the model clocks. Timing functions based on clock cycle counting are not feasible.
- In a first phase the model is allowed to use the actual system time for those calculations if this is not deductible form event time stamps.
- The Basic Runtime System will keep the actual system time applied to the model as an input unchanged within the same Basic Runtime System cycle.
- The actual system time will be strictly monotonic increasing between two subsequent Basic Runtime System cycles
- The interval between two subsequent Basic Runtime System cycles should not be critical for the application. Measures done on the partial Scade model indicate that a cycle of 1 ms could be achieved.
- Practically, the actual system time could be used in the model for the implementation of time intervals.

3.3 Time stamp on the physical System

In addition to the "cyclic execution" principle, the application must take into account that the various peripheral boards/sub-systems in charge of the input/output treatment are not synchronized and that their performances are not identical, nor constant in time;

- e.g. time delay due to filtering and smoothing of analog data such as MMU sensors, train digital inputs, ...
- e.g. variable propagation delay on bus systems depending on performance (chained treatments) and arbitration criteria.

Therefore,

- The input data present at the interface of the Onboard ETCS/EVC will be provided with a variable delay to the application (fluctuation of delay duration, jitter);
- The application output data are provided to the EVC/Onboard ETCS interface with a variable delay (non deterministic delay, jitter).

For each input and for each output data, a *T_min* (minimum delay of EVC input/output treatment) and a *T_max* (maximum delay of EVC input/output treatment) will be defined as shown on the diagrams in figure 2 :

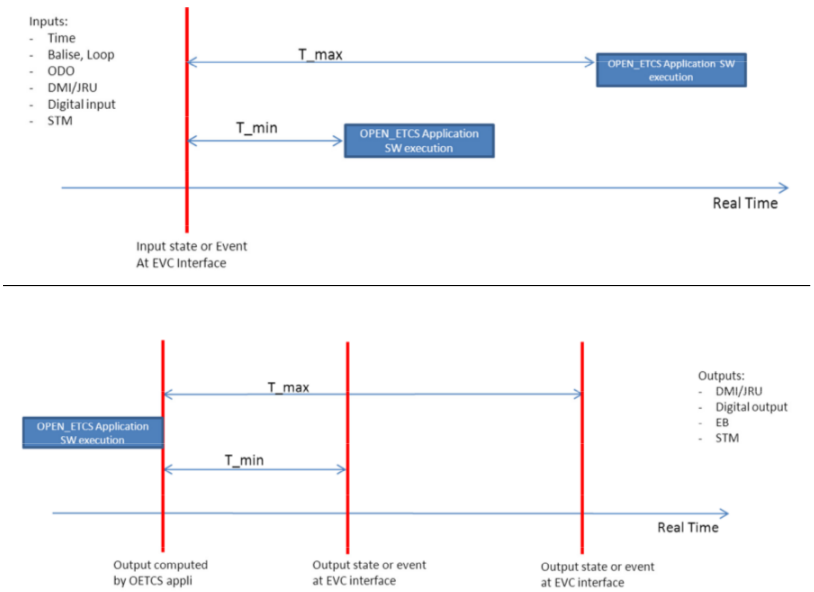


Figure 2. ***Managing delays with time stamps***

Notice that when the input data are time-stamped at the moment they are produced (by the source), the receiving applications may apply a correction in order to manage the delay of transmission of the data; assuming that the clocks of the various calculators are synchronized (e.g. the accuracy of the clock synchronization within the Model EVC is 1ms).

4 OB subsystem context

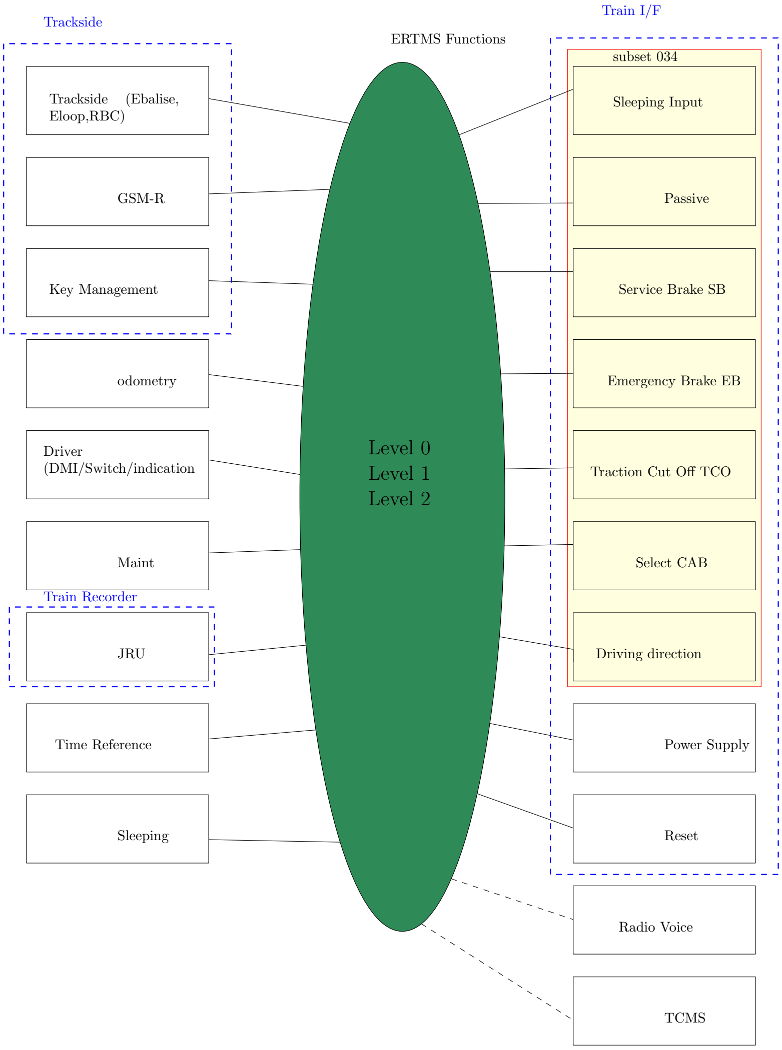


Figure 3. OB context diagram

5 Elements of the ETCS OB Architecture

The ERTMS/ETCS OB sub-system comprises at least following equipments:

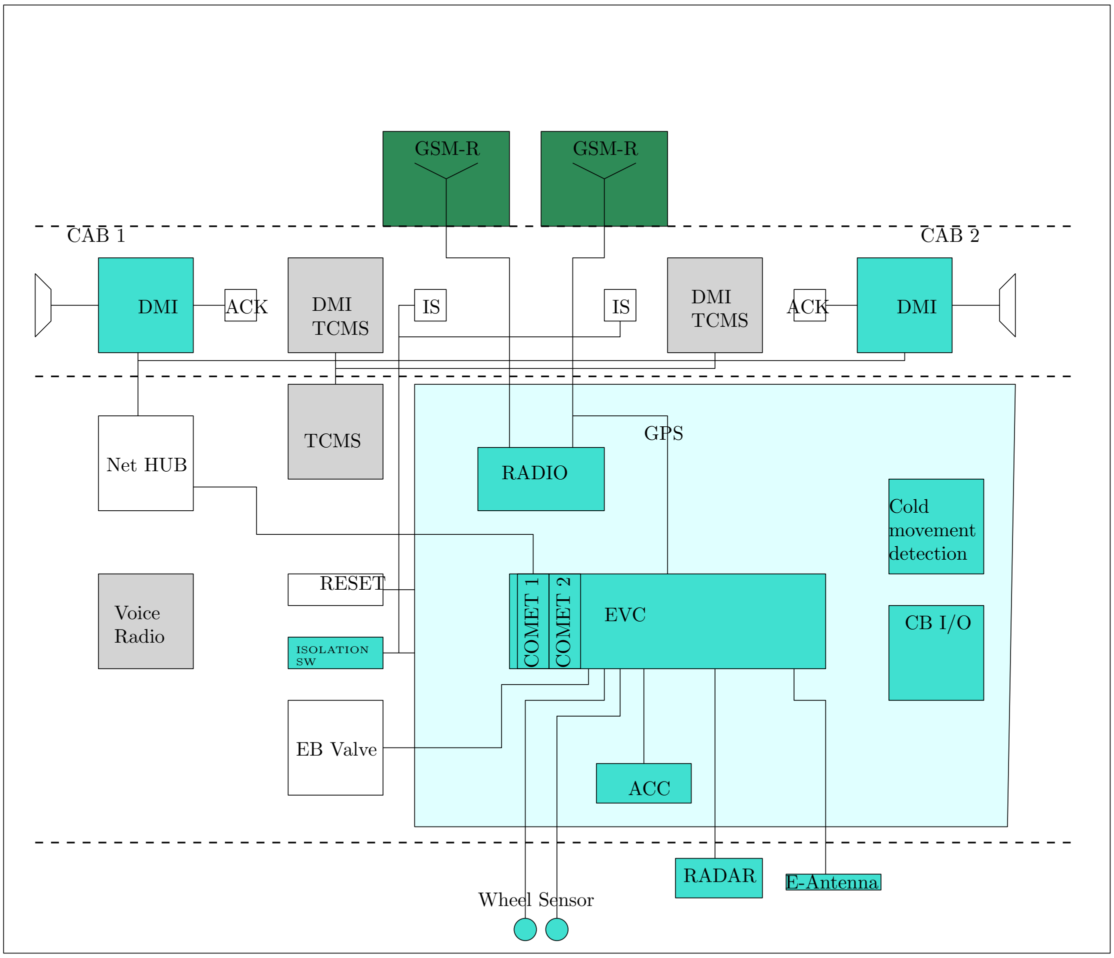


Figure 4. OB architecture Overview

The ETCS cubicle is realized as one mechanical assembly with a standard housing IP20 and will include :

- the EVC (SIL4 equipment);
- an internal recorder memory for Juridical Data and diagnostic data;
- one GSM-R Data Radio module for GSM-R communication with trackside
- relaying devices (contactors, circuit-breakers, relays);
- fan modules for the cooling function ;
- one accelerometer for the speed and distance measurement;
- a Cold Movement Detector to detect train movements when ETCS is powered off.

The EVC includes among others the MMU providing the odometry function. The odometry implies a complex vital function non mapped on specific Unisig requirements.

Other components installed elsewhere are listed below:

- The Isolation switch aims to isolate the ETCS OB sub-system from the train braking and traction interfaces when ETCS OB is excluded; associated bypass and isolation circuitry is installed to perform the isolation function. An Isolation indication must be visible to the driver in case of ETCS OB isolation.
- Reset button A periodical reset of the ETCS OB sub-system is necessary in order to ensure its required level of availability and safety.
- One ETCS driver-machine interface (DMI) (including display, loudspeaker and associated acknowledgement button) is installed in each cab. Other components (EVC, odometry, radio, ...) can be unique for both cabs or can be installed separately for each cab"
- The Eurobalise/Euroloop antenna is installed for Eurobalise and Euroloop reception.
- A radar device is installed for the speed and distance measurement.
- Wheel Sensors are installed for the speed and distance measurement.
- Redundant antennas are installed for GSM-R data communication and GPS signal acquisition.

A redundant network (bus, switch) is used for the communication between EVC, DMI(s), Train Recording Unit (TRU), and the other OB subsystems.

Train Recorder Unit The Train Recorder Unit shall record information from the ETCS OB (diagnostic and juridical data) and from the train interfaces. The GPS signal (from a GSM-R/GPS combined antenna) will be received for the data time stamping. The TRU will also compute and record the train speed. The Train Control Management System (TCMS) will exchange inputs/outputs information with the On-board (OB) ETCS OB subsystem

The TCMS will manage the TCMS DMIs and the data communication at train level (between the different vehicles in a multiple configuration). As a remark the TCMS and voice radio systems are not part of the ERTMS/ETCS system.

Braking electro-valves, train devices The braking and traction devices commanded by the ETC OB (emergency brakes, service brake and traction cut-off) are part of the Rolling Stock subsystem.

Appendix A: Technical compendium on selected topics

A.1 Concepts and principles of calculation used to estimate train position

A.1.1 conventions and variables related to oriented quantities

The basic convention for the MMU, related to movement detection is given by the rule establishing that the odometric counter values will increase, when the train is moving towards cab A. The Cab A is defined permanently by hard wiring during installation of the on-board unit (OBU) system. The counter will decrease for the opposite motion, which is also said to be in direction of Cab B. This rule, non depending on any operational setup, makes the counter increase if Cab A is preceding in the movement Cab B when crossing a given track point and vice versa makes the counter decrease when Cab B is preceding cab A.

Counter values of MMU are signed, fixed point (1 digit) numbers going from $-15 \cdot 10^6$ to $15 \cdot 10^6$ to be understood as a measure of traveled meters in a given direction, as seen from OBU's (wheel turns) point of view. It should be kept in mind that the value variation of the counter after a given traveled path will fluctuate depending mainly on slip and slide effects. Repeating the train ride on the same track path (length) many times will produce different counter variations at each trial. In general it can be said that the expected fluctuation of the relative counter value will increase with the length of the traveled path.

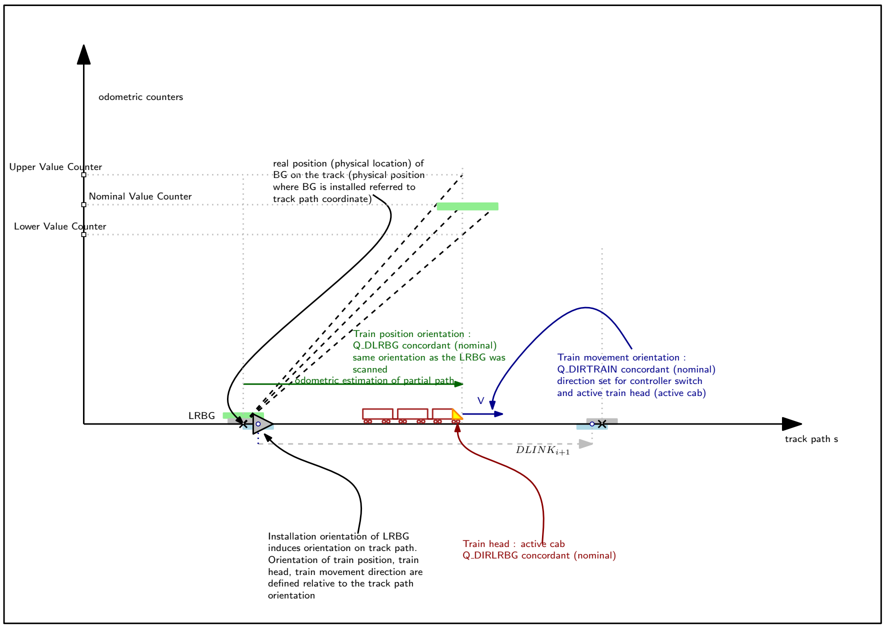


Figure A1. *orientation convention*

The MMU is making also movement detection and movement direction detection as reflected by API outputs MOTION_STATE and MOTION_DIRECTION. No train movement detection (train standing still) implies also that there will be no train movement direction detection, in such a case the detected train direction is said to be unknown. This outputs will be delayed by the window function needed to stabilize reliable detection. The MMU orientation establishing the counter behavior is non volatile and non switchable. If in the engine there are two OBUs installed then each OBU will have it's own wired Cab A irrespective of the configuration adopted for the other OBU. The two OBUs will never compete for MMU data.

When the OBU is in a mode referencing to a LRBG, the directions relevant for train driving are given by the following triple (relative to LRBG orientation):

- Q_DIRLRBG Train Orientation => active cabin, => Front End => headlights
- Q_DIRTRAIN direction of train movement (not the detected motion direction but the direction as set by direction controller switch. When reversing is allowed this orientation can be reverted operating on the direction switch allowing the train to proceed in reverse direction) possible values are : nominal (e.g. train oriented nominal and direction switch forward), reverse (e.g. train orientation nominal and direction switch reverse) unknown direction switch in neutral position. The ETCS system will supervise the direction of motion and allow to switch in the reverse direction (opposite to train direction) only under foreseen conditions e.g. during emergency in tunnel.
- Q_DLRBG "scanning" orientation at time of over-passing the LRBG and therefore "sign" of the coordinate relative to LRBG (positive coordinate direction means train has scanned LRBG in nominal direction reading increasing N_PIGs and is now on the arrow tip side of LRBG). The absolute value of this coordinate is given by the variable D_LRBG.

The above three quantities have to be considered as software variables, that need to be initialized and may change depending on the actual state and inputs elaborated. They have only meaning if the direction of the LRBG is given.

other relevant Direction qualifier are :

- Q_DIR message orientation (relative to sending BG) nominal (message valid for nominal scanning/overpassing of the BG) reverse and neutral (valid for both directions) In case of RBC sending the message the reference orientation is given by the one of the LRBG. This direction qualifier is most relevant for giving the relative position of a MA (Packet 12) transmitted by a BG in respect to relevant train end. This gives the permitted movement direction of the train, and if an opposite movement direction is detected, by evaluating the MMU output MOTION_DIRECTION a brake command should be triggered.
- Q_LINKORIENTATION part of a linking vector (packet 5) indicates if the linked BG pointed by the vector will be over-passed in nominal or reverse direction.

other Direction qualifier are :

- Q_MPOSITION orientation of kilometer counting convention
- Q_ORIENTATION orientation given by RBC to a passed LRBG

A.1.2 Rail-routed path and empirical odometric map

The estimation of the train position is currently based on the odometric measure of the distance traveled by the train made with an on board equipment and by the detection of reference balises installed at known positions on the track, when the Balise receiver antenna detects them. The distance is measured along the rail or track path. Actually track routes can have variable branches (when crossing switch points from blade side). The on-board odometry is not able to map different branches and will only trace an one-dimensional path.

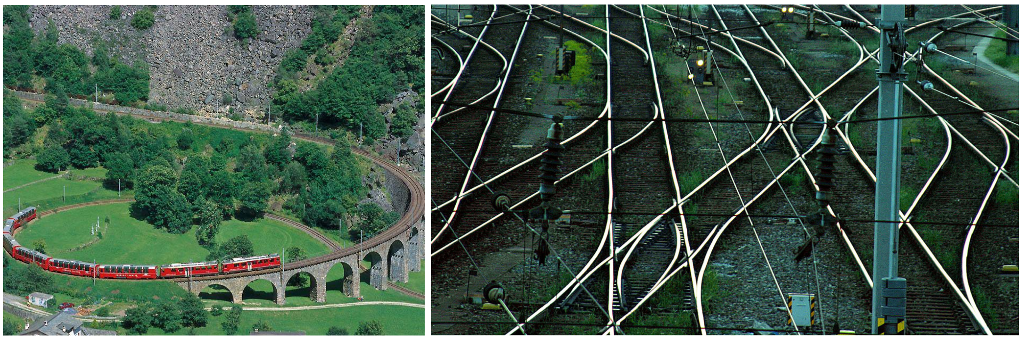


Figure A2. ***Examples of train path***

The picture in figure A2 gives two illustrative examples of rail path. Even if the train route can follow different branches depending on the position of the switches the coordinate system available on-board is always one-dimensional and needs to get updated track data for the selected route each time the train passes over a switch point from blade side.

This is done mainly through the so-called repositioning information[1] available at a BG immediately after the branch point. Such a balise group is usually announced by a weak link [2] not including the id[3] of the BG and applying a relaxed expectation window. Note: In cases where

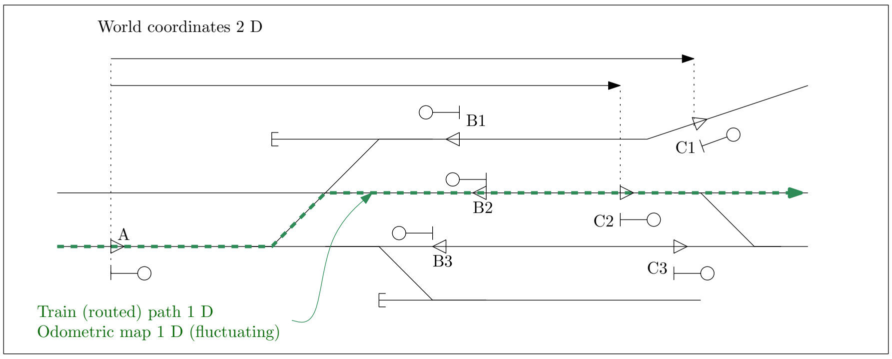


Figure A3. ***Example of repositioning data.***

a balise group contains repositioning [4] information, the term linked also applies since the balise group is announced, marked as linked and contains repositioning information marked accordingly. For clarity sometimes we will use the term weakly linked. Repositioning information contained in a balise group message shall only be evaluated if linking information has announced a following balise group as unknown but containing repositioning information. A balise group message containing a movement authority shall not contain repositioning information for the same direction.

[1] see system requirement specification (SRS) 026 3.8.5.3.5.
[2] 3.4.4.2. and 3.4.4.4.
[3] id number 16383
[4] By convention *relocation* indicates a coordinate shift along the the train path, while *repositioning* indicates a displacement in the second dimension, which is not tracked by the on-board equipment

In the balise group A the following information is given:

- The most restrictive track description from all routes (which could be a combination from the routes);
- The linking distance given to the farthest balise group containing repositioning information, the identification of the repositioning balise group is not known;
- For a given aspect of signal A, the most restrictive MA from all routes (the shortest sections from the routes and the lowest target speed at the End Of Authority);
- If some sections are time limited, the most restrictive timer.

Balise groups B (B 1 or B 2) give the following static information (repositioning information):

- Linking to the next balise group C
- The distance to the end of the current section (i.e. the distance to the end of section B1 - C1, or the distance to the end of section B2 - C2)
- The track description related to this track.

A.1.2.1 Mitigation of balise cross-talk while expecting repositioning information

If repositioning is announced and the expected repositioning balise group has been found, the ERTMS/ETCS on-board equipment shall keep looking for a balise group that satisfies the same criteria as this previously expected and already found repositioning balise group, until one of the following events occurs:

- the on-board antenna leaves the expectation window of the repositioning balise group that was announced and already found
- a linked balise group that has been announced with known identity is found.

If a second balise group is found that satisfies the same criteria as the previously expected and already found repositioning balise group, the ERTMS/ETCS on-board equipment shall command the service brake and the driver shall be informed. At standstill, the location based information stored on-board shall be shortened to the current position. Refer to appendix Subset 026 A.3.4 for the exhaustive list of information, which shall be shortened.

Note: this function is independent from linking function, i.e. the rules related to linking always apply. This means that once a repositioning balise group has been found and if this latter contains new linking information, the ERTMS/ETCS on-board equipment will start expecting the first balise group announced in this new linking information in parallel with the monitoring specified for mitigating balise reception degradation

A.1.3 Correlating two linked BG detections

If the odometric difference between two linked BG detections differs from nominal distance in a significant way (at least more than the odometric error of 5 % in plus or in minus) this supplemental information can be used to constrain the reciprocal confidence interval of the second detected BG.

A.1.4 Odometry from Wheel motion

In the following we try to adopt an uniform symbol convention, based on the assumption that there are quantities measured at fixed time intervals and quantities measured at variable, speed dependent time intervals. Fixed time intervals are :

$$\tau^\star = 1\,\mu s,\ \tau^\circ = 50\,ms,\ \tau^\bullet = N^\bullet\tau^\circ = 500\,ms \tag{A1}$$

Variable time intervals are given by the time between the detection of two odometer teeth or two odometer virtual teeth.

$N^\otimes = 50\,p.p.r.$ number of physical teeth on wheel sensor

$N^\ominus = \frac{N^\otimes}{5} = 10\,v.p.r.$ number of virtual teeth on wheel sensor

$N^\bullet = 10$ number of odo cycles in a smoothing window

$\tau^\star = 1\,\mu s$ clock tick

$\tau^\circ = 50\,ms$ odo cycle

$\tau^\bullet = N^\bullet\tau^\circ = 10\cdot\tau^\circ = 500\,ms$ averaging window size

$$\frac{\Delta^\circ_n x}{\tau^\circ} \approx \left(\frac{\mathrm{d}x}{\mathrm{d}t}\right)_{t=t_n}$$

$$\frac{\Delta^\bullet_n x}{\tau^\bullet} \approx \left\langle\frac{\mathrm{d}x}{\mathrm{d}t}\right\rangle_{t=t_n-\tau^\bullet/2}$$

$\Delta^\circ_n x$ increment of quantity x during interval τ° of odo cycle n

$\Delta^\circ_n\theta = \theta_{n^\circ}$ increment of (wheel) angle during one odo cycle n of size τ°

$\Delta^\circ_n N^\otimes = N^\otimes_{n^\circ}$ increment of physical teeth during interval τ° on odo cycle n

$\Delta^\circ_n N^\ominus = N^\ominus_{n^\circ}$ increment of virtual teeth on odo cycle n of size τ° (n. of samples collected)

$\Delta^\circ_n\theta = \left(\frac{2\pi}{N^\ominus}\right)\Delta^\circ_n N^\ominus = 0.2\pi\Delta^\circ_n N^\ominus = \left(\frac{2\pi}{N^\otimes}\right)\Delta^\circ_n N^\otimes$

$N^\ominus_{n^\circ} \sim N^\ominus_n$ number of virtual teeth counted in odo cycle n

$N^\otimes_{n^\circ} \sim N^\otimes_n$ number of physical teeth counted in odo cycle n

$\delta^\otimes_k\theta = \theta_{,k^\otimes}$ increment of (wheel) angle from tooth k-1 to tooth k (fixed quantity)

$\delta^\ominus_k\theta = \theta_{,k^\ominus}$ increment of (wheel) angle from virtual tooth k-1 to tooth k (fixed quantity)

$\delta^\ominus_k\theta = \left(\frac{2\pi}{N^\ominus}\right)\cdot\delta^\ominus_k N^\ominus = \left(\frac{2\pi}{N^\ominus}\right)\cdot 1 = 0.314 = 0.2\pi\,\mathrm{rad}$ (constant)

$\delta^\otimes_k\lambda^\star_{n^\circ} = \lambda^\star_{n^\circ,k^\otimes} \sim \lambda^\star_{n,k^\otimes}$ clock sample in cycle n from physical tooth $k-1$ to tooth k

$\delta^\ominus_k\lambda^\star_{n^\circ} = \lambda^\star_{n^\circ,k^\ominus} \sim \lambda^\star_{n,k^\ominus}$ clock sample in cycle n from virtual tooth $k-1$ to tooth k

$\Delta^\bullet N^\ominus = \sum_{j=0}^{N^\bullet-1}\Delta^\circ_{n-j}N^\ominus$ Number of samples (teeth) over $N^\bullet$ odo cycles (500 ms)

$$[\Delta^\bullet N^\ominus]_W = N^\ominus\left\lfloor\frac{\sum_{j=0}^{N^\bullet-1}\Delta^\circ_{n-j}N^\ominus}{N^\ominus}\right\rfloor$$

Number of samples (teeth) over $N^\bullet$ odo cycles (500 ms) and completing a number of full wheel turns

$$\langle v\rangle^\bullet_n = \frac{1}{N^\bullet}\sum_{j=0}^{N^\bullet-1}v_{n-j}$$

moving average over $N^\bullet$ cycles trailing and including current cycle n.

$\widetilde{v}^\bullet_n$ speed smoothed over a window $\tau^\bullet$ trailing odo cycle n

$T_\kappa(v) = \delta^\ominus_\kappa\lambda\tau^\star$ Time interval between tooth $k-1$ and tooth k

$$\frac{\delta^\otimes x}{\delta^\otimes_k\lambda\tau^\star} \approx \left(\frac{\mathrm{d}x}{\mathrm{d}t}\right)_{\mathrm{tooth}k}$$

$$\frac{\delta^\ominus x}{\delta^\ominus_k\lambda\tau^\star} \approx \left(\frac{\mathrm{d}x}{\mathrm{d}t}\right)_{\mathrm{vtooth}k}$$

$\Delta^\circ_n x$ increment of x over sampling interval n of fixed size τ°

$\delta^\otimes_k x$ increment of x over interval between detection of tooth $k-1$ and k of variable size $\lambda^\star_k\cdot\tau^\star$ where λ_k is the count of clock pulses between the two physical teeth detections

Figure A4. ***explanation to symbols used in following text***

A.1.4.1 Basic principles for odometry modeling

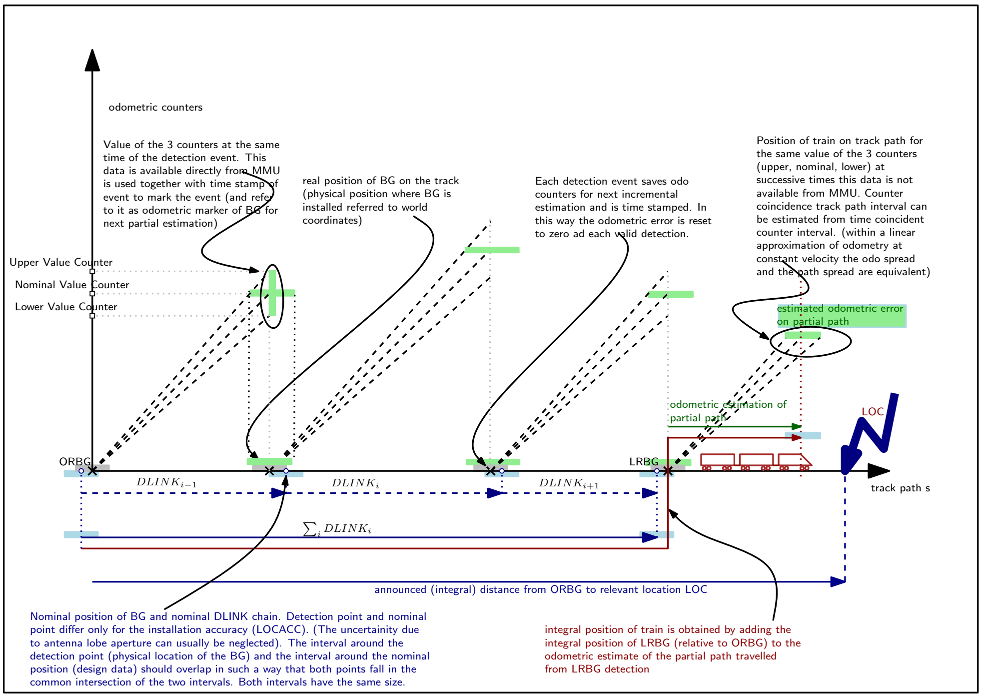


Figure A5. ***odometric conversion***

The main role/function of the MMU involves acquiring the measured physical quantities on the various sensors, to process them and to supply, in safety, to each EVC main unit the following set of information:

- A speed range [speed min , speed nom , speed max].
- A cumulated distance range [distance min , distance nom , distance max].
- An acceleration range [acceleration min , acceleration nom , acceleration max].
- The direction of travel (Direction of Travel (DOT)).
- The motion information.
- The adhesion loss detection.

A.1.4.2 Odometry model performance

Measure / Estimate	examples of range / value for an odometry model
Distance accuracy	±(4m + 5%)
Speed accuracy	± 2 km/h for v< 30 km/h, then increasing linearly up to ± 12 km/h at v = 500 km/h
Speed *resolution*	$\frac{2\pi R}{N^{\ominus}\cdot\tau^{\circ}} = \frac{0.314m}{.05s}$ ($\approx 6.28\,m/s \approx 22.6\,km/h$) minimal speed to have at least one new sample each cycle
data acquisition cycle τ°	50 ms between measurement
Cycle time	Data transmission to Core each 100 ms
Distance resolution	0.34 m (min for standstill detection = 0.5 m)
Speed resolution (GATC DESG 0638)	0,025 m/s ($\approx$ 0.1 km/h)
Speed range	0 to 500 km/h
Direction of travel (DOT)	Mandatory in objective SIL4
Motion detection	Motion detected if v > 0,5 km/h max or d > 0,5 m max
Safety Integrity Level	SIL4

A.1.4.3 Encoders and WS

Incremental encoders provide a specific number of equally spaced pulses per revolution (PPR) or per inch or millimeter of linear motion. A single channel output is used for applications where sensing the direction of movement is not important. Where direction sensing is required,

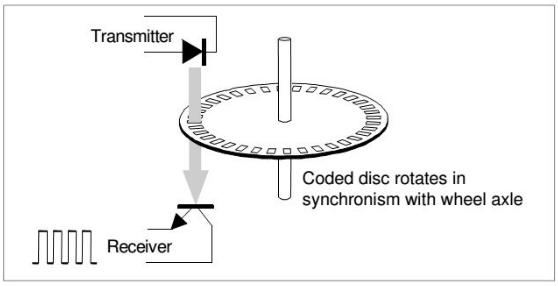


Figure A6. ***wheel sensor***

quadrature output is used, with two channels 90 electrical degrees out of phase; circuitry determines direction of movement based on the phase relationship between them. This is useful for processes that can reverse, or must maintain net position when standing still or mechanically oscillating. For example, machine vibration while stopped could cause a unidirectional encoder to produce a stream of pulses that would be erroneously counted as motion. The controller would not be fooled when quadrature counting is used.

When more resolution is needed, it is possible for the counter to count the leading and trailing edges of the pulse train from one channel, which doubles (x2) the number of pulses counted for one rotation or inch of motion. Counting both leading and trailing edges of both channels will give 4x resolution. An incremental encoders output indicates motion. To determine position, its pulses must be accumulated by a counter. The count is subject to loss during a power interruption or corruption by electrical transients. When starting up, the equipment must be driven to a reference or home position to initialize the position counters. Some incremental encoders also produce another signal known as the marker, index, or Z channel. This signal, produced once

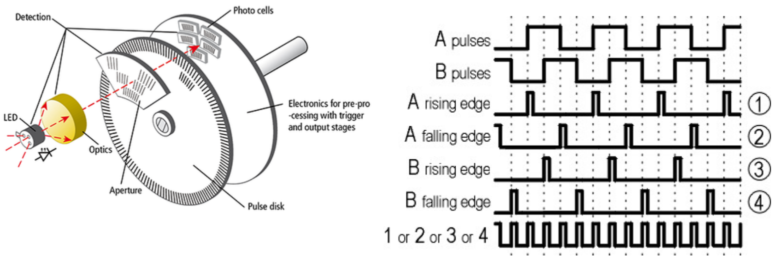


Figure A7. ***encoder output***

per revolution of a shaft encoder or at precisely-known points on a linear scale, is often used to locate a specific position, especially during a homing sequence. Resolution is the number of measuring segments or units in one revolution of an encoder shaft or one inch or mm of a linear scale. Shaft encoders are available with resolutions up to 10,000 pulses per revolution (PPR) directly, and 40,000 PPR by edge-detection of the A and B channels, while linear encoders are available with resolutions measured in microns. The bottom line is, the selected encoder must have resolution equal to or better than that required by the application. But resolution is not the whole story. Accuracy and resolution are different, and it is possible to have one without the other. This figure shows a distance X divided into 24 increments or bits. If X represents 360 deg of shaft rotation, then one revolution has been resolved into 24 parts. While there are 24 bits of

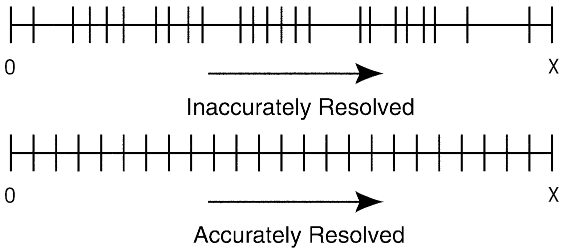


Figure A8. ***accuracy***

resolution, the 24 parts are not uniform. This transducer could not be used to measure position, velocity or acceleration with any accuracy. On the other hand, in this figure the distance X is divided into 24 equal parts. Each increment represents exactly 1/24 of a revolution. This transducer operates with accuracy as well as resolution. Accuracy, however, can be independent of resolution. A transducer may have a resolution of only two parts per revolution, yet its accuracy could be ±6 arc seconds.

System Accuracy: An encoder performance is typically stated as resolution, rather than accuracy of measurement. The encoder may be able to resolve movement into precise bits very accurately, but the accuracy of each bit is limited by the quality of the machine motion being monitored. For example, if there are deflections of machine elements under load, or if there is a drive screw with 0.1 inch of play, using a 1000 count-per-turn encoder with an output reading to 0.001 inch will not improve the 0.1 inch tolerance on the measurement. The encoder only reports position; it cannot improve on the basic accuracy of the shaft motion from which the position is sensed.

System Repeatability: Repeatability is the tolerance to which the controlled machine element can be repeatedly positioned to the same point in its travel. Repeatability is generally less than system resolution, but somewhat better than system accuracy. 10,000 pulses per turn can be generated from a 2500 cycle, two channel encoder. Typically with a Dynapar encoder, this 4x signal will be accurate to better than ±1 count.

Resolution is the ability to 'resolve' differences; that is, to draw a distinction between two things. High resolution means being able to resolve small differences. In a digital system, resolution means the smallest increment or step that can be taken or seen. In an analog system, it means the smallest step or difference that can be reliably observed.

The accuracy of a system refers to how much the system, whether in measurement or control, deviates from the truth. To be meaningful, accuracy must really refer to 'worst case accuracy'. As a trivial example, a stopped, unworking clock is still correct twice a day! We must describe this unworking clock by its worst case accuracy, which would be +/- 6 hours. Sometimes the '+/-' is dropped and the extent of the inaccuracy is reported: for this clock, it would be 12 hours. Beyond describing the worst case, we must also consider resolution. The following is very important: The accuracy of a system can never exceed its resolution! This follows from the worst case requirement.

Repeatability is the variance in repeated measurements on an unchanging sample. If measured statistically, it is sometimes referred to as the noise level in the measurement. In general, high repeatability means a low noise level. Precision is used in different ways depending on context. By design, measurement resolution is normally set better than accuracy. In the case of contact angle measurements, the resolution might be 0.01 degree but the absolute accuracy 1 degree. Why have resolution better than accuracy? One reason is that often accuracy is limited by the availability of calibration standards to show or prove the accuracy. If the best available standard is +/- 0.5 degrees, you can not claim an accuracy better than that. However, the ability to measure small differences between samples, whatever the absolute number, is useful and for this reason a higher resolution is provided. In general, instruments can measure repeatedly the same sample with a variance of a few units of resolution. This is called the noise level of the measurement. If we report a resolution of 0.01 degree, we expect to measure a static sample with a repeatability of, say, +/-0.02 or 0.03 degrees.

A.1.5 Speed measure

The[5] classical and probably the simplest method to measure rotor speed is the direct measure of the frequency of the encoder pulses.

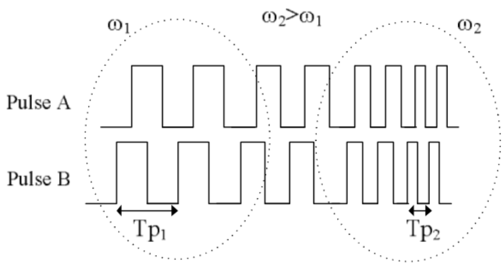


Figure A9. ***encoder output while speed increases***

Typically the number of the observed pulses inside a given and constant-width time-window is counted. Angular speed is then approximated to the discrete incremental ratio, that is constant speed is considered inside the observation windows:

$$\omega = \frac{d\theta}{dt} \approx \frac{\Delta_n^\circ \theta}{\tau^\circ} = \left(\frac{2\pi}{N^\otimes}\right)\frac{\Delta_n^\circ N^\otimes}{\tau^\circ} \tag{A2}$$

[5] Petrella Speed Measurement Algorithms for Low-Resolution Incremental Encoder Equipped Drives: a Comparative Analysis [**?**]

Where $N^{\otimes}$ is the number of pulses per revolution, $\Delta_n^{\circ}\theta$ is the angular increment and $\Delta_n^{\circ}N$ the corresponding number of pulses counted within the cycle n corresponding (observation window of fixed size τ°). As a concrete model we assume to have following values $N^{\otimes} = 50\,$p.p.r. (pulses per revolution) a wheel diameter of about $R \approx 1$ m (ICE trains have about 980 mm, while Euro-sprinter are between 1.1 and 1.2m) at $v_{train} = 5$km/h $= 1.4\,$m/s whith $\tau^{\circ} = 50\,ms$ we get:

$$\boxed{\Delta_n^{\circ}N^{\otimes} = v_{train}[m/s] \cdot \left(\frac{\tau^{\circ}}{2\pi R}\right) \cdot N^{\otimes} = 1.4 \cdot \frac{0.05}{3.14} \cdot 50 = 1.4 \cdot (0.796) = 1.1} \qquad \textbf{(A3)}$$

We observe that at low speeds there will be no increment in pulse counter $\Delta_n^{\circ}N^{\otimes}$ within the observation window τ° (no variation is detected in one single MMU cycle, the counter may vary after more than one MMU cycle).

speed v [km/h]	speed v [m/s]	approximate pulse (teeth) count in a 50 ms time window ($v \cdot 0.796$)	virtual teeth $\Delta_n^{\circ}N/5$
$v_{train} \approx 1\,$km/h	0.28 m/s	$\Delta_n^{\circ}N^{\otimes} \approx 0.22$	$\Delta_n^{\circ}N^{\ominus} \approx .044$
$v_{train} \approx 3.6\,$km/h	1 m/s	$\Delta_n^{\circ}N^{\otimes} \approx 0.796$	$\Delta_n^{\circ}N^{\ominus} \approx .159$
$v_{train} \approx 5\,$km/h	1.4 m/s	$\Delta_n^{\circ}N^{\otimes} \approx 1.1$	$\Delta_n^{\circ}N^{\ominus} \approx .22$
$v_{train} \approx 10\,$km/h	2.8 m/s	$\Delta_n^{\circ}N^{\otimes} \approx 2.2$	$\Delta_n^{\circ}N^{\ominus} \approx .44$
$v_{train} \approx 11.3\,$km/h	3.14 m/s	$\Delta_n^{\circ}N^{\otimes} \approx 2.5$	$\Delta_n^{\circ}N^{\ominus} \approx .5$
$v_{train} \approx 22.6\,$km/h	6,28 m/s	$\Delta_n^{\circ}N^{\otimes} \approx 5$	$\Delta_n^{\circ}N^{\ominus} \approx 1$
$v_{train} \approx 36\,$km/h	10 m/s	$\Delta_n^{\circ}N^{\otimes} \approx 7.96$	$\Delta_n^{\circ}N^{\ominus} \approx 1.59$
$v_{train} \approx 50\,$km/h	14 m/s	$\Delta_n^{\circ}N^{\otimes} \approx 11$	$\Delta_n^{\circ}N^{\ominus} \approx 2.2$
$v_{train} \approx 72\,$km/h	20 m/s	$\Delta_n^{\circ}N^{\otimes} \approx 15.92$	$\Delta_n^{\circ}N^{\ominus} \approx 3.18$
$v_{train} \approx 100\,$km/h	28 m/s	$\Delta_n^{\circ}N^{\otimes} \approx 22$	$\Delta_n^{\circ}N^{\ominus} \approx 4.4$
$v_{train} \approx 150\,$km/h	42 m/s	$\Delta_n^{\circ}N^{\otimes} \approx 33$	$\Delta_n^{\circ}N^{\ominus} \approx 6.6$
$v_{train} \approx 160\,$km/h	44.8 m/s	$\Delta_n^{\circ}N^{\otimes} \approx 35.66$	$\Delta_n^{\circ}N^{\ominus} \approx 7.132$
$v_{train} \approx 300\,$km/h	84 m/s	$\Delta_n^{\circ}N^{\otimes} \approx 66.864$	$\Delta_n^{\circ}N^{\ominus} \approx 13.372$
$v_{train} \approx 360\,$km/h	100 m/s	$\Delta_n^{\circ}N^{\otimes} \approx 79.6$	$\Delta_n^{\circ}N^{\ominus} \approx 15.92$

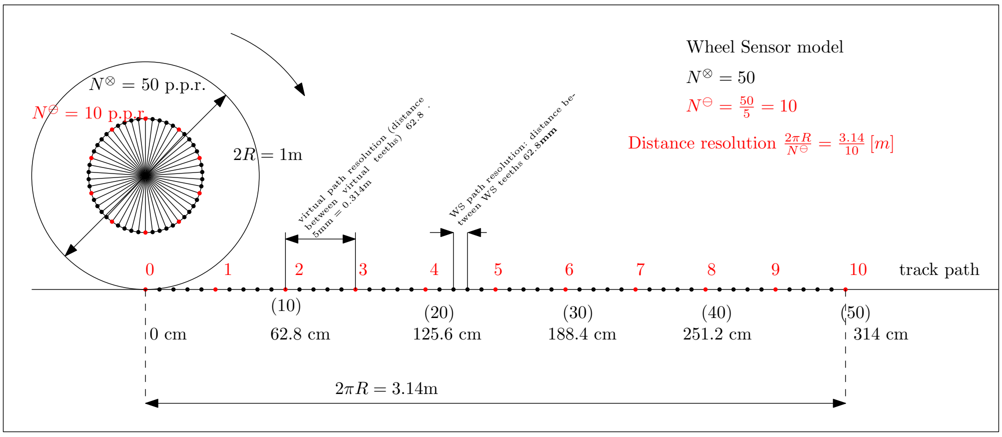


Figure A10. ***virtual teeth and path resolution***

An alternative solution to improve the measurement performance at lower speed is to switch from frequency to period measurement e.g. below a certain level of (low) speed. The measurement is realized by counting the number of periods of a high frequency signal inside one (or more) encoder pulses, Fig. A11. The following formula is obtained under the hypothesis that motor speed is constant and only one period $\delta_k^{\ominus} N^{\ominus} = 1$ of the encoder signal is considered:

$$\omega = \frac{\mathrm{d}\theta}{\mathrm{d}t} \approx \frac{\delta_k^{\ominus}\theta}{\delta_k^{\ominus}\lambda^{\star} \cdot \tau^{\star}} = \left(\frac{2\pi}{N^{\ominus}}\right) \frac{1}{\lambda^{\star}_{,k^{\ominus}} \cdot \tau^{\star}} \tag{A4}$$

A speed sample λ_k can be accumulated each period of the encoder signals. The speed sampling period is thus a function of wheel speed:

$$T_k^{\ominus}(\omega) = \delta_k^{\ominus}\lambda^{\star} \cdot \tau^{\star} = \left(\frac{2\pi}{N^{\ominus}}\right) \frac{1}{\omega} = \frac{T_{WS}(\omega)}{N^{\ominus}} \tag{A5}$$

where ω is the angular speed of the wheel and T_{WS} would be the period of a complete revolution of the wheel sensor at constant angular speed. The accuracy of the measurement is related to the ratio between the period of the high frequency counter and that of the encoder signals, that is not an integer value as it depends on motor speed. A calculation of absolute and percentage errors is extremely difficult as it involves non-linear rounding functions. A worst-case condition error could be calculated instead by considering the absolute error of one high frequency pulse, leading to a maximum percentage error limiting locus given by the following formula, i.e. roughly linear with speed:

$$e_k(\omega) = \frac{\tau^{\star}}{\frac{2\pi}{N^{\otimes}\cdot\omega} - \tau^{\star}} = \frac{\omega N^{\ominus}\tau^{\star}}{2\pi\left(1 - N^{\ominus} \cdot \frac{\tau^{\star}}{T_{WS}}\right)} \approx \frac{\omega \cdot N^{\ominus} \cdot \tau^{\star}}{2\pi} = \frac{N^{\ominus}\tau^{\star}}{T_{WS}(\omega)} = \frac{\tau^{\star}}{T_k^{\ominus}(\omega)} \tag{A6}$$

(the factor 100 is omitted)

At very low speed the number of high frequency pulses can be extremely high and saturation of the digital timer employed for measurement can occur. Also a speed sample is not available each speed control period, needing an adaptation of the control parameters. In that situation, the quadrature decoding of the encoder pulses can be exploited in order to reduce the width of the measuring window by a factor of four or a reduction of the frequency of the timer can be considered.

The former solution allows to reduce the speed sampling period and improve the control performance of the drive, but makes the measuring system more sensitive to sensor nonidealities, including variations in the transition locations from their nominal values and phasing errors between encoder channels. When low-cost and low-resolution sensors are employed, nonidealities play the major role in the determination of period measuring errors and has to be carefully analyzed in drive design phase. The latter solution needs to switch on-line the frequency of the timer and adapt the coefficients of the equations above.

The implementation of the period measuring method is also straightforward as it requires a simple timer capture unit, commonly found inside recent micro-controllers. Specific hardware subsystems can be considered to overcome digital timer saturation problems and proper algorithms are needed to filter out the errors introduced by sensor nonidealities. The problem of speed calculation error generated at motor reversion can be easily solved by the identification of motor direction during time measurement and correction of the timer value, but best performance are obtained by means of specific hardware solutions.

A.1.5.1 sensor data

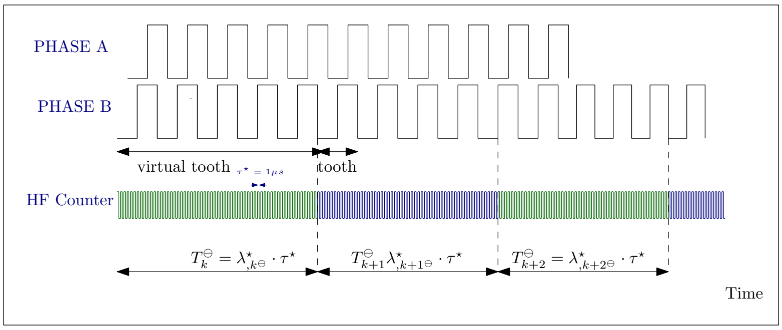


Figure A11. *period measure*

A.1.5.1.1 Delta path (MMU cycle path increment)

[6] The MMU processor executes a fetch cycle each $\tau^\circ = 50\,\mathrm{m}s$. The samples which are available as raw data are used to calculate the path traveled on the last 50 ms. this is also called Delta-distance or path increment during cycle n.

$$\Delta_n^\circ s = R\Delta_n^\circ \theta = \left(\frac{2\pi R}{N^\ominus}\right)\Delta_n^\circ N^\ominus \tag{A7}$$

where $\Delta_n^\circ N$ [7] indicates the number of samples cumulated in the MMU cycle n.

A.1.5.1.2 current speed

From the samples $\lambda_{n,k}$ where $n \in [0, \infty]$ is the index of the MMU cycle while $k \in [0, \Delta_n^\circ N^\ominus - 1]$ is the index of the samples cumulated in the last cycle at virtual tooth border within the counter array,[8] the current speed on the cycle n can now be calculated:

$$v_{n^\circ} = \frac{\Delta_n^\circ s}{\sum\limits_{k=0}^{\Delta_n^\circ N^\ominus - 1} \lambda^\star_{n,k}\tau^\star} \tag{A8}$$

A.1.5.1.3 smoothed speed

To avoid small oscillations of the speed due to the eccentricity of the fitted sensor relative to axle of rotation, the speed is smoothed over a full number of wheel revolutions while keeping a delay of 250 ms for the synchronization. The raw samples of are transferred to a local buffer together with samples of previous cycles. The sum of the last $N_{\tau^\bullet}$ raw samples (buffered locally) is chosen so that the sum of samples times $\tau^\star$ is immediately less than 500 ms (10 MMU cycles) and $N_{\tau^\bullet}$ is a multiple of the number of virtual teeth $N^\ominus$ of the sensor.

$$\Delta_n^\bullet N^\ominus = N^\ominus \left\lfloor \frac{\sum\limits_{j=0}^{N^\bullet - 1} \Delta_{n-j}^\circ N^\ominus}{N^\ominus} \right\rfloor \tag{A9}$$

[6] called also delta distance

[7] Later on where the context is clear we will drop the Delta symbol Δ and simply write $N_{n^\circ}^\ominus = N_n^\ominus$ to indicate the number os samples acquired in the MMU cycle n.

[8] The array has a size NB_SAMPLES_MAX, which limits the maximum number of samples, that can be stored in one cycle

Where the parenthesis ⌊⌋ indicate integer part (floor). To obtain the raw speed, $\Delta_n^\bullet N$ is multiplied by the distance resolution and divided by this sum:

$$\widetilde{v}_{n^\bullet} = \frac{\Delta_n^\bullet N^\ominus \left(\frac{2\pi R}{N^\ominus}\right)}{\sum_{j=0}^{N^\bullet-1} \sum_{k=0}^{\Delta_n^\circ N^\ominus-1} \lambda^\star_{n-j,k}\tau^\star} \tag{A10}$$

$$\widetilde{v}_{n^\bullet} = \frac{\Delta_n^\bullet N^\ominus \left(\frac{2\pi R}{N^\ominus}\right)}{\sum_{\kappa=0}^{\Delta_n^\bullet N^\ominus-1} \lambda^\star_{,\kappa}\tau^\star} \tag{A11}$$

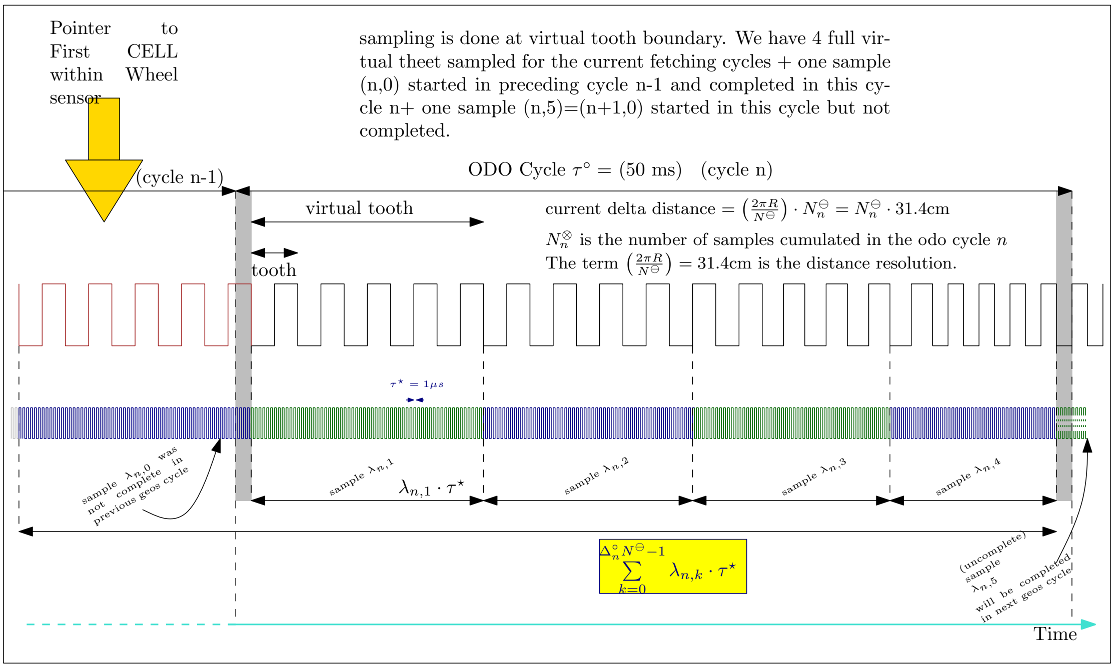


Figure A12. ***sampling on a MMU cycle***

A.1.5.1.4 moving average over delta path

The sensors output are to be synchronized every 250 ms. This is achieved by using a moving average on a period of time = (500ms latency time). The latency time of the sensors has to be less than 250ms. Currently, the used sensors have a negligible latency time and so, the moving average is computed on 500 ms. The speed estimation is computed using a smoothing mechanism over the speed measurements obtained during the last 500 ms leading to a delay of 250 ms on the speed estimation. The purpose of this mechanism is to smooth the speed measurement and to synchronize at 250 ms the speed estimations between the different sensors.

The delta-path ($\Delta_n^\circ s$) will now be averaged using a moving window over the last $N^\bullet$ sampled and stored $\Delta_{n-j}^\circ s$ values (impulses times wheel sensor resolution) acquired in the last 500 ms ($N^\bullet$ cycles = 10 data values). The expression used for the moving average if we consider also the current value ($j = 0$) is given by the formula:

$$\langle\Delta s\rangle_n^\bullet = \frac{1}{N^\bullet} \sum_{j=0}^{N^\bullet-1} \Delta_{n-j}^\circ s \tag{A12}$$

which can be updated recurrently at each cycle:

$$\langle\Delta s\rangle_{n+1}^\bullet = \langle\Delta s\rangle_n^\bullet + \frac{1}{N^\bullet}\left(\Delta_{n+1}^\circ s - \Delta_{n-N+1}^\circ s\right) \tag{A13}$$

where $N^\bullet = 10$ is the size of the sampling buffer and the computation is extended over $N^\bullet$ values including the current data value ($j = 0$). The index n on the angular brackets indicates that the sample mean is done on the running buffer containing the data sample trailing the data value acquired at cycle n. The index j gives the relative delay of the data point compared to the current data point n, while the index i gives the number of the cycle where the data point was acquired, where the number of the fetching cycle has to be considered from the startup of the program, Usually such an index is not available and actual calculations should be made by using the relative index j at each cycle n The life span of a odometry increment is thus equal to 10 cycles. This allows to improve the resolution of the delta-path. The handling of vote mechanism for distance will be than performed voting over the values $\langle \Delta s \rangle_n^\bullet$ obtained using the formula above. The corresponding time value should be the one of the center of the averaging window, which is delayed by $j = N/2$ cycles relative to current fetch cycle. The correct direction has to be associated to the distance increment.

A.1.5.2 estimating acceleration from speed measures

The acceleration can be estimated from the speed data sampled in the current buffer (sample n including current speed measure v_n) by implementing a linear regression on the sample. Two possible linear estimators[9] are given by :

$$\hat{v}(t_i) = \hat{v}_n + \hat{a}_n (t_i - t_n) \quad \text{(A14)}$$

$$\hat{v}(t_i) = \hat{v}_n^\spadesuit + \hat{a}_n (t_i - \langle t \rangle_n) \quad \text{(A15)}$$

where the index n indicates the value currently acquired and the associated trailing buffer storing another $N = 2M$ sampled data. The total sample will have $N^\bullet$ data points (v_i, t_i). The hat symbol will indicate estimated quantities e.g. $\hat{a}_n$ indicates the estimated acceleration over the sample n. There are two parameters to estimate $\hat{v}_n$ and $\hat{a}_n$. The angular brackets indicate the mean value over the sample n obtained by averaging over the $N + 1$ data points available at fetching cycle n. For the time value this is the arithmetic mean value (there is no statistical spread over the t variable and the fluctuation on the cycle time τ is considered negligible). Therefore this value is given only by the buffer size (time window) and can be calculated even without data.

$$\begin{aligned}
\langle t \rangle_n^\bullet &= \frac{1}{N^\bullet} \sum_{i=n}^{n-N^\bullet+1} t_i \\
&= \frac{1}{N^\bullet} \sum_{j=0}^{N^\bullet-1} t_{n-j} \\
&= t_{n-N^\bullet/2} \\
&= t_n - \frac{N^\bullet}{2} \tau^\circ
\end{aligned} \quad \text{(A16)}$$

$$\boxed{\langle t \rangle_n^\bullet = t_n - \frac{N^\bullet}{2} \tau} \quad \text{(A17)}$$

and recurrently:

$$\langle t \rangle_{n+1}^\bullet = \langle t \rangle_n^\bullet + \tau^\bullet \quad \text{(A18)}$$

[9] *regression* meaning the prevailing of the average over the fluctuations. The concept of *regression to mediocrity*, introduced by Sir Galton reflects the behavior of data when numerous small effects influence and produce fluctuations superposed to systematic behavior. This behavior would certainly not be applicable to a random walk process.

where $M\tau = N/2\tau$ is the delay of the center of the buffering window (average delay) with respect to the current cycle. in a similar way for the speed we have:

$$\langle v \rangle_n^\bullet = \frac{1}{N^\bullet} \sum_{i=n}^{n-N^\bullet+1} v_i = \frac{1}{N^\bullet} \sum_{j=0}^{N^\bullet-1} v_{n-j} \tag{A19}$$

which can be updated recurrently by:

$$\boxed{\langle v \rangle_{n+1}^\bullet = \langle v \rangle_n^\bullet + \frac{1}{N^\bullet} \left(v_{n+1} - v_{n-N^\bullet+1}\right)} \tag{A20}$$

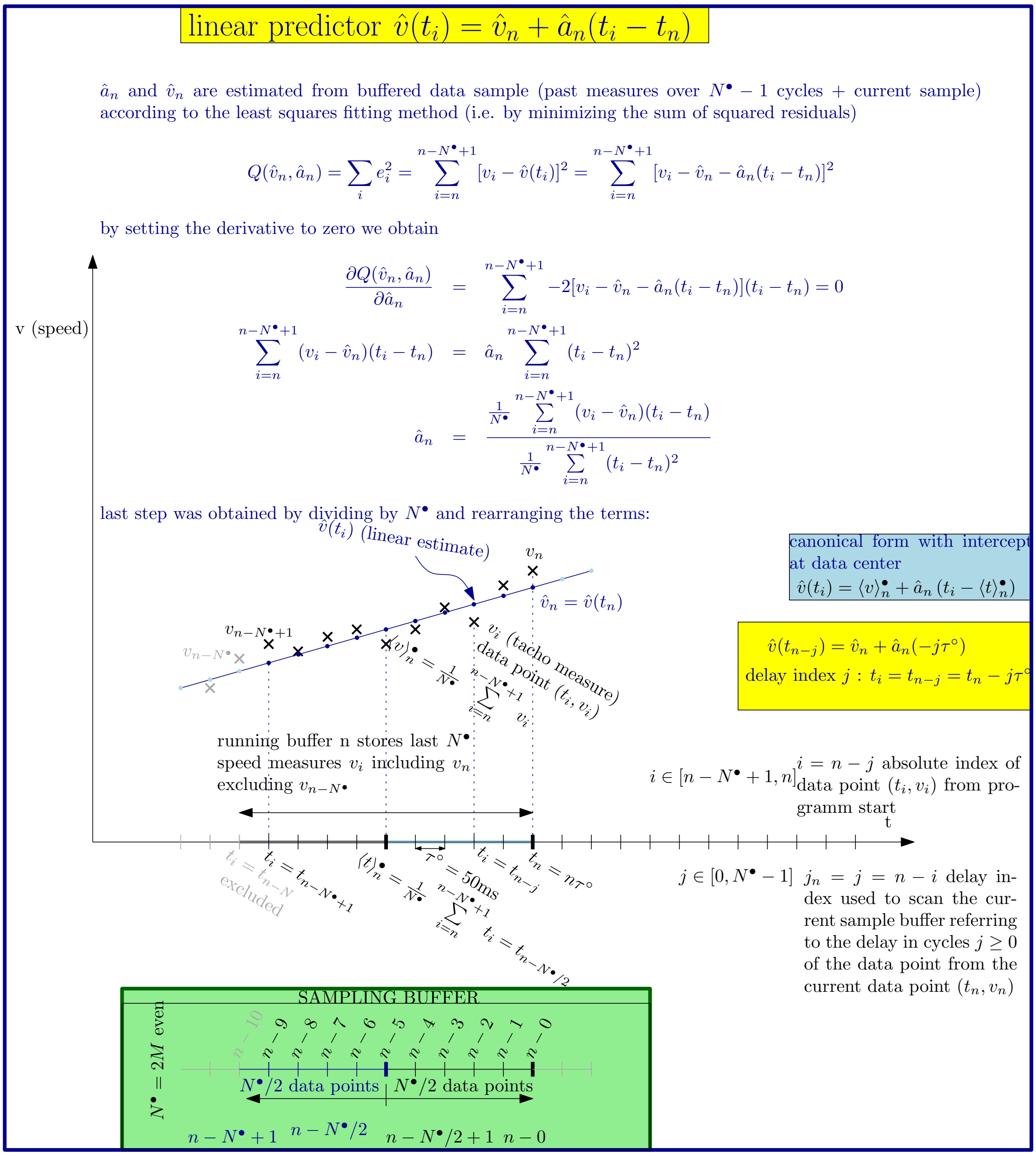


Figure A13. ***estimate acceleration applying linear regression***

A way to estimate the parameters of the linear estimator consists in building an error function, aimed to produce a metric by summing up the distance or residual error between acquired data points and estimated data markers [6]. This metric will allow to optimize the parameters for the smallest error. One of the most common error function is given by the sum of squared errors, the confidence interval of all data points being assumed identical. This is known as *least square estimation*, and can be applied to a vast spectrum of (not only linear) functions containing parameters e.g. it si applicable for the coefficient estimation of any Fourier sum.

$$Q(\hat{v}_n, \hat{a}_n) = \sum_{i=n}^{n-N^\bullet+1} e_i^2 = \sum_{i=n}^{n-N^\bullet+1} [v_i - \hat{v}(t_i)]^2 = \sum_{i=n}^{n-N^\bullet+1} [v_i - \hat{v}_n - \hat{a}_n(t_i - t_n)]^2 \tag{A21}$$

We insert one of the selected estimators (A14) in the Q function, derive it and put the result to zero to locate the minimum.

$$\frac{\partial Q(\hat{v}_n, \hat{a}_n)}{\partial \hat{v}_n} = \sum_{i=n}^{n-N^\bullet+1} 2\frac{\partial \hat{v}(t_i)}{\partial \hat{v}_n}[v_i - \hat{v}_n - \hat{a}_n(t_i - t_n)] = 0 \tag{A22}$$

dividing by $N^\bullet$ this gives:

$$\begin{aligned} \hat{v}_n &= \frac{1}{N^\bullet} \sum_{i=n}^{n-N^\bullet+1} [v_i - \hat{a}_n(t_i - t_n)] \\ &= \langle v \rangle_n^\bullet - \hat{a}_n \left[\langle t \rangle_n^\bullet - t_n\right] \end{aligned} \tag{A23}$$

In a similar way inserting the estimator (A15) in the Q function we obtain after derivation:

$$\begin{aligned} \hat{v}_n^\star &= \frac{1}{N^\bullet} \sum_{i=n}^{n-N^\bullet+1} [v_i - \hat{a}_n(t_i - \langle t \rangle_n^\bullet)] \\ &= \langle v \rangle_n^\bullet \end{aligned} \tag{A24}$$

If we substitute the parameter $\hat{v}_n$ in the original function we obtain in both cases:

$$\boxed{\hat{v}(t_i) = \langle v \rangle_n^\bullet + \hat{a}_n \left(t_i - \langle t \rangle_n^\bullet\right)} \tag{A25}$$

centered on the center of data points sample $(\langle v \rangle_n^\bullet, \langle t \rangle_n^\bullet)$.[10] This is a canonical form for the linear predictor reflects the fact that the linear estimation function always includes the center of the data sample. The second parameter $\hat{a}_n$ being the inclination of the line pivoting about the data sample center. The estimated value for the last sample is given by adding the estimated acceleration over the half window period to the window center velocity value $v(\langle t \rangle_n) = \langle v \rangle_n$

$$\hat{v}(t_n) = \langle v \rangle_n^\bullet + \hat{a}_n \frac{N^\bullet}{2} \tau^\circ \tag{A26}$$

The error function becomes:

$$\boxed{Q(\hat{v}_n, \hat{a}_n) = \sum_{i=n}^{n-N^\bullet+1} [v_i - \langle v \rangle_n^\bullet - \hat{a}_n \left(t_i - \langle t \rangle_n^\bullet\right)]^2} \tag{A27}$$

We use now this expression to get the estimation of the second parameter $\hat{a}_n$

$$\frac{\partial Q(\hat{v}_n, \hat{a}_n)}{\partial \hat{a}_n} = \sum_{i=n}^{n-N^\bullet+1} 2\frac{\partial \hat{v}(t_i)}{\partial \hat{a}_n}[v_i - \langle v \rangle_n^\bullet - \hat{a}_n(t_i - \langle t \rangle_n^\bullet)] = 0 \tag{A28}$$

[10] With increasing values of n we get $N^\bullet$ different estimates $\hat{v}(t_i)$ for the same sample v_i, as the averaging window is moving forward in time.

dividing by $N^\bullet$ and reordering we get

$$\frac{1}{N^\bullet}\sum_{i=n}^{n-N^\bullet+1}[v_i - \langle v\rangle_n^\bullet - \hat{a}_n(t_i - \langle t\rangle_n^\bullet)](t_i - \langle t\rangle_n^\bullet) = 0 \tag{A29}$$

$$\boxed{\hat{a}_n = \frac{\frac{1}{N^\bullet}\sum_{i=n}^{n-N^\bullet+1}(v_i - \langle v\rangle_n^\bullet)(t_i - \langle t\rangle_n^\bullet)}{\frac{1}{N^\bullet}\sum_{i=n}^{n-N^\bullet+1}(t_i - \langle t\rangle_n^\bullet)^2} = \frac{\langle v\cdot t\rangle_n^\bullet - \langle v\rangle_n^\bullet\langle t\rangle_n^\bullet}{\langle t^2\rangle_n^\bullet - \langle t\rangle_n^{\bullet 2}}} \tag{A30}$$

the expression for $\langle t\rangle_n$ was already given in A16. We still need the expression for $\langle t^2\rangle_n$. Both $\langle t^2\rangle_n^\bullet$ and $\langle(t - \langle t^2\rangle_n^\bullet)^2\rangle_n^\bullet$ are characterized only by the buffer size and are data independent, while $\langle v\cdot t\rangle_n^\bullet$ and $\langle v\rangle_n^\bullet$ have to be evaluated from data.

$$\begin{aligned}\frac{1}{N^\bullet}\sum_{i=n}^{n-N^\bullet+1}(t_i - \langle t\rangle_n^\bullet)^2 &= \langle t^2\rangle_n^\bullet - \langle t\rangle_n^{\bullet 2} \\ &= \langle(t - \langle t\rangle_n^\bullet)^2\rangle_n^\bullet\end{aligned} \tag{A31}$$

Recalling that $\langle t\rangle_n^\bullet = t_n - \frac{N^\bullet}{2}\tau^\circ$ we get:

$$\begin{aligned}\frac{1}{N^\bullet}\sum_{i=n}^{n-N^\bullet+1}\tau^{\circ 2}\left(i - n + \frac{N^\bullet}{2}\right)^2 &= \\ \frac{\tau^{\circ 2}}{N^\bullet}\sum_{j=0}^{N^\bullet-1}(\frac{N^\bullet}{2} - j)^2 &= \tau^{\circ 2}\frac{(N^\bullet - 1)(N^\bullet + 1)}{12}\end{aligned} \tag{A32}$$

where we have changed the index $i - n = -j$

$$\boxed{\frac{1}{N^\bullet}\sum_{i=n}^{n-N^\bullet+1}(t_i - \langle t\rangle_n^\bullet)^2 = \tau^{\circ 2}\frac{(N^\bullet - 1)(N^\bullet + 1)}{12}} \tag{A33}$$

In a similar fashion :

$$\begin{aligned}\frac{1}{N^\bullet}\sum_{i=n}^{n-N^\bullet+1}(t_i - \langle t\rangle_n^\bullet) &= \langle t - \langle t\rangle_n^\bullet\rangle_n^\bullet \\ &= \langle t\rangle_n^\bullet - \langle t\rangle_n^\bullet \\ &= 0\end{aligned} \tag{A34}$$

and so finally the equation A30 becomes:

$\Leftarrow$ by using:

$$\langle v\rangle_n \sum_{i=n}^{n-N}(t_i - \langle t\rangle_n) = 0$$

$$\begin{aligned}
\hat{a}_n &= \frac{\frac{1}{N^\bullet}\sum\limits_{i=n}^{n-N^\bullet+1}(v_i - \langle v\rangle_n^\bullet)(t_i - \langle t\rangle_n^\bullet)}{\frac{1}{N^\bullet}\sum\limits_{i=n}^{n-N^\bullet+1}(t_i - \langle t\rangle_n^\bullet)^2} \\
&= \frac{\frac{1}{N^\bullet}\sum\limits_{i=n}^{n-N^\bullet+1} v_i(t_i - \langle t\rangle_n^\bullet)}{\frac{\tau^{\circ 2}(N^\bullet-1)(N^\bullet+1)}{12}} \\
&= \frac{\frac{1}{N^\bullet}\sum\limits_{i=n}^{n-N^\bullet+1} v_i t_i - \frac{\langle t\rangle_n^\bullet}{N^\bullet}\sum\limits_{i=n}^{n-N^\bullet+1} v_i}{\frac{\tau^{\circ 2}(N^\bullet-1)(N^\bullet+1)}{12}} \\
&= \frac{\frac{1}{N^\bullet}\sum\limits_{j=0}^{N^\bullet-1} v_{n-j}t_{n-j} - \langle t\rangle_n^\bullet\langle v\rangle_n^\bullet}{\frac{\tau^{\circ 2}(N^\bullet-1)(N^\bullet+1)}{12}} \\
&= \frac{\frac{1}{N^\bullet}\sum\limits_{j=0}^{N^\bullet-1} v_{n-j}(t_n - j\tau^\circ) - \langle t\rangle_n^\bullet\langle v\rangle_n^\bullet}{\frac{\tau^{\circ 2}(N^\bullet-1)(N^\bullet+1)}{12}} \\
&= \frac{\frac{t_n}{N^\bullet}\sum\limits_{j=0}^{N^\bullet-1} v_{n-j} - \frac{\tau^\circ}{N^\bullet}\sum\limits_{j=0}^{N^\bullet-1} j\cdot v_{n-j} - \langle t\rangle_n^\bullet\langle v\rangle_n^\bullet}{\frac{\tau^{\circ 2}(N^\bullet-1)(N^\bullet+1)}{12}} \\
&= \frac{\langle v\rangle_n^\bullet(t_n - \langle t\rangle_n^\bullet) - \frac{\tau^\circ}{N^\bullet}\sum\limits_{j=0}^{N^\bullet-1} j\cdot v_{n-j}}{\frac{\tau^{\circ 2}(N^\bullet-1)(N^\bullet+1)}{12}} \\
&= \frac{\langle v\rangle_n^\bullet\frac{\tau^\circ N^\bullet}{2} - \frac{\tau^\circ}{N^\bullet}\sum\limits_{j=0}^{N^\bullet-1} j\cdot v_{n-j}}{\frac{\tau^{\circ 2}(N^\bullet-1)(N^\bullet+1)}{12}}
\end{aligned} \tag{A35}$$

$$\boxed{\hat{a}_n = \left(\langle v\rangle_n^\bullet\frac{\tau^\circ N^\bullet}{2} - \frac{\tau^\circ}{N^\bullet}\sum_{j=0}^{N^\bullet-1} j\cdot v_{n-j}\right)\frac{12}{\tau^{\circ 2}(N^\bullet-1)(N^\bullet+1)}} \tag{A36}$$

which can be evaluated recurrently:

$$\boxed{\hat{a}_{n+1} = \hat{a}_n + \frac{6}{\tau^\circ}\frac{(N^\bullet-1)v_{n+1} - 2N^\bullet\langle v\rangle_n^\bullet + (N^\bullet+1)v_{n-N^\bullet}}{(N^\bullet-1)N^\bullet(N^\bullet+1)}} \tag{A37}$$

To obtain this relation simply substitute $n \rightarrow n+1$ in the definition A36

$$
\begin{aligned}
\hat{a}_{n+1} &= \left(\langle v \rangle^{\bullet}_{n+1} \frac{N^{\bullet}}{2} - \frac{1}{N^{\bullet}} \sum_{j=0}^{N^{\bullet}-1} j \cdot v_{n+1-j} \right) \frac{12}{\tau^{\circ}(N^{\bullet}-1)(N^{\bullet}+1)} \\
&= \left(\left(\langle v \rangle^{\bullet}_{n} + \frac{v_{n+1} - v_{n-N^{\bullet}+1}}{N^{\bullet}} \right) \frac{N^{\bullet}}{2} - \frac{1}{N^{\bullet}} \sum_{j=0}^{N^{\bullet}-1} j \cdot v_{n-(j-1)} \right) \frac{12}{\tau^{\circ}(N^{\bullet}-1)(N^{\bullet}+1)} \\
&= \left(\left(\langle v \rangle^{\bullet}_{n} + \frac{v_{n+1} - v_{n-N^{\bullet}+1}}{N^{\bullet}} \right) \frac{N^{\bullet}}{2} - \frac{1}{N^{\bullet}} \sum_{j=-1}^{N^{\bullet}-2} (j+1) \cdot v_{n-j} \right) \frac{12}{\tau^{\circ}(N^{\bullet}-1)(N^{\bullet}+1)} \\
&= \left(\left(\langle v \rangle^{\bullet}_{n} + \frac{v_{n+1} - v_{n-N^{\bullet}+1}}{N^{\bullet}} \right) \frac{N^{\bullet}}{2} - \frac{\sum\limits_{j=0}^{N^{\bullet}-1} j \cdot v_{n-j}}{N^{\bullet}} + v_{n-N^{\bullet}+1} - \langle v \rangle^{\bullet}_{n} \right) \frac{12}{\tau^{\circ}(N^{\bullet}-1)(N^{\bullet}+1)} \\
&= \hat{a}_n + \left(\frac{v_{n+1} - v_{n-N^{\bullet}+1}}{N^{\bullet}} \frac{N^{\bullet}}{2} + v_{n-N^{\bullet}+1} - \langle v \rangle^{\bullet}_{n} \right) \frac{12}{\tau^{\circ}(N^{\bullet}-1)(N^{\bullet}+1)} \\
&= \hat{a}_n + \left(\frac{\frac{N^{\bullet}}{2} v_{n+1} + \left(\frac{N^{\bullet}}{2} + 1 \right) v_{n-N^{\bullet}+1} - (N^{\bullet}) \langle v \rangle^{\bullet}_{n}}{N^{\bullet}} \right) \frac{12}{\tau^{\circ}(N^{\bullet}-1)(N^{\bullet}+1)}
\end{aligned}
\tag{A38}
$$

A.1.5.3 Goodness of fit

If the deviations from the mean follow a Gaussian statistics, the probability of making any one observation v_i at time t_i is given by:

$$
P(v_i) = \frac{1}{\sigma_i \sqrt{2\pi}} \exp\left(-\frac{[v_i - \mu(t_i)]^2}{2\sigma_i^2} \right) \approx \frac{1}{\sigma_i \sqrt{2\pi}} \exp\left(-\frac{[v_i - \hat{v}(t_i)]^2}{2\sigma_i^2} \right) \tag{A39}
$$

where we use the linear estimate $v(t_i)$ as estimate of the mean value $\mu(t_i)$ of the Gaussian being the value of velocity if measured at time t_i with an hypothetical error-free device. The total probability of obtaining a set of $N+1$ measurements,$\{t_i, v_i\}$, is equal to the product of the probabilities for each data point:

$$
\prod_{i=N-n}^{n} P(v_i) \approx \prod_{i=N-n}^{n} \left\{ \frac{1}{\sigma_i \sqrt{2\pi}} \exp\left(-\frac{[v_i - \hat{v}(t_i)]^2}{2\sigma_i^2} \right) \right\} \tag{A40}
$$

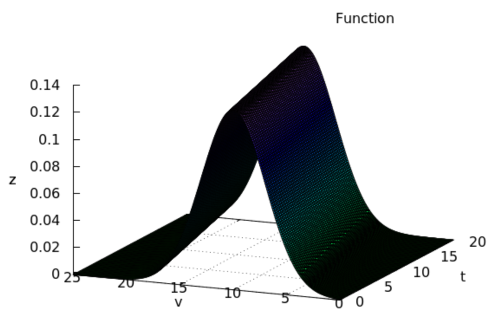


Figure A14. ***Gaussian family with "drifting" mean and constant variance***

Maximizing the probability is equivalent to minimize the sum in the exponential term of $P(\{v,t\})$, specifically the sum of the deviations weighted by σ_i^2,

$$\sum_{i=N-n}^{n} \frac{e_i^2}{2\sigma_i^2} = \sum_{i=N-n}^{n} \frac{[v_i - \hat{v}(t_i)]^2}{2\sigma_i^2} \tag{A41}$$

[11] In the special case where all the σ_i are identical we are we are led back to the least squares principle. The chi-square statistic is defined by following sum where we replace the absolute index i with the relative index j ($i = n - j$):

$$\boxed{\chi^2 = \sum_{j=0}^{N} \frac{e_{n-j}^2}{\sigma_{n-j}^2} = \sum_{j=0}^{N} \frac{\left[v_{n-j} - \hat{v}(t_{n-j})\right]^2}{\sigma_{n-j}^2}} \tag{A44}$$

If the terms of the sum are of the order of ≈ 1 we get that χ increases with the number of observations $N + 1$. The method of least squares is built on the hypothesis that the optimum description of a set of data is one which minimizes the weighted sum of squares of deviations, e_i, between the data, v_i , and the fitting function $\hat{v}(t_i)$.

The sum of squares of deviations is characterized by the estimated variance of the fit, s^2 , which is an estimate of the variance of the parent distribution, σ^2.

$$\begin{aligned} s^2 &= \frac{1}{N+1} \sum_{j=0}^{N} (v_{n-j} - \hat{v}(t_{n-j}))^2 \\ \sigma^2 &= \lim_{N\to\infty} \frac{1}{N+1} \sum_{j=0}^{N} (v_{n-j} - \mu(t_{n-j}))^2 \end{aligned} \tag{A45}$$

The ratio of s^2/σ^2 can be estimated by χ^2/ν, where $\nu = N + 1 - p$ is called degree of freedom, $N + 1$ is the number of observations and p is the number of fitting parameters. The expression χ^2/ν is called the reduced chi-square statistic. For $p = 2$ we get $\nu = N - 1$:

$$\boxed{\frac{\chi^2}{\nu} = \frac{1}{N-1} \sum_{j=0}^{N} \frac{\left[v_{n-j} - \hat{v}(t_{n-j})\right]^2}{\sigma_{n-j}^2}} \tag{A46}$$

If the fitting function $\hat{v}(t_{n-j})$ accurately predicts the means $\mu(t_{n-j})$[12] of the parent distribution, then the estimated variance, s^2 , should agree well with the variance of the parent distribution,

[11] In case of variable variance σ_i the expression to be maximized is

$$\prod_{i=N-n}^{n} \frac{1}{\sqrt{2\pi}} exp\left(-\frac{[v_i - \hat{v}(t_i)]^2}{2\sigma_i^2} - \log\sigma_i\right) \tag{A42}$$

and therefore the exponent to be minimized is:

$$\sum_{i=N-n}^{n} \left(\frac{[v_i - \hat{v}(t_i)]^2}{2\sigma_i^2} + \log\sigma_i\right) \tag{A43}$$

But as we parametrize only the model estimating the mean value of $v(\hat{t_i})$ the term $\log\sigma_i$ is a constant and can be ignored. The model predicts only the mean value and tries not to guess the variance interval.

[12] the velocity at time t_{n-j} measured with a much better device, having a much smaller error

σ^2 , and their ratio should be close to one. This explains the origin of the rule of thumb for chi-square fitting that states that a good fit is achieved when the reduced chi-square equals one. The Chi-Square (χ^2) distribution function gives the probability distribution for any quantity which is the sum of the squares of independent, normally-distributed variables with unit variance. In is used in the method of maximum likelihood for testing the functional relationship between measured quantities.

$$\boxed{p(\chi^2, \nu) = \frac{1}{2^{\frac{\nu}{2}}\Gamma(\frac{\nu}{2})} \left(\chi^2\right)^{\frac{\nu-2}{2}} \exp\left[-\frac{\chi^2}{2}\right]} \tag{A47}$$

Resuming the least squares method gives a general principle to obtain a fitting between a N-diemnsional vector (v_i) of measured data with an N-dimensional vector produced by a model using 2 or more parameters. The parameters are used to assemble a linear combination of the base functions of the model space, they can be polynomial, periodic function or ...in the simplest case two parameters are used to produce a straight line. (linearity of the model space and linearity of function). There is no reference to any distribution or likelihood until here simply a minimum distance principle in the N-dimensional data space between observed vector $(v_0, v_1, v_2 \dots v_N)$ and the synthetic vector $(\hat{v}(t_0), \hat{v}(t_1), \hat{v}(t_2), \dots \hat{v}(t_N))$ It emerges that if the data vector can be assumed to be subjected to an Gaussian error with constant variance, the least squares principle satisfies also a maximum likelihood principle. And the vector distance normalized by the factor σ obeys a χ^2 probability distribution function, which can be used to evaluate the estimate the confidence interval. We define as above the error between the sampled speed and the speed value estimated with the linear regression:

$$e_i = v_i - \hat{v}(t_i) = v_i - \langle v \rangle_n + \hat{a}_n \left(t_i - \langle t \rangle_n\right) \tag{A48}$$

(see above A25). and squaring:

$$e_i^2 = (v_i - \langle v \rangle_n)^2 + 2(v_i - \langle v \rangle_n)(\hat{a}_n \left(t_i - \langle t \rangle_n\right)) + \hat{a}_n^2 \left(t_i - \langle t \rangle_n\right)^2 \tag{A49}$$

$$\begin{aligned} \frac{1}{N+1}\sum_{j=0}^{N} e_{n-j}^2 &= \frac{1}{N+1}\sum_{j=0}^{N} (v_{n-j} - \langle v \rangle_n)^2 \\ &+ \frac{2}{N+1}\sum_{j=0}^{N} (v_{n-j} - \langle v \rangle_n)(\hat{a}_n \left(t_{n-j} - \langle t \rangle_n\right)) \\ &+ \frac{1}{N+1}\sum_{j=0}^{N} \hat{a}_n^2 \left(t_{n-j} - \langle t \rangle_n\right)^2 \end{aligned} \tag{A50}$$

$$\begin{aligned} \frac{1}{N+1}\sum_{j=0}^{N} e_{n-j}^2 &= \langle (v - \langle v \rangle_n)^2 \rangle_n \\ &+ 2\hat{a}_n \langle (v - \langle v \rangle_n) \cdot (t - \langle t \rangle_n) \rangle_n \\ &+ \hat{a}_n^2 \langle (t - \langle t \rangle_n)^2 \rangle_n \end{aligned} \tag{A51}$$

the chi square indicates the variance of the error

$$\chi_{red}^2(e) = \frac{1}{N-1}\sum_{j=0}^{N} e_{n-j}^2 \tag{A52}$$

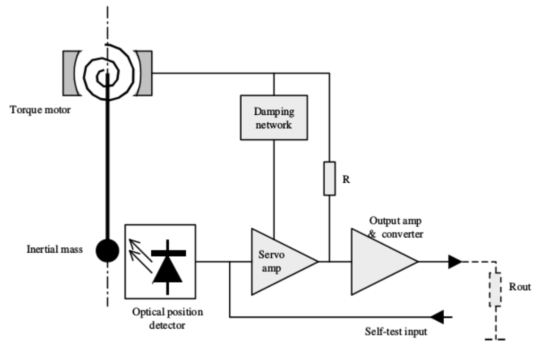


Figure A15. ***sketch of accelerometer***

A.1.6 measuring accelerations

The accelerometer is intended to measure the acceleration along the train path (in tangent direction to the train path). The device is calibrated in defined conditions when the train is in horizontal position and it is set up to have an equal elongation range for acceleration and for deceleration on a rectilinear route.

It should be noticed, that if the train is moving freely downhill the acceleration measured by this device will be null, according to equivalence principle. On the other side if the train stands still with applied brake on a track with gradient the device will measure a deceleration given by $g \cdot \sin\iota \approx g \cdot \iota$, which will be of the order of $0.x\ m/s^2$ for typical gradient values. This effect corresponds to the braking force applied to the train to hold it still.

This force is not producing work in the rest frame because there is no displacement and it is not producing work in any other reference because in the still stand the braking force compensates exactly the gradient force.

If the train is accelerating by effect of traction force the measured value shall correspond to the (averaged) acceleration due to the traction. In other words the measured acceleration is only the contribution due to applied traction or brake but not the contribution from the gradient. In order to know the tangential acceleration component due to gradient we should (at least in theory) measure the acceleration component normal to the rail $g \cdot \cos\iota \approx g \cdot \left(1 - \frac{1}{2}\iota^2\right)$.

The loss of weight would be only of the second order in the gradient and indicate a corresponding downhill acceleration or an uphill deceleration.

Appendix: References

[1] Jan Welvaarts and Baseliyos Jacob. *Requirements on openETCS API*, 11.05.2014 edition, May 2014. https://github.com/openETCS/requirements/blob/master/D2.7-Technical_Appendix/2014-05-13-Munich-Meeting/Bullit%20point%20openETCS%20requirements_20140511.pdf.

[2] Alstom Transport. *Appendix application layer*, v1.2 edition, 2014. https://github.com/openETCS/requirements/blob/master/D2.7-Technical_Appendix/OETCS_API%20Requirements_appendix_application_layer_v1.2.pdf.

[3] Alstom Transport. *Appendix Functional Data Dictionary*, v1.1 edition, 2014. https://github.com/openETCS/requirements/blob/master/D2.7-Technical_Appendix/OETCS_API%20Requirements_appendix_functional_data_dictionary_v1.1.pdf.

[4] Nicolas Boverie. *API Requirements for OpenETCS*. Alstom Transport, v1.4 edition, September 2014. https://github.com/openETCS/requirements/blob/master/D2.7-Technical_Appendix/OETCS_API%20Requirements_v1.4.pdf.

[5] ERA. *System Requirements Specification, SUBSET-026*, v3.3.0 edition, March 2012.

[6] Philip R. Bevington D. Keith Robinson. *Data Reduction and Error Analysis for Physical Sciences*, 3rd edition, 2003.

Appendix: Acronyms

API	application programming interface
DMI	driver-machine interface
DOT	Direction of Travel
EVC	European Vital Computer
MMU	Movement Measuring Unit
OB	on-board
OBU	on-board unit
SRS	system requirement specification
TCMS	Train Control Management System
TRU	Train Recording Unit
WS	Wheel Sensor