GPS Technical Support

Manual

0311376 Issue: Jan. 1999

Table of Contents

1. INTRODUCTION

1.1 FOREWORD

This manual aims at providing users with the necessary information to make the best possible use of DSNP receivers. In particular, it includes information on the various options available and a description of the functions of the new product line. Also included are technical data that may be helpful in the event of any problem you can quickly isolate and fix yourself.

Providing a detailed description of the GPS system is beyond the scope of this manual and unnecessary insofar as scores of publications on the subject have already been disseminated worldwide. For further explanations on the GPS system as such, refer to the existing literature.

Because the technology used and the high level of integration make it very difficult to isolate and replace any faulty component, no detailed description is provided of the different boards making up the new products. Instead, this manual contains descriptions of the functions and of communications between the various devices.

1.2 BBS No. and E-MAIL ADDRESS

As a result of our commitment to providing the best possible technical support, we have implemented communications means for the users to quickly get a reply to any request they make. These means consist of an E-mail address and two telephone numbers attached to a Bulletin Board System (BBS). You can send comments, request or messages, etc. at any time. The Customer Support team will make every effort to provide the fastest possible response.

E-Mail address : technical.dsnp@sercel.fr

BBS : 33 2 40 30 59 21.
33 2 40 30 59 22.

1.3 TECHNICAL SUPPORT

Naturally, in addition to these communication means, the Customer Support team is still at your disposal for any information you may need. Feel free to contact us:

DSNP

B.P.439 44474 CARQUEFOU CEDEX (FRANCE)

Tel : (33) 2 40 30 59 00

FAX : (33) 2 40 30 58 92

2. INTRODUCTION TO THE GPS SYSTEM

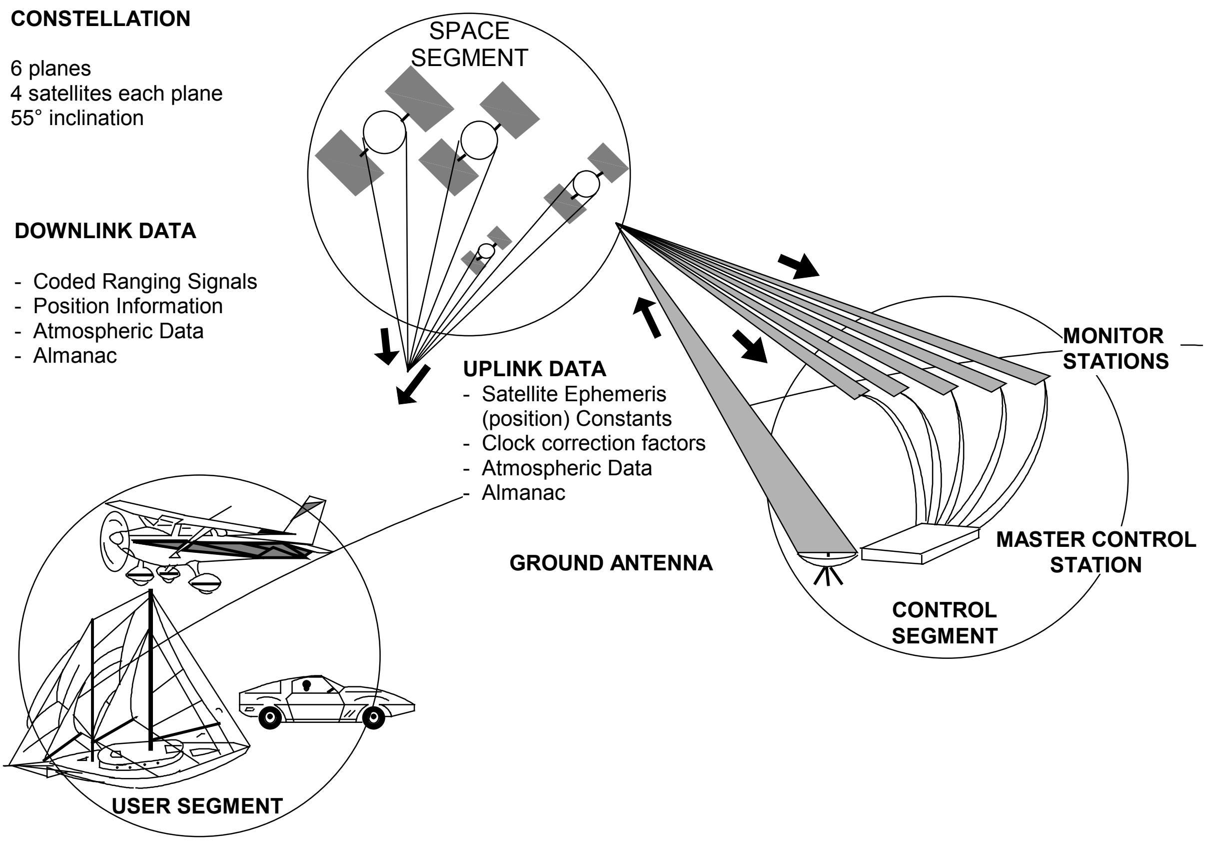
2.1 GPS CONSTELLATION

The GPS system (Global Positioning System) consists of three segments:

Space segment

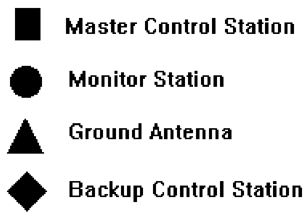
Control segment

User segment

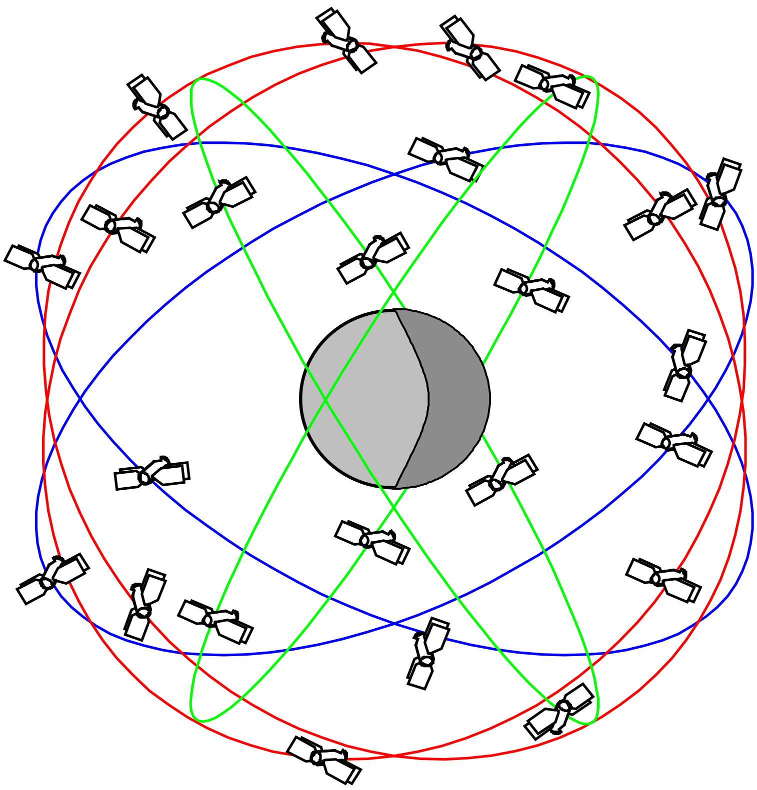


The major segments of the GPS system

The Control segment is made up of monitoring stations distributed along the equator. They are used to pick up the signals from the satellites and relay the data they convey to a master station located in Colorado Springs (USA). The data collected are processed, corrected, filtered and finally uploaded to the satellites that broadcast them through a navigation message (ephemerides, almanacs, clock corrections).



The Space segment consists of 24 satellites (often referred to as "SVs" which is an abbreviation for Space Vehicles) orbiting approximately 20200 km above the earth's surface, so that at least four satellites can be simultaneously in view, round the clock, anywhere on earth. The satellites are distributed over 6 orbit planes inclined 55° with respect to the equatorial plane. Each satellite completes an orbit once every 12 hours approximately. From any point on earth, a satellite remains in view for 5 hours (maximum) above the horizon.



The user segment is naturally that which means most to us. It is made up of all the marine, land or air-borne applications deciphering and using the signals received from the satellites. From a user's point of view, the user segment consists of a receiver capable of recording the GPS information so that it can be processed at a later date or a receiver computing a position in real time with an accuracy depending on the signals used.

2.2 SIGNALS

The signals transmitted by the satellites fall into two categories: signals used to control the system, and signals used for measurements within receivers (user segment).

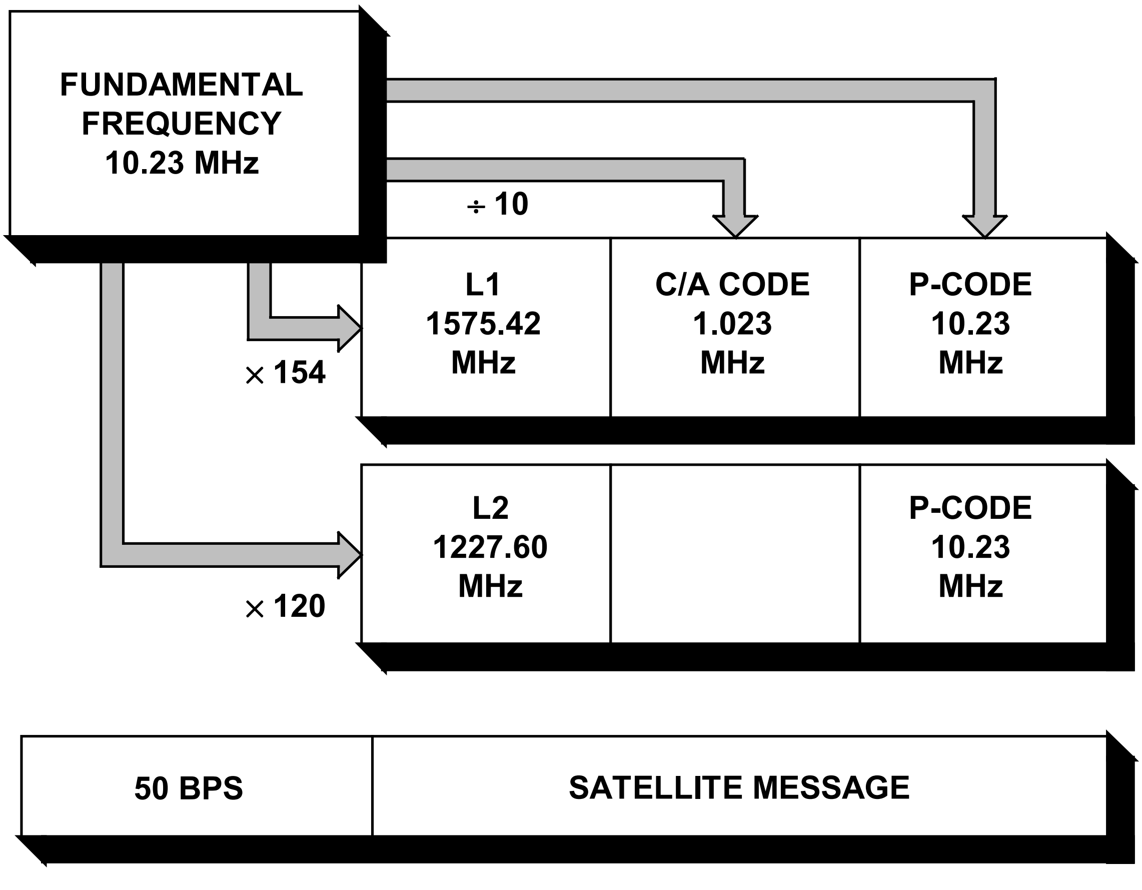
The first type of signal is transmitted in the S-band on the following frequencies:

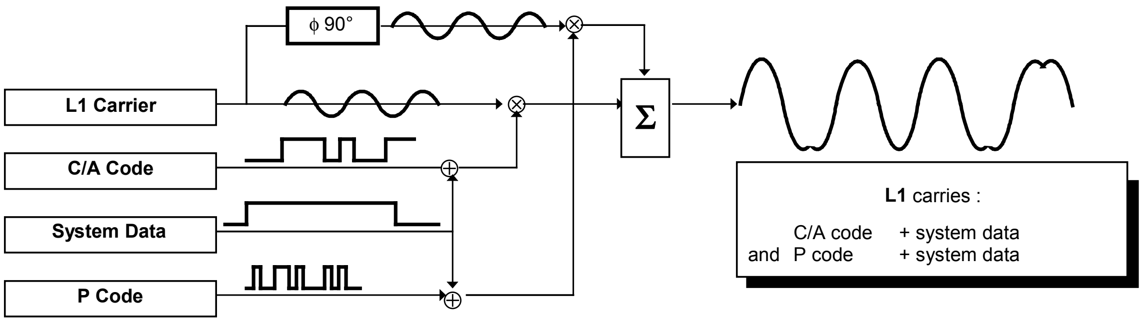
- 1 783.74 kHz for links from the control station to the satellites,
- 2 227.5 kHz for links from the satellites to the monitoring stations.

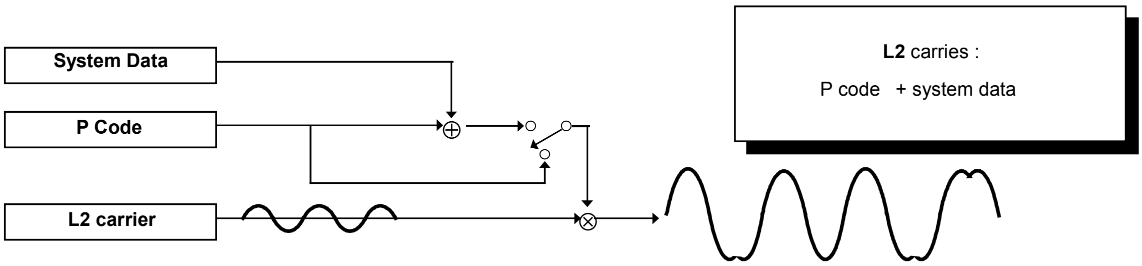
The second type of signal is for signals known as L1 and L2, transmitted in the L-band, on the following frequencies:

- L1: 1 575.42 kHz
- L2: 1 227.6 kHz

SATELLITE SIGNAL



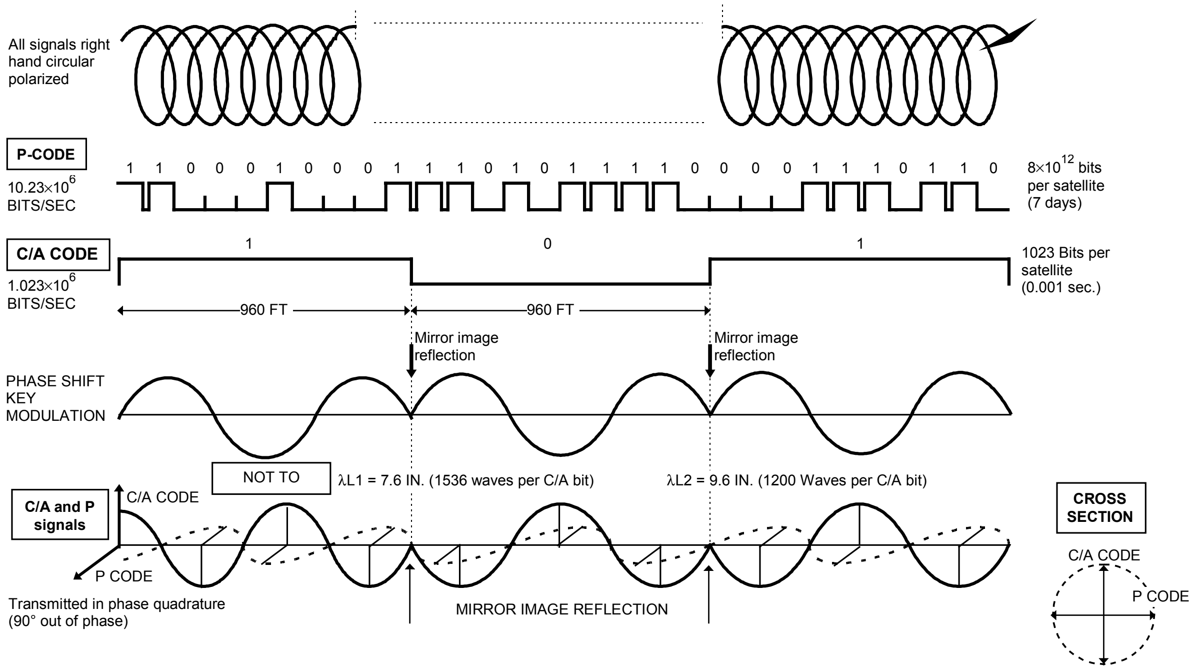

ϕ 90°
L1 Carrier
C/A Code
System Data
P Code
Σ
L1 carries :
C/A code + system data
and P code + system data


L2 carries :
P code + system data
System Data
P Code
L2 carrier

The two frequencies generated from a fundamental frequency of 10.23 MHz go through a modulator circuitry that enables them to convey the necessary information for determining the position.

Two types of information are contained in the modulated signal:

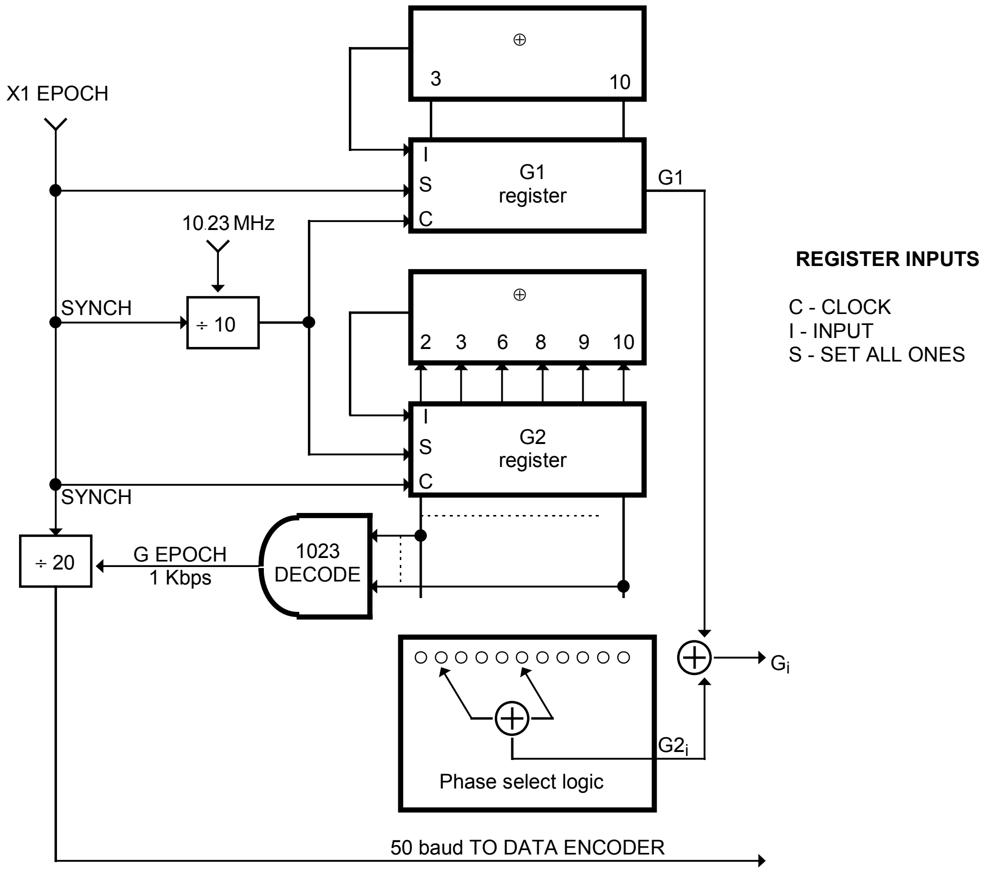
- Pseudo-Random sequences known as C/A code or P code,
- Navigation data.

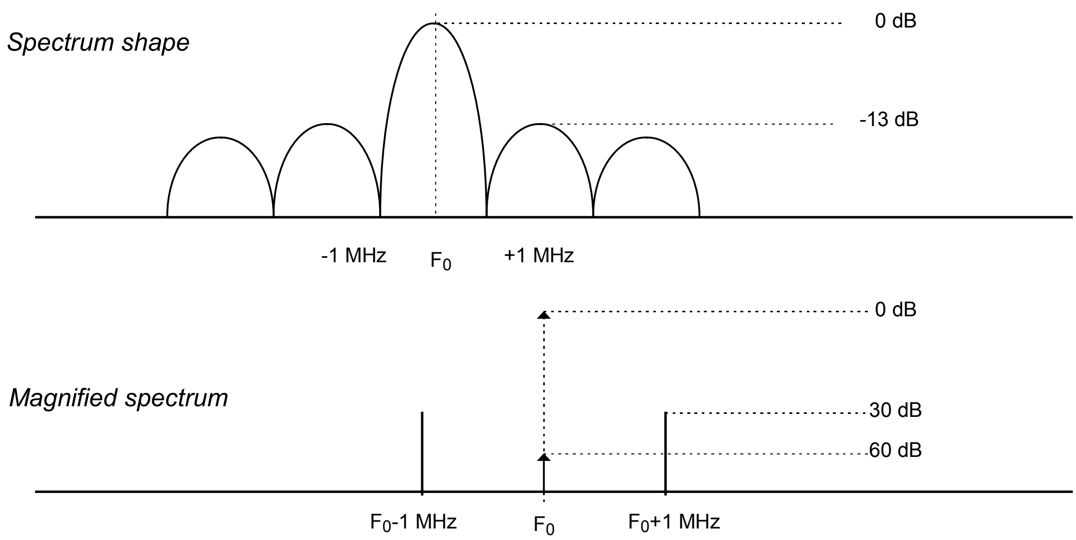


SIGNAL MODULATION TECHNIQUES

The Pseudo-Random sequences are generated by linear-feedback shift registers. The C/A (Coarse Acquisition) code sequence is clocked by a 1.023 MHz basic signal and is 1 ms long. It is available to any user. (This service is known as SPS or Standard Positioning Service).

The P code sequence is clocked by a 10.23 MHz basic signal and is 267 days long. In fact it is reset every week. In addition it is modulated by another code (W) to generate an encrypted code (Y) which denies access to unauthorized users. (This is known as PPS or Precise Positioning Service).


X1 EPOCH
10.23 MHz
SYNCH
÷ 10
3
10
I
S
C
G1
register
G1
2 3 6 8 9 10
G2
register
SYNCH
÷ 20
G EPOCH
1 Kbps
1023
DECODE
Gi
G2i
Phase select logic
50 baud TO DATA ENCODER
REGISTER INPUTS
C - CLOCK
I - INPUT
S - SET ALL ONES


Spectrum shape
0 dB
-13 dB
-1 MHz
F0
+1 MHz
0 dB
Magnified spectrum
30 dB
60 dB
F0-1 MHz
F0
F0+1 MHz

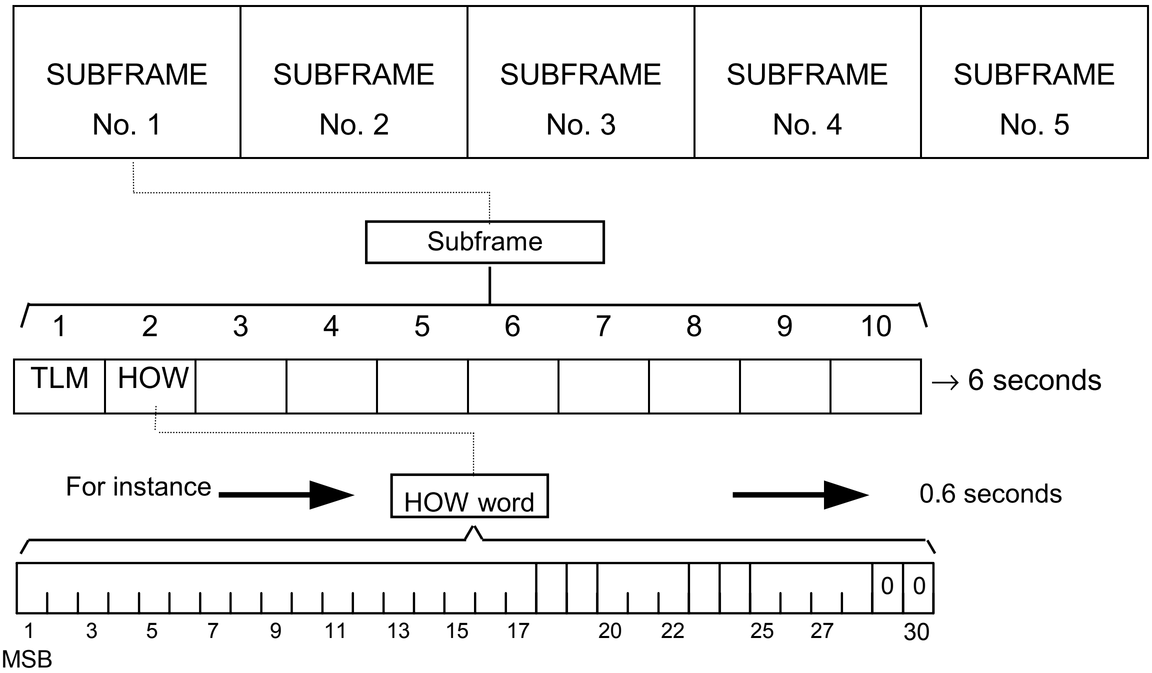
2.3 NAVIGATION MESSAGE

The Navigation Message contains the necessary information for the description of the constellation and for the position computation. The message includes orbital Keplerian parameters precisely defining the orbits of the satellites. It also includes parameters used to partially correct system errors (e. g. signal propagation errors, satellite clock errors, etc.).

The complete message is contained in a data frame that is 1500 bits long, with a total duration of 30 seconds (i. e. the data transmission clock rate is 50 bits/second). The 1500-bit frame is divided into five 300-bit subframes, each with a 6-second duration. Each subframe consists of 10 words of 30 bits each. Each word takes 0.6 second to transmit.

The content of subframes 4 and 5 changes on a page-roll basis: it changes on every frame and repeats every 25 frames. As a result, it takes at least 12 1/2 minutes to log the entire navigation message.

Below is a description of each subframe.



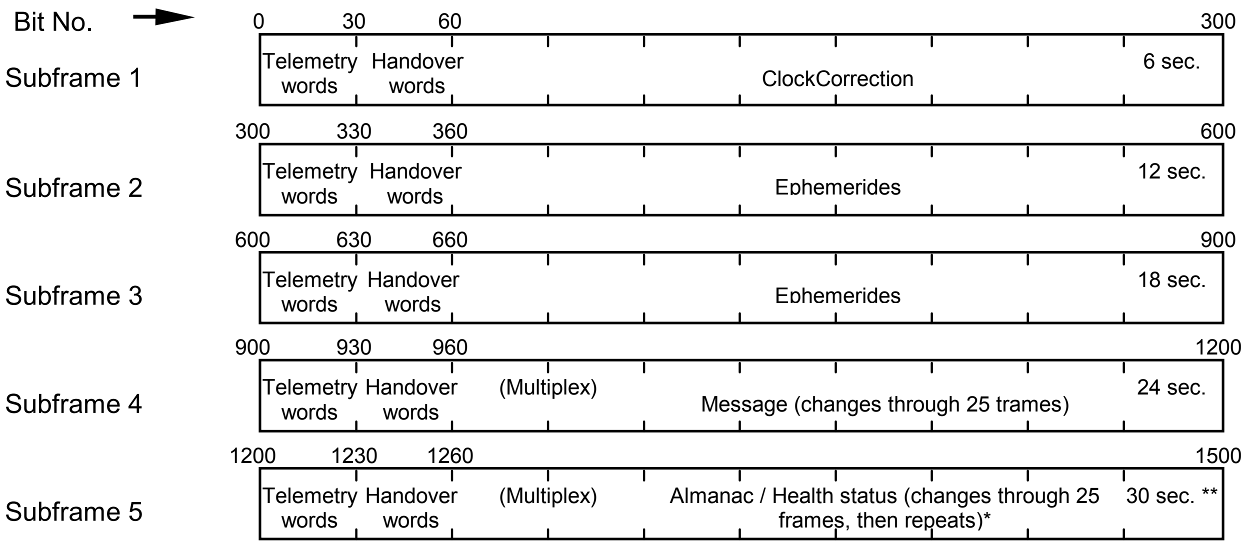
Subframe No. 1
- Satellite clock parameters
- GPS week
- Ionospheric correction param.
- URA

Subframe No. 2, Subframe No. 3 } Satellite Ephemerides

Subframe No. 4, Subframe No. 5 } Almanac

Navigation message structure

All subframes start with the TLM word, followed by the HOW word.



* *Format of frame twenty-five changes.*

** *12.5 minutes before the entire message repeats.*

PARAMETERS	UNITS	COMMENTS
Code(s) on L2 channel	dimensionless	Indicates whether L2 is modulated by P or C/A
Week number	week	Value contained in z-count
Data Flag for L2 P-code	dimensionless	Indicates whether the P-code is modulated by the navigation data on L2 channel
Satellite accuracy	dimensionless	Gives the predicted User Range Accuracy to the unauthorized user
Satellite health	dimensionless	Summary of the health of the navigation data and of the signals (additional data are given in Data Blocks 4 and 5)
T_{GD}	seconds	L1 - L2 correction term for the benefit of single-frequency users
AODC	seconds	Age Of Data, Clock: indicates the issue number of the clock correction data
t_{0c}	seconds	Reference time of satellite clock data. Max. value 604784
af_2	sec/sec^2	Polynomial coefficient used to compute satellite clock correction data
af_1	sec/sec	Polynomial coefficient used to compute satellite clock correction data
af_0	seconds	Polynomial coefficient used to compute satellite clock correction data

DATA BLOCK-1 PARAMETERS

Parameter	Unit	Comments
AODE	seconds	Age of Data, Ephemeris Appears in both subframes
C_{rs}	metres	Amplitude of the sine harmonic correction term to the orbit radius
Δ_n	semicircles/s	Mean motion difference from computed value
M_o	semicircles	Mean anomaly at reference time
C_{uc}	radians	Amplitude of the cosine harmonic correction term to the argument of latitude
e	dimensionless	Eccentricity (= 0.03)
C_{us}	radians	Amplitude of the sine harmonic correction term to the argument of latitude
$(A)^{1/2}$	metres	root of the semi-major axis
t_{oe}	seconds	Ephemeris reference time Maxvalue = 604784
C_{ic}	radians	Amplitude of the cosine harmonic correction term to the angle of inclination
Ω_o	semicircles	Right ascension at reference time
C_{is}	radians	Amplitude of the sine harmonic correction term to the angle of inclination
i_o	semicircles	Inclination angle at reference time
C_{rc}	metres	Amplitude of the cosine harmonic correction term to the orbit radius
ω	semicircles	Argument of perigee
OMEGADOT	semicircles/s	Rate of right ascension
IDOT	semicircles/s	Rate of inclination angle

EPHEMERIS PARAMETERS (Data blocks 2 and 3)

DATA BLOCKS 4 AND 5

Parameter	Unit	Comment
e	dimensionless	Eccentricity
t_{oa}	seconds	Almanac reference time
δ_i	semicircles	Inclination correction (truncated value that can also be computed) relative to i_o = 0 3
OMEGADOT	semicircles/s	Rate of right ascension
$(A)^{1/2}$	$metres^{1/2}$	Square root of semi-major axis
Ω	semicircles	Right ascension at reference time
ω	semicircles	Argument of perigee
M_o	semicircles	Mean anomaly at reference time
af_o	seconds	polynomial coefficient used to compute satellite clock correction data (less accurate than that of data block 1)
af_1	sec/sec	polynomial coefficient used to compute satellite clock correction data (less accurate than that of data block 1)

Almanac parameters

Parameter	Unit	Comment
A_o	seconds	Polynomial coefficient used for the computation
A_1	sec/sec	Polynomial coefficient used for the computation
Δ_{TLS}	seconds	Delta time due to Leap Seconds
t_{ot}	seconds ~	Reference time of UTC data. Max. value: 602112
WN_t	weeks	UTC reference week number
WN_{LSF}	weeks	Week number of future leap second effectivity
DN	days	Day number of future Leap Second effectivity (max. = 7)
Δt_{LSF}	seconds	Delta time due to future Leap Seconds

GPS/UTC-time relationship parameters

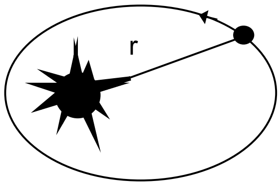
Parameter	Unit	Comment
α_0, β_0	seconds	Coefficients to be used in the single-frequency user correction model
α_1, β_1	sec/semicircle	Coefficients to be used in the single-frequency user correction model
α_2, β_2	sec/semicircle^2	Coefficients to be used in the single-frequency user correction model
α_3, β_3	sec/semicircle^3	Coefficients to be used in the single-frequency user correction model

Ionospheric Correction Parameters

CONTENT OF SUBFRAMES 4 AND 5 AS A FUNCTION OF PAGE No.

Subframe	Page No.	Content
4	1, 6, 11, 16 and 21	Reserved
	2, 3, 4, 5, 7, 8, 9, 10	Almanac data for satellites 25 through 32 respectively
	12, 19, 20, 22, 23, 24	Reserved
	13, 14, 15	Spares
	17	Special messages
	18	UTC and ionospheric data
	25	Antispoof configurations for 32 satellites plus satellite health data for satellites 25 through 32

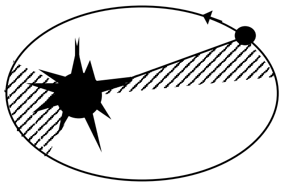
KEPLER'S FIRST LAW



Each planet moves along an elliptical orbit with the sun at one focus.

$$r = \frac{a(1-e^2)}{1+e\cos\theta}$$

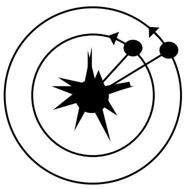
KEPLER'S SECOND LAW



The line joining the sun and the planet sweeps out equal areas in equal times.

$$h = r^2\frac{d\theta}{dt} = \text{constant}$$

KEPLER'S THRID LAW

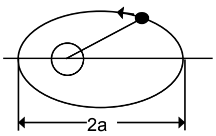


The squares of the periods of the planets are proportional to the cubes of their mean distances from the sun.

$$T = 2\pi\sqrt{\frac{a^3}{\mu}}$$ or $n^2a^3 = \mu^5$

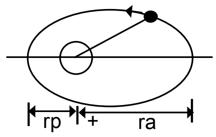
KEPLER'S THREE LAWS OF ORBITAL MOTION

a : SEMI MAJOR AXIS



The half-length of the ellipse

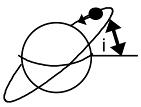
e : ORBITAL ECCENTRICITY



The "oblateness" of the orbit

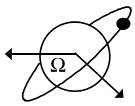
$$e = \frac{ra - rp}{ra + rp}$$

i : ORBITAL INCLINATION



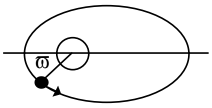
The angle between the equatorial plane and the orbital plane

Ω : ASCENDING NODE



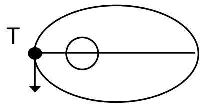
The longitude of the ascending node (equatorial crossing)

ϖ : ARGUMENT OF PERIGEE



The longitude of the perigee

T : TIME OF PERIGEE PASSAGE



The time of passage of the point of closest approach

The GPS ephemeris coordinates include 4 extra elements for clock errors and about 10 others for orbital perturbations.

THE SIX TRADITIONAL KEPLERIAN ORBITAL ELEMENTS a, e, i, Ω, ϖ, T

2.4 SIGNAL PROCESSING

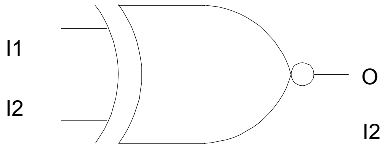
2.4.1 Pseudo-Range measurement

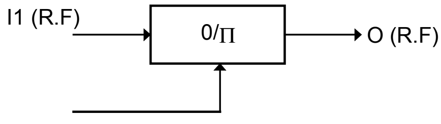
The GPS measurement is derived from the time it takes for the signal to travel from a satellite to the user receiver. As the wave propagation speed is known, the distance is computed using the following expression:

Distance = Wave propagation speed × Travel time.

The signal travel time is determined through a special operation –correlation– that pseudo-random sequences allow us to perform. The receiver performs the correlation operation by generating a pseudo-random code identical to that transmitted by a satellite and compares it to the code received to determine its time shift. The signal resulting from the comparison looks like the one shown below.

The time shift between the two pseudo-random codes is a measure of the time elapsed since the code was transmitted by the satellite.

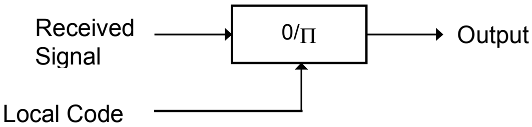




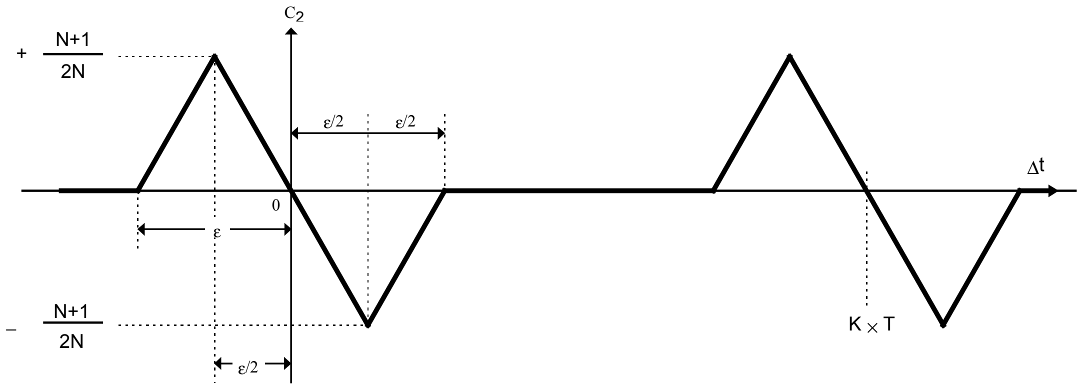
I_1	I_2	O
0	1	0
0	0	1
1	1	1
1	0	0

I_1 (phase)	I_2 (level)	O (phase)
0	1	π
0	0	0
π	1	0
π	0	π

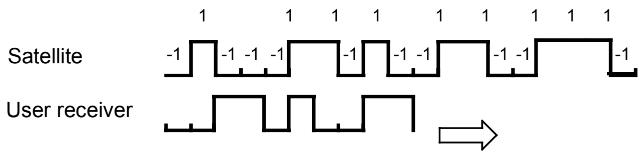
The received signal, modulated by a pseudo-random code, is fed to a phase inverter controlled by a local pseudo-random code identical to the incoming one.



That correlation function can be represented as shown below



The autocorrelation function is illustrated below.



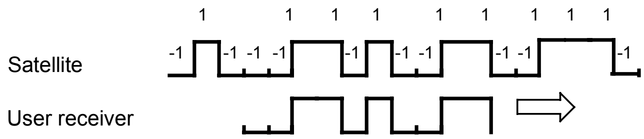
Autocorrelation function:

$$\frac{1}{N}\int_0^T X(t)*X(t-\tau)\,dt = \frac{1}{N}\sum_{i=1}^{N} X_i * X_{i-\tau}$$

In the above case :

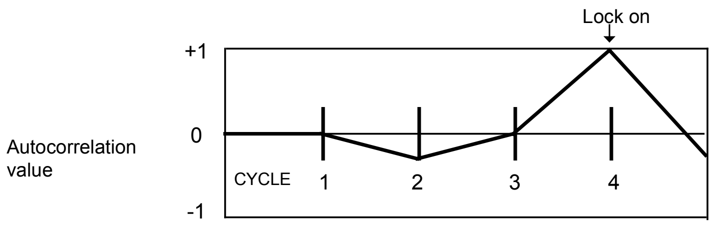
$$\frac{1}{10}\sum_{I=1}^{10} X_I * X_{I-3} = \frac{1}{10}\,[(-1)(-1)+(1)(-1)+(-1)(1)+(-1)(1)+(-1)(-1)+(1)(1)+(1)(-1)+(-1)(-1)+(1)(1)+(-1)(1)]$$

$$= \frac{1}{10}\,[+1-1-1-1+1+1-1+1+1-1]=0$$



$$\frac{1}{10}\sum_{i=1}^{10} X_i * X_{i-3} = \frac{1}{10}\,[(-1)(-1)+(-1)(-1)+(1)(1)+(1)(1)+(-1)(-1)+(1)(1)+(-1)(-1)+(-1)(-1)+(1)(1)+(1)(1)]$$

$$= \frac{1}{10}\,[1+1+1+1+1+1+1+1+1-1]=1$$



2.4.2 Doppler measurement

The relative speed of a GPS satellite seen from the earth surface may be as high as ± 700 m/s as it rises above the horizon. Because of the resulting Doppler effect on the frequency, the signal should be observed over a 20-kHz band. The theory of radiocommunications teaches us that the width of the reception band of a receiver determines the sensitivity of the receiver. Assuming we can anticipate the relative position of a satellite from a given location on earth, at a given time, we can easily compute the expected Doppler for this satellite. This allows the receiver bandwidth to be narrower, which increases the receiver sensitivity.

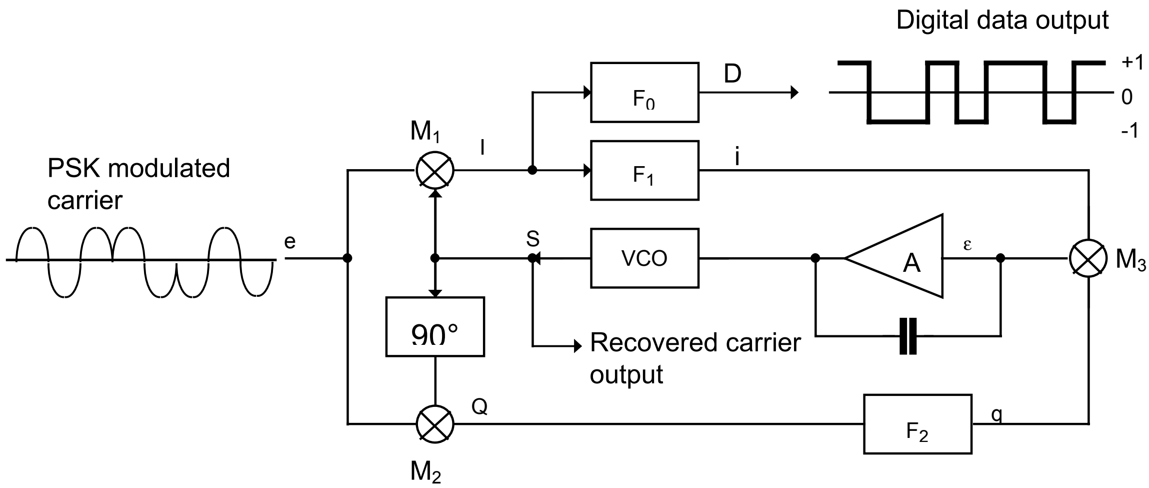
2.4.3 Phase measurement

Measuring the phase of the carrier of the GPS signal provides an efficient means of fine-tuning the pseudo-range measurement. The phase measurement is achieved through a circuitry known as Costas loop, with two functions:

1 - Locking the phase of the local master oscillator to that of the input signal, regardless of the modulation,

2 - Decoding the navigation message.

A description of the theory of operation of a Costas loop is provided below.



Input signal: $e = \sqrt{2\,\mathrm{A}}\cos(\omega_0 t + \theta p)$

The VCO generates a signal: $S = \sqrt{2}\cos(\omega_0 t + \theta r)$

At M1 multiplier output we have:

$I = e \times S = A\cos(2\,\omega ot+\theta+\theta r) + A\cos(\theta p-\theta r)$ where $\theta p-\theta r = \varphi$ stands for the system phase error.

Now, like in any phase-lock loop $\varphi = 0$ hence $\cos(\theta p-\theta r) = 1$

The 2 ωot term is eliminated by F1 and F0. At output D is a binary data waveform (A) and

$i = A\cos(\theta p-\theta r) = \cos\varphi$.

Likewise, in the quadrature circuitry (in the lower part of the diagram)

$Q = A\sin(\theta p-\theta r)= A\sin\varphi$; and $\in = A^2 \sin\varphi \cos\varphi = A^2/2 \sin 2\varphi$

2.4.4 Code smoothing by the carrier phase

We have seen earlier how the distance measurement is performed, derived from the pseudo-random codes (through correlation of the signals received and replicas generated by the receiver). The techniques in use today allow measurements to be made with a resolution corresponding to 1/100 the signal to be measured. Considering the characteristics of the GPS signal, This leads to the following figures:

$$\lambda = C/F$$

where

λ = signal wave length (metres)

C = wave propagation speed (approximately the speed of light)

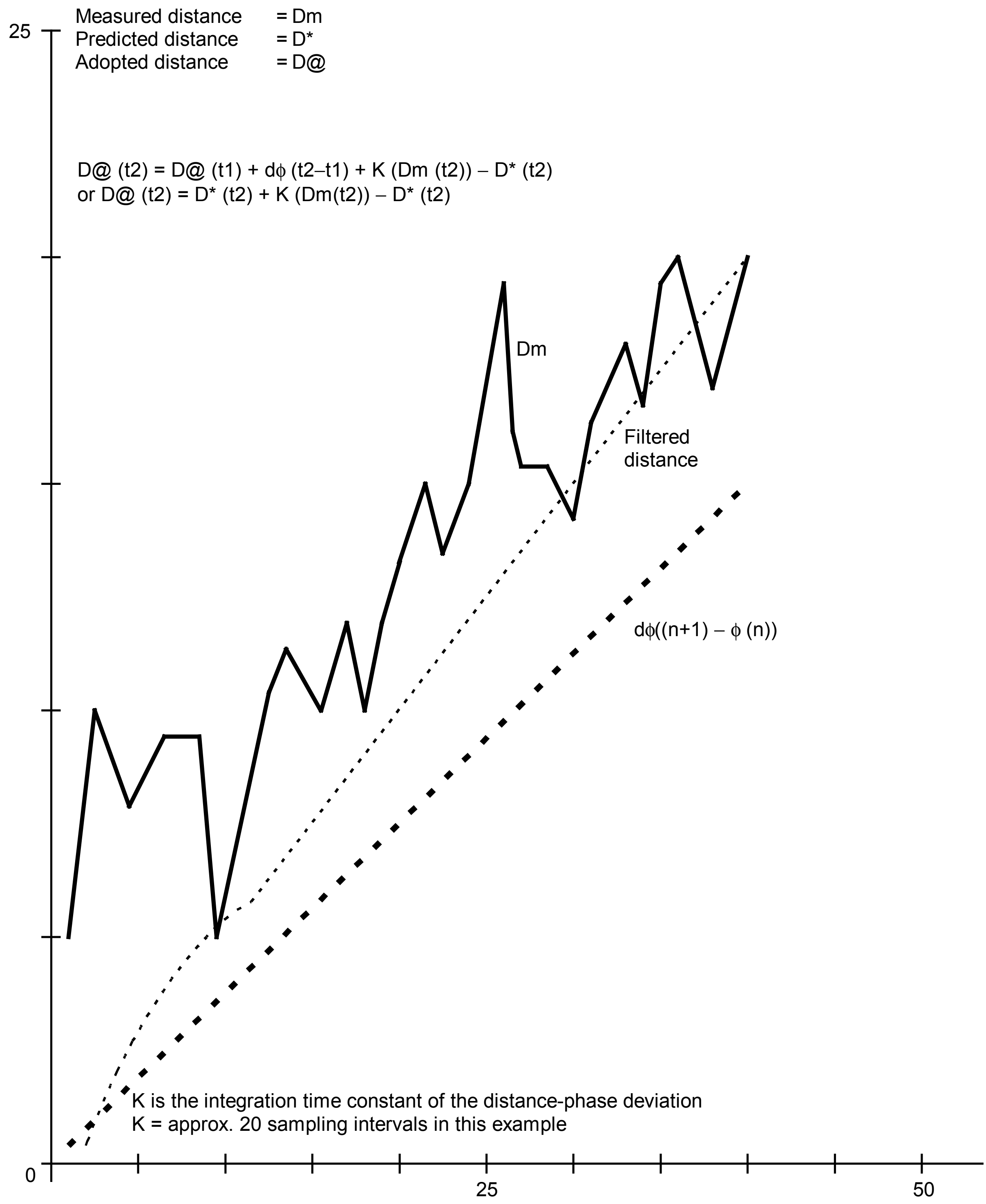
F = signal frequency (Hz)

For the C/A code (1 KHz) : λ = 300 Km

For L1 (1.57542 GHz) : λ = 19 cm.

For L2 (1.22760 GHz) : λ = 24 cm.

The resolution is at the metre level for the C/A code, meaning that a noise of several metres is to be expected, whereas the resolution for the L1 or L2 phase measurement is at the millimetre level. As a result, measuring the phase of the carrier naturally seems an efficient means of decreasing the noise on the code measurement. This technique is very helpful in GPS receivers.


Measured distance = Dm
Predicted distance = D*
Adopted distance = D@
D@ (t2) = D@ (t1) + dϕ (t2–t1) + K (Dm (t2)) – D* (t2)
or D@ (t2) = D* (t2) + K (Dm(t2)) – D* (t2)
Dm
Filtered distance
dϕ((n+1) – ϕ (n))
K is the integration time constant of the distance-phase deviation
K = approx. 20 sampling intervals in this example
25
0
25
50

2

2.5 POSITION DETERMINATION

The position is determined from the pseudo-range measurements. The pseudo-range can be expressed as:

Rij = ρij + C(dtj – dti) + Δρiono,j + Δρ tropo,j

$$\rho_{ij} = \sqrt{(Xj - Xi)^2 + (Yj - Yi)^2 + (Zj - Zi)^2}$$

(Xi ; Yi ; Zi) are the unknown coordinates of the point.

(Xj ; Yj ; Zj) are the known coordinates of the satellites, derived from the navigation message.

Δρ iono,j ; Δρ tropo,j are respectively the ionospheric and tropospheric errors corrected according to the models.

dtj stands for the satellite clock error derived from the parameters contained in the navigation message.

dti stands for the receiver clock error (unknown).

As there are four unknowns, (Xi ; Yi ; Zi ; dti), we need four equations to compute a three-dimension position solution.

As the equation system is not linear, the processor in the receiver computes a least-square solution.

Dn = PRn × C – Rn-1 – H n-1×C

where

Dn	: innovation at t.
PRn	: pseudo-range measurement at t.
C	: speed (m/s)
Rn-1	: satellite-to-receiver distance (estimated at t-1)
H n-1	: receiver clock error as compared to the GPS time, at t-1.

Matrix representation

$$\begin{bmatrix} D_1 \\ D_2 \\ . \\ D_j \end{bmatrix} = \begin{bmatrix} -\cos EV_1 \cos AZ_1 & -\cos EV_1 \sin AZ_1 & -\sin EV_1 & -1 \\ -\cos EV_2 \cos AZ_2 & -\cos EV_2 \sin AZ_2 & -\sin EV_2 & -1 \\ . \quad . \quad . & . \quad . & . & . \\ -\cos EV_j \cos AZ_1 & -\cos EV_j \sin AZ_j & -\sin EV_j & -1 \end{bmatrix} \times \begin{bmatrix} \Delta X \\ \Delta Y \\ \Delta Z \\ \Delta Ck \times C \end{bmatrix}$$

Matrix A , called transfer matrix

The solution can be expressed as:

$$\begin{bmatrix} \Delta X \\ \Delta Y \\ \Delta Z \\ \Delta Ck \times C \end{bmatrix} = (A^T A)^{-1} A^T [D]$$

ΔX = East/West updating relative to estimated preceding measurement

ΔY = North/South updating relative to estimated preceding measurement

ΔZ = Altitude updating relative to estimated preceding measurement

ΔCk = Receiver clock updating relative to estimated preceding measurement

The above expressions show that the measurement accuracy depends on the geometry of the constellation. From the variance/covariance matrix, we can compute a coefficient, known as DOP (Dilution Of Precision), reflecting the precision of the measurement.

With the variance/covariance matrix expressed as follows:

$$Q_X = \begin{bmatrix} \sigma^2{}_X & \sigma_{XY} & \sigma_{XZ} & \sigma_{Xt} \\ \sigma_{YX} & \sigma^2{}_Y & \sigma_{YZ} & \sigma_{Yt} \\ \sigma_{ZX} & \sigma_{ZY} & \sigma^2{}_Z & \sigma_{Zt} \\ \sigma_{tX} & \sigma_{tY} & \sigma_{tZ} & \sigma^2{}_t \end{bmatrix}$$

the geometric coefficients are computed as follows:

$$VDOP = \sigma_Z / \sigma_0$$

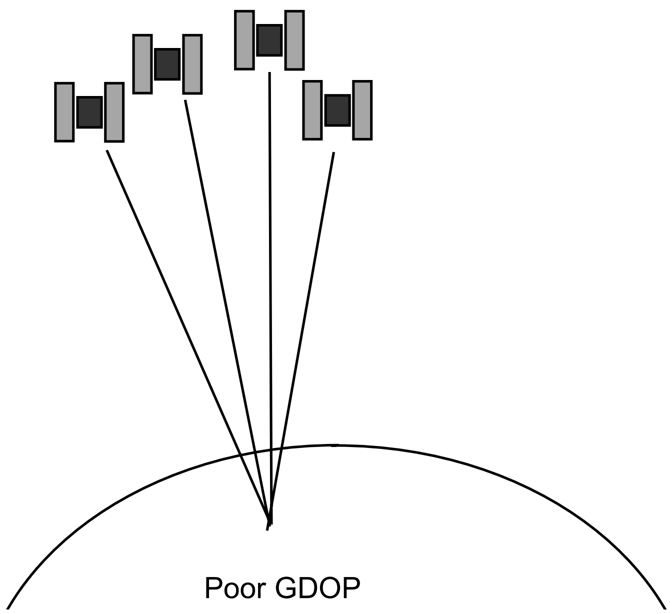
$$HDOP = 1/\sigma_0 \sqrt{\sigma_X^2 + \sigma_Y^2}$$

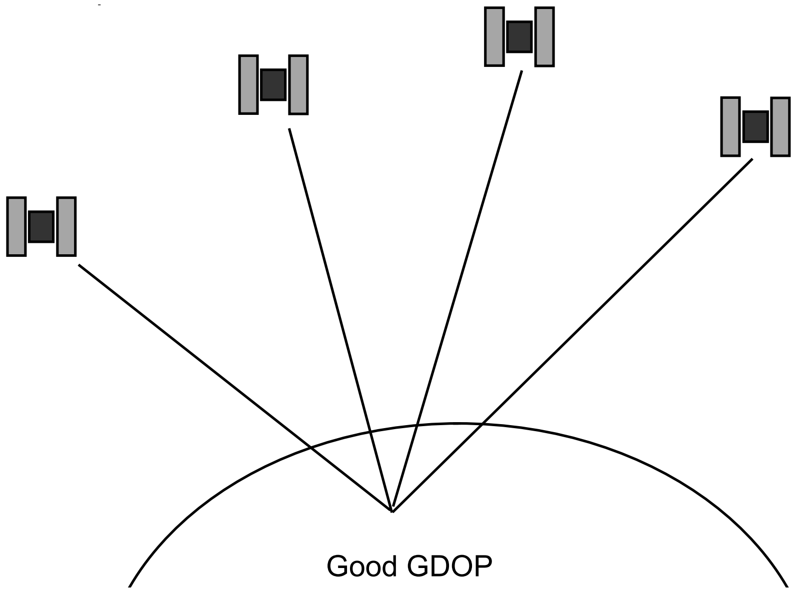
$$PDOP = 1/\sigma_0 \sqrt{\sigma_X^2 + \sigma_Y^2 + \sigma_Z^2}$$

$$TDOP = \sigma_t / \sigma_0$$

$$GDOP = 1/\sigma_0 \sqrt{\sigma_X^2 + \sigma_Y^2 + \sigma_Z^2 + \sigma_t^2}$$

σ_o stands for the standard deviation of the pseudo-range measurement.


Poor GDOP


Good GDOP

2.6 ERRORS

Measurement errors fall into four major categories:

Errors attributable to the satellites,

Errors attributable to wave propagation,

Errors attributable to the receiver itself,

Intentional degradation of the signal (Selective Availability).

Satellite errors

Some of the errors originating from the satellites are attributable to the clocks used to generate the signals. Others are due to the perturbations affecting the orbits of the satellites (effect of the gravity of the earth and moon and sun upon a satellite's orbit).

Models of those errors are generated by the monitoring stations (control segment) that determine the corrections to be broadcasted via the navigation message. However, those errors will not be totally eliminated.

Propagation errors

Ionospheric refraction

The ionospheric layer is the atmospheric layer ranging from 50 km to several hundred km overhead. This is a dispersive medium through which radio frequency waves are accelerated or delayed depending on the signal frequency. The delay on the measurement may range from 0 to 50 m. It can partly be eliminated by combining the measurements performed on the two frequencies transmitted by each satellite.

Troposheric refraction

The troposphere is made up of the low atmospheric layers. Its thickness depends on where you are on the earth surface. This is a non-dispersive medium, causing refraction manifested by a delay on the measured signal regardless of the signal frequency. The delay depends on the temperature, humidity, pressure and on the satellite elevation.

Multipath effect

The environment near the receiving antenna or any nearby obstructions (buildings, vegetation, etc.) may cause reflections or attenuate the signal, causing errors on measurements. Those errors vary over time and cannot be represented by any model. They mostly affect stationary receivers. The major ways or reducing or precluding multipath effects are the following:

- Installing the GPS receiving antenna in an open area, clear of any obstructions all round,

- Choosing the appropriate antenna, equipped with a ground plane if required,

- Choosing the appropriate minimum elevation angle required of the satellites to be used. (Multipath is more likely on low-elevation satellites).

Receiver errors

Receiver errors are directly connected with the quality of the receiver used, depending on:

- the noise figure (sensitivity, measurement noise),

- the measurement resolution,

- computation algorithms, etc.

2.7 DIFFERENTIAL GPS

When used straight away the GPS may not meet the accuracy requirements of some applications. It should be borne in mind that because of the degradation, whether or not intentional, inherent in the system the accuracy achieved with the C/A code is only ± 100 metres, 95% of time. The measurements relying on the P code achieve an accuracy of about ten metres, unfortunately the P code is not directly available to any user. The differential mode, a technique well known and widely used in the radionavigation area, allows the accuracy on the position computation to be substantially enhanced. The differential technique can be applied to the code measurements or to the phase measurements.

2.7.1 Pseudo-range corrections

A stationary receiver, referred to as 'reference', with known coordinates is used to compute the times of transmission of the signals from the satellites and to determine the times of arrival of those signals. Therefore the reference receiver is capable of determining corrections for each satellite in view, to compensate for the errors mentioned above. The mobile receiver applies the corrections to its own measurements, which enables it to compute a more accurate position.

2.7.2 Phase measurement

As seen earlier, the phase measurement is less noisy than the code measurement (millimetric resolution for L1 or L2 phase, as opposed to metric resolution for the C/A code or decimetric for the P code).

The receiver compares the phase of the received signal with that of its own signal and determines a phase shift lying between 0 and 1 phase cycle (i. e. from 0 to 19 cm for the L1 signal and from 0 to 24 cm for L2). This is only a measure of the fractional part of the phase. The total phase measurement can be expressed by the following equation:

$$\phi i(tR) - \phi J(tE) = \Delta\varphi(tR) + N\ i\ J\ (tR)$$

where $\phi i(tR)$: Phase of receiver signal at time of reception.

$\phi J(tE)$: Phase of satellite signal at time of transmission.

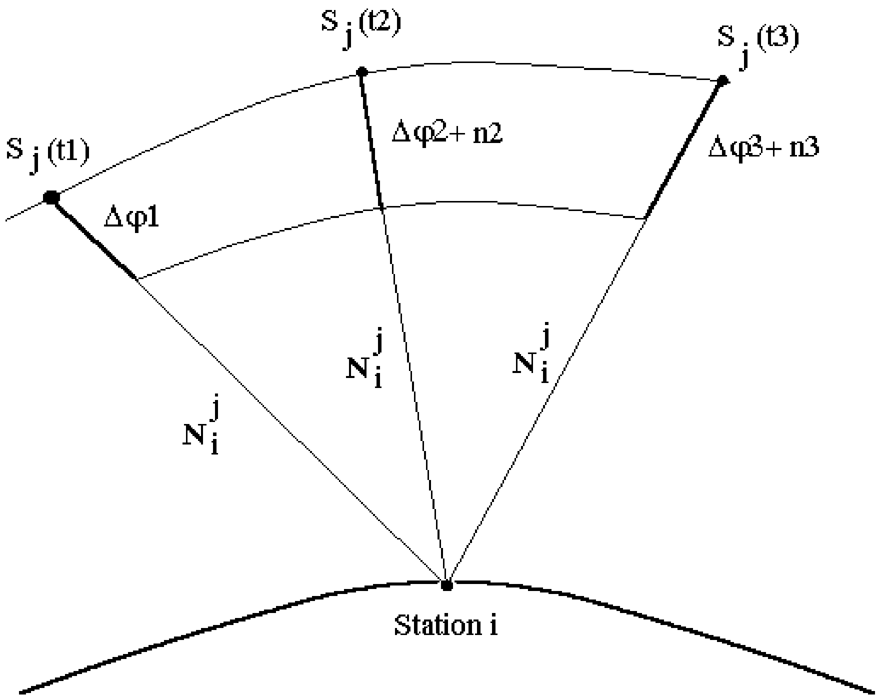
$\Delta\varphi(tR)$: Phase shift at time of signal reception.

$N\ i\ J\ (tR)$: Number of complete phase cycles, unknown.

In addition, the receiver also counts the number of whole phase cycles of the received signal completed since the first measurement was taken. Assuming t1 denotes the time of the first measurement, unless the signal is interrupted the subsequent measurements can be expressed by:

$$N\ i\ J\ (tR) = N\ i\ J\ (t1) + n(tR)$$

where *n(tR)* stands for the number of whole cycles since initialization



For a given satellite, the unknown number of whole cycles remains unchanged so long as the signal is not interrupted. The measurement performed on the code does not allow us to determine the unknown integer.

Taking account of the clock errors of the receiver and satellite, the global measurement of the distance is expressed by:

$$\phi ij = f\,(dtj - dti\,) + (f/c)\,\rho ij - Nij$$

where dtj : satellite clock error

dti : receiver clock error

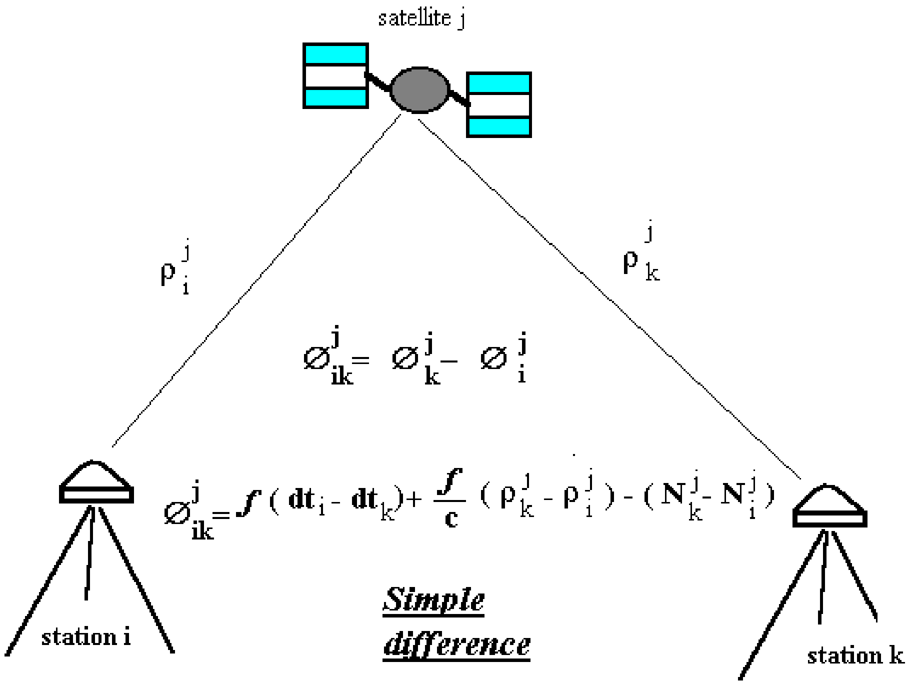
(f/c) ρij : measurement of satellit-to-receiver distance expressed in phase cycles

Nij : unknown number of whole cycles

The above expression is not directly used in the receiver. Instead, in order to get rid of the unknown terms, the following differences are computed:

Single difference

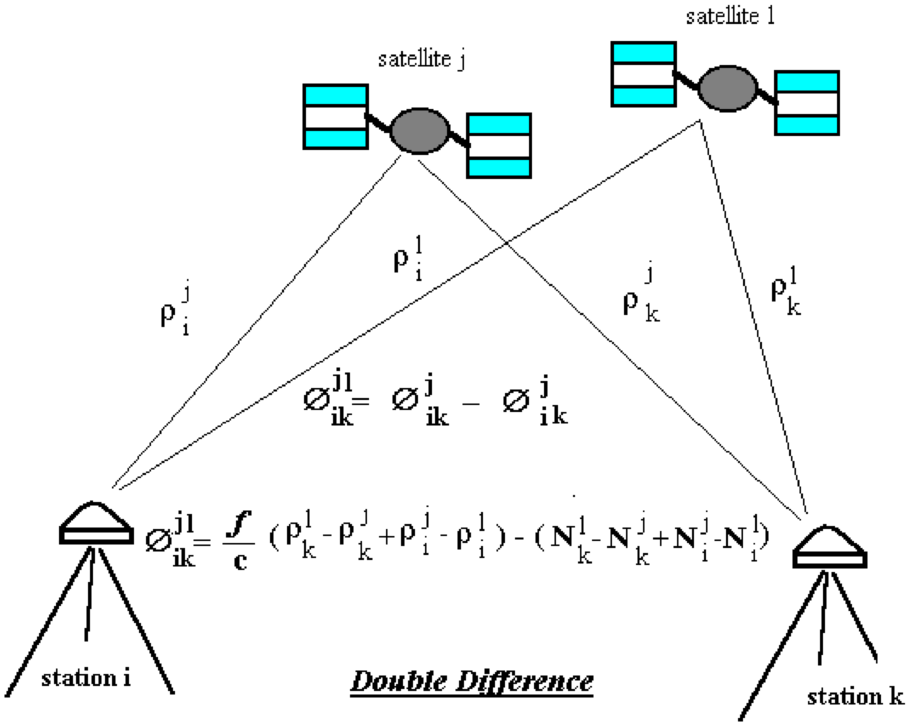
A given satellite is observed from two distinct locations simultaneously. The difference of the observations is expressed as:



This shows that the single difference causes the satellite clock error to vanish.

Double difference

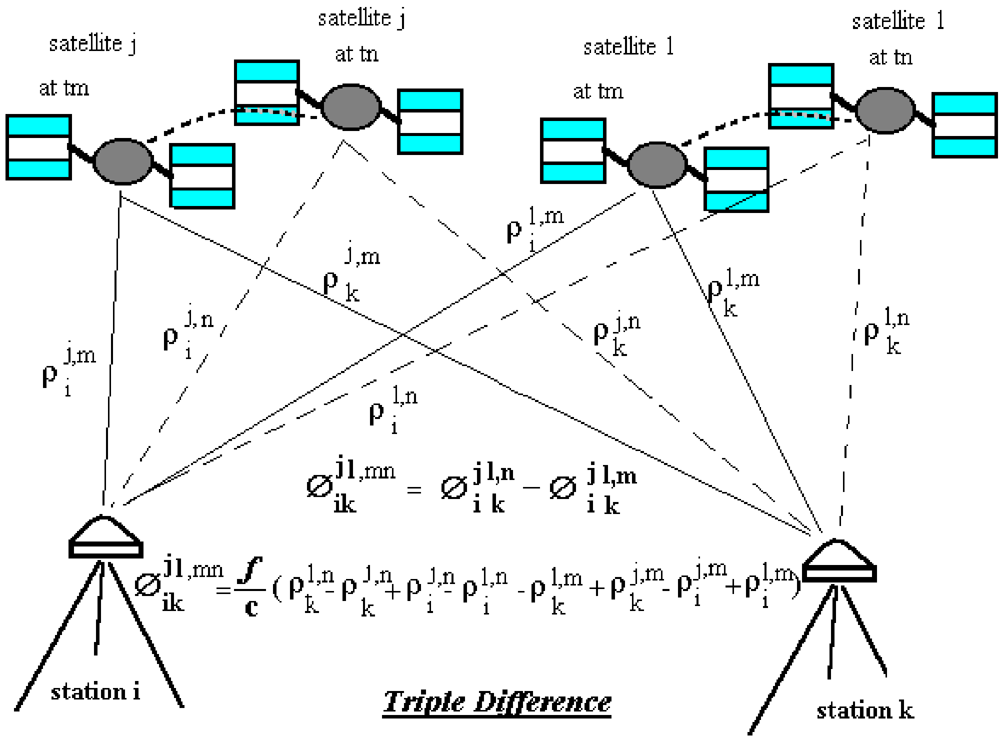
Two satellites are observed from two distinct locations simultaneously. The double difference of the observations is expressed as:



The double difference causes the receiver clock errors to vanish.

Triple difference

Two satellites are observed from two distinct locations simultaneously over two separate time spans. The triple difference of the observations is expressed as:



This shows that all ambiguous terms have vanished. The unknown integer is constant over time so long as the observation is uninterrupted. This method is used to detect any phase cycle slips in measurements.

2.7.3 Postprocessing

For postprocessing, the received data is recorded by both a reference station and the mobile receiver, whether on an internal or on an external record media. The data is processed at a later date through special software that returns the trackline of the mobile receiver or the coordinates of the surveyed point. The postprocessing software packages currently (Nov 1997) available from DSNP are GPSWINRS and 3SPACK.

GPSWINRS only processes the data from the L1 signal with a Rapid Static computation method. 3SPACK is capable of computing single- or dual-frequency data either in Rapid Static mode, for a surveyed point, or in Kinematic mode for a trajectory.

2.7.4 Real Time

In some applications, the position needs to be computed in real time (navigation, guidance, etc.). Besides, you may need to know whether or not results are consistent immediately rather than wait until postprocessing is completed. This requires that the system be capable of achieving high-accuracy instantaneous positioning while supplying a quality control figure.

The Differential GPS system relies on two parts: a reference station that broadcasts the necessary information over the area of interest, and a receiver capable of computing the position in real time and assessing the quality of that position.

2.7.5 Reference station

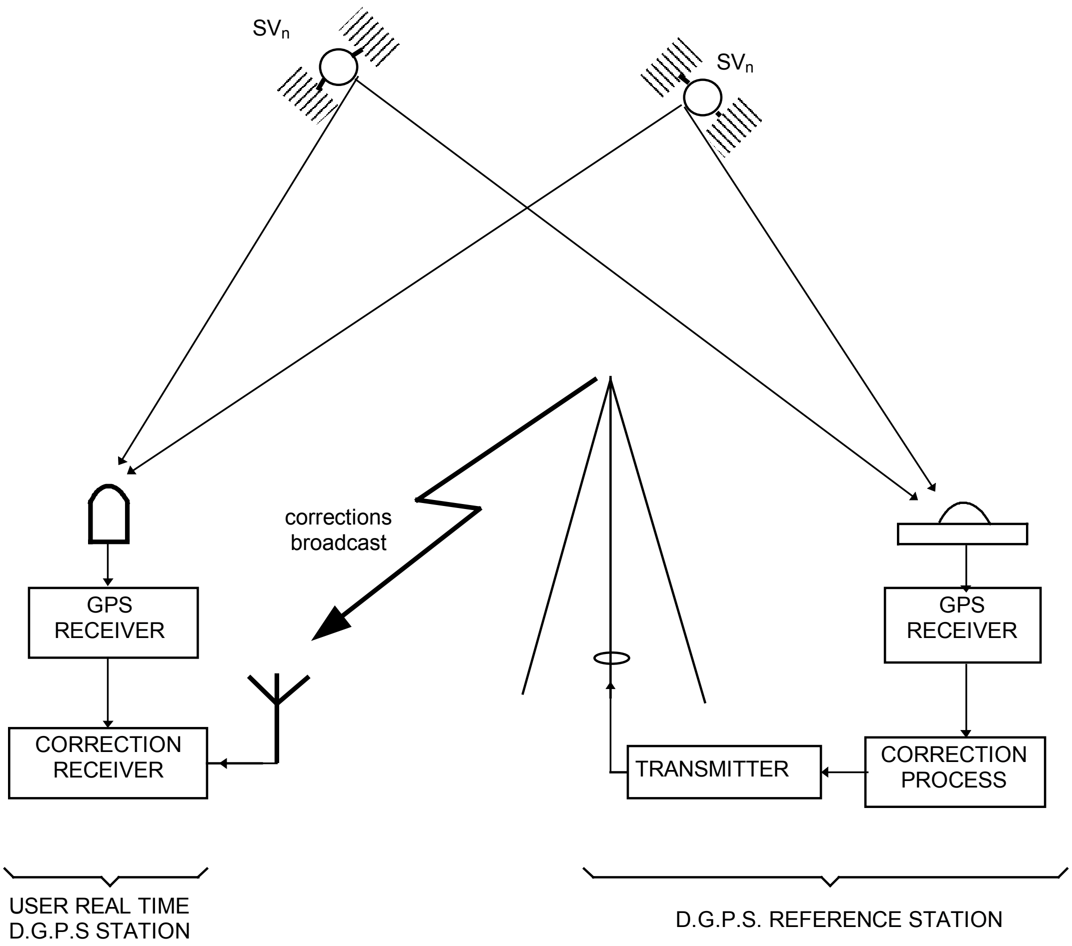
To describe a real-time differential system the following aspects need to be considered:

- Amount of data to be transmitted to the mobile
- Data update rate
- Area to be covered
- Cost of the equipment

The amount of data to be transmitted and the update rate will determine the frequency of the clock to be used to generate the corrections messages. The spectrum width required by that clock depending on the type of modulation should be consistent with the transmission frequency. Besides, the propagation of radio frequency signals depends on the carrier frequency. The area covered by the reference station depends on the propagation conditions and on the transmission power level. Therefore, those different parameters should be considered in selecting a frequency with a view to meeting the users requirements.

For several years DSNP has been providing differential stations operating on two frequency bands: HF band (from 1.6 MHz to 3.4 MHz) and UHF band (from 410 MHz to 470 MHz).

A differential station consists of a GPS receiver capable of computing pseudo-range corrections and measuring the phase of the received signal. The resulting data is fed to a formatter board that generates the corrections messages to be broadcasted by the power transmitter.



2.8 KART / LRK

2.8.1 KART

• Theory of operation

Typically, the algorithms used to determine the values of ambiguous terms go through the following steps:

- Defining a search volume, built on the basis of the approximate position, according to its uncertainty (covariance),
- Computing all possible solutions in the search volume,
- Choosing the best possible solution (minimum variance),
- Validating the solution (through comparison with the second best solution).

Some conditions and tests are applied at different levels so that a priori inconsistent solutions can be rejected.

This scheme can be used for dual-frequency receivers because the number of possible solutions remains relatively low within the search volume. In single-frequency receivers, it is not reasonably applicable (in real time) due to the large number of possible solutions.

This is why the scheme used in the KART system is radically different:

- Recursive computation of an approximate solution, using the phase and pseudo-ranges triple difference. The solution tends towards the actual solution over time.

- On each measurement epoch, computation of a double-difference solution, with a fixed value for ambiguous terms (integers). The solution results from successive updatings, starting from the approximate solution, first fixing the integers for the satellite pair less sensitive to the initial position error and finishing with the most sensitive pair.

- Validation of these solutions with fixed ambiguous terms (traditional test on residuals, complemented by a test for their changes over time).

- Check for the repeatability of the solution over a given period.

This scheme dramatically differs from the traditional methods. The following differences can be pointed out:

- No search volume is required, meaning that the method does not rely on any hypothesis on the quality of the approximate solution, so the actual solution does not risk being left outside a search volume that may be improperly estimated.

- No statistical analysis on multiple solutions.

- No Kalman filter or statistical analysis. More generally, no stochastic models or statistical distributions need to be a priori determined.

The methods based on statistical tests are naturally attractive, but they are only efficient on condition that the stochastic models can be determined properly. Unfortunately, it is very difficult to determine precise models for the errors affecting the GPS measurements. The multipath effects or the propagation errors, to name but a few, are not easy to evaluate. As a result, the so-called 'optimal' methods are not so optimal in practice, as they are to much dependent on the a priori models.

- **Characteristics**

The maximum range and the initialization time are the main two aspects to be considered in order to achieve the best possible results with a real-time kinematic system.

The maximum range is defined as the maximum distance from the reference station within which the user will fully benefit from the performance of the system. Two major factors may affect the maximum range:

- Physical phenomena: in the case of KART, just like in any single-frequency system, the maximum range is mostly limited by the uncorrelation of ionospheric errors. This phenomenon is significant at distances in excess of 10 to 15 km (for a medium latitude). Generally such distances can be covered with the KART even though longer initialization times are required.

- Data transmission and the techniques implemented to provide a reliable link from the reference station to the mobile. The traditional UHF DGPS systems from DSNP have a long record of field-proven reliability in excess of the requirements of the KART.

To sum up, it is advisable to install both the GPS and UHF antennas of the reference station at a location clear of any obstructions and high enough to cover the area of interest and preclude any multipath effects that otherwise might affect the performance of the system. Even if the UHF link allows operations over longer ranges, initializations should be performed within no more than 10 to 15 km of the reference station.

2.8.2 LRK

The advantages of using the two GPS frequencies lie in that this makes it possible to cover longer ranges and cut down the initialization time while increasing the reliability of the initialization process.

● As far as maximum range is concerned, the uncorrelation of ionospheric errors no longer prohibits operations more than 15 km away from the reference station.

We can compensate for uncorrelation by combining the measurements taken on L1 and L2 (ionospheric delays are inversely proportional to the squared frequency).

Then the remaining limitations arise from the uncorrelation of tropospheric errors and from the errors on the broadcasted ephemerides, but those errors have a much smaller impact in terms of amplitude. As a result, the only practical limitation is dependent on the quality of the radio link (several tens kilometres in the case of the DSNP UHF link).

● When it comes to solving for the ambiguous terms in carrier phase measurements, resorting to two frequency is still more beneficial. Combining the two frequencies makes it possible to get a wavelength of 86 cm (L2-L1), which decreases the number of solutions that may be retained around the approximate position and increases the separation between any two possible solutions. All those features allow the initialization time to be substantially reduced, and they significantly increase the reliability of the solution.

As many tests have been conducted that reveal the merits of the technique used in the KART system, the same processes can be used with the following enhancements:

- Recursive computation of the approximate solution: in addition to the pseudo-ranges (C/A code) and phase triple difference on L1, a triple difference is computed on the phase of L2 and on L1/L2 pseudo-ranges (P/Y code). This leads to a faster convergence towards the actual solution.

- On each epoch, computation of a double-difference solution —only one— with resolved integers. With the LRK method, however, the position solution is computed in two steps. Starting from the approximate solution, in the first step, a «coarse» solution is computed through a linear combination of the measurements made on L1 and L2 (86 cm integer ambiguity instead of 19 cm), on wide windows. The solution, more accurate —albeit coarse— than the approximate position, is used to solve for integers in the second step with higher accuracy.

- Final computation of the position: over long distances, using the two frequencies compensates for ionosppheric delays and increases the covered area.

- Solution validation: testing the residudals on L1 and L2.

- The checking for the repeatability of the fitting solution is used only in the case of critical conditions (e. g. only four satellites received).

To sum up, the LRK system can be used farther from the reference station than the KART and allows shorter initialization times to be achieved, meaning that it can be used in critical environments (forests, urban areas, harbours, etc.). However, this does not relieve the user of the necessity of installing the reference station antenna in an open area, high enough to cover the desired range.

2.9 GNSS

2.9.1 GENERAL DESCRIPTION

Satellite navigation systems are now used in scores of applications worldwide. The best known two systems in operation as of today are:

The US GPS (Global Positioning System) which is the most complete,

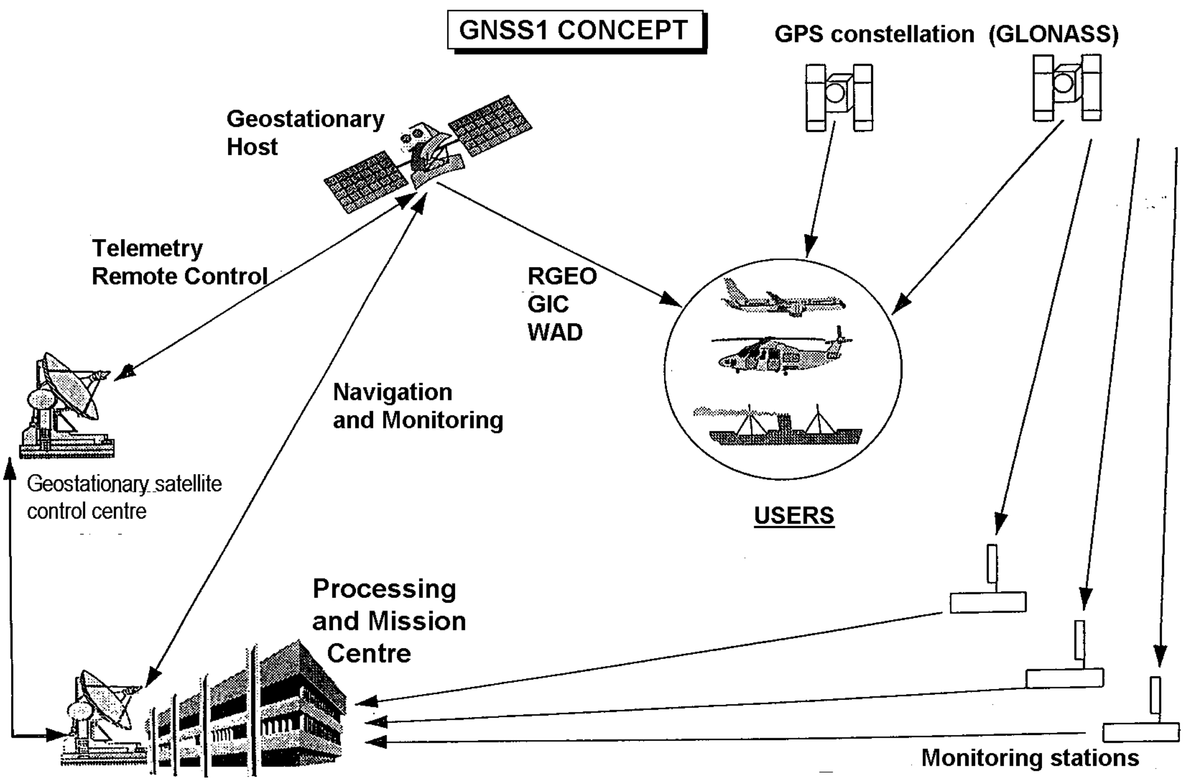
The Russian GLONASS (GLObal NAvigation Satellite System).

As both these systems are originally designed for military applications, they are entirely under the control of the respective Defence Department of the two countries. As a result, civilians cannot be sure of being allowed full access to the signals in critical periods of time. Moreover, the accuracy achieved using the non-encrypted signals is only on the order of a few tens metres.

All those aspects led the civilian community to devise a totally new system known as GNSS (Global Navigation Satellite System).

In future, a complete constellation —GNSS2— should provide civilian users with signals and data allowing them to compensate for any shortcomings in the navigation systems at sea, on land or in the air.

The current GNSS1 is the first phase in that scheme, based on the augmentation of the GPS service through geostationary satellites.



2.9.2 Purpose

The GNSS scheme serves three major purposes:

Complementing the range measurements with geostationary satellites (R_GEO),

Controlling the integrity of the navigation system (GIC),

Broadcasting differential corrections over a wide area (WAD).

2.9.3 GNSS concept

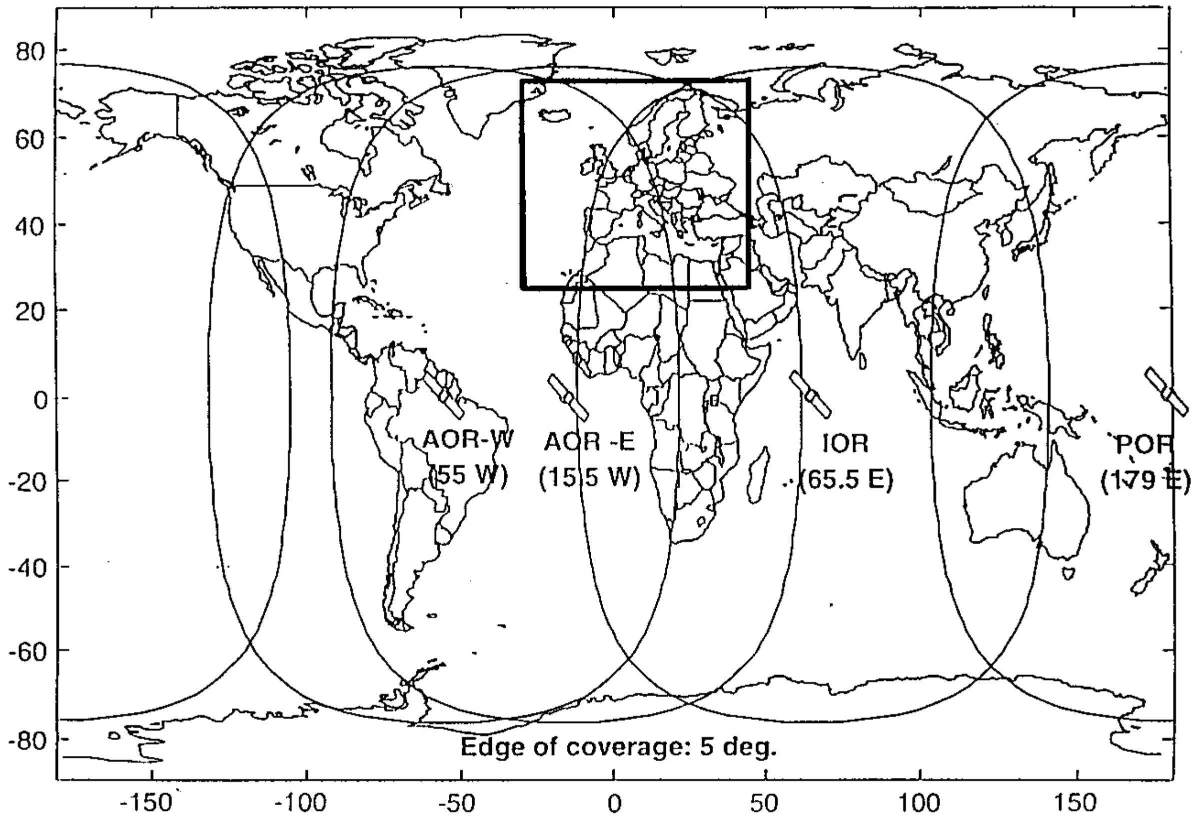
The GNSS system consists of the following elements:

Stations monitoring the navigation system (GPS, GLONASS), distributed over the area to be covered, allowing continuous monitoring of the system,

A Processing and Mission Centre that collects and computes the data required for the performance of the system,

A control centre for the geostationary satellites, uploading the necessary data to the geostationary satellites,

One or more geostationary satellites broadcasting the data (R_GEO, GIC, WAD) over the area to be covered.



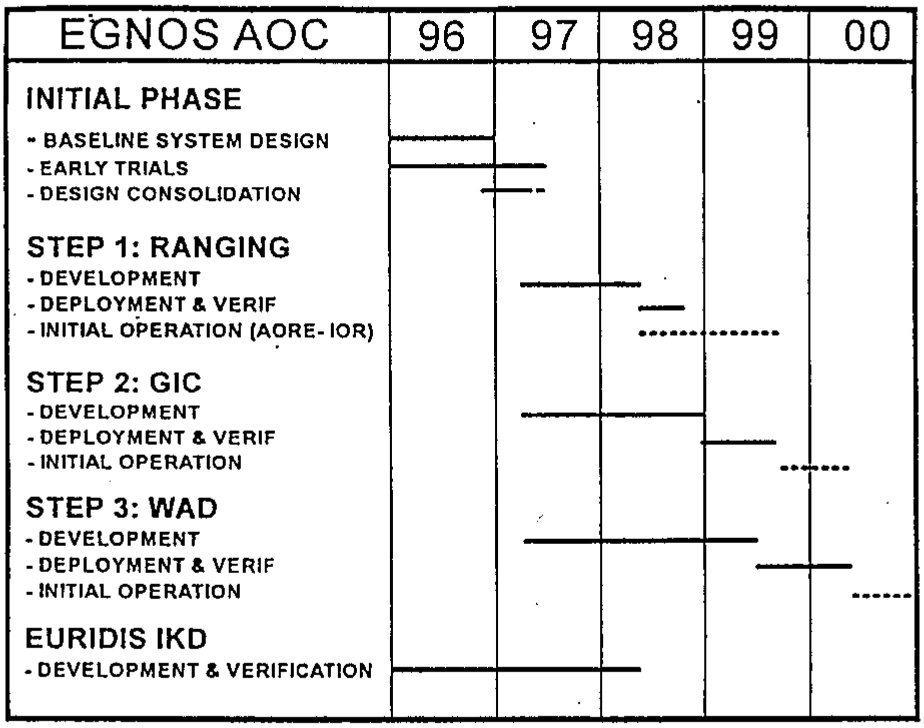
2.9.4 The different systems

There are three systems as of today (January 1998):

For the American continent: WAAS (Wide Area Augmentation System)

For Europe: EGNOS (European Geostationary Navigation Overlay System)

For Asia: MSAT



EGNOS PLANNING

3. AQUARIUS 5000 RECEIVER

3.1 GENERAL DESCRIPTION

The DGNSS receiver can be thought of as a 'black box'. This is a watertight, ruggedized, high-tech receiver that comes in two versions, single- or dual-frequency. It can be changed from L1 receiver to L1/L2. It is intended for positioning systems, machinery guidance or for use as sensor in integrated navigation systems.

It may be optionally composed of:

- A C/A GPS Core module (receiver for L1 C/A code GPS signals) in the single-frequency version of products.
- A P/Y GPS Core module (receiver for GPS L1/L2 frequencies, C/A and P/Y codes) in the dual-frequency version of products.

The DGNSS receiver makes it possible to use 12 C/A or C/A and P/Y channels primarily dedicated to receiving the signals from the GPS constellation satellites, and four C/A channels dedicated to receiving the signals from the geostationary satellites complementing the GPS (WAAS, EGNOS) and pseudolites.

It relies on modern techniques such as correlation (FFT), multipath processing, use of different types of DGPS corrections (WASGPS from the WAAS, digital RTCM-SC104 LADGPS, kinematic-processing precise DGPS), quality control (UKOOA).

The basic receiver is a position and raw data sensor with a variety of options capable of meeting multiple needs:

- Option for generating, transmitting or receiving and using DGPS data on a UHF radio link, by way of an add-on transmitting or receiving module that allows operation as a 'reference station' or a 'mobile receiver'.
- Option for recording data to a PCMCIA memory card, with record sessions scheduling capability and management of sleep/wake-up functions.
- Option for increasing input/output capabilities, with precise time tagging of events, and 1PPS output.
- Provision for upgrading taking account of current and future GPS applications:
    - Communications with an outside device, whether for a mobile station or for a reference station, and capability of supporting all functions of the applications (kinematic DGPS processing, user coordinates transformation system, user geoid model).
    - Capability of using DGPS data from sources supporting formats other than RTCM-SC104 (INMARSAT demodulator, MF receiver, etc.).
    - Operation as a reference station generating DGPS data to be fed to an outside data transmitting device.

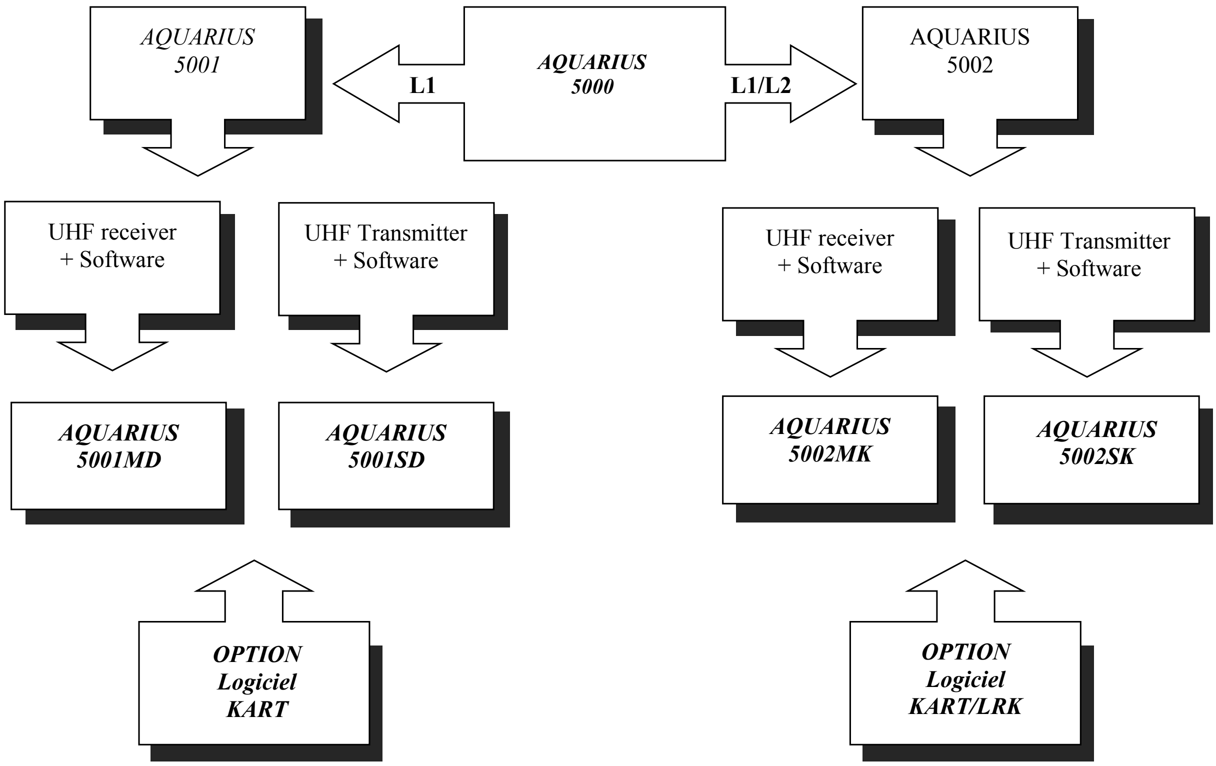
The receiver does not have any built-in human/computer interface but it comes with user-friendly control and monitoring software running on a standard PC to be attached to the receiver through a serial line.

The receiver is controlled and monitored through one of its serial ports with a PC and specific software accompanying the receiver, or with any software tool capable of using the set of available remote control words (integrated navigation system, data acquisition system).

Also, it can be operated without any control device, on the basis of configuration data saved in its non-volatile memory. The configuration data are loaded from a PC using the Conf Pack software tool.

If no outside human/computer interface is used, the configuration and performance of the receiver can still be checked on an elementary built-in display panel.

All the functions of the receiver are integrated in a single housing, except the GNSS receiving antenna, the UHF DGPS data transmitting or receiving antenna and the optional outside power supply battery. All the connectors required for communications with the outside are available on the front panel and rear panel.

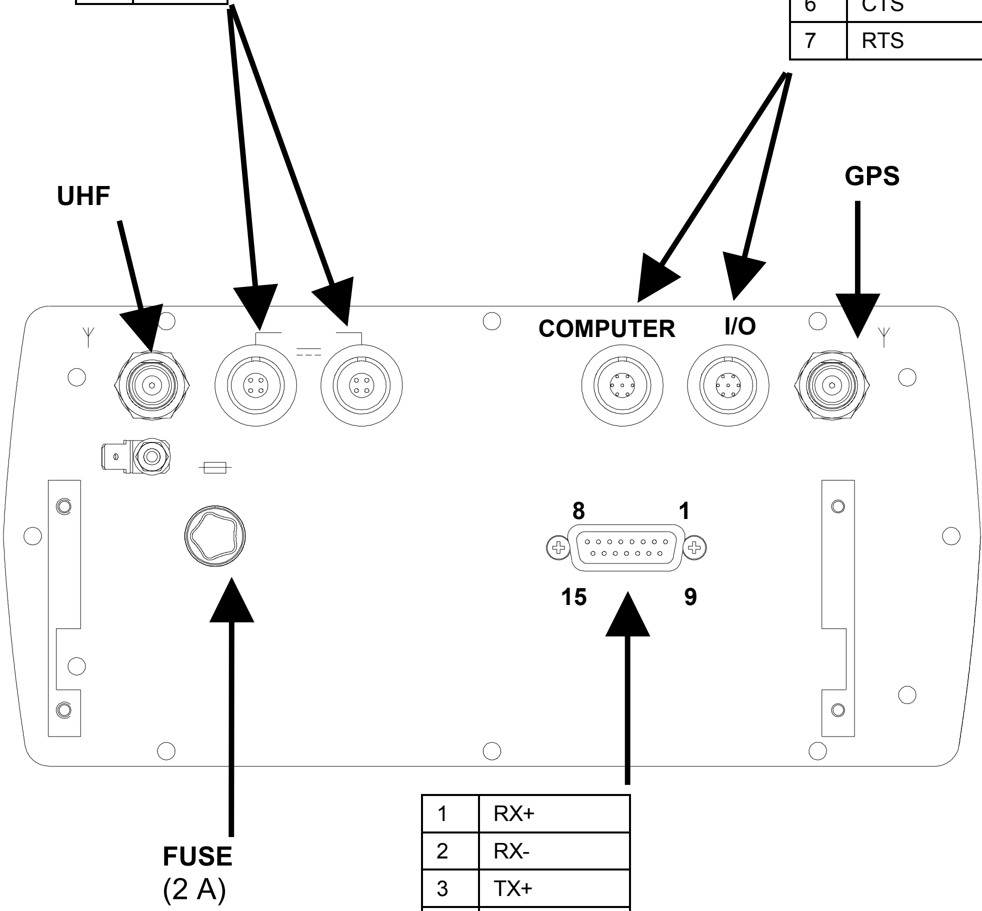

AQUARIUS 5001
AQUARIUS 5000
L1
L1/L2
AQUARIUS 5002
UHF receiver + Software
UHF Transmitter + Software
UHF receiver + Software
UHF Transmitter + Software
AQUARIUS 5001MD
AQUARIUS 5001SD
AQUARIUS 5002MK
AQUARIUS 5002SK
OPTION Logiciel KART
OPTION Logiciel KART/LRK

POWER

1	+VIN
2	+VIN
3	-VIN
4	-VIN

RS232

1	+12V
2	TX
3	RX
4	ON
5	GND
6	CTS
7	RTS



RS422 INPUT/OUTPUT

1	RX+
2	RX-
3	TX+
4	TX-
5	
6	GND
7	GND
8	VIN
9	
10	
11	
12	
12	
14	
15	

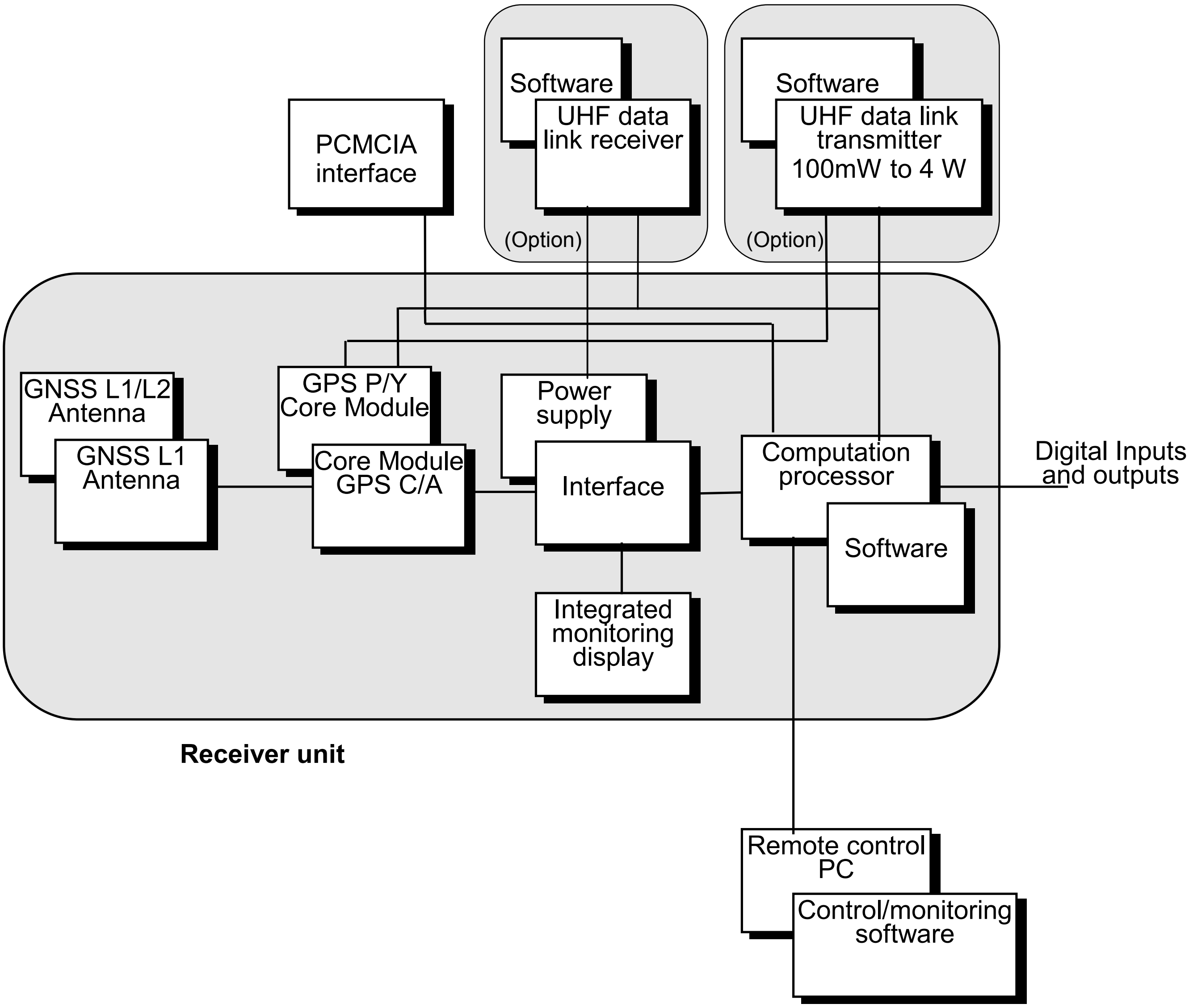
3.2 SPECIFICATIONS

Receiver with no integrated data transmission or reception unit

Operating temperature	:	-20 to +55°C
Storage temperature	:	-40 to +70°C
Consumption	:	8 to 15 Watts with P/Y core module.
Maximum volume	:	130×260×220 mm.
Maximum weight	:	3.7 kg with P/Y core module.
Sealing	:	IP66 (totally protected against solid bodies, heavy sea watertight)
Mounting	:	Bare, rack-mount or fit-in (with mount accessories)
		On stand (tripod) or support device.

Integrated display for daylight (full sun light) and night.

3.3 BLOCK DIAGRAM



3.4 POWER SUPPLY CONNECTORS

3.4.1 Power supply

A floating-type power supply voltage is required, lying between 9 and 36 VDC. The equipment is protected against wrong polarity and overvoltage on the power supply input. Two power supply connectors are provided, on the rear panel. They can be used interchangeably. The required input power depends on which options are connected.

✦**CAUTION**: If a UHF transmitting module is attached to the equipment, then the power supply voltage is limited to **16 VDC**, non-floating. See the installation instructions relating to the UHF transmission module.

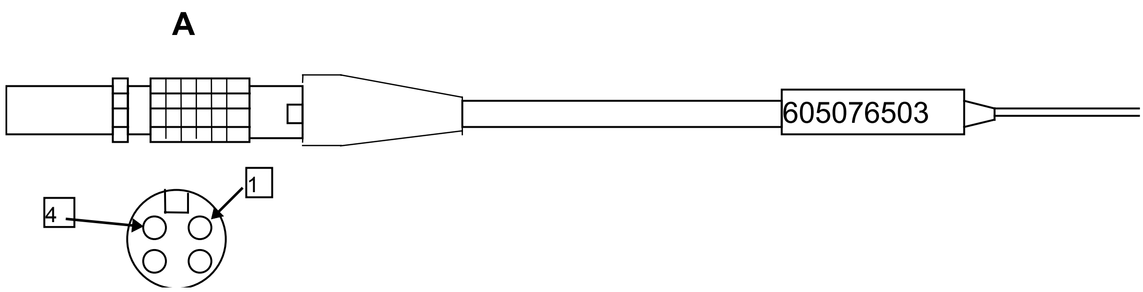
Power drain table

OPTIONS INSTALLED	POWER DRAIN
L1 receiver	11 W
L1/L2 receiver	13.6 W
L1 receiver + UHF receiver	15 W
L1/L2 receiver + UHF receiver	17 W
L1 receiver + UHF transmitter (1)	24 W
L1/L2 receiver + UHF transmitter (1)	26 W

(1) In receivers equipped with a UHF transmitter, the power drain depends on the transmission recurrence rate. In the above table the power drain is specified for a maximum recurrence rate.

3.4.2 Power Supply cable

Part number: 2640076503



A is a 4-contact JKX FD 1G 04 MSSDSM (5011252) plug with JBX1 MPN (5080359) sleeve.Manufacturer: FCI. Shielded cable, CRTB 2x0.93 mm2, red/black (6030103), 2 m long. Manufacturer: FILOTEX.

Plug A

1	+VIN
2	
3	-VIN
4	

3.5 RS232 INPUT/OUTPUT PORTS

3.5.1 General description

Serial ports (basically three, optionally four) and a PCMCIA interface are used for data inputs/outputs.

The inputs on the serial ports convey the following remote control words and differential data or corrections:

- The remote control words, in NMEA0183 format, fall into three categories:
  - All remote control words required to put the equipment into operation, using the human/computer interface software running on an outside PC for control and performance check (position initialization, time setup, ephemerides, selecting output messages, enabling functions, sending queries, etc.).
  - Configuration loading commmands.
  - Application software loading commands.
- The differential corrections or data may be in the following formats:
  - Single- or multi-station, 6 of 8 RTCM-SC104 digital format,
  - 6 of 8 UHF RTCM-SC104 format, through the built-in UHF receiver option (NDS100MKII compatible),
  - Single- or multi-station DSNP UHF format, through the built-in UHF receiver option (NDS100MKII compatible),
  - DSNP format, through the built-in UHF receiver option,
  - User data to be relayed over the UHF link.

No interleaving of input messages is permitted, except for NMEA0183 messages within a 6 of 8 RTCM-SC104 format bit stream.

All functions are accessible through any of the serial ports. However, port A is preferably dedicated to connection to a PC used as outside human/computer interface, and port D is dedicated to connection to a transmitter or to a built-in DGPS corrections receiver (the power supply to this equipment is provided).

The outputs on the serial ports convey the following information:

- Replies to remote control words and, more generally, any information required by the outside human/computer interface.
- GPS measurements and raw data.
- Computed GPS data.
- User data received on the UHF data link.
- For a reference station, differential data and corrections intended for an outside data transmitting device, in 6 of 8 RTCM-SC104 digital format.

Interleaving of output messages on the same port is not permitted.

The output buffers containing the messages are emptied as the messages are read out. In case of jamming (i. e. if the output buffers fail to be emptied) whole messages may be lost. If any information is lost, this is reported in an anomaly message.

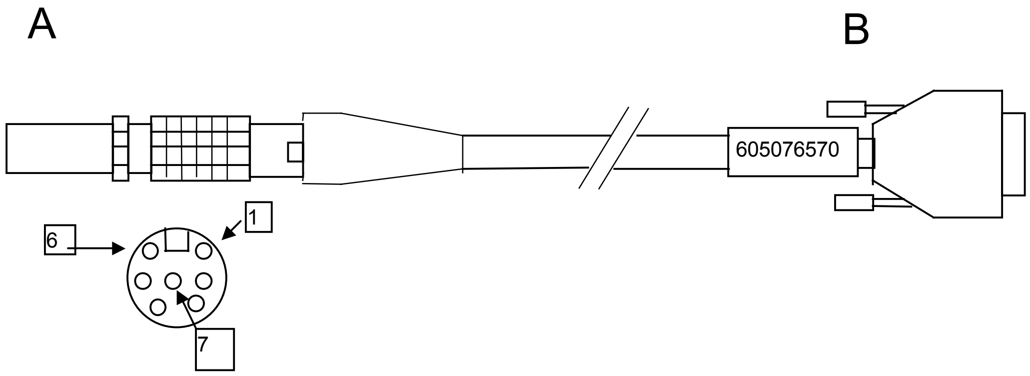
Output message latency:

Data type	Output latency
Computed data	
Raw data	

3.5.2 RS232 cable

According to NMEA0183 Version 2.2, an RS422-type interface should be used. However, many devices connecting to GPS receivers (such as PCs, plotters, etc.) only have RS232 ports. For that reason, our receivers are still equipped with RS232 ports so that they can easily interface with existing equipment.

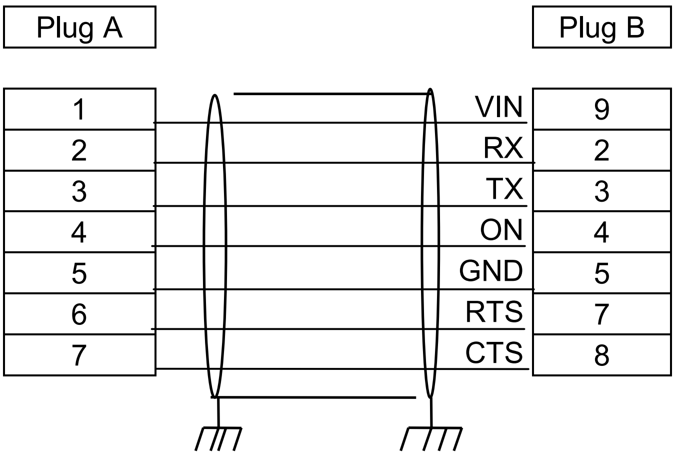
Part number: 26H1076570



A is a 7-contact JKX FD 1G 07 MSSDSM (5011253) plug with JBX1 MPN (5080359) sleeve. Manufacturer: FCI.

B is a 9-contact female subD DE-9S (5030357) connector with metal cover 8655MH09-11 (5080357). Manufacturer: FCI.

Shielded cable, 4-pair, FMA2R (6030097). Overall length 2 m.



3.6 L1 AND L1/L2 ANTENNAS

3.6.1 Power supply

The power supply voltage required for the antenna preamplifier is available on the GPS signal input connector (12 VDC, 200 mA).

3.6.2 Coaxial cables

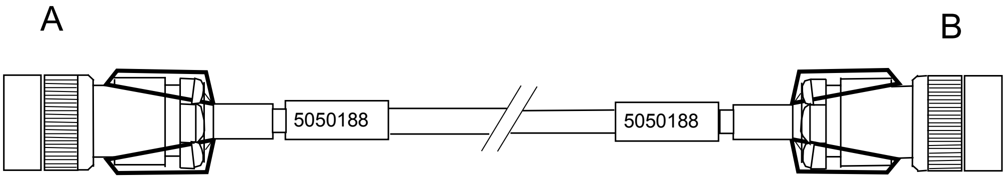
The signal loss between the antenna and the receiver should not exceed 24 dB. As a result, the cable length should not exceed a maximum determined as a function of the type of cable.

Below are a few examples of maximum cable lengths:

Cable Type	Att. @ 1.6 GHz for 100 m (dB)	Max. length (m)
RG223	83	30
RG214 (kx13)	32	75
GEDELEX 4.6/50CC	13	185
GEDELEX 4.8/50CC-FP	9.5	250
GEDELEX 8.5/50CC	7	340
GEDELEX 9.3/50CC	5.5	430
LDF 4-50A	10	240
LDF 5-50A	6	400
LDF 2-50	14	170

- **GPS antenna-to-receiver cable**

Part number: 5050188



A and B are male TNC-type coax plugs with bushings (type R143008 or equivalent)

Coaxial cable: RG223, 30 m long.

3.6.3 L1 antenna

Part No.26E1076311

- **Specifications**

Right circular polarization antenna

Impedance	:	50 **Ω**.
Power supply	:	5 to 13Volts through the TNC socket.
Consumption	:	40 mA
Gain	:	39 dB±1 dB
Noise factor	:	≤ 4 dB @ 25°C
Selectivity	:	-26 dBm @ 1555 MHz
		-08 dBm @ 1615 MHz
		-06 dBm @ 1625 MHz
		-04 dBm @ 1675 MHz
		For example, a standard Immarsat transmitter (C, 20 W at 1625 MHz) generates +7 dBm 1 m away, -7 dBm 5 m away.
		The L1 antenna can tolerate the presence of such a transmitter if it is at least 2.50 m away.
Phase centre position	:	24 mm from fixing plane
Operating temperature	:	-40° to +65° C.
Storage temperature	:	-40° to +70° C.
Shocks	:	IT2521
Sealing	:	IP67
Weight	:	342 g

3.6.4 L1/L2 antenna

(Part number: 26E1076208)

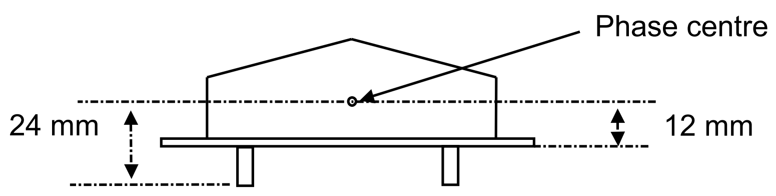
- **Specifications**

Right circular polarization antenna

Impedance	:	50 **Ω**.
Power supply	:	5 to 13Volts through the TNC socket.
Consumption	:	55 mA

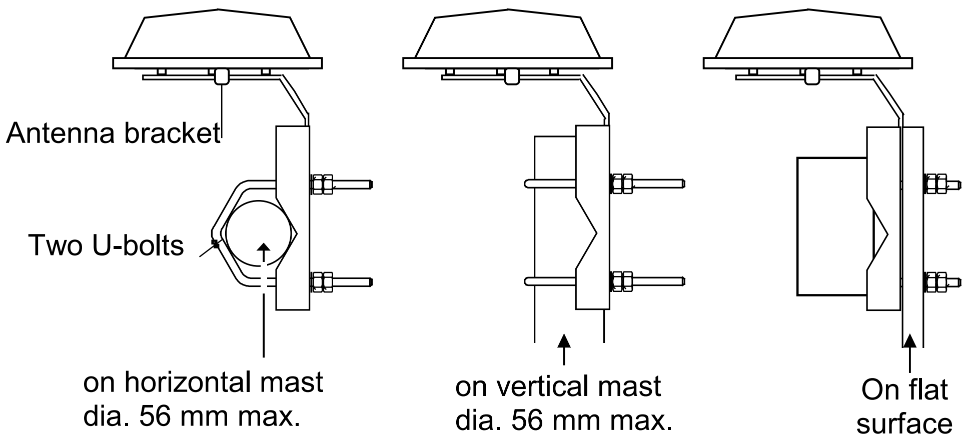
Gain	:	39 dB. ±1 dB
Noise factor	:	≤ 4 dB.@ 1575 MHz
		≤ 4 dB.@ 1227 MHz
Selectivity	:	-26 dBm @ 1555 MHz
		-08 dBm @ 1615 MHz
		-06 dBm @ 1625 MHz
		-04 dBm @
		For example, a standard Immarsat transmitter (C, 20 W at 1625 MHz) generates +7 dBm 1 m away, -7 dBm 5 m away.
		The L1 antenna can tolerate the presence of such a transmitter if it is at least 2.50 m away.
Phase centre position	:	24 mm from fixing plane
Operating temperature	:	-40° to +65° C.
Storage temperature	:	-40° to +70° C.
Shocks	:	IT2521
Sealing	:	IP67

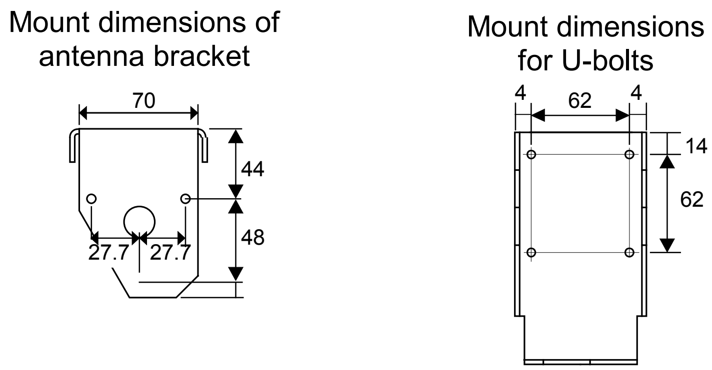
Weight: 356 g.



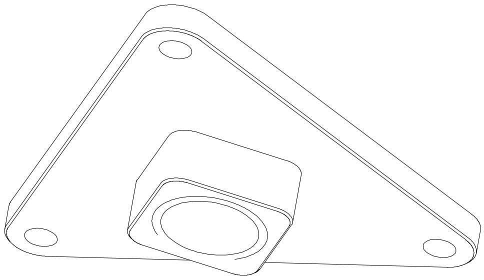
3.6.5 Antenna stands and brackets

- **AQUARIUS antenna bracket**

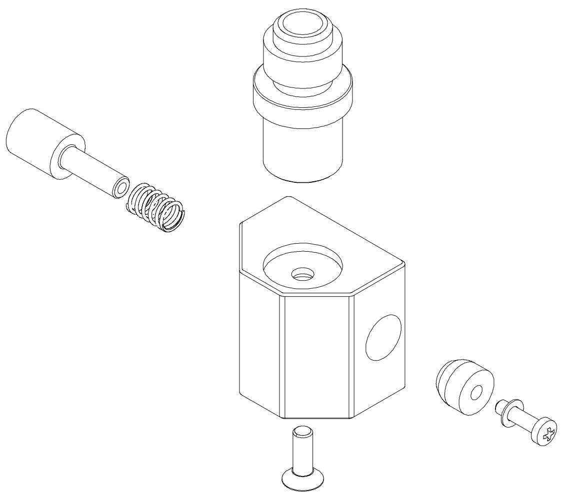




- ## Antenna stand accessories



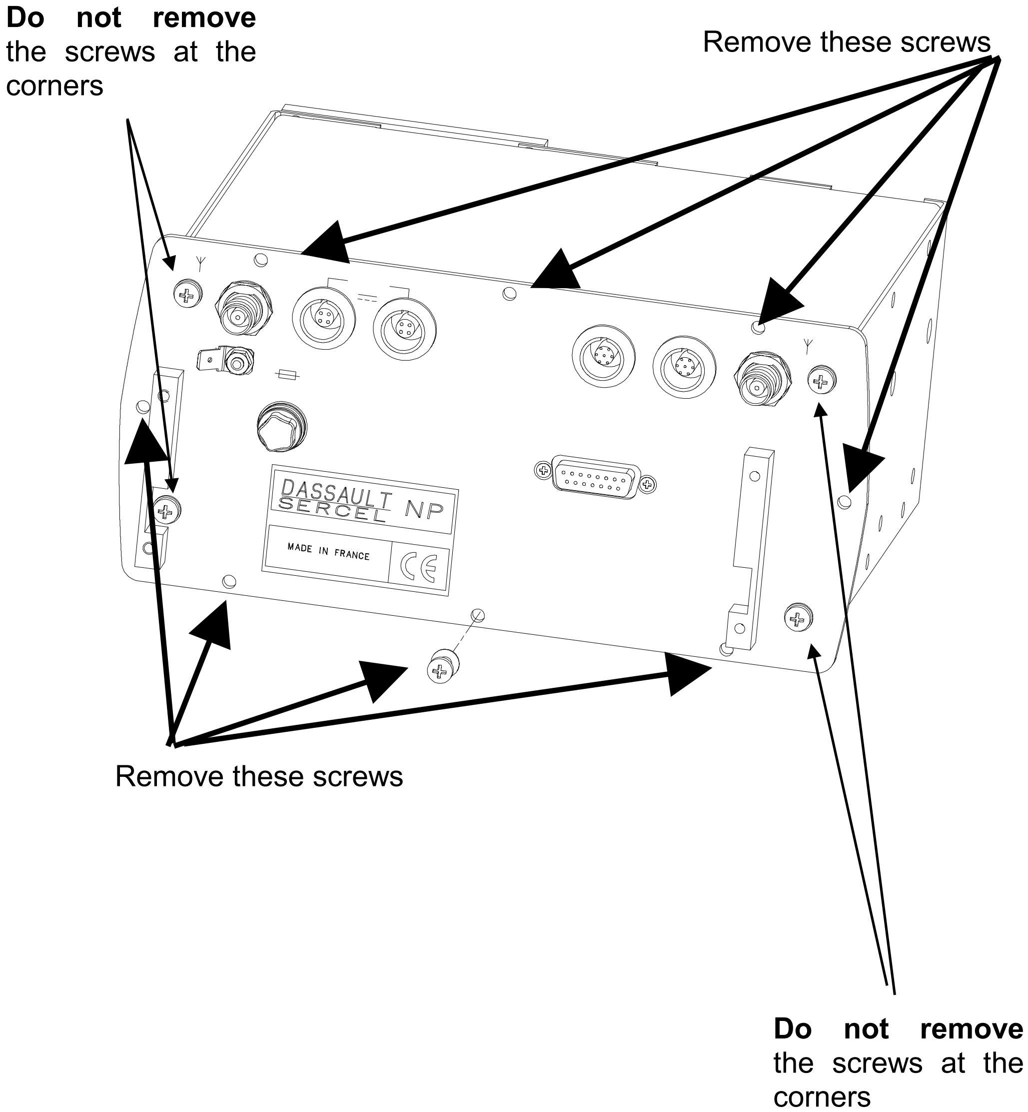
Support (Part Number 26__076577)

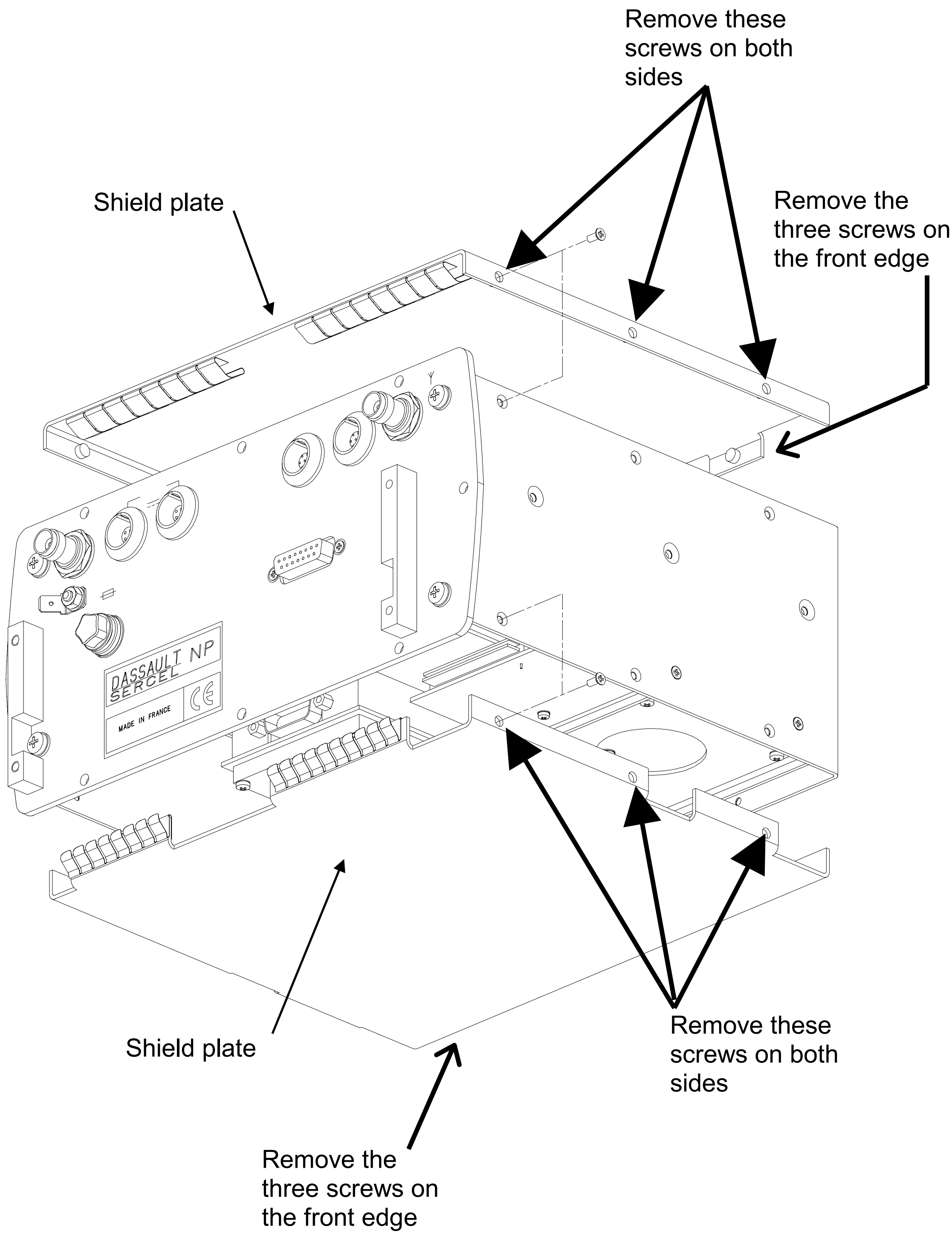


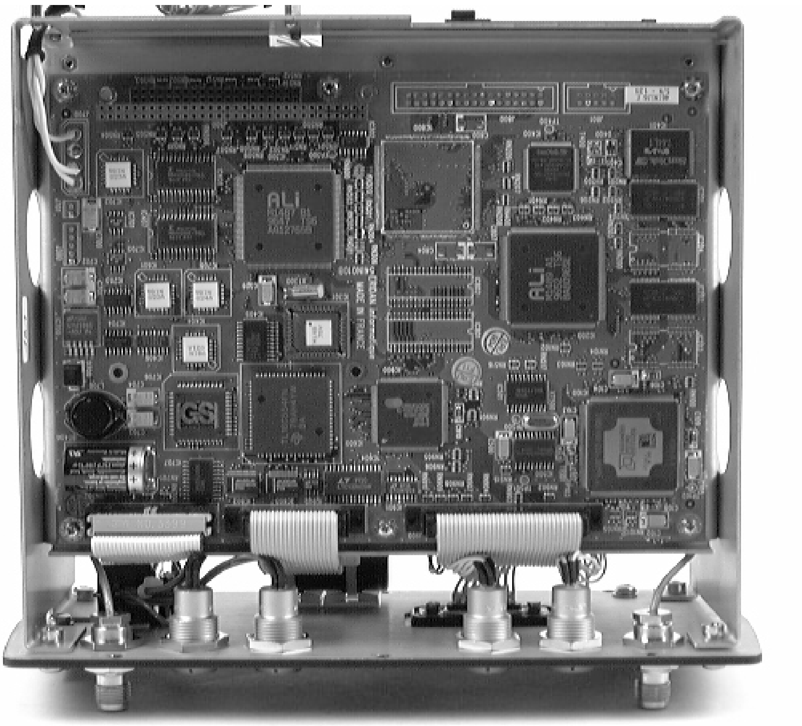
Quick Release (Part No. 26__076528)

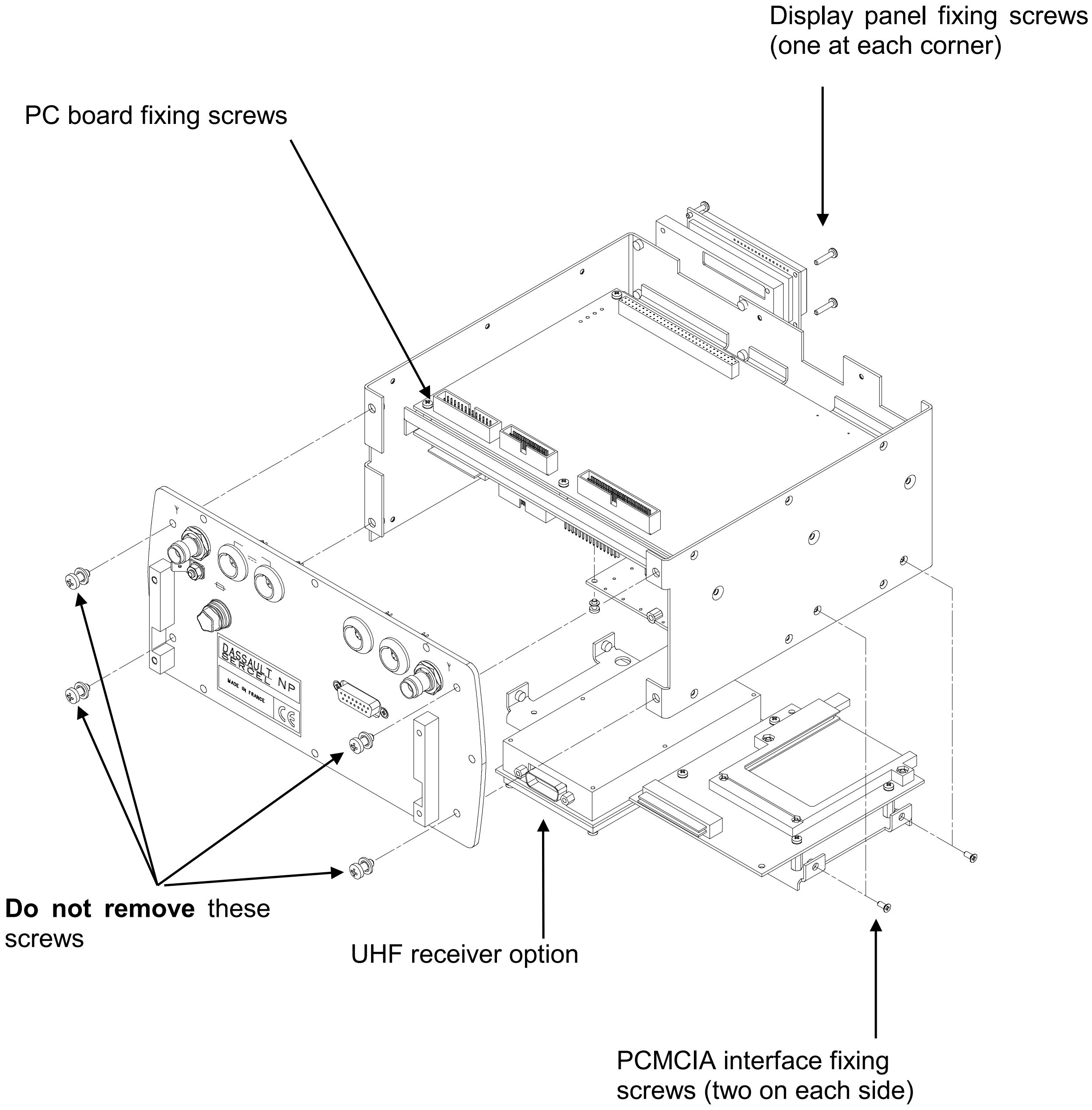
3.7 HOW TO OPEN THE UNIT

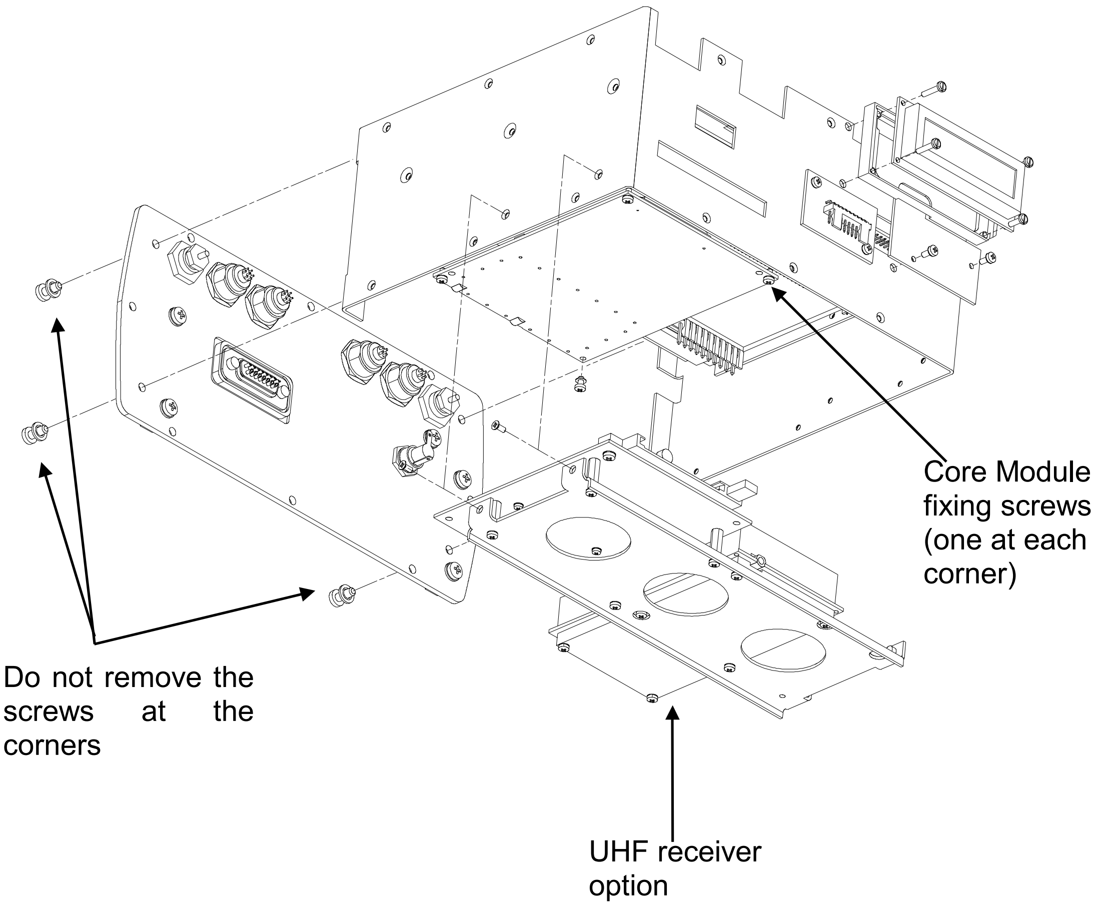
Reverse the unit and remove the eight securing screws located on the rear panel. Pull out the board assembly from the rear and unplug the pushbutton cable.




Remove these screws on both sides
Shield plate
Remove the three screws on the front edge
DASSAULT NP
SERCEL
MADE IN FRANCE
CE
Shield plate
Remove these screws on both sides
Remove the three screws on the front edge




Display panel fixing screws (one at each corner)
PC board fixing screws
Do not remove these screws
UHF receiver option
PCMCIA interface fixing screws (two on each side)
DASSAULT SERCEL NP
MADE IN FRANCE


Core Module fixing screws (one at each corner)
Do not remove the screws at the corners
UHF receiver option

Bottom View

3.8 PRINTED CIRCUIT BOARDS

3.8.1 INTERFACE board

The Interface board performs two major functions:

- Interfacing the other boards with the front and rear panels of the unit.
- Generating and monitoring all the power supply voltages required for the different modules.

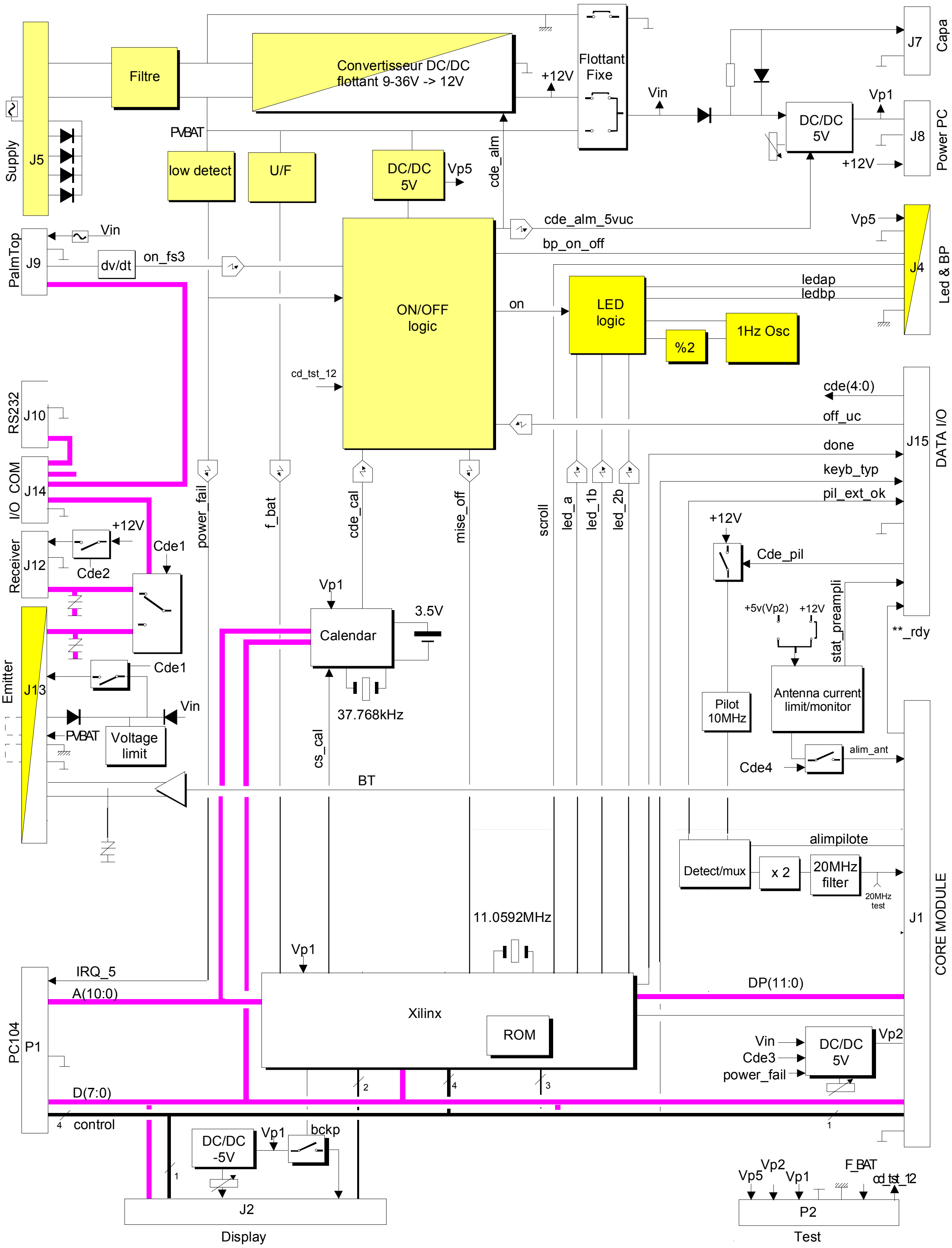
- **Power supply circuitry**.

The unit is equipped with two power supply inputs protected in the event of wrong polarity of the input voltage. The input voltage is filtered and fed to a DC/DC 9-36v/12v converter. It is also fed to a DC/DC 12v/5v converter (VP5) that generates the power supply voltage for the control logic circuitry of the Interface board. The input voltage is also fed to a monitoring circuit that transmits the measured value to the PC board. In addition, the input voltage is used for the power supply to the transmitting module. If a transmitting module is present, the DC/DC 9-36v/12v converter is bypassed because it cannot supply the necessary power to both the receiving module and the transmitting module. In that case, the input voltage is limited to 16 VDC and the receiver is protected against overvoltages (above 18 V the protection circuitry causes the fuse to blow).

If a transmitting module is present, the input converter's output voltage (VIN/VINF) is fed to two DC/DC 5 V converters generating a 5v VP2 line and a 5v VP3 line as power supply voltages to the Core Module and PC486 boards respectively.

The voltages from the UHF receiving module, from the digital output ports (P12EXT available on the rear panel connectors) and from the built-in oscillator are picked up on the VIN/VINF output and monitored by the receiver's software.

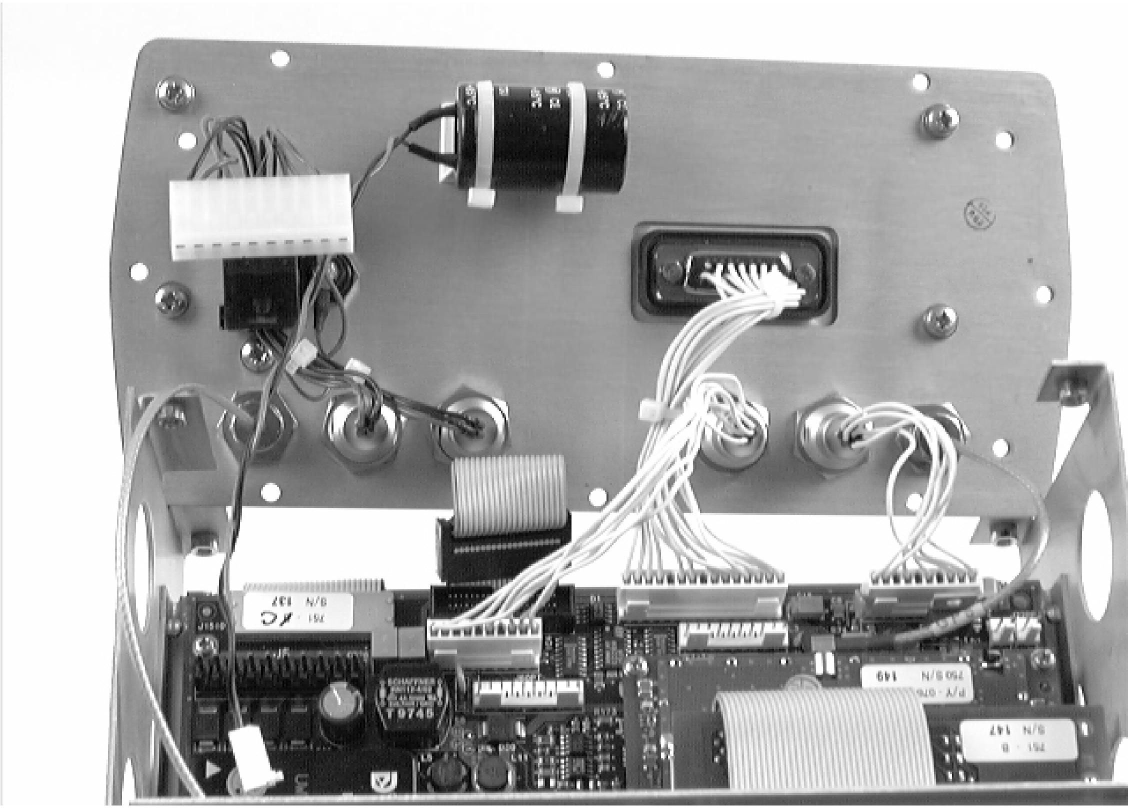
The power supply to the antenna preamplifier is generated and monitored on the Interface board. It is fed to the antenna via the Core Module board.


Supply
J5
Filtre
Convertisseur DC/DC
flottant 9-36V -> 12V
+12V
Flottant
Fixe
Vin
J7
Capa
Vp1
DC/DC
5V
J8
Power PC
+12V
PVBAT
low detect
U/F
DC/DC
5V
Vp5
cde_alm
PalmTop
J9
Vin
dv/dt
on_fs3
cde_alm_5vuc
Vp5
bp_on_off
J4
Led & BP
ledap
ledbp
ON/OFF
logic
on
LED
logic
%2
1Hz Osc
cd_tst_12
RS232
J10
cde(4:0)
off_uc
done
keyb_typ
pil_ext_ok
J15
DATA I/O
I/O COM
J14
power_fail
f_bat
cde_cal
mise_off
scroll
led_a
led_1b
led_2b
+12V
Cde_pil
Receiver
J12
+12V
Cde1
Cde2
Vp1
Calendar
3.5V
+5v(Vp2)
+12V
stat_preampli
**_rdy
Emitter
J13
Cde1
Vin
PVBAT
Voltage
limit
37.768kHz
cs_cal
Pilot
10MHz
Antenna current
limit/monitor
alim_ant
Cde4
BT
alimpilote
Detect/mux
x 2
20MHz
filter
20MHz
test
J1
CORE MODULE
11.0592MHz
Vp1
PC104
P1
IRQ_5
A(10:0)
Xilinx
ROM
DP(11:0)
Vin
Cde3
power_fail
DC/DC
5V
Vp2
D(7:0)
control
DC/DC
-5V
Vp1
bckp
J2
Display
Vp5
Vp2
Vp1
F_BAT
cd_tst_12
P2
Test

• Interface circuitry

The PC486 (Central Unit) board is equipped with four input/output ports that are not connected direct to the rear panel of the unit. Instead they are relayed via the Interface board. Ports 1, 2 respectively correspond to ports A (Computer, RS232) and B (I/0, RS232). Port 4 (RS422) is fed to an analog switch (MAX314). The data to or from the PC486 board are then fed either to the transmitting module or to the receiving module, depending on the software request.

A XILINX logic circuit (IC29) with software loaded from the PC486 board at power-on handles all the data transmitted to the display board and all the power supply control signals.



The interconnections of the Interface board with the other parts of the unit are given below.

- **J1 ⇔ Core Module (see Core Module, par. 3.8.2)**
- **J2 ⇔ Display**

1	+12V
2	N.U.
3	N.U.
4	N.U.
5	VP1
6	V0
7	Ground
8	BCKM
9	E_V
10	RS_V
11	10W
12	D0
13	D1
14	D2
15	D3
16	D4
17	D5
18	D6
19	D7
20	Ground

- **J4 ⇔ Pushbuttons**

1	VP5
2	VP5
3	LEDAP
4	LEDBP
5	BP_ON_OFF
6	BP_DEF
7	Ground
8	N.U.
9	N.U.
10	GND1

- **J5 ⇔ Power supply rear panel**

1	+VIN
2	PV1BAT1
3	+VIN
4	PV2BAT1
5	+VIN
6	PV1BAT2
7	+VIN
8	PV2BAT2
9	PVBATF
10	PVBAT_IN

- **J7 ⇔ capacitor**

1	+V
2	Ground

- **J8 ⇔ CU & Power supply**

1	VP3 (5 V)
2	Ground
3	N.U.
4	+12 V

• J9 ⇔ PALMTOP RS (rear panel)

1	+VIN (12 V)
2	RXD1
3	N.U.
4	TXD1
5	N.U.
6	CTS1
7	N.U.
8	RTS1
9	ON
10	Ground

• J10 ⇔ I/O (rear panel)

1	+VIN (12 V)
2	RXD2
3	N.U.
4	TXD2
5	N.U.
6	CTS2
7	N.U.
8	RTS2
9	N.U.
10	Ground

• J12 ⇔ UHF receiver

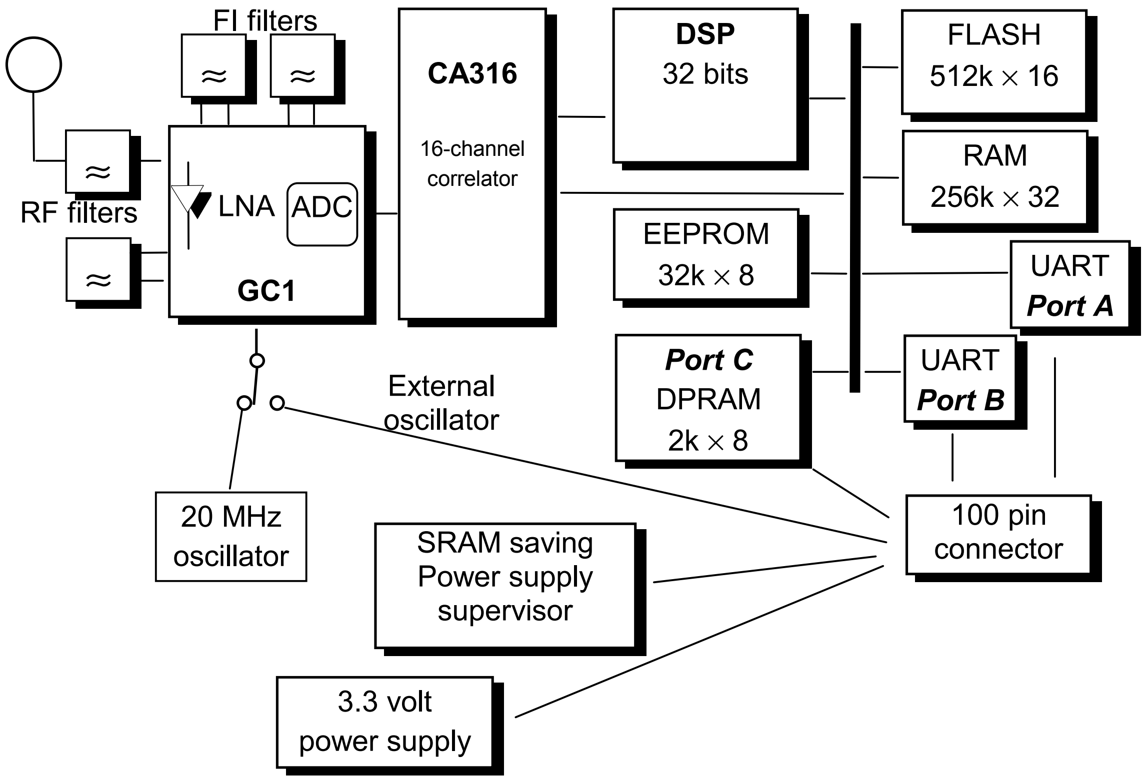
1	P12V_TDR
2	RXD4A+
3	RXD4A—
4	TXD4A+
5	TXD4A—
6	
7	
8	
9	
10	Ground

3.8.2 CORE MODULE board

One of the following two types of boards may be installed in the receiver:

- **C/A GPS Core Module**

The C/A GPS Core Module board is equipped with 16 parallel channels capable of receiving the L1 C/A code GPS signals and processing the WAAS, EGNOS and pseudolites signals. Twelve of those channels are typically used to receive the GPS signals whereas the other four are dedicated to receiving the GNSS geostationary satellites.

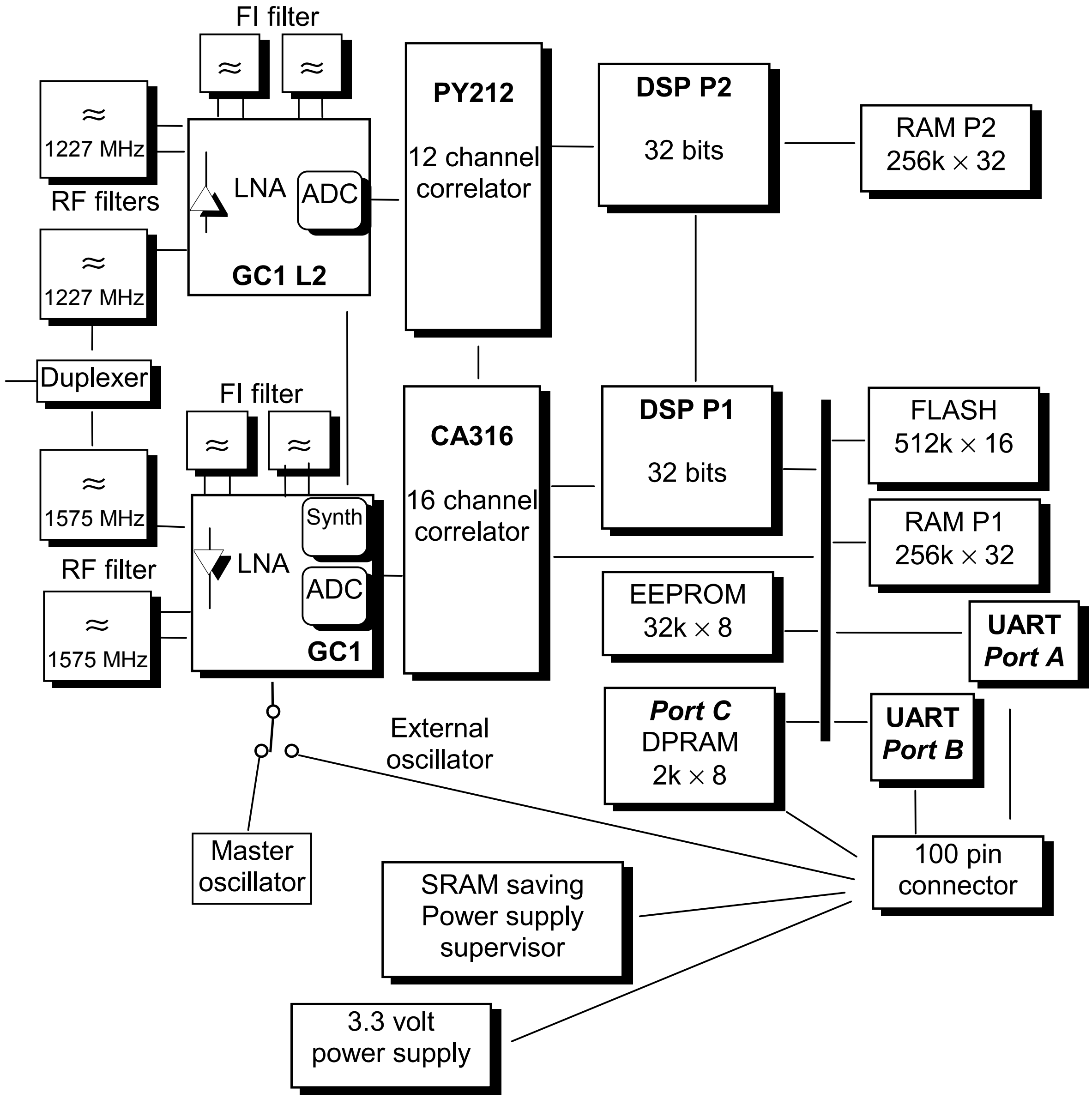


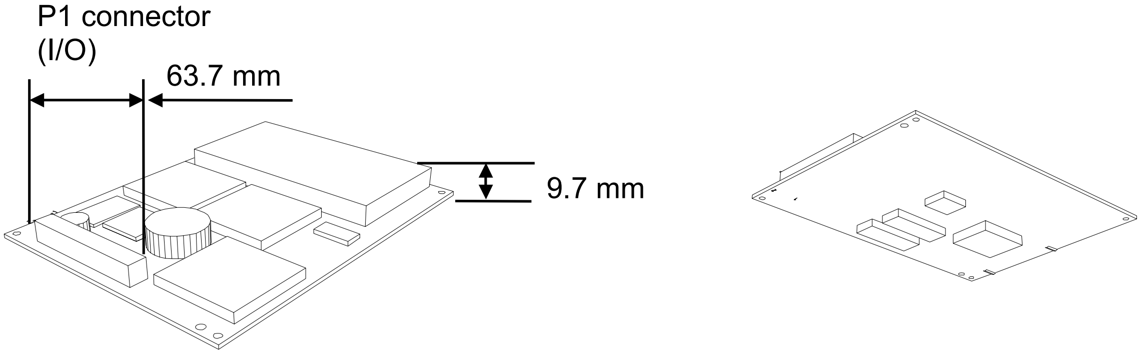
- ## P/Y GPS Core Module

The P/Y GPS Core Module board is capable of processing both L1 and L2 GPS signals. It is equipped with 12 parallel channels used to process the C/A and P codes, and four channels for processing the signals from the geostationary satellites and pseudolites.

This board also generates a PPS signal with an accuracy of 100 ns (including SA effects). It features an input for an external event to trigger a position message output time-tagged with an accuracy of 100 ns (including SA effects), and an input time tagged to within 1 ms.

Communications with the other boards are achieved through a parallel port and three serial ports.





CORE Module

1	+5V	printed circuit board's power supply	**2**	+5V	printed circuit board's power supply
3	0V	ground	**4**	0V	ground
5	H3	clock for emulator	**6**	EMU0	DSP emulation
7	EMU1	DSP emulation	**8**	EMU2	DSP emulation
9	EMU3	DSP emulation	**10**	PRGBOOT	flash 'boot' programming enable
11	0V	ground	**12**	0V	ground
13	DEFMOD	initialization of serial links	**14**	EVT2	time-tag external event input
15	SIN1	serial input 1	**16**	RTS1	enable serial link 1 inputs
17	CTS1	enable serial link 1 outputs	**18**	SOUT1	serial link 1 output
19	SIN2	serial input 2	**20**	RTS2	enable serial link 2 inputs
21	CTS2	enable serial link 2 output	**22**	SOUT2	serial link 1 output
23	0V	ground	**24**	0V	ground
25	SD0	parallel link LSB, DPRAM	**26**	SD4	
27	SD1		**28**	SD5	} parallel link data, DPRAM
29	SD2	} parallel link data, DPRAM	**30**	SD6	
31	SD3		**32**	SD7	parallel link MSB, DPRAM
33	0V	ground	**34**	0V	ground
35	CEL	DPRAM 'chip enable' input	**36**	SA4	
37	RWL	DPRAM 'read/write' input	**38**	SA5	} DPRAM parallel link addresses
39	SA10	DPRAM parallel link address	**40**	SA6	
41	OEL	DPRAM 'output enable' input	**42**	INTL	DPRAM 'interrupt' output
43	SA0	DPRAM parallel link address LSB	**44**	BSYL	DRA 'busy' output
45	SA1		**46**	SA7	
47	SA2	} DPRAM parallel link addresses	**48**	SA8	} DPRAM parallel link addresses
49	SA3		**50**	SA9	
51	0V	ground	**52**	0V	ground
53	UN_PPS	1Hz output	**54**	EVT1	time-tag external event input
55	EXSTA1		**56**	EXSTA2	
57	PROCRDY	'processor ready' monostable output	**58**	GPSRDY	'GPS ready' monostable output
59	MR	general 'reset' input	**60**	+3V	internal power supply test
61	0V	ground	**62**	BT	
63	IBBTST1		**64**	IBBTST2	
65	IBBTST3		**66**	CK25T	
67	TST25M		**68**	0V	ground
69	0V	ground	**70**	PILOT	oscillator output used
71	ALIMPILOT	internal/external oscillator selection	**72**	0V	ground
73	0V	ground	**74**	PILOTEXT	external oscillator input
75	VTUNE		**76**	0V	ground
77	0v	ground	**78**	CKPCFG	
79	CKINT		**80**	CKP+	
81	0V	ground	**82**	0v	ground
83	CAMPL1		**84**	ALIMANT	power supply to antenna preamp
85	CAMPL2		**86**	0V	ground
87	0V	ground	**88**	IBBL1TST1	
89	IBBL1TST2		**90**	IBBL1TST3	
91	CK25L2TST		**92**	IBBL2TST1	
93	IBBL2TST2		**94**	IBBL2TST2	
95	nu	(reserved)	**96**	nu	(reserved)
97	0V	ground	**98**	0V	ground
99	+5V	printed circuit board's power supply	**100**	+5V	printed circuit board's power supply

100-pin P1 connector

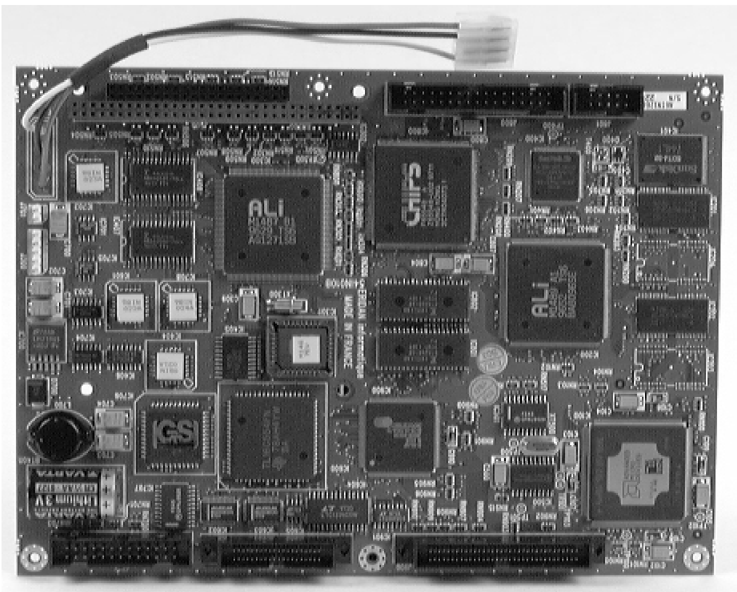
3.8.3 PC BOARD

The PC board is based on a 486-type microprocessor. It performs the following functions:

- Computing the position solution in the selected operating mode (straight GPS, DGPS, KART, LRK).
- Acquiring and checking the DGPS data in the different formats supported (DSNP, RTCM-SC104).
- Transforming the coordinates to a local geodetic system.
- Processing the altitude (global geoid model or user model).
- Navigation functions.
- Digital input/output management.
- Remote control management.
- Generating the standard output messages in NME183 format .
- Generating the user-configured output messages.
- Generating the GPS raw data messages.
- Controlling write and read operations on PCMCIA memory cards.

On the PC board are also the files containing the different configuration parameters and any information required for the unit to operate properly.

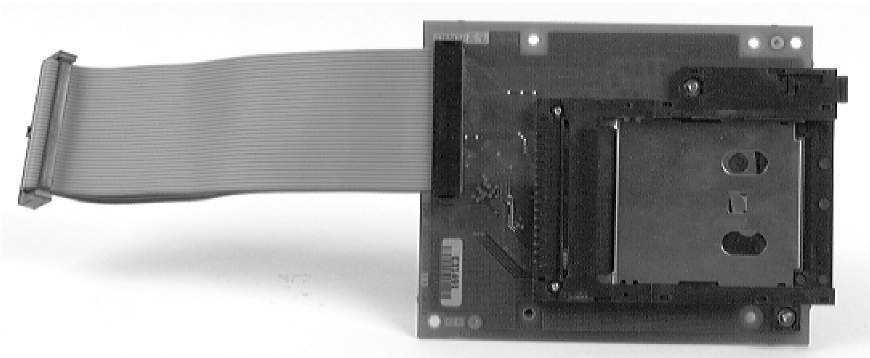
Communications with the other circuits in the receiver are achieved via a PC104 bus and four serial ports.



3.8.4 PCMCIA interface

The PCMCIA interface is connected direct to the PC board via an IDE bus. It allows interfacing with an ATA IDE standard, Type I or II format PCMCIA card.

The interface is used for communications with a PCMCIA card, whether to read from the card (new configuration, new software release, stake-out points, etc.) or to write data to the card (raw data, computed data).



3.8.5 Integrated Display

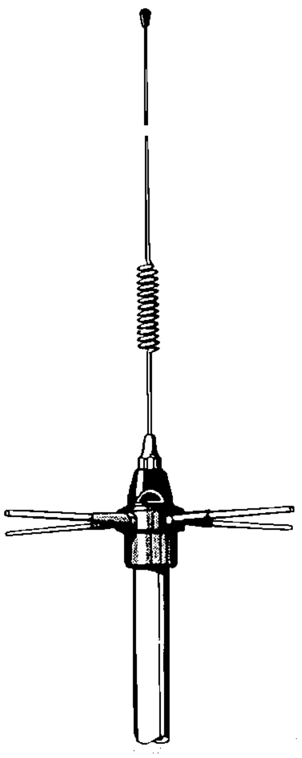
The integrated display allows quick performance checks to be made without the need to connect any communications device. The screen is composed of two 16-character rows. By repeatedly pressing the pushbutton on the front panel the operator can read the following data:

- **Installation parameters:**
  - Power supply voltage in worst-case conditions (e.g. during UHF transmission, for a station).
  - Fault indicator for the power supply of GNSS and UHF antennas.
  - WGS84 or local reference position (for a station).
  - Geodetic system identification (if local system).
  - UHF reception or transmission frequency, station No., format, cycle, sequence.
  - Transmission power (for a DGPS station).
- **Application hardware and software configuration:**
  - Identification of installed options and serial numbers of boards (if applicable).
  - Software identifications and versions.
  - Identification of the configuration of printed circuit boards.
- **Performance checks:**
  - Status of GNSS reception, Status of received satellites with Signal-to Noise Ratio.
  - Quality of UHF reception for all received stations.
  - Number of corrections transmitted and statistics on correction values.
  - Local time and *[local time−UTC]* delta.
  - Master oscillator drift speed.
  - Configuration of operating and recording sessions.
  - Status of logging memory (if the option is enabled).
  - Results from self-tests, and faults identified.

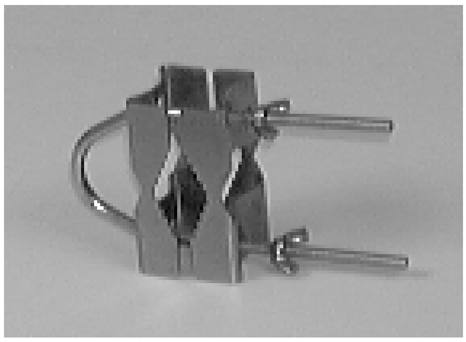
4. UHF RECEIVER OPTION (5001MD and 5002MK mobiles)

4.1 DESCRIPTION

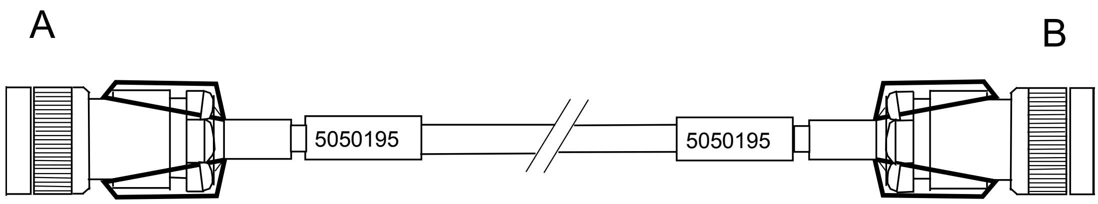
The UHF reception optional module consists of the following elements:



UHF GP 450-3 PROCOM antenna (3310202).



Antenna mount bracket



KX15 cable, 30m long (Part number : 5050195)

A and B are male TNC-type coax plugs with bushings (type R143008 or equivalent)

4.2 INSTALLATION

The unit should be powered off, with no cables connected.

Remove the cover as described above (see Par. 3.7)

Connect the cable from the UHF receiver module to the J12 connector on the interface board.

4.3 SOFTWARE SETUP

The Aquarius receiver automatically detects the presence of the UHF receiving module. However, the way the UHF module is controlled and its data are used depends on the software options enabled in the unit (e. g. KART or LRK function, etc.).

After you enable the desired options you can set the reception parameters in two ways:

- either using the CONFPACK configuring software,
- or by sending the necessary remote control words, using a communications software such as WinComm. The following four commands are required:

  $PDAS, FIXMOD

  $PDAS, DGPS,MODE

  $PDAS, DGPS,STATION

  $PDAS, NAVSEL

For detailed information on commands, see Par. 8.6.

4.4 THEORY OF OPERATION

1200 bits/s QPSK demodulation is performed, providing compatibility with earlier-design DSNP UHF stations. 4800 Baud GMSK demodulation is also performed so that the messages for high-accuracy receivers can be decoded (especially for the dual-frequency LRK).

The UHF signal is fed to a 3-pole helix filter (T1) that determines the input bandwidth and provides protection from possible jammers without noticeably impairing the noise factor of the receiver. The loss brought about by this filter is about 3 dB.

A low-noise preamplifier (Q1) with a gain of 19 dB compensates for the noise factor of the next stages.

A second UHF filter (T2), identical to the first one, is used to reject the image frequency (-80 dB at Fol-21.4 MHz). The 3 dB global bandwidth is about 8 MHz at 440 MHz.

A first frequency change is performed by the balanced dual-mixer (IC4).

Impedance matching between the mixer and the crystal-controlled filter (IC5) is performed by amplifier Q2.

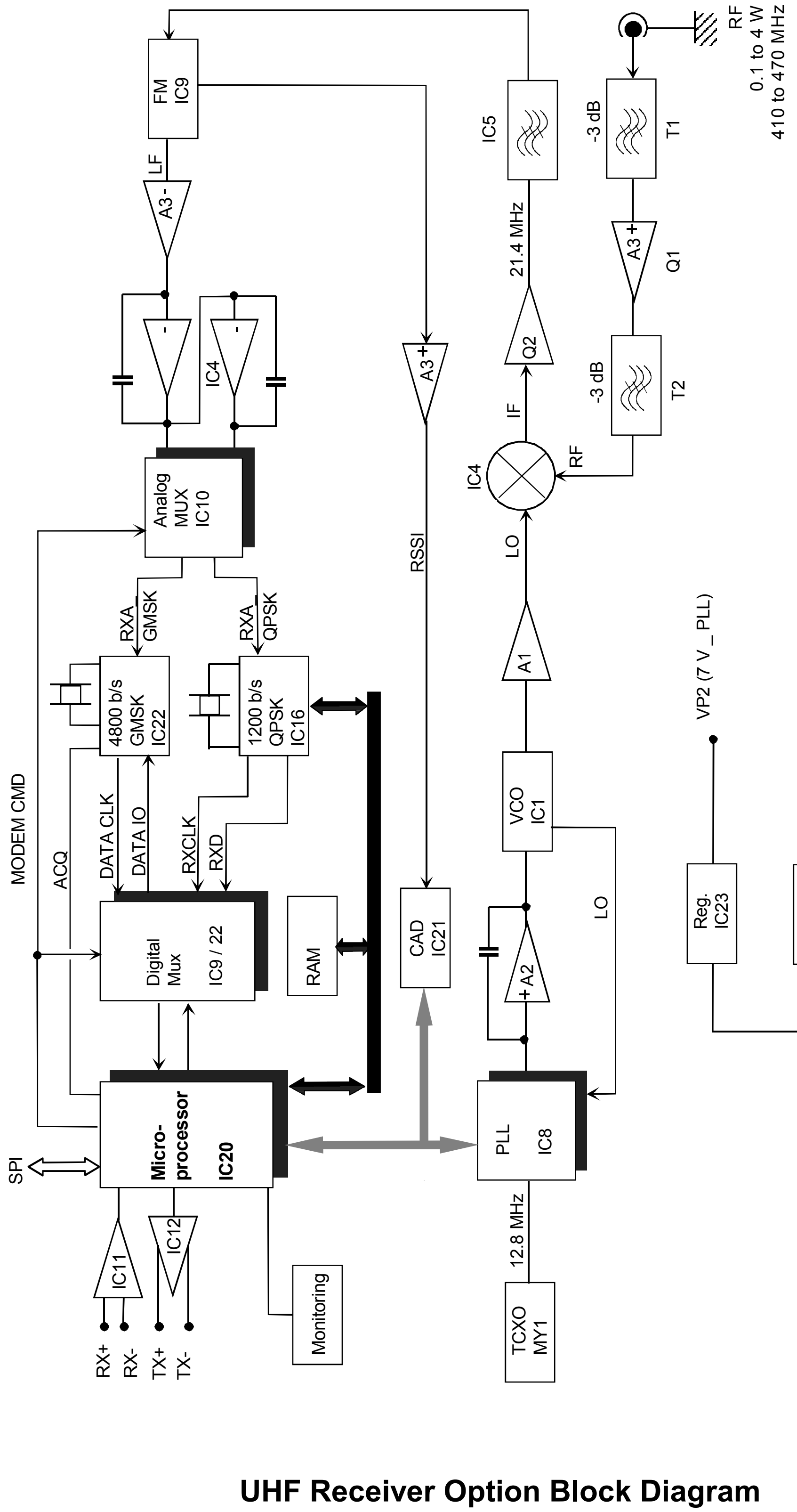
The IC9 custom-design integrated circuit performs the second frequency change at 455 kHz and frequency demodulation after limiting the signal to 4555 kHz.

The demodulated low-frequency signal is filtered, reshaped for the QPSK (IC16) or GMSK (IC22) MODEMs, then it is fed to the appropriate MODEM via the analog multiplxer circuit (IC10). Each MODEM extracts the bit clock and recovers the digital signal. This signal is fed to the microprocessor via the digital multiplexer circuit (IC14/IC15) as a function of the selected demodulation.

Then the data is relayed to the RS232 port. The microprocessor handles the command from the digital inputs. Some parameters (e. g. modulation type, frequency) are saved in the EEPROM (IC13).

The analog-to-digital converter (IC21) measures the reception level a the antenna, using the RSSI voltage from integrated circuit IC9. For an input level exceeding -60 dBm the measurement is irrelevant (above this level the RSSI voltage is no longer linear).

The DC/DC converter (IC24) converts the input voltage (10 to 16 V) to 5 volts, which is the power supply voltage required for the different circuits.



UHF Receiver Option Block Diagram

4.5 ADJUSTING THE RECEIVING BAND

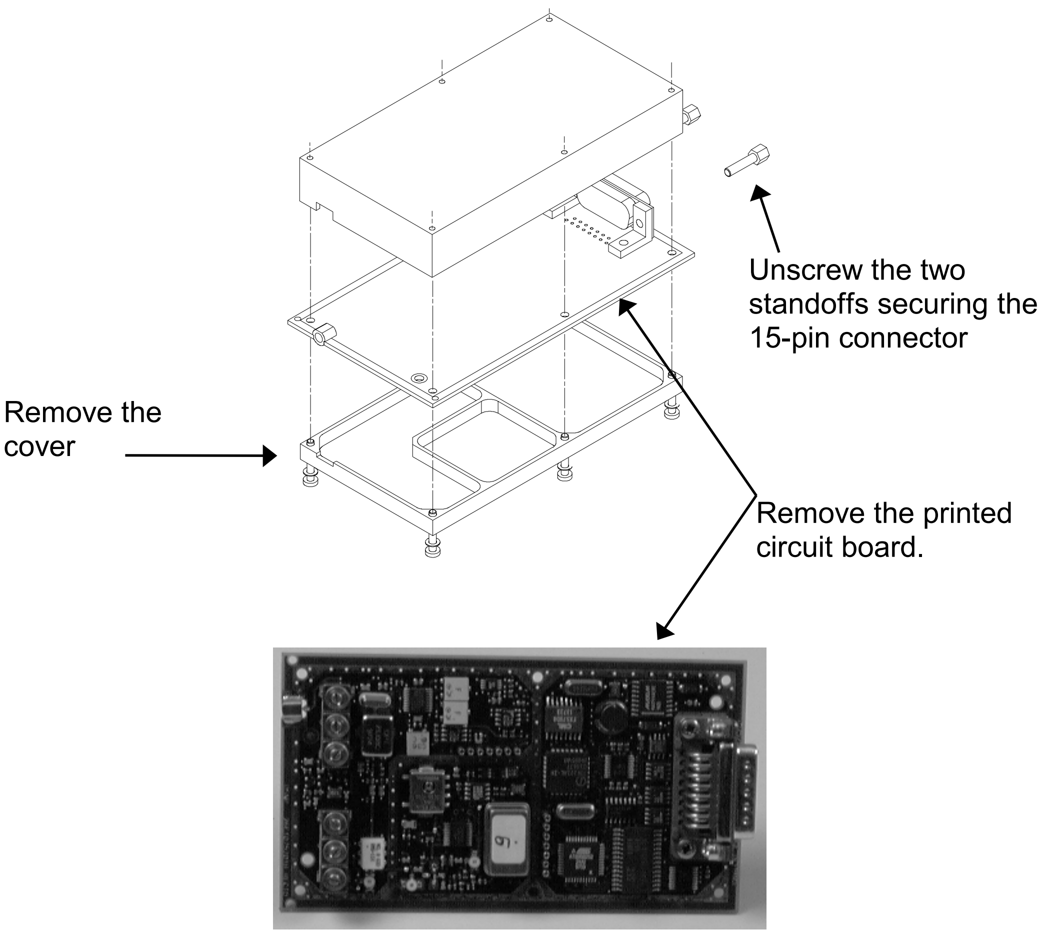
The total receiving band of the UHF receiver unit is from 407 MHz to 470 MHz but the input filters feature a bandwidth of 4 MHz on either side of the nominal frequency. The nominal frequency is factory-set to 446 MHz (unless otherwise specified by the user on ordering). As a result you will be able to use the receiver within a frequency band from 442 MHz to 450 MHz without any significant loss in sensitivity. Outside that band you should contact your vendor in order to re-adjust the operating frequency.

- **Equipment required:**
  - Scalar spectrum analyzer
  - 12 V, 0.5 A power supply
  - coaxial cables.

- **Adjustment procedure**

- Open the AQUARIUS unit as described above (see Par. 3.7)

- Remove the UHF receiver from the unit and

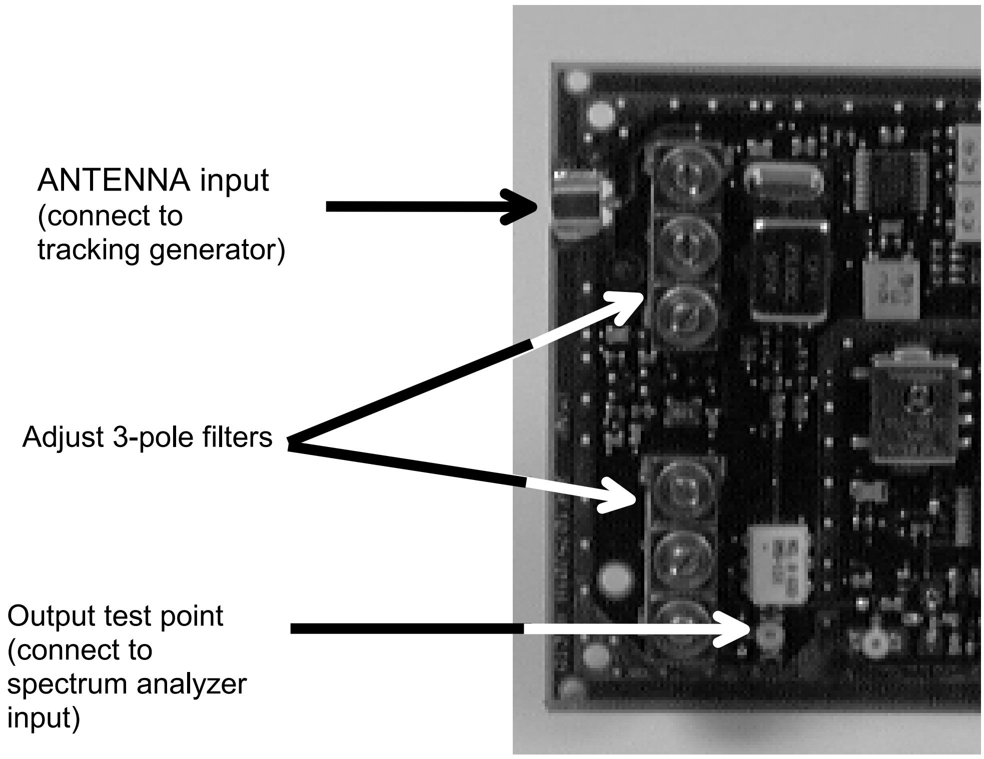


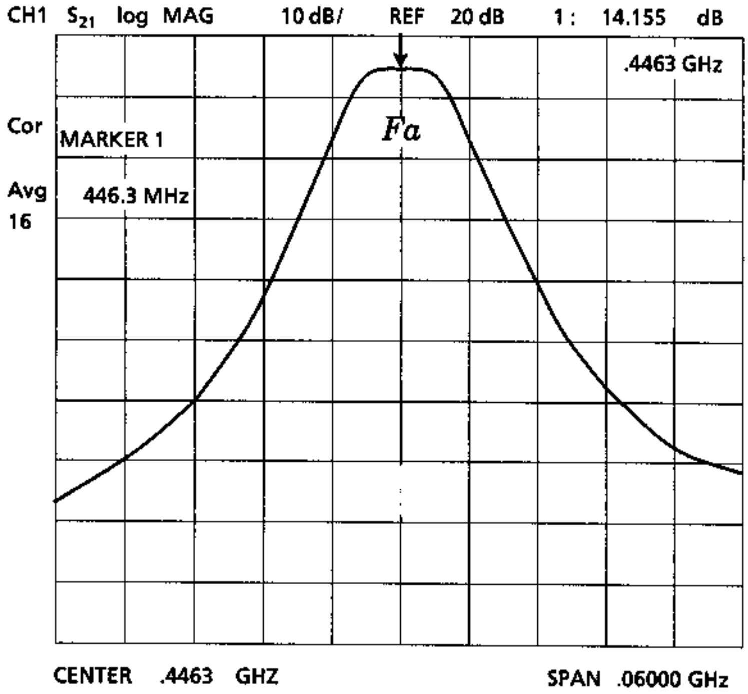
- Connect the signal generator to the UHF Antenna input and connect the spectrum analyzer to test point .

- Tune the generator and the spectrum analyzer to the desired nominal frequency.

- Feed a 12 VDC power supply voltage to the UHF receiver board, either via the AQUARIUS unit or using an external 12 V, 0.5 A power supply, with the + terminal connected to pin 8 of the 15-pin CANNON connector and the - terminal connected to pin 7 .

- Adjust the T1 and T3 three-pole filters (see below) for the maximum level on the spectrum analyzer display.





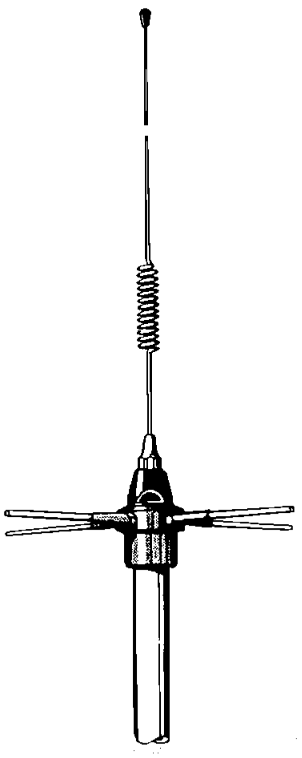
- Re-assemble the UHF receiver, complete with its cover, and put it back in place in the AQUARIUS unit. Take care to fully tighten the fixing screws in order to achieve proper grounding.

- Write down the new operating frequency on the receiver housing.

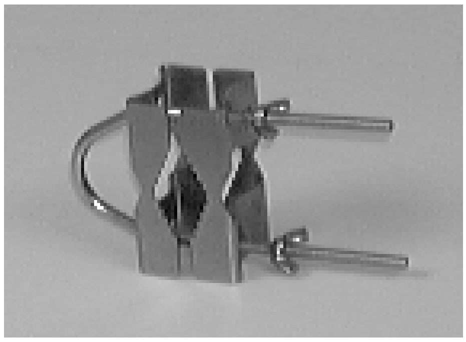
5. UHF TRANSMITTER OPTION (5001SD and 5002SK stations)

5.1 DESCRIPTION

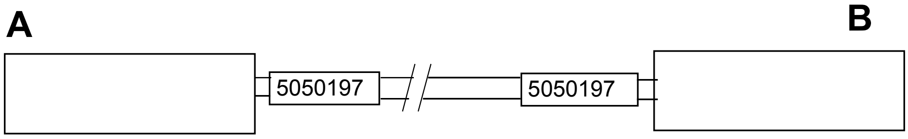
The UHF transmission optional module consists of the following elements:



UHF GP 450-3 PROCOM antenna (3310202).



Antenna mount bracket



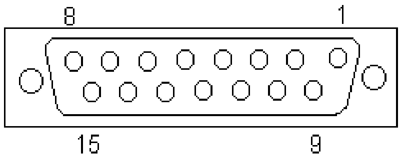
KX15 coaxial cable, 6.5 m long (50500197)

A and B are male N-type coax plugs (RADIALL R161062.000 type or equivalent)

5.2 INSTALLATION

The UHF transmitter module is fitted on the rear panel of the unit and secured by four screws. The power suppy voltage and commands are relayed through the 15-pin connector.

Pin number	Signal
1	TX+
2	TX-
3	RX+
4	RX-
5	
6	GND
7	GND
8	VIN+
9	
10	
11	SYNCHRO+
12	SYNCHRO-
13	PSLITE +
14	PSLITE -
15	



5.3 SOFTWARE SETUP

The Aquarius receiver automatically detects the presence of the UHF transmitting module. However, the way the UHF module is controlled and its data are transmitted depends on the software options enabled in the unit (e. g. Reference Station mode, etc.).

After you enable the desired options you can set the reception parameters in two ways:

- either using the reference station control software (see AQUARIUS User's Manual),
- or by sending the necessary remote control words, using a communications software such as WinComm. The following five commands are required:

  $PDAS, FIXMOD

  $PDAS, DGPDAT

  $PDAS, DGPS,MODE

  $PDAS, DGPS,STATION

  $PDAS, PREFLL

For detailed information on commands, see Par. 8.6.

5.4 THEORY OF OPERATION

The 1200 bits/s QPSK modulation is supported, for the sake of compatibility with earlier-design DSNP products. For the station to transmit the necessary messages for dual-frequency receivers, a faster modulation is used: GMSK modulation at 4800 Bauds (Gaussian Minimum Shift Keying).

The frequency bandwidth of the transmitter ranges from 410 MHz to 470 MHz in 12.5-kHz steps, complying with the ETS 300-113 standard. This requires a very high spectral purity of the UHF local oscillator and leads to the breaking off from the traditional technology relying on bipolar class-C operation power transistors.

For the spectral occupation requirements to be consistent with a bit rate of 4800 bits/s, a basic modulation needs to be employed. The GMSK modulation meets those requirements while achieving an excellent reception sensitivity.

The digital information from the GPS receiver is input to the micro-controller via the RS232 port. The micro-controller sets the data to the appropriate format for the enabled MODEM. It can use the PPS signal to trigger a modulo 1 second synchronous transmission. In addition, it processes the incoming remote control words from the RS232 ports (requests for version number, frequencies, modulation type, etc.). The parameters are saved in an EEPROM memory (IC19).The micro-controller sets the parameters of the MODEM and UHF synthesizer according to the transmission channel.

The digital multiplexer (IC9/IC22) relays the digital data to the enabled MODEM.

The filtered digital signal from the GMSK MODEM (IC21) is handled on two channels. One channel modulates the VCO (IC1) for the high frequencies and the other channel is fed to the VTCXO (MY1) for the low frequencies.

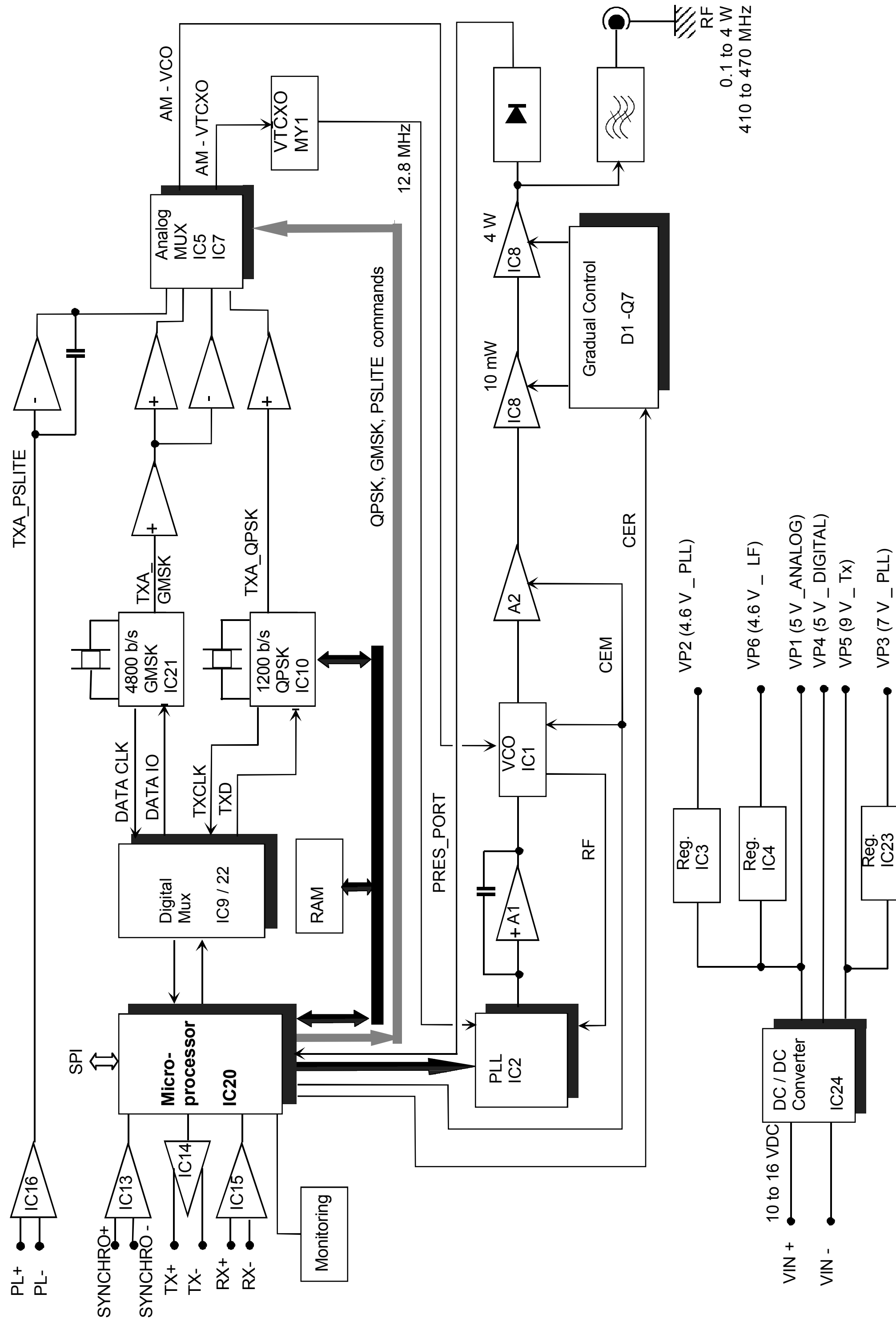
The QPSK modulation, with a 1.2 kHz subcarrier, is only fed to the VCO. The modulation type is selected by analog multiplexers IC5 and IC7.

The modulated UHF signal generated by the VCO is fed to amplifier A2 which provides 'anti-pulling' protection for the power stages. The power of the output stages is controlled by the circuit made up of D1/Q7. The output power of the power module is 4 W (+36 dBm). It is software-adjustable from 0.1 W to 4 W.

A diode (D2) detects the ouput signal to inform the microprocessor of the presence of a carrier frequency.

A high-cut filter made up of three pi networks is used to reject harmonics from the output stage.

The power supply voltages are generated by two DC/DC converters. From the 10 to 16 VDC input voltage IC24 generates the 9 volt (1.5 A) voltage required for the power module. The efficiency of this module is 90%. Q8 generates the logic and analog 5 volt power supply voltages.



UHF Transmitter Option Block Diagram

5.5 ADJUSTING THE OUTPUT POWER LEVEL

You may need to adjust the output power level in one of the following two cases:

- The output power level exceeds the maximum level allowed in the country to comply with the radio-communications regulations.
- The output power level is not linear over the whole frequency band.

• Equipment Required

- Wattmeter
- 15-pin extender cable)

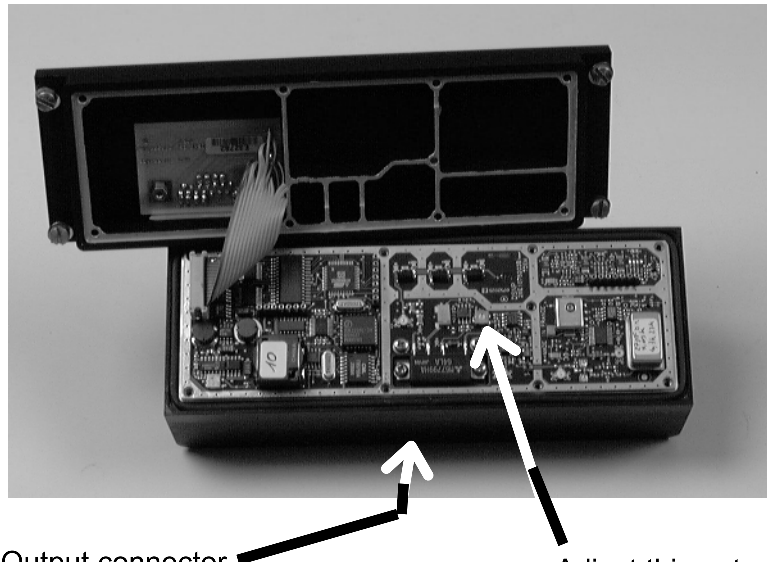
• Procedure

- Disconnect the UHF transmitter from the AQUARIUS unit

- Remove the rear cover.

- Connect the wattmeter to the output connector

- Apply the power supply to the transmitter (from the AQUARIUS unit via the extender cable). The AQUARIUS unit must have been configured to work at the desired operating frequency.



- Turn off the AQUARIUS unit.

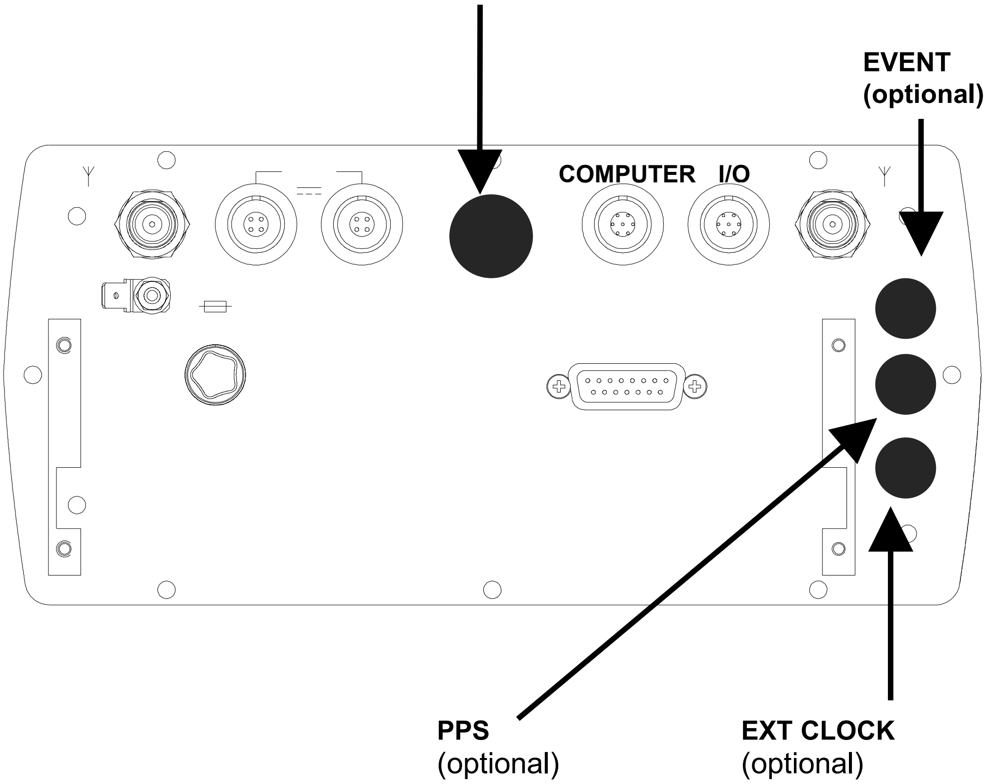
- Put the rear cover back in place on the UHF transmitter. See that the unit is properly sealed up (with the BS rings).

6. MISCELLANEOUS OPTIONS

6.1 RS422 PORT, PPS OUTPUT, EXTERNAL CLOCK

RS422 I/O
(optional)

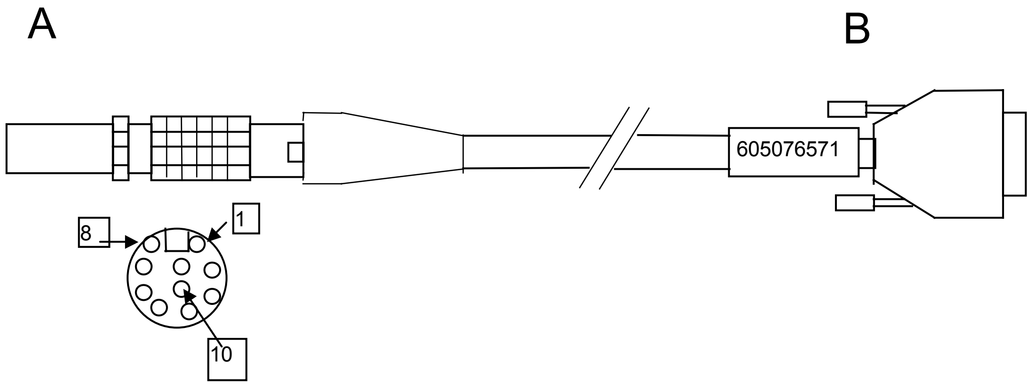
1	12 V
2	TX+
3	TX-
4	RX+
5	RX-
6	RTS+
7	RTS-
8	CTS+
9	CTS-
10	Ground



6.1.1 RS422 port

Basically, the unit is equipped with two RS232 ports. If required, for example in applications where the peripheral is far away from the receiver, an RS422 can be added. This requires that the unit be fitted with a different type of rear panel.

Cable Part number: 26E1076571



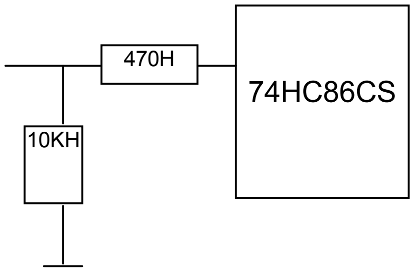
A is a 10-contact FGG-1K-310.CTAK (5011265) plug. Manufacturer: LEMO

B is a 15-contact female subD DA-15S (5010832) connector with metal cover DA121073-150 and 250-8501-013(2) (5080329). Manufacturer: CANNON.

Plug A	Plug B	
1	2	12V
2	10	RXD+
3	11	RXD-
4	14	TXD+
5	15	TXD-
6	8	CTS+
7	9	CTS-
8	13	RTS+
9	12	RTS-
10	3	GND

6.1.2 External event input (Trigger)

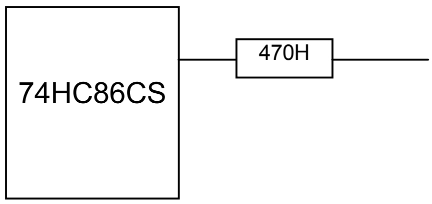
This input allows a time tag to be attached to an external event. The time tagging accuracy is ±100 ns (not including the error on the time solution). Time tagging is triggered on the high-to-low transition of the external signal.



6.1.3 PPS output

The PPS output is available as an option. It consists of a 1-Hz, 0-5 volt square wave, with a duty cycle of ½, available on the PPS connector on the rear panel.

The accuracy of the PPS signal is ± 10 ns (not including SA-induced error) on the high-to-low transition, after acquisition of the GNSS signals and computation of the position.



6.1.4 External Clock

This input is used to feed an accurate, stable frequency reference to the GNSS reception and acquisition circuitry if the master oscillator located on the GPS C/A or P/Y core module board fails to meet the accuracy requirements of any application. This mostly applies to reference receiver applications for stations generating DGPS corrections.

The presence of an external oscillator is detected automatically.

The specifications of the external oscillator should be better than those of the built-in master oscillator:

Frequency: 10 MHz
Stability: $\pm 2\times10^{-8}$ per day.
$\pm 2\times10^{-7}$ per year
Level: 0 DBm
Impedance: 50 ohm

6.2 PCMCIA MEMORY CARD.

The PCMCIA card is used for the following functions:

- Re-loading a GPS configuration (file generated by the Conf Pack configuration software).
- Loading new software releases to the unit.
- Recording the GPS raw data for post-processing.
- Storing stake-out points files or reference points files.
- Recording the computed data, in stake-out mode or in position logging mode.

The PCMCIA card consists of a flash memory, type I or II, agreeing with the ATA IDE standard, with a 3-volt or 5-volt programming voltage. The unit does not support earlier-design cards featuring a 12 volt programming voltage. The write-mode bit rate is 15 kbit/s or higher.

The writing and reading operations on the PCMCIA card are performed using the KISS application from the 3SPACK software or using the Con Pack configuration software. (See the USER'S MANUAL of the software used).

PCMCIA card available from DSNP:

Part number : 4660039

Type :

Size : 4 MB.

7. CONFIGURATION

As the basic unit has no built-in user interface but a simple readout used for monitoring and performance checks, the user has to resort to remote control words to select the desired data if he wishes to change the operating parameters (e. g. geodetic system, output messages, etc.). In cases where the remote control words have no effect on the contents of the messages or no parameters can be added, the Conf Pack configuration software has to be employed (see the corresponding manual).

Three types of configurations are saved in the unit:

- ***Default configuration***, resident in the firmware. This configuration cannot be modified. It resets all parameters in the unit to known values (operating mode, RS port configuration, output messages, etc.)
- ***Initial configuration***, saved in a non-volatile memory. It can be modified using the configuration software. It contains the necessary parameter settings for the reference configuration of an application or for any particular operating mode (mobile, reference station, etc.).
- ***Current configuration***, saved in a non-volatile memory. This configuration is modified by the operator's actions (through the human/computer interface or through remote control words).

The **Default configuration** can be loaded in place of the current configuration by pressing the pushbutton on the integrated display and holding it depressed, at power-on, or by sending the command **$PDAS,CONFIG,RESET**.

If, at power-on, a **PCMCIA** card containing a configuration file is found in the unit and you press the **SCROLL** button then the configuration file is loaded from the PCMCIA card to the **Current configuration**.

The command **$PDAS,CONFIG,INIT** can be used to load the **Initial configuration** in place of the current configuration.

8. PERFORMANCE CHECKS

8.1 TEST REMOTE CONTROL WORDS

8.1.1 Commmands format

The format of commands is based on the NMEA 0183 standard. The NMEA 0183 standard, originally designed for interfacing with radionavigation systems and maritime radiocommunications equipment, has been adopted by most GPS equipment manufacturers. It has been retained by international organizations:

- The complete NMEA 0183 standard has been included in the CEI 1162-1 standard by the Geneva international electrotechnology commission.
- The RTCM-SC104 committee retained the NMEA 0183 standard in defining the communications between the units of a DGPS reference and monitoring station, and it has been adopted by the USCG (US Coast Guards) for covering american inner and outer waters.
- In its July 1995 issue, under the title 'INNOVATION' the GPS WORLD magazine describes the NMEA 0183 standard as the standard for digital outputs in GPS receivers.
- The NMEA 0183 standard has already been used by DSNP for controlling the NDR104 UHF receiver.

Compliance with the NMEA 0183 standard, for both inputs and outputs, makes it easy for receivers to interface with any navigation equipment (Radar, track plotters, electronic charts, etc.).

The NMEA 0183 standard makes provision for three types of messages:

- ***APPROVED SENTENCES***: general-purpose, standard sentences with predefined contents, made up of an address field used to identify the unit that sent the sentence, and of the sentence as such.
- ***QUERY SENTENCES***: used to query the unit and read the current contents of a standard sentence.
- ***PROPRIETARY SENTENCES***: used for all operations not covered by standard sentences. The contents of proprietary sentences are defined by the owner. Proprietary sentences should include an address field with a 3-character code used to identify the sentence owner.

The identification code granted to DSNP by the NMEA Standards Committee, to identify DSNP proprietary sentences, is

DAS

• Format NMEA 0183 approved sentences

$aaccc,x.x,c--c,cc,xx,aa,hh,A,llll.ll,yyyyy.yy,hhmmss.ss*hh<CR><LF>

$	Sentence beginning
aa	Transmitter identification (e. g. GP for GPS receiver)
Ccc	Sentence identification (e. g. GLL for Geographic Latitude Longitude)
,x.x	Numeric field, integers or rational numbers, variable length
,h--h	Numeric field, hexadecimal characters, variable length
,c--c	Alphanumeric character field, variable length
,cc	Alphanumeric character field, fixed length
,xx	Numeric character field, fixed length
,aa	Alphabetic character field, fixed length
,hh	Hexadecimal character field, fixed length
,A	One character field
,llll.ll	Field dedicated to Latitude, variable or fixed length
,yyyyy.yy	Field dedicated to Longitude, variable or fixed length
,hhmmss.ss	Field dedicated to hours/minutes/seconds, variable or fixed length
*hh	Checksum field (optional)
	Hexadecimal encoding of "XOR" of the 8 data bits of all the characters in the sentence, from “$” to “*” NOT inclusive
<CR><LF>	Sentence end

Example :

$GPGLL,4716.01784,N,00129.83456,W,101430.3,A<CR><LF>

Geographic position	47º16.01784’ Noth
	1º29.83456’ West
	at 10:14:30.3”
	Valid Position

If any field has no data, the field is still present but it is blank.

Example with no data in the "Time" field:

$GPGLL,4716.01784,N,00129.83456,W,,A<CR><LF>

The data fields at the end of the sentence may be omitted.

Example with no "Time" and "Valid Position" fields:

$GPGLL,4716.01784,N,00129.83456,W<CR><LF>

- ## Format of NMEA 0183 Query sentences

The format of query sentences for approved sentences is as follows:

$aabbQ,ccc *hh<CR><LF>

$	Sentence beginning
Aa	Identification of querying transmitter (e.g. EC for Electronic system Card)
Bb	Identification of destination transmitter (e.g.: GP for GPS receiver)
Q	Identification of a query sentence
,ccc	Identification of the requested sentence (ex: GLL for Geographic Latitude Longitude)
*hh	Checksum field (optional)
<CR><LF>	Sentence end

Example :

$ECGPQ,GLL<CR><LF>

The reply to this query is the GPGLL approved sentence described above.

• Format of NMEA 0183 Proprietary

The format of DSNP proprietary sentences is compatible with that defined for the C/A and P/Y.Core Module.

$PDAS,c--c,...........................*hh<CR><LF>

$	Sentence beginning
P	Indicator for proprietary sentence
DAS	DSNP manufacturer code
,c--c	Identification of the proprietary sentence (2 to 6 alphanumeric characters)
,..........	Data field, with the same rules as for approved sentences whenever possible.
*hh	Checksum field (optional)
<CR><LF>	Sentence end

Example: PRANGE command in the Core Module (Configuration of pseudo-range output).

$PDAS,PRANGE,x,a,x,x.x,x,x.x,x.x,x.x<CR><LF>

See description in Par.8.38

With proprietary sentences, queries are made by simply sending the desired command with no parameters.

Examples :

$PDAS,PRANGE<CR><LF>

returns the scripts of all output messages.

$PDAS,PRANGE,2<CR><LF>

returns the script of pseudo-ranges output message No. 2.

8.1.2 Command summary table

Command	Function
$PDAS,AGECOR	Changes/reads maximum age of corrections
$PDAS,ALTI	Changes/reads altitude correction mode
$PDAS,COMMNT	Reads comment present in configuration
$PDAS,CONFIG	Reads data from the current configuration
$PDAS,CONFIG,INIT	Makes initial configuration the receiver's new current configuration
$PDAS,CONFIG,LOAD	Loads configuration from PCMCIA into the receiver to be its new initial configuration
$PDAS,CONFIG,READ	Reads data from initial configuration
$PDAS,CONFIG,RESET	Makes default configuration the receiver's new current configuration
$PDAS,DEFLT	Reports/acknowledges errors, if any
$PDAS,DELSES	Deletes the specified sessions
$PDAS,DGPS,DELSTA	Cancels a DGPS transmitting station in the receiver
$PDAS,DGPS,MODE	Controls DGPS transmit or receive channel
$PDAS,DGPS,STATION	Describes/lists DGPS transmitting stations
$PDAS,DGPDAT	Edits definitions of DGPS raw data outputs
$PDAS,EXPSES	Edits mode linked to sessions
$PDAS,FILTER	Edits speed filtering time constant
$PDAS,FIXMOD	Edits fix mode & associated reference station
$PDAS,GEO	Edits the coordinate system used
$PDAS,GEODAT	Edits definitions of WAAS/EGNOS data outputs
$__GPQ,GLL **$__GLL**	Edits estimated position
$PDAS,GNOS	Enables/disables operation with WAAS/EGNOS; also used to specify PRNs of GEOs tracked if chosen selection mode is “Manual”
$_ _GPQ,_ _ _	Returns the current value of the specified parameter (NMEA0183 compliant)
$PDAS,GPSDAT	Edits definition of GPS raw data outputs
$PDAS,HARDRS	Edits settings of serial ports
$PDAS,HEALTH	Edits health status of reference station
$PDAS,IDENT	Reads identification of harware and software parts
$PDAS,LOG	Edits or erases error reports
$PDAS,MEMORY	Reads amount of free memory space on PCMCIA card
$PDAS,MEMORY,DIR	Provides characteristics of files stored on PCMCIA card
$PDAS,NAVSEL	Edits the currently selected navigation mode
$PDAS,OUTMES	Edits definitions of computed data outputs
$PDAS,OUTON **$PDAS,OUTOFF**	Respectively enables and disables data outputs on the serial ports
$PDAS,PRANGE	Edits/adds definitions of pseudorange-data outputs
$PDAS,PREFLL	Changes/lists coordinates of reference station
$PDAS,PREFNE	Changes/lists projected coords of reference station

$PDAS,QC	Deals with Quality Control in the receiver
$PDAS,RAZALM	Deletes the specified type of almanac
$PDAS,RESET	Resets software
$PDAS,SELGEO	Edits the currently used coordinate system
$PDAS,SESSN	Edits/adds definitions of sessions
$PDAS,SVDSEL	Deals with rejected SVs & elevation threshold
$PDAS,TR	Triggers data output in RS232 mode on the specified port
$PDAS,UNIT	Edits receiver identification number
$PDAS,XSOFT **$PDAS,XSOFT,C** **$PDAS,XSOFT,E** **$PDAS,XSOFT,P**	Re-loads software into the unit
$__GPQ,ZDA **$__ZDA**	Respectively changes and reads receiver date & time

8.2 NMEA 0183 APPROVED SENTENCES

8.3 $_GLL and $_GPQ,GLL

- **Function**
  - Edits the estimated position used in the initial position-speed-time processing or displays the latest position solution.

- **Syntax**
  - Complete command:

$--GLL,a,b,c,d,e,f[*hh]<CR><LF>

  - QUERY command:

$--GPQ,GLL[*hh]<CR><LF>

- **Parameters**

ref.	format	
a	llll.lll	: Latitude of estimated position
b	a	: North or South latitude (N or S)
c	yyyyy.yyy	: Longitude of estimated position
d	a	: East or West Longitude (E or W)
e	hhmmss.ss	: UTC time
f	a	: Data status A: data valid V: data not valid
***hh**		: Checksum (optional)
<CR><LF>		: End of command

- **Examples**

$ECGPQ,GLL **QUERY**
$GPGLL,4716.091395,N,00129.463318,W,180449.00,A*14 (Reply)

$ECGLL,3940,N,00415,E **Initializing position**

$ECGPQ,GLL **QUERY**
$GPGLL,3940.000000,N,00415.000000,E,180731.00,A*06 (Reply)

8.4 $--GPQ,---

- **Function**

  - Returns the current values of the parameters whose generic code accompanies the command. All replies are compliant with the approved sentences of the NMEA 0183 standard (vers. 2.30, March 1, 1998).

- **Syntax**

$--GPQ,a[*hh]<CR><LF>

- **Parameters**

ref. format

a c--c : NMEA code corresponding to the parameters for which you want the receiver to return their current values. The codes list is given below (entry is also possible, in the NMEA standard, for underlined data).

ALM	GPS Almanac Data
DTM	<u>Datum Reference</u>
GGA	GPS Fix Data
GLL	<u>Geographic Position - Latitude/Longitude</u>
GRS	GNSS Range Residuals
GSA	<u>GNSS DOP and Active Satellites</u>
GST	GNSS Pseudorange Error Statistics
GSV	<u>GNSS Satellites in view</u>
RMC	Recommended Minimum Navigation Information
VTG	Course Over Ground and Ground Speed
ZDA	<u>Time & Date</u>

***hh** : Checksum (optional)

<CR><LF> : End of command

- **Examples**

$ECGPQ,DTM
$GPDTM,W84,,0.0000,N,0.0000,E,0.0,W84*6F

$ECGPQ,GLL
$GPGLL,4716.104353,N,00129.454296,W,134944.00,A*1F

$ECGPQ,GSA
$GPGSA,A,3,30,,,23,02,26,07,08,21,09,05,,,,,,2.5,1.3,-1.0*1F

$ECGPQ,GSV
$GPGSV,3,1,09,30,03,223,31,23,35,270,44,02,13,051,38,26,51,152,49*75
$GPGSV,3,2,09,07,33,077,43,08,37,278,45,21,23,309,42,09,74,306,49*7D
$GPGSV,3,3,09,05,44,217,48*4D

$ECGPQ,ZDA
$GPZDA,135127.00,8,12,1998,+00,00*7C

8.5 $_ZDA and $_GPQ,ZDA

- **Function**
  - Respectively changes and reads the receiver date & time.
- **Syntax**
  - Change command:

$--ZDA,a,b,c,d,e,f[*hh]<CR><LF>

  - Read command:

$--GPQ,ZDA[*hh]<CR><LF>

- **Parameters**

ref.	format		
a	hhmmss.ss	:	UTC time
b	xx	:	Day (01 to 31)
c	xx	:	Month (01 to 12)
d	xx	:	Year (4 char.)
e	xx	:	Local time offset (in hours) compared with UTC time (-13 to +13)
f	xx	:	Local time offset (in minutes) compared with UTC time (00 to 59)
***hh**		:	Checksum (optional)
<CR><LF>		:	End of command

- **Examples**

$ECGPQ,ZDA **QUERY**
$GPZDA,180919.00,17,2,1998,+00,00*78
(Reply)
$ECZDA,082100,18,12,1997,-1,00 **Changing time**
$ECGPQ,ZDA **Checking new time**
$GPZDA,082117.00,18,12,1997,-01,00*4B

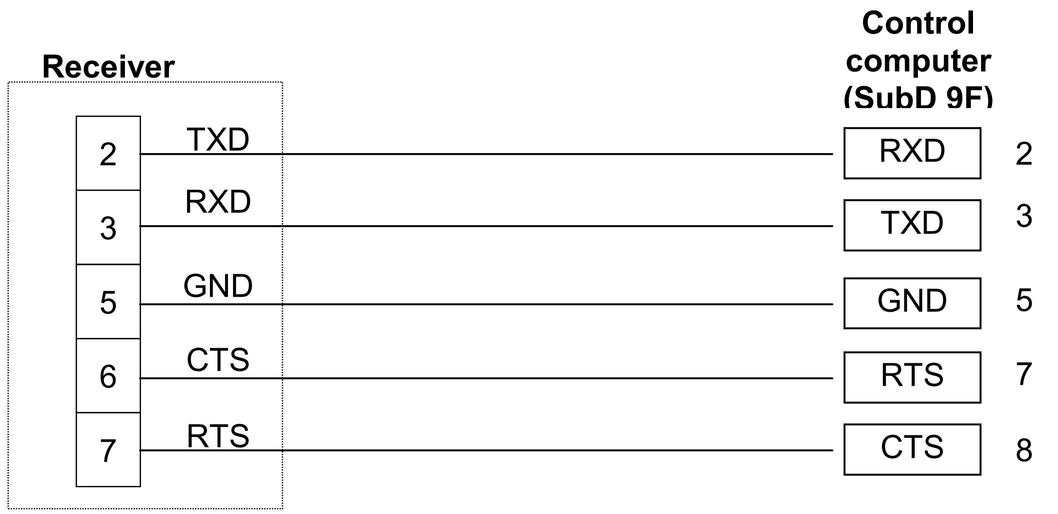
8.6 DSNP PRORIETARY COMMANDS

The DSNP proprietary commands are intended for the control and configuration of your DGNSS receiver. The commands are presented in alphabetical order, and described outside their use contexts (emphasis is placed on the syntax rather than on the use context).

For each command, the description plan is always the same: function, syntax, parameters, examples.

To apply commands to a DGNSS receiver an RS232 serial line is used between the control terminal (usually a PC computer) and the receiver (port A, COMPUTER). Commands are generated using the computer in terminal mode or better, using DSNP *Win Comm* software.

If for any reason you do not have the serial cable required, you can prepare a cable according to the following wiring instructions:

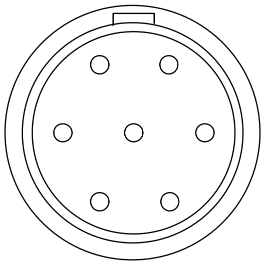


Pinout of COMPUTER connector on DGNSS receiver rear panel:

COMPUTER connector

(RS232 Port A)

type: JKX FD1G 07 MSSDSM, plug: JBX1 MPN, manufacturer: FCI,



Pin	Signal	
1	+12V	
2	TXD	(Output)
3	RXD	(Input)
4	Remote ON	
5	GND	
6	CTS	(Output)
7	RTS	(Input)

8.6.1 Format

- The format of all the commands available complies with the NMEA 0183 standard.
- DSNP was assigned a manufacurer code by the NMEA 0183 Committee for all its proprietary sentences (which we call here "commands").

  This code is "**DAS**".

  As a consequence, the first field in any DSNP proprietary command will therefore be:

$PDAS,

  where:

  - **$** indicates the beginning of a command

    P identifies a proprietary command

    DAS is our manufacturer code
- The beginning of any field is denoted by a comma (**,**). This character is the only way to detect and identify a new field.
- Most fields containing numerical data are of variable length.
- Although from version 2.1 of the NMEA standard the cheksum field is compulsory, it is optional in all DSNP proprietary sentences in order that commands can be sent from a simple, "non-intelligent" terminal or communications utility.
- When the checksum field is present and the test on this checksum fails, the command is rejected.
- Any command that you send can contain empty fields. If a field data is missing, it is assumed to keep its current value.

8.6.2 Conventions

The following symbols and conventions are used in the description of the DSNP proprietary commands:

Square-brackets []	:	used to bound optional parameters
x.x	:	designates the format of any numerical data, signed or not, with or without decimal point and decimal places, and with an integer part of variable length
a	:	designates a one-letter parameter (example: A)
x	:	designates the format of any numerical data which is necessarily an integer
xx	:	Numerical data, fixed length
c--c	:	Character string, variable length
cc	:	Character string, fixed length
a--a	:	Keyword
hhmmss.ss	:	Time
llll.lllll	:	latitude (ddmm.mmmmm)
yyyyy.yyyyyy	:	longitude (dddmm.mmmmmm)
[y]x	:	Field containing two one-figure parameters the first of which is optional

In the examples given at the end of each description, the following fonts are used:

- **Bold Times New Roman** for commands sent from the control computer
- Normal Times New Roman for replies to these commands returned by the DGNSS receiver.

8.7 $PDAS,AGECOR

- **Function**
  - Edits the maximum age permitted for DGPS corrections.
- **Syntax**
  - Complete command:

$PDAS,AGECOR,a[*hh]<CR><LF>

  - QUERY command:

$PDAS,AGECOR[*hh]<CR><LF>

- **Command identification**
  - $PDAS,AGECOR
- **Parameters**

ref.	format		
a	x.x	:	Maximum age of corrections, in seconds (default: 40 s)
***hh**		:	Checksum (optional)
<CR><LF>		:	End of command

- **Examples**

$PDAS,AGECOR	**Reading current max. age of corrections**
$PDAS,AGECOR,40*1E	**(40 seconds)**
$PDAS,AGECOR,50	**Changing max. age of corrections (50 s)**
$PDAS,AGECOR	**Checking new max. age of corrections**
$PDAS,AGECOR,50*1F	

8.8 $PDAS,ALTI

- **Function**
  - Edits the altitude processing mode and the altitude correction mode.
- **Syntax**
  - Complete command:

$PDAS,ALTI,a,b,c[*hh]<CR><LF>

  - QUERY command:

$PDAS,ALTI[*hh]<CR><LF>

- **Command identification**
  - $PDAS,ALTI
- **Parameters**

ref. format

a x : Altitude processing mode (0 to 3)

0: $H_{\text{user}}=H_{\text{WGS84 ellips}} - MSL_{\text{Stanag}} - EMSL_{\text{Local}} - offset$

1: $H_{\text{user}}=H_{\text{WGS84 ellips}} - EMSL_{\text{Local}} - offset$

2: Huser=Htransfo geoid $- EMSL_{\text{Local}} - offset$

3: $H_{\text{user}}=H_{\text{WGS84 ellips}} - MSL_{\text{User}} - EMSL_{\text{Local}} - offset$

where *MSL: Geoidal separation*

b	x.x	:	Offset altitude (from -999.999... to +999.999... m; default: 0.00 m). This parameter describes the height of the antenna phase center with respect to the reference surface.
c	x	:	$EMSL_{\text{Local}}$ altitude correction mode (0 to 9).
		0:	no altitude correction ($EMSL_{\text{Local}}$=0.0)
		> 0:	model used for altitude correction (for future applications)
***hh**		:	Checksum (optional)
<CR><LF>		:	End of command

- **Examples**

$PDAS,ALTI **Reading current correction mode**
$PDAS,ALTI,0,2.000,0*0A

$PDAS,ALTI,1,1.9,0 **Changing correction mode**
$PDAS,ALTI **Re-reading current correction mode**
$PDAS,ALTI,1,1.900,0*01

8.9 $PDAS,COMMNT

- **Functions**

  - Reads the "comment" field present in the current configuration (one or more lines). This field is assumed to identify the configuration.

- **Syntax**

$PDAS,COMMNT[*hh]<CR><LF>

- **Command identification**

  - $PDAS,COMMNT

- **Parameters**

(none)		
***hh**	:	Checksum (optional)
<CR><LF>	:	End of command

- **Examples**

$PDAS,COMMNT
$PDAS,COMMNT,2,1,AQUARIUS 5000 SERIES*14
$PDAS,COMMNT,2,2,DEFAULT CONFIGURATION*2B

8.10 $PDAS,CONFIG

- **Function**
  - Reads the data from the current configuration.
- **Syntax**

$PDAS,CONFIG[*hh]<CR><LF>

- **Command identification**
  - $PDAS,CONFIG
- **Parameters**

(none)

***hh**	:	Checksum (optional)
<CR><LF>	:	End of command

- **Examples**

$PDAS,CONFIG **Reading the data from the current configuration**

$PDAS,CONFIG,BEGIN,52*60 (Reply)
$PDAS,COMMNT,2,1,AQUARIUS 5000 SERIES*14
$PDAS,COMMNT,2,2,DEFAULT CONFIGURATION*2B
$PDAS,LANG,EN,f,1,1*43
$PDAS,AGECOR,040.0*31
$PDAS,ALTI,1,0.000,0*39
$PDAS,FILTER,6.00*1E
$PDAS,DOPMAX,40.0*13
$PDAS,SVDSEL,5.0,0*2A
$PDAS,SELGEO,0*21
...
$PDAS,CONFIG,END,0001C81B*66

8.11 $PDAS,CONFIG,INIT

- **Function**

  - Performs internal loading of the initial configuration so as to make it the receiver's new current configuration. The "current" configuration is referred to as the active configuration in the receiver.

    The receiver is automatically re-initialized after running this command.

- **Syntax**

  $PDAS,CONFIG,INIT[*hh]<CR><LF>

- **Command identification**

  - $PDAS,CONFIG,INIT

- **Parameters**

(none)		
***hh**	:	Checksum (optional)
<CR><LF>	:	End of command

- **Examples**

$PDAS,COMMNT
$PDAS,COMMNT,2,1,AQUARIUS 5000 SERIES*14
$PDAS,COMMNT,2,2,DEFAULT CONFIGURATION*2B
$PDAS,CONFIG,INIT
$PDAS,COMMNT
$PDAS,COMMNT,1,1,CONFIG PALMTOP*61

8.12 $PDAS,CONFIG,LOAD

- **Function**

  - Loads the configuration from the PCMCIA into the receiver so as to make it the receiver's new initial *and* current configurations.

    The file read from the PCMCIA card is necessarily CONFIG.CFG.

    The receiver is automatically re-initialized after running this command.

- **Syntax**

$PDAS,CONFIG,LOAD[*hh]<CR><LF>

- **Command identification**

  - $PDAS,CONFIG,LOAD

- **Parameters**

(none)		
***hh**	:	Checksum (optional)
<CR><LF>	:	End of command

- **Examples**

$PDAS,COMMNT
$PDAS,COMMNT,2,1,AQUARIUS 5000 SERIES*14
$PDAS,COMMNT,2,2,DEFAULT CONFIGURATION*2B
$PDAS,CONFIG,LOAD
$PDAS,COMMNT
$PDAS,COMMNT,1,1,CONFIG PALMTOP*61

8.13 $PDAS,CONFIG,READ

- **Function**
  - Reads the data from the initial configuration.
- **Syntax**

$PDAS,CONFIG,READ[*hh]<CR><LF>

- **Command identification**
  - $PDAS,CONFIG,READ
- **Parameters**

(none)

***hh**	:	Checksum (optional)
<CR><LF>	:	End of command

- **Examples**

$PDAS,CONFIG,READ
$PDAS,CONFIG,BEGIN,40*63 **(Reply)**
$PDAS,COMMNT,1,1,CONFIG PALMTOP*61
$PDAS,LANG,EN,f,1,1*43
$PDAS,AGECOR,040.0*31
$PDAS,ALTI,0,2.000,0*3A
$PDAS,FILTER,6.00*1E
$PDAS,DOPMAX,40.0*13
$PDAS,SVDSEL,5.0,0*2A
$PDAS,SELGEO,0*21
...
$PDAS,CONFIG,END,00015678*62

8.14 $PDAS,CONFIG,RESET

- **Function**

  - Performs internal loading of the default configuration so as to make it the receiver's new current configuration. The "current" configuration is referred to as the active configuration in the receiver.

    The receiver is automatically re-initialized after running this command.

- **Syntax**

  $PDAS,CONFIG,RESET[*hh]<CR><LF>

- **Command identification**

  - $PDAS,CONFIG,RESET

- **Parameters**

(none)		
***hh**	:	Checksum (optional)
<CR><LF>	:	End of command

- **Examples**

$PDAS,CONFIG,RESET

$PDAS,COMMNT
$PDAS,COMMNT,1,1,CONFIG PALMTOP*61
$PDAS,CONFIG,RESET
$PDAS,COMMNT
$PDAS,COMMNT,2,1,AQUARIUS 5000 SERIES*14
$PDAS,COMMNT,2,2,DEFAULT CONFIGURATION*2B

8.15 $PDAS,DEFLT

• Functions

- Reports the errors, if any, detected by the receiver. Errors are listed from the latest to the earliest.
- Can acknowledge these errors (they are then removed from the list), unless they are still persisting.

• Syntax

- Complete command:

$PDAS,DEFLT,a,b[*hh]<CR><LF>

- QUERY command:

$PDAS,DEFLT[*hh]<CR><LF>

• Command identification

- $PDAS,DEFLT

• Parameters

ref	format		
a	x.x	:	Error code to be listed (1 to 104) If **b** is absent and **a**=0: all errors, except those still persisting, are acknowledged
b	x.x	:	Error code to be acknowledged
***hh**		:	Checksum (optional)
<CR><LF>		:	End of command

- Receiver reply to a QUERY command:

$PDAS,DEFLT,A,B,C,D,E,F[*hh]<CR><LF>

reply data	format	
A	x	: Error code (0 to 100)
B	x	: Error extra-code (1 to 256)
C	a--a	: Keyword (TD, SYSTM, CONFG, POSIT, NAVIG, I/O, CM, IHM, DGPS, INTRF, GEODY, NONE)
D	x	: Day (1 to 31)
E	hhmmss.ss	: Time of first occurrence of the error
F	hhmmss.ss	: Time of latest occurrence of the error
***hh**		: Checksum (optional)
<CR><LF>		: End of command

- **Examples**

$PDAS,DEFLT **Listing all detected errors**
$PDAS,DEFLT,23,0,I/O,18,174909,174910*6C
$PDAS,DEFLT,24,0,I/O,18,174835,175045*6D
$PDAS,DEFLT,103,1,I/O,18,174827,174828*59
$PDAS,DEFLT,102,4,I/O,18,174827,174828*5D
$PDAS,DEFLT,8,1003,CM,18,174826,174827*49

$PDAS,DEFLT,103 **Reading error 103**
$PDAS,DEFLT,103,1,I/O,18,174827,174828*59

$PDAS,DEFLT,0 **Acknowledging all errors**

$PDAS,DEFLT **Re-listing errors**
$PDAS,DEFLT,24,0,I/O,18,174835,175045*6D
(error 24 persisting)

8.16 $PDAS,DELSES

- **Function**
  - Deletes the specified programmed sessions. All sessions invoked by the last EXPSES command run cannot be deleted.

- **Syntax**
  - Complete command:

$PDAS,DELSES,a,b,...[*hh]<CR><LF>

  - Command deleting all sessions that can be deleted:

$PDAS,DELSES,0[*hh]<CR><LF>

- **Command identification**
  - $PDAS,DELSES

- **Parameters**

ref	format	
a	x	: Number of the session first deleted
		a=0: all existing programmed sessions are deleted
b,...	x	: Number(s) of the session(s) next deleted, if any
***hh**		: Checksum (optional)
<CR><LF>		: End of command

• Examples

$PDAS,SESSN	**Listing the existing programmed sessions**
$PDAS,SESSN,1,111500.00,140000.00,1,ESSAI1*23 (reply)	
$PDAS,SESSN,2,154500.00,173000.00,1,ESSAI2*22	
$PDAS,SESSN,3,180000.00,203000.00,1,TEST1*73	
$PDAS,DELSES,1,3	**Deleting sessions 1 and 3**
$PDAS,SESSN	**Checking the new list of sessions**
$PDAS,SESSN,2,154500.00,173000.00,1,ESSAI2*22	
$PDAS,DELSES,0	**Deleting all sessions**
$PDAS,SESSN	**Checking the new list of sessions**
$PDAS,SESSN,0*6E	**(None left)**

8.17 $PDAS,DGPS,DELSTA

- **Function**
  - Erases a DGPS transmitting station in the receiver.
- **Syntax**
  - Complete command:

    $PDAS,DGPS,DELSTA,a,b,... [*hh]<CR><LF>

  - Shortened command (cancels all stations):

    $PDAS,DGPS,DELSTA[*hh]<CR><LF>

- **Command identification**
  - $PDAS,DGPS,DELSTA
- **Parameters**

ref.	format	
a	x	: Station number (0 to 1023)
b	x	: Station number (0 to 1023)
...	x	: Station number (0 to 1023)
***hh**		: Checksum (optional)
<CR><LF>		: End of command

• Examples

$PDAS,DGPS,STATION **(Listing all known stations)**
$PDAS,DGPS,STATION,1,LRK1,4716.28,N,00129.23,W,UHF,446532000.0,50.00,,,4800.0,GN*4E
$PDAS,DGPS,STATION,2,LRK2,4728.45,N,00148.19,W,UHF,446532000.0,45.00,,,4800.0,GN*42
$PDAS,DGPS,STATION,11,DSNP1,4710.00,N,00030.00,E,UHF,443550000.0,35.00,,,1200.0,DN*3B
$PDAS,DGPS,STATION,12,DSNP2,4630.00,N,00100.00,E,UHF,443550000.0,35.00,,,1200.0,DN*3A

$PDAS,DGPS,DELSTA,2,12 **(Deleting stations 2 and 12)**

$PDAS,DGPS,STATION **(Re-listing all known stations)**
$PDAS,DGPS,STATION,1,LRK1,4716.28,N,00129.23,W,UHF,446532000.0,50.00,,,4800.0,GN*4E
$PDAS,DGPS,STATION,11,DSNP1,4710.00,N,00030.00,E,UHF,443550000.0,35.00,,,1200.0,DN*3B

8.18 $PDAS,DGPS,MODE

(for receivers used as corrections generators and so connected to a transmitter)

- **Function**
  - Defines the receiver's serial port as a DGPS transmit channel.
- **Syntax**
  - Complete command:

$PDAS,DGPS,MODE,a,b,E,e,f[*hh]<CR><LF>

  - QUERY command:

$PDAS,DGPS,MODE,a[*hh]<CR><LF>

  - QUERY command (all lines are read):

$PDAS,DGPS,MODE[*hh]<CR><LF>

- **Command identification**
  - $PDAS,DGPS,MODE

• Parameters

ref.	format	
a	x	: Line number (1 to 3)
b	a	: Port identification (A, B, etc.)
E	a	: "E" for "Transmitter". The other setting (R) for this third parameter is discussed in the next command description
d	x	: Transmitter identification number, as referenced in $PDAS,STATION. If **d** is omitted, corrections are simply made available on the specified port (no transmitter control provided)
e	x.x	: Transmission programming (1): 0: free mode 1 to 6: transmission rate in seconds (synchronous mode)
	x.x	: In synchronous mode (**e**=1 to 6), **f** is the transmit slot number (1 to 6)
***hh**		: Checksum (optional)
<CR><LF>		: End of command

• Examples

$PDAS,DGPS,STATION **(Listing all known stations)**
$PDAS,DGPS,STATION,1,LRK1,4716.28,N,00129.23,W,UHF,446532000.0,50.00,,,4800.0,GN*4E
$PDAS,DGPS,STATION,11,DSNP1,4710.00,N,00030.00,E,UHF,443550000.0,35.00,,,1200.0,DN*3B

$PDAS,DGPS,MODE,1,D,E,1,3,2 **(Writing description line 1)**

- According to this description line (line 1), the receiver will transmit corrections via transmitter No. 1 in slot 2, at a transmit format of 3 seconds
- (No receiver reply)

$PDAS,DGPS,MODE **(Listing all the description lines)**
$PDAS,DGPS,MODE,1,D,E,1,3,2*05
$PDAS,DGPS,MODE,2,N*79
$PDAS,DGPS,MODE,3,N*78

$PDAS,DGPS,MODE,1,D,E,11,0 **(Re-programming line 1)**

- According to this line (line 1), the receiver will transmit corrections via transmitting station No. 11 in free-running mode.
- (No receiver reply)

$PDAS,DGPS,MODE,1 **(Checking the content of line 1)**
$PDAS,DGPS,MODE,1,D,E,11,3,2*34

8.19 $PDAS,DGPS,MODE

(for receivers processing corrections received from a reference station — via a transmitter).

- **Function**
  - Defines the receiver's serial port as a DGPS receive channel.
- **Syntax**
  - Complete command:

    $PDAS,DGPS,MODE,a,b,R,d,e,f,g,h,i,j[*hh]<CR><LF>

  - QUERY command (only the specified line is read):

    $PDAS,DGPS,MODE,a[*hh]<CR><LF>

  - QUERY command (all lines are read):

    $PDAS,DGPS,MODE[*hh]<CR><LF>

- **Command identification**
  - $PDAS,DGPS,MODE
- **Parameters**

ref.	format	
a	x	: Line number (1 to 3)
b	a	: Port identification (A, B, etc.)
R	a	: "R" for "Receiver". The other setting (E) for this third parameter is discussed in the previous command description
d	x	: Transmitter identification number, as referenced in $PDAS,STATION. If **d** is omitted, corrections are simply allowed to be fed to the specified port (no receiver control provided)
e	x.x	: (Empty field)
f	x.x	: (Empty field)
g	x.x	Identification of the reference station from which corrections should be processed in priority (0 to 1023). If **g** is omitted, received corrections are processed without checking the reference station number

h	x.x	Identification of the reference station from which corrections should be processed in second priority (0 to 1023), optional
i	x.x	Identification of the reference station from which corrections should be processed in third priority (0 to 1023), optional
h	x.x	Identification of the reference station from which corrections should be processed in fourth priority (0 to 1023), optional
***hh**		: Checksum (optional)
<CR><LF>		: End of command

• Examples

$PDAS,DGPS,STATION **(Listing all known stations)**
$PDAS,DGPS,STATION,11,DSNP1,4710.00,N,00030.00,E,UHF,443550000.0,35.00,,,1200.0,DN*3B
$PDAS,DGPS,STATION,12,DSNP2,4630.00,N,00100.00,E,UHF,443550000.0,35.00,,,1200.0,DN*3A

$PDAS,DGPS,MODE,1,D,R,11,,,11,12 **(Writing description line 1)**

- According to this line (line 1), the receiver will receive (R) corrections via its port D from transmitter No. 11. These corrections will be generated by reference stations Nos. 11 and 12.

$PDAS,DGPS,MODE,1 **(Checking the content of description line 1)**
$PDAS,DGPS,MODE,1,D,R,11,,,11,12*21 **(Reply)**

$PDAS,DGPS,MODE,2,B,R,,,,712,713 **(Writing line 2)**

- According to this line (line 2), the receiver will receive (R) corrections from an external receiver (4th field blank) via its port B. These corrections will be generated by stations Nos. 712 and 713.

$PDAS,DGPS,MODE **(Listing all description lines)**
$PDAS,DGPS,MODE,1,D,R,11,,,11,12*21 **(Reply)**
$PDAS,DGPS,MODE,2,B,R,,,,712,713*26
$PDAS,DGPS,MODE,3,N*78

8.20 $PDAS,DGPS,STATION

• Functions

- Allows you to enter the complete description (including decryption code C3) of each of the usable reference stations.
- Allows you to list the description of each of them (or all of them).

• Syntax

- Complete command:

$PDAS,DGPS,STATION,a,b,c,d,e,... n[*hh]<CR><LF>

- QUERY command (only the specified station is reported):

$PDAS,DGPS,STATION,a[*hh]<CR><LF>

- QUERY command (all stations are listed):

$PDAS,DGPS,STATION[*hh]<CR><LF>

• Command identification

- $PDAS,DGPS,STATION

• Parameters

ref.	format	
a	x	: Transmitter identification number (0 to 1023)
b	c--c	: Transmitter name (12 char. max.)
c	llll.ll	: Reference latitude
d	a	: North or South latitude (N or S)
e	yyyyy.yy	: Reference longitude
f	a	: East or West longitude (E or W)
g	c--c	: Band of first transmission frequency (UHF)
h	x.x	: First transmission frequency, in Hz
i	x.x	: Range, in km
j	c--c	: Band of second transmission frequency (for future development)
k	x.x	: Second transmission frequency, in Hz (for future development)
l	x.x	: Baud rate (1200 or 4800 Bd)

m	cc	:	Character string containing the following information: Modulation type: D for DQPSK, G for GMSK Encrypted/non-encrypted corrections: C for encrypted, N for non-encrypted
n	c--c	:	If encrypted corrections, decryption code C3 (for future development)
***hh**		:	Checksum (optional)
<CR><LF>		:	End of command

- ## Examples

$PDAS,DGPS,STATION **(Listing all known stations)**
$PDAS,DGPS,STATION,,NONE*56 (Reply: none)

$PDAS,DGPS,STATION,1,LRK1,4716.28,N,00129.23,W,UHF,446532000,50,,,4800,GN
$PDAS,DGPS,STATION,11,DSNP1,4710,N,00030,E,UHF,443550000,35,,,1200,DN

$PDAS,DGPS,STATION **(Re-listing all known stations)**
$PDAS,DGPS,STATION,1,LRK1,4716.28,N,00129.23,W,UHF,446532000.0,50.00,,,4800.0,GN*4E
$PDAS,DGPS,STATION,11,DSNP1,4710.00,N,00030.00,E,UHF,443550000.0,35.00,,,1200.0,DN*3B

8.21 $PDAS,DGPDAT

- **Function**
  - Edits the definitions of the DGPS raw data outputs.
- **Syntax**
  - Complete command:

$PDAS,DGPDAT,a,b,c,d,e,f,... [*hh]<CR><LF>

  - QUERY command:

$PDAS,DGPDAT,a[*hh]<CR><LF>

- **Command identification**
  - $PDAS,DGPDAT
- **Parameters**

ref.	format	
a	x	: Output number (1 or 2) If **a**=0, all description lines are cleared
b	a	: Output port identification (A, B, etc.)
c	x	: Output mode (0 to 3) 0: stopped 1: time 2: trigger 3: immediate
d	x.x	: Output rate If **c**=1 (time), **d** is the data output rate expressed in units of 0.1 seconds If **c**=2 (trigger), then: **d**=1, next data block following EVENT is output **d**=3, next data block following 1PPS is output
e	x	: Data type (x) 1: RTCM-SC104 2: LRK 3: DSNP UHF 4: SVAR 5: Relayed user data

f, g,... x	: If **e**=1, RTCM messages of the type **f, g,...** are generated
	: If **e**=3, DSNP UHF messages of the type **f, g,...** are generated
***hh**	: Checksum (optional)
<CR><LF>	: End of command

- **RTCM correction types**

Type		Corrections
1 and 9	:	Corrections
2	:	Correction Delta
3	:	Parameters of the reference station
5	:	Constellation Health
16	:	Mail
18	:	Carrier phase measurement (for future development)
19	:	Code phase measurement (for future development)

- **DSNP UHF correction types**

Type		Corrections
1	:	Code corrections (type C)
2	:	Phase corrections (type P)

- **Examples**

$PDAS,DGPDAT **Listing DGPS raw data outputs**
$PDAS,DGPDAT,1,N*57 **(Reply: none)**
$PDAS,DGPDAT,2,N*54

$PDAS,DGPDAT,1,D,1,10,3,1,2 **Defining DGPS raw data output 1:**
- To port D (to UHF transmitter), "Time" output mode, 1-sec. output rate, DSNP UHF data, type C and P

$PDAS,DGPDAT,1 **Checking definition of output 1**
$PDAS,DGPDAT,1,D,1,10,3,1,2*71

$PDAS,DGPDAT,2,A,1,100,4 **Defining DGPS raw data output 2:**
- To port A, "Time" output mode, 10-sec. output rate, SVAR data

$PDAS,DGPDAT **Listing definitions of outputs 1 & 2**
$PDAS,DGPDAT,1,D,1,10,3,1,2*71
$PDAS,DGPDAT,2,A,1,100,4*43

- If a display terminal is connected to port A (this may be the terminal from which you sent the preceding commands), then data blocks of the following type are now received:

```
!D,945,329190.1
%R,14,,0
*3,5.9,0.33,0.0,201
*17,8.0,-0.19,0.0,183
*19,32.2,-0.28,0.0,224
*21,-40.5,0.14,0.0,204
*22,-2.6,-0.39,0.0,51
*23,-17.9,0.51,0.0,75
*27,-23.3,-0.22,0.0,228
*31,29.8,0.12,0.0,153
*15,12.5,0.13,0.0,50
```

$PDAS,DGPDAT,2,A,1,50,1,2,3,5,9,16 **Re-defining output 2:**

- To port A, "Time" output mode, 5-sec. output rate, RTCM-SC104 data, messages Nos 2, 3, 5, 9, 16

$PDAS,DGPDAT **Re-listing definitions of outputs 1 & 2**

$PDAS,DGPDAT,1,D,1,10,3,1,2*71

$PDAS,DGPDAT,2,A,1,50,1,2,3,5,9,16*54

- Again, if a display terminal is connected to port A, then data blocks of the following type are now received:


```
fAC\fEr~fRXnzdUO|orxDs~ICSnYOnY^}cTzCiXaOOu{MouRjpL@]ZPN@CzPM@mI_puAOulCosdYn}cp

ET{bo{|}Ym[qfLi@Dp{\GpzWyC@KsMfQB\jEXsb_DCBey[pfLZGDDbxOEhFL_L_fQB\OzoB]IDCbpZLL

YsOGNDDGpzW\t^LdYn}cpy_tbIDCbVcpfLRGMDQGpzWy[AIswYn}cUFhG]@DCbcXTMIss`cWJgxOEhFX

]vLJfQB\jy[pbj{|m_cgpvLY_bdFnxOEhF`lpLQfQB\OF\@]w{|m}y[svLy`MXe`xOEX]WNwL~
```

8.22 $PDAS,EXPSES

- **Function**
  - Edits the operating mode linked to sessions.
- **Syntax**
  - Complete command:

$PDAS,EXPSES,a,b,c,d,... [*hh]<CR><LF>

  - QUERY command:

$PDAS,EXPSES[*hh]<CR><LF>

- **Command identification**
  - $PDAS,EXPSES
- **Parameters**

ref.	format		
a	c--c	:	Session type: END: All sessions disabled, or ends the immediate session ON: Programmed sessions are run once CYCLE: Programmed sessions will be repeated, if this may happen IMMED: Will cause the receiver to start operating and recording data on PCMCIA (if option installed) on reception of the command.
b	c--c	:	Power control type: AUTPW: Receiver power supply controlled by the programmed sessions MANPW: Receiver power supply controlled only by the operator
c	x	:	Number of the session to be run first. Sessions are numbered from 1 to 8 corresponding to the description lines in which they are defined (see SESSN command)
d,...	x	:	Number of the session to be run in the second, third,... place (up to 8 sessions in sequence)
***hh**		:	Checksum (optional)
<CR><LF>		:	End of command

• Information supplement

The reply to the Query command will vary with time, depending first on the progress of the sessions, but also on the operating options chosen with this command.

- With ON selected as the session type, session numbers will disappear after the corresponding sessions have been run.
- With CYCLE selected as the session type, sessions will be run in the specified order. After a session has been run, the number corresponding to that session will be moved to the end of the session sequence.
- With AUTPW selected as the power control type, the receiver will be turned on 5 minutes before the beginning of the session, and turned off 30 seconds after the end of the session.
- With MANPW selected as the power control type, the receiver will be turned on 5 minutes before the beginning of the first session, unless it is already ON at that time. From that moment, the receiver will operate continuously until somebody turns it off.
- A receiver cannot be turned off with the ON/OFF button if a session (with/without recording) is in progress.
- The SCROLL front panel indicator will blink if a session is pending (with ON or CYCLE selected). It will be lit while the session is being run.

• Examples

$PDAS,EXPSES	**QUERY**
$PDAS,EXPSES,END,MANPW*28	**No session pending**
$PDAS,EXPSES,IMMED	**Running an immediate session**
$PDAS,EXPSES	**QUERY**
$PDAS,EXPSES,IMMED,MANPW*2F	**Immediate session in progress**
$PDAS,EXPSES,END	**Ending an immediate session**
$PDAS,EXPSES	**QUERY**
$PDAS,EXPSES,END,MANPW*28	

$PDAS,SESSN	**Listing the programmed sessions**

$PDAS,SESSN,1,103000.00,120000.00,1,RECORD1*63
$PDAS,SESSN,2,140000.00,153000.00,1,RECORD2*60
$PDAS,SESSN,3,070000.00,200000.00,0, *78

$PDAS,EXPSES,ON,MANPW,1,2	**Validating sessions 1 and 2 (in this order), run once (ON), manual power control (MANPW)**
$PDAS,EXPSES	**QUERY (checking the validated sessions)**

$PDAS,EXPSES,ON,MANPW,1,2*65

$PDAS,EXPSES,CYCLE,AUTPW,3	**Cycling on session 3, in auto power mode (AUTPW)**
$PDAS,EXPSES	**QUERY (checking the validated session)**

$PDAS,EXPSES,CYCLE,AUTPW,3*2A

8.23 $PDAS,FILTER

- **Function**
  - Edits the speed filtering time constant.
- **Syntax**
  - Complete command:

    $PDAS,FILTER,a[*hh]<CR><LF>

  - QUERY command:

    $PDAS,FILTER[*hh]<CR><LF>

- **Command identification**
  - $PDAS,FILTER
- **Parameters**

ref.	format	
a	x.x	: Speed filtering time constant (default: 6 seconds)
***hh**		: Checksum (optional)
<CR><LF>		: End of command

- **Examples**

$PDAS,FILTER	**QUERY**
$PDAS,FILTER,6.000000*1E	**(Reply:6 seconds)**

8.24 $PDAS,FIXMOD

• Function

- Edits the fix mode and the associated DGPS reference station or WAAS/EGNOS GEO.

  About the selection of the associated reference station, this command will require prior execution of $PDAS,DGPS,MODE.
- Wherever a reference position is required (for example at a reference station or for KART or LRK initialization), use the $PDAS,PREFLL or $PDAS,PREFNE command to enter that position.

• Syntax

- Complete command:

$PDAS,FIXMOD,a,b,c,... [*hh]<CR><LF>

- QUERY command:

$PDAS,FIXMOD[*hh]<CR><LF>

• Command identification

- $PDAS,FIXMOD

• Parameters

ref.	format	
a	x	: Selects GPS fix mode: 0: no fix computation 1: Residuals computation in transmitting reference station mode 2: Residuals computation in monitoring reference station mode (*for future use*) 3: "Straight" GPS fix mode 4: Single-station DGPS fix mode 5: Multiple-station DGPS fix mode (MDGPS) *(for future use)* 6: EDGPS fix mode (Enhanced DGPS) 7: KART or LRK fix mode (with OTF initialization) 8: KART or LRK fix mode (with Static initialization) 9: KART or LRK fix mode (with Z-FIXED initialization) 10: KART or LRK fix mode (initialization from a known point)

b	x	: Selects the source of corrections (none, from WAAS/EGNOS or from DGPS reference station) and simultaneously, enables or disables the use of pseudoranges from the selected GEO (if any): 0: Straight GPS without WAAS/EGNOS pseudoranges 1: DGPS/KART/LRK without WAAS/EGNOS pseudoranges 2: WADGPS without WAAS/EGNOS pseudoranges 10: Straight GPS with WAAS/EGNOS pseudoranges 11: DGPS/KART/LRK with WAAS/EGNOS pseudoranges 12: WADGPS with WAAS/EGNOS pseudoranges
c,...	x	: If **b**=1 or 11, **c,...**: Identification(s) of DGPS reference station(s)
		: If **b**=2 or 12, **c**: PRN No. of WAAS/EGNOS GEO If **c** is not mentioned, then the corrections data from the closest WAAS/EGNOS GEO is used
***hh**		: Checksum (optional)
<CR><LF>		: End of command

- **Examples**

$PDAS,FIXMOD	**QUERY**
$PDAS,FIXMOD,3,1*39	**(Reply: "Straight" GPS fix mode, DGPS station)**
$PDAS,FIXMOD,1,1	**Changing fix mode**
$PDAS,FIXMOD	**QUERY**
$PDAS,FIXMOD,1,1*3B	**(Reply: transmitting reference station)**
$PDAS,FIXMOD,4,1,12	**Changing fix mode**
$PDAS,FIXMOD	**QUERY**
$PDAS,FIXMOD,4,1,12*11	**(Reply:Single-station DGPS mode, DGPS station No. 12)**
$PDAS,FIXMOD,4,12,128	**Changing fix mode. WADGPS selected using GEO PRN No. 128, WAAS/EGNOS pseudoranges involved in position processing**

8.25 $PDAS,GEO

• Function

- Edits the characteristics the specified coordinate system (datum & projection).
- Lists the characteristics of all or specified coordinate systems.

• Syntax

- Complete commands:

$PDAS,GEO,a,b,c,d [*hh]<CR><LF>
$PDAS,GEO,a,b,e,f [*hh]<CR><LF>
$PDAS,GEO,a,b,A,1/F,S,j [*hh]<CR><LF>
$PDAS,GEO,a,b,Dx,Dy,Dz,n [*hh]<CR><LF>
$PDAS,GEO,a,b,Ax,Ay,Az,r [*hh]<CR><LF>
$PDAS,GEO,a,b,s,t [*hh]<CR><LF>
$PDAS,GEO,a,b,u,v,w,... [*hh]<CR><LF>

- QUERY command:

$PDAS,GEO,e[*hh]<CR><LF>

• Command identification

- $PDAS,GEO

• Parameters

ref.	format	
a	x.x	: Count of lines required to describe the specified coordinate system
b	x.x	: Number of the present line
c	x.x	: GPS week number (optional)
d	x.x	: GPS time within week, in sec. (optional)
e	x.x	: Coordinate system number (0 to 10) (default: 0)
f	c--c	: Coordinate system name (10 characters max.)
A	x.x	: Semi-major axis ("A," placed before)
1/F	x.x	: Inverse flattening ("1/F," placed before)
S	x.x	: Scale factor ("S," placed before)
j	x	: Unit code (see table below)

Dx	x.x	: X deviation ("Dx," placed before)
Dy	x.x	: Y deviation ("Dy," placed before)
Dz	x.x	: Z deviation ("Dz," placed before)
n	x	: Unit code (see table below)
Ax	x.x	: X angular deviation ("Ax," placed before)
Ay	x.x	: Y angular deviation ("Ay," placed before)
Az	x.x	: Z angular deviation ("Az," placed before)
r	a	: Unit code (see table below)
s	x.x	: Projection number (1.. 99)
t	c--c	: Projection name (12 characters max.)
u,...		: Projection parameters
...		
***hh**		: Checksum (optional)
<CR><LF>		: End of command

- **Examples**

$PDAS,GEO,2
$PDAS,GEO,8,1,0,0*6E
$PDAS,GEO,8,2,02,NTF*33
$PDAS,GEO,8,3,A,6378249.145,1/F,293.465000,S,1.00000000,1*1E
$PDAS,GEO,8,4,Dx,-168.000000,Dy,72.000000,Dz,318.500000,1*4F
$PDAS,GEO,8,5,Ax,0.000000,Ay,0.000000,Az,0.554000,e*03
$PDAS,GEO,8,6,02,LambII*49
$PDAS,GEO,8,7,Lori,0.81681408993,Gori,0.04079233948,Eori,600000.000,Nori,20
0000.
000,d1*11
$PDAS,GEO,8,8,Ko,0.999877420*6A

8.26 $PDAS,GEODAT

• Function

- Edits the definitions of the SBIN@W or SVAR@W data outputs. This type of data is received from WAAS/EGNOS GEOs.
- Adds new definitions of SBIN@W or SVAR@W data outputs.

• Syntax

- Complete command:

$PDAS,GEODAT,a,b,c[*hh]<CR><LF>

- QUERY command (all output definitions are returned):

$PDAS,GEODAT[*hh]<CR><LF>

- QUERY command (only the specified output is returned):

$PDAS,GEODAT,a[*hh]<CR><LF>

• Command identification

- $PDAS,GEODAT

• Parameters

ref.	format	
a	x.x	: Output number (1, 2, etc.) (default: 1)
b	a	: Output port identification (A, B, etc.)
c	x	: Data output control: 0: No output (invalidated) 1: Output of SBIN@W data, at regular intervals of time 3: Output of SVAR@W data, at regular intervals of time
***hh**		: Checksum (optional)
<CR><LF>		: End of command

- ## Examples

$PDAS,GEODAT,1	**QUERY (about output 1 definition)**
$PDAS,GEODAT,1,A,1	**(Reply: output 1 enabled, delivers SBIN@W data on port A)**
$PDAS,GEODAT,1,A,0	**Invalidates output 1** **(no reply)**

8.27 $PDAS,GNOS

• Functions

- Enables/disables the tracking of the WAAS or EGNOS satellite
- Specifies the way WAAS/EGNOS GEOs should be selected by the receiver (Auto/Manual).
- Provides the receiver with the PRNs of the GEOs to be used in case of Manual selection.

• Syntax

- Complete command:

$PDAS,GNOS,a,b,c[*hh]<CR><LF>

- QUERY command:

$PDAS,GNOS[*hh]<CR><LF>

• Command identification

- $PDAS,GNOS

• Parameters

ref.	format	
a	x	: Controls the tracking of the WAAS/EGNOS system in the receiver and the way the receiver selects GEOs (possible values for **a**: 0, 1 or 2; default:1): 0: Use of WAAS/EGNOS disabled 1: Automatic selection of the WAAS/EGNOS GEO: the receiver will be allowed to choose the GEOs with which to work (nothing then needs to be specified in fields **b** and **c**). 2: Manual selection of the WAAS/EGNOS GEOs: the receiver will work with the GEOs whose PRNs are specified in fields **b** and **c** below.
b	a	: If **a**=2, **b** is the PRN of the 1st WAAS/EGNOS GEO to be tracked (120 ≤ **b** ≤ 138) (irrelevant for the other values of **a**)
c	a	: If **a**=2, **c** is the PRN of the 2nd WAAS/EGNOS GEO to be tracked (120 ≤ **b** ≤ 138) (irrelevant for the other values of **a**). See also comments below.
***hh**		: Checksum (optional)
<CR><LF>		: End of command

- ## Examples

$PDAS,GNOS	**QUERY**
$PDAS,GNOS,0	**Reply: Use of WAAS/EGNOS currently disabled**
$PDAS,GNOS,1	**Command allowing the use of the WAAS or EGNOS system; GEOs are selected automatically by the receiver**
$PDAS,GNOS,2,122	**Command allowing the use of the WAAS or EGNOS system; The selected GEO is PRN 122 (Manual selection mode).**

- ## Comments

Although in the *Aquarius* receiver, two channels can be reserved for the reception of one GEO each, we do not recommend users to select two GEOs in the case of Manual selection.

In future, users should be allowed to run such a command in which two GEOs are selected manually:

$PDAS,GNOS,2,122,138

8.28 $PDAS,GPSDAT

• Functions

- Edits the definitions of the GPS raw data outputs.
- Adds new definitions of GPS raw data outputs.

• Syntax

- Complete command:

$PDAS,GPSDAT,a,b,c,d,e,f[*hh]<CR><LF>

- QUERY command (all output definitions are returned):

$PDAS,GPSDAT[*hh]<CR><LF>

- QUERY command (only the specified output is returned):

$PDAS,GPSDAT,a[*hh]<CR><LF>

• Command identification

- $PDAS,GPSDAT

• Parameters

ref.	format	
a	x	: Output number (1 or 2)
b	a	: Output port identification (A, B, etc.)
c	x	: Ephemeris data output: 0: none 1: at regular intervals, in SBIN@E binary format 2: on request, in SBIN@E binary format 3: at regular intervals, in SVAR!E format 4: on request, in SVAR!E format
d	x	: Almanac data output: 0: none 1: at regular intervals, in SBIN@A binary format 2: on request, in SBIN@A binary format 3: at regular intervals, in SVAR!A format 4: on request, in SVAR!A format

e	x	: Iono-UTC data output: 0: none 1: at regular intervals, in SBIN@U binary format 2: on request, in SBIN@U binary format 3: at regular intervals, in SVAR!U format 4: on request, in SVAR!U format
f	x	: Health & A/S data output: 0: none 1: at regular intervals, in SBIN@S binary format 2: on request, in SBIN@S binary format 3: at regular intervals, in SVAR!S format 4: on request, in SVAR!S format
***hh**		: Checksum (optional)
<CR><LF>		: End of command

- **Examples**

$PDAS,GPSDAT	**QUERY**
$PDAS,GPSDAT,1,B,-3,-3,-3,-3*4C	
$PDAS,GPSDAT,2,N*43	**(Reply: a single output defined, output 1, on port B, all GPS data blocks programmed in this output are invalidated)**

$PDAS,GPSDAT,1,B,3,3,3,3	**Validating GPS data blocks in output 1**
$PDAS,GPSDAT	**QUERY**
$PDAS,GPSDAT,1,B,3,3,3,3*4C	**(Reply: 2 lines)**
$PDAS,GPSDAT,2,N*43	

$PDAS,GPSDAT,2,A,0,0,4,0	**Adding output 2 on port A (iono-utc data)**

- If a display terminal is connected to port A (this may be the terminal from which you sent the preceding commands), then data blocks of the following type are now received:

```
!U,945,378367.0
780F00,FF0136,FEFC03,000032,000000,0F90B1,0C9002,0CAAAA
```

8.29 $PDAS,HARDRS

- **Function**
  - Edits the settings of the receiver's serial ports.
- **Syntax**
  - Complete command:

    $PDAS,HARDRS,a,b,c,d,e,f,g[*hh]<CR><LF>

  - QUERY command:

    $PDAS,HARDRS[*hh]<CR><LF>
- **Command identification**
  - $PDAS,HARDRS
- **Parameters**

ref.	format	
a	x	: Count of lines containing definitions of serial ports
b	x	: Line number (from 1 to **a**)
c	a	: Port identification (A, B, etc.)
d	x.x	: Baud rate (1200, 2400, 4800, 9600, 19200) (default: 9600 Bd)
e	x	: Number of data bits (6, 7, 8) (default: 8)
f	x.x	: Number of stop bits (1, 1.5, 2) (default: 2)
g	a	: Parity control ("N" for None, "O" for Odd, "E" for Even, "M" for Mark, "S" for Space) (default: N)
***hh**		: Checksum (optional)
<CR><LF>		: End of command

• Examples

$PDAS,HARDRS **QUERY**
$PDAS,HARDRS,6,1,A,9600,8,1.0,N*08
$PDAS,HARDRS,6,2,B,9600,8,1.0,N*08
$PDAS,HARDRS,6,3,C,9600,8,2.0,N*0B
$PDAS,HARDRS,6,4,D,19200,8,1.0,N*3D

$PDAS,HARDRS,,,B,19200,7,1,0 **Changing port B settings**

$PDAS,HARDRS **QUERY**
$PDAS,HARDRS,6,1,A,9600,8,1.0,N*08
$PDAS,HARDRS,6,2,B,19200,7,1.0,N*33
$PDAS,HARDRS,6,3,C,9600,8,2.0,N*0B
$PDAS,HARDRS,6,4,D,19200,8,1.0,N*3D

8.30 $PDAS,HEALTH *(for future use)*

- **Function**
  - Edits the health status of the reference station (information delivered at a monitoring station).
- **Syntax**
  - Complete command:

    $PDAS,HEALTH,a[*hh]<CR><LF>

  - QUERY command:

    $PDAS,HEALTH[*hh]<CR><LF>

- **Command identification**
  - $PDAS,HEALTH

Parameters

ref.	format	
a	x	: Health status (0 to 7) (default: 6 or 7) RTCM-SC104 Health conventions: 7: station not working 6: station not monitored 5: UDRE scale factor is 0.1 4: UDRE scale factor is 0.2 3: UDRE scale factor is 0.3 2: UDRE scale factor is 0.4 1: UDRE scale factor is between 0.5 & 0.75 0: UDRE scale factor is 1
***hh**		: Checksum (optional)
<CR><LF>		: End of command

- **Examples**

$PDAS,HEALTH	**QUERY**
$PDAS,HEALTH,0*2A	**(Reply)**
$PDAS,HEALTH,6	**Initializing health status for a working station**
$PDAS,HEALTH	**QUERY**
$PDAS,HEALTH,6*2C	**(Reply)**

8.31 $PDAS,IDENT

- **Function**

  - Reads the identification of each of the hardware and software parts in the receiver.

- **Command syntax (a QUERY command only)**

$PDAS,IDENT[*hh]<CR><LF>

- **Receiver Reply syntax**

$PDAS,IDENT,a,b,c,d[*hh]<CR><LF>

- **Parameters returned in the reply**

ref.	format	
a	x.x	: Total count of reply lines
b	x.x	: Line number
c	cccc	: Subassembly hardware identification Always 4 characters: **c1**, **c2**, **c3**, **c4** where:

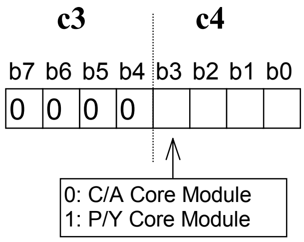
➪ **c1c2** are the 2 characters identifying the subassembly:

c1c2 = CM ⇒ Core Module

c1c2 = TD ⇒ Data Transmission

c1c2 = UC ⇒ Application Central Unit

➪ **c3c4** are the 2 characters identifying the hardware version of the subassembly:

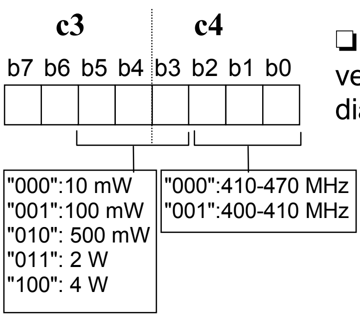


❑ If **c1c2** = CM, then **c3** identifies the type of board:

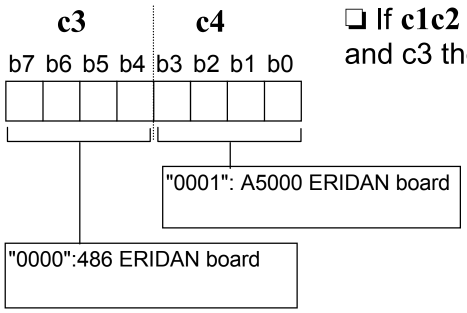
c3 = 0 ⇒ Core Module Type I

and **c4** represents the 4 bits read from the IC device

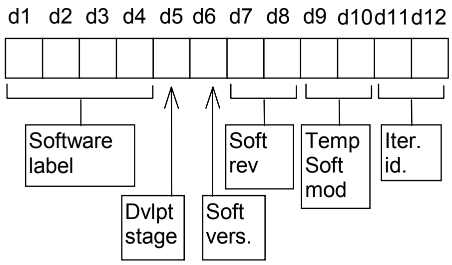
If **c3** = 0, see in the opposite diagram how to interpret bit b3 in character **c4**



❑ If **c1c2** = TD, bits 2, 1 and 0 (character **c4**) identify the PCB version and bits 5, 4 and 3 the transmission power (see opposite diagram).



❑ If **c1c2** = UC, c4 identifies the PCB version of the Central Unit and c3 the type of Central Unit (see opposite diagram)



d	cc	: Subassembly software identification (always 10 or 12 characters) (see diagram just above)

➪ **d1** to **d4**: Software label:
- CMBL: Core Module Boot Loader
- CMCA: C/A L1 Core Module
- CMPY: C/A & P/Y L1 & L2 Core Module
- UCBS: BIOS application
- UCBL: Application Boot Loader
- UCBN: "Black Box" Application
- EUHF: UHF transmitter (data link)
- RUHF: UHF receiver (data link)

➪ **d5**: Development stage (B for β-test version, V for production version, X for development version)

➪ **d6**: Identification of software version or standard:
- 0: S0, S0+, S0.2 and S0.3 (Core Module)
- 1: E1 state (application)
- 2: E2 state (application)
- 3: E3 state (application)

➪ **d7** & **d8**: Revision of the software version

➪ **d9** & **d10**: Temporary or On-Site software modification

➪ **d11** & **d12**: Iteration identification (optional, applies to CM only)

***hh** : Checksum (optional)

<CR><LF> : End of reply

- ## Examples

Query:

$PDAS,IDENT

Reply from a dual-frequency station:

$PDAS,IDENT,6,1,TD20,EUHFV10300
$PDAS,IDENT,6,2,CM08,CMPYV0020107
$PDAS,IDENT,6,3,CM08,CMBLV0020107
$PDAS,IDENT,6,4,UC01,UCBNV12000
$PDAS,IDENT,6,5,UC01,UCBLV10000
$PDAS,IDENT,6,6,UC01,UCBSV20000

8.32 $PDAS,LOG

- **Function**

Edits or erases the error reports or anomaly context reports recorded in the log file.

- **Syntax**

$PDAS,LOG,a,b,...[*hh]<CR><LF>

QUERY command:

[*hh]<CR><LF>

edits the entire log file.

- **Parameters**

Ref	Format		
,a	**x.x**	:	0 erases all the data or the list specified
			other than 0 : type of recording requested
***hh**		:	Checksum (optional)
<CR><LF>		:	End of command

- **Command identification**

$PDAS,LOG

- **Examples**

$PDAS,LOG	**Reads all the data**
$PDAS,DEFAULT,...	**(Reply)**
$PDAS,LOG,0	**Erases all the data**
$PDAS,LOG,3	**Requests the recording of data levels 1,2 and 3.**

8.33 $PDAS,MEMORY

- **Function**

  - Reads the amount of free memory space on the PCMCIA card (result returned in bytes).

- **Syntax**

$PDAS,MEMORY[*hh]<CR><LF>

- **Receiver Reply syntax**

$PDAS,MEMORY,DIR,a,b,c,d,e,f[*hh]<CR><LF>

- **Reply Parameters**

ref.	format	
a	x	: Total count of reply lines
b	x	: Reply line number
c	a	: PCMCIA card label (**c** omitted if no card)
d	x	: Total count of bytes used
e	x	: Total count of free bytes
***hh**		: Checksum (optional)
<CR><LF>		: End of reply

- **Examples**

$PDAS,MEMORY **QUERY**
$PDAS,MEMORY,DIR,1,1,,122880,3858432,0

$PDAS,MEMORY **QUERY**
$PDAS,MEMORY,DIR,1,1,0 (no PCMCIA inserted)

8.34 $PDAS,MEMORY,DIR

- **Function**
  - Provides the characteristics of a file (or of all files) stored on the PCMCIA card.
- **Syntax**

$PDAS,MEMORY,DIR,a[*hh]<CR><LF>

- **Receiver Reply syntax**

$PDAS,MEMORY,DIR,a,b,c,d,e,f[*hh]**<CR><LF> **$PDAS,MEMORY,DIR,a,b,g,h,i,j,k,l[*hh]<CR><LF> **...** **$PDAS,MEMORY,DIR,a,b,g,h,i,j,k,l[*hh]**<CR><LF>

- **Reply Parameters**

ref.	format	
(first line)		
a	x	: Total count of reply lines
b	x	: Number of reply line (1 necessarily)
c	a	: PCMCIA card label (**c**=0 if no card)
d	x	: Total count of bytes used (omitted if no card)
e	x	: Total count of free bytes (omitted if no card)
f	x	: Total count of files (omitted if no card)

ref.	format		
(next lines)			
a	x	:	Total count of reply lines
b	x	:	Reply line number
g	a	:	File name
h	x	:	File size, in bytes
i	xx		File creation day (dd)
j	xx		File creation month (mm)
k	xx		File creation year (yyyy)
l	hhmmss.s		File creation time
***hh**		:	Checksum (optional)
<CR><LF>		:	End of reply

- **Examples**

Query:

$PDAS,MEMORY,DIR

Reply without PCMCIA card:

$PDAS,MEMORY,DIR,1,1,0

Reply with a PCMCIA card inserted (example):

$PDAS,MEMORY,DIR,3,1,,122880,3858432,2
$PDAS,MEMORY,DIR,3,2,sesimmed.d00,69444,18,02,1998,092034.0
$PDAS,MEMORY,DIR,3,3,sesimmed.d01,30304,18,02,1998,092230.0

8.35 $PDAS,NAVSEL

- **Function**
  - Edits the navigation mode currently selected.
- **Syntax**
  - Complete command:

$PDAS,NAVSEL,a,b,c,d[*hh]<CR><LF>

  - QUERY command:

$PDAS,NAVSEL[*hh]<CR><LF>

- **Command identification**
  - $PDAS,NAVSEL
- **Parameters**

ref.	format		
a	x	:	Type of fix used for navigation (1 to 4) 1: (D)GPS or WADGPS 2: EDGPS 3: KART A 4: KART R
b	x	:	Navigation mode used (1 to 4) 1: Position (none) 2: Heading (future use) 3: Heading along a specified direction (future use) 4: Route (future use)
c	c--c	:	Navigation instructions (8 characters max.) If **b**= 2 or 3, **c**=label of target waypoint If **b**= 4, **c**=label of the route to follow
d	x	:	Direction of travel along the route If **d**= 1, direct If **d**= 0, inverse
***hh**		:	Checksum (optional)
<CR><LF>		:	End of command

• Examples

$PDAS,NAVSEL	**QUERY**
$PDAS,NAVSEL,1,1*29	**(DGPS, Position mode)**
$PDAS,NAVSEL,2	**Changing fix used for navigation**
$PDAS,NAVSEL	**QUERY**
$PDAS,NAVSEL,2,1*2A	**(EDGPS, Position mode)**

8.36 $PDAS,OUTMES

- **Functions**
  - Edits the definitions of the computed-data outputs.
  - Adds new definitions of computed-data outputs.
- **Syntax**
  - Complete command:

$PDAS,OUTMES,a,b,c,d,e[,...,n][*hh]<CR><LF>

  - QUERY command:

$PDAS,OUTMES,a,b[*hh]<CR><LF>

- **Command identification:**
  - $PDAS,OUTMES
- **Parameters**

ref	format	
a	x.x	: message number (0 to 20)
b	a	: Port identification (A, B, C, D, P)
c	x	: Trigger mode (-8 to 8): 0: disables the output (no possibility to know what the former trigger mode was for this output, as opposed to the "-" sign; see below) 1: Time 2: External Event is the triggering signal 3: (*reserved*) 4: 1pps is the triggering signal 5: Manual (future development) 6: by $PDAS,TR command 7 & 8: for future development A negative value will cause the output to be disabled (but the trigger mode information will still be present in the output definition for further use)
d	x.x	: if **c**=1, then **d** is the trigger rate expressed in 100-ms units if **c**=2 or 4, then **d** is the trigger rate expressed in count of events
e, ..., n	x.x	: numbers of the formats which will generate the message being defined
***hh**		: Checksum (optional)
<CR><LF>		: End of command

• Examples

$PDAS,OUTMES **Querying the receiver to obtain the list of its computed data outputs**

$PDAS,OUTMES,1,A,-1,10,1,5,7,8,9,10,20*4F
$PDAS,OUTMES,2,A,-1,10,2,7,8,5,11,20*5B
$PDAS,OUTMES,3,A,-1,10,12,20*51
$PDAS,OUTMES,4,A,-1,10,13,20*57
$PDAS,OUTMES,5,A,-1,10,3,5,6,20*64
$PDAS,OUTMES,6,A,-1,10,4,5,11,7,8,14,20*70
$PDAS,OUTMES,7,B,-2,1,15,20,16,20,17,20,18,20,19*4C

$PDAS,OUTMES,2,B,4 **Changing output 2**
$PDAS,OUTMES,2 **Checking new output 2**
$PDAS,OUTMES,2,B,4,10,2,7,8,5,11,20*70

$PDAS,OUTMES,2,B,-4 **Invalidating output 2 (trigger information setting preserved)**
$PDAS,OUTMES,2 **Checking output 2**
$PDAS,OUTMES,2,B,-4,10,2,7,8,5,11,20*5D

$PDAS,OUTMES,2,B,0 **Stopping output 2 (trigger information setting lost)**
$PDAS,OUTMES,2 **Checking output 2**
$PDAS,OUTMES,2,B,0,10,2,7,8,5,11,20*74

$PDAS,OUTMES,8,C,1,20,1,5,7,8,9,10,20 **Creating output 8**
$PDAS,OUTMES,8 **Checking output 8**
$PDAS,OUTMES,8,C,1,20,1,5,7,8,9,10,20*6A

$PDAS,OUTMES,1,A,1,100 **Creating/Validating output 1 on the port through which this command is sent**

Data blocks, such as those below, are returned to your display terminal:

```
$GPGGA,104849.99,4716.12353,N,00129.44097,W,0,09,1,85.99,M,0.00,M,-1.0,0000
$GPGGA,104859.99,4716.12259,N,00129.43925,W,0,09,1,80.49,M,0.00,M,-1.0,0000
$GPGGA,104909.99,4716.12146,N,00129.43786,W,0,09,1,75.71,M,0.00,M,-1.0,0000
$GPGGA,104919.99,4716.12013,N,00129.43679,W,0,09,1,71.56,M,0.00,M,-1.0,0000
$GPGGA,104929.99,4716.11865,N,00129.43614,W,0,09,1,68.01,M,0.00,M,-1.0,0000
$GPGGA,104939.99,4716.11713,N,00129.43585,W,0,09,1,65.04,M,0.00,M,-1.0,0000
```

8.37 $PDAS,OUTON and $PDAS,OUTOFF

- **Functions**
  - Respectively enables/disables data outputs on the planned serial ports

    These commands have no effect on the port currently used as far as the dialog between PC and receiver is concerned.

- **Syntax**
  - Output disabling command:

$PDAS,OUTOFF[*hh]<CR><LF>

  - Output (re)enabling command:

$PDAS,OUTON[*hh]<CR><LF>

- **Parameters**

(none)

***hh**	:	Checksum (optional)
<CR><LF>	:	End of command

- **Examples**

$PDAS,OUTOFF	**All data outputs suspended** **(No reply)**
$PDAS,OUTON	**All data outputs resumed** **(No reply)**

8.38 $PDAS,PRANGE

- **Functions**
  - Edits the definitions of the pseudorange-data outputs.
  - Adds definitions of pseudorange-data outputs.
- **Syntax**
  - Complete command:

$PDAS,PRANGE,a,b,c,d,e,f,g,h[*hh]<CR><LF>

  - QUERY command (only the specified line is returned):

$PDAS,PRANGE,a<CR><LF>

  - QUERY command (all output definitions are returned):

$PDAS,PRANGE<CR><LF>

- **Command identification:**
  - $PDAS,PRANGE
- **Parameters**

	format	
a	x	: Output number (1 or 2)
b	a	: Output port identification (A, B, C, etc.)
c	x	: Output mode (1, 2, etc.) 0: stopped 1: time 2: trigger
d	x.x	: Output rate: if **c**=1, output rate in units of 0.1 second if **c**=2: 1: Data block following External Event is output 3: Data block following 1pps is output

e [y]x : Data type (1 to 5) (*see Par. 9.1 and 9.2*):

1: SBIN@r
2: SBIN@R
3: SBIN@Q
4: SVAR!R
5: SVAR!Q

and [y]: Multi-Core data (0 or 1)

0:Master Core Module (default)
1:All Core Modules (future use)

f x.x : GPS carrier/code filtering time constant, in seconds (0 to 600)

g x.x : WAAS carrier/code filtering time constant, in seconds (0 to 600)

h x.x : Pseudolite carrier/code filtering time constant, in seconds (0 to 600)

***hh** : Checksum (optional)

<CR><LF> : End of command

- **Examples**

$PDAS,PRANGE **QUERY**
$PDAS,PRANGE,1,B,-1,10,4,0,0,0*4F **(Reply, 2 lines)**
$PDAS,PRANGE,2,N*59

$PDAS,PRANGE,1,A,1,40 **Validating SVAR!R data blocks on port A, in time mode every 4 seconds**

Data blocks, such as the one below, are returned to your display terminal (if you sent the command through port A):

```
!R,945,393528.0
&P,30
*0,4,3118942645,2536448,2407692,38,A,70,BF,-16,-100,,0,,83,EF
*1,5,3019470900,3296845,2821288,47,2,00,8F,-27,-43,-119,6985938,2198376,01,7F
*2,14,3154649176,3909466,1131044,26,A,9C,FF,-500,,,0,,00,00
*3,8,2993264420,3359515,3040848,49,2,01,8F,-35,-2,-81,6011982,2369436,01,7F
*4,9,3111375648,3690470,4080896,42,2,21,AF,-35,-36,4,9813601,3179876,41,AF
*6,29,3056856053,7688995,-162032,45,2,23,9F,-20,-28,-139,1925823,-126280,22,9F
*7,25,3045055618,2261374,-2272900,46,2,16,9F,-31,-124,-231,7378134,-1771120,22,8F
*8,30,2968080800,5981519,718444,50,2,01,8F,-35,43,-76,9875600,559820,01,6F
*9,1,3035062400,3442959,2495792,45,2,03,9F,-39,-197,-244,9392251,1944616,21,9F
*A,6,3002300530,8653059,-1492552,49,2,00,8F,-31,-63,-139,7324873,-1163000,02,6F
*B,24,3062858884,5679502,186896,44,2,00,AF,4,-6,-29,1506867,145640,42,9F
```

8.39 $PDAS,PREFLL

- **Functions**

  - In a reference station, allows you to enter the precise latitude and longitude of this station.
  - In a mobile receiver, allows you to enter the precise latitude and longitude of the position from which the receiver will be initialized.

    In both cases, the command is used to enter a reference position. See also $PDAS,FIXMOD.

- **Syntax**

  - Complete command:

    $PDAS,PREFLL,a,b,c,d,e,f[*hh]<CR><LF>

  - QUERY command:

    $PDAS,PREFLL[*hh]<CR><LF>

- **Command identification:**

  - $PDAS,PREFLL

- **Parameters**

ref.	format	
a	x	: Coordinate system number (1 to 10) (default: 0)
b	llll.lllll	: Reference station latitude (with centimetric accuracy)
c	a	: Latitude direction (N or S)
d	yyyyy.yyyyyy	: Reference station longitude (with centimetric accuracy)
e	a	: Longitude direction (E or W)
f	x.x	: Reference station altitude, in metres (*centimetric accuracy* required for this parameter)
***hh**		: Checksum (optional)
<CR><LF>		: End of command

• Examples

$PDAS,PREFLL **QUERY**
$PDAS,PREFLL,0,0000.000000,N,00000.000000,E,0.00*3E

Changing the coordinates of the reference station:

$PDAS,PREFLL,0,3835.448532,S,01020.993478,E,93.833
$PDAS,PREFLL **QUERY**
$PDAS,PREFLL,0,3835.448532,S,01020.993478,E,93.83*18

8.40 $PDAS,PREFNE

• Function

- In a reference station, allows you to enter the precise projected coordinates of this station.
- In a mobile receiver, allows you to enter the precise projected coordinates of the position from which the receiver will be initialized.

  In both cases, the command is used to enter a reference position. See also $PDAS,FIXMOD and $PADS,PREFLL.

• Syntax

- Complete command:

$PDAS,PREFNE,a,b,c,d[*hh]<CR><LF>

- QUERY command:

$PDAS,PREFNE[*hh]<CR><LF>

• Command identification:

- $PDAS,PREFNE

• Parameters

ref.	format	
a	x	: Coordinate system number (1 to 10) (default: 0)
b	x.x	: Reference station Northing (*centimetric accuracy* required)
c	x.x	: Reference station Easting (*centimetric accuracy* required)
d	x.x	: Reference station altitude, in metres (*centimetric accuracy* required)
***hh**		: Checksum (optional)
<CR><LF>		: End of command

- **Examples**

$PDAS,PREFNE **QUERY**
$PDAS,PREFLL,0,0000.000000,N,00000.000000,E,0.00*3E
(No projection)

$PDAS,SELGEO,2 **Changing coord. syst**
$PDAS,PREFNE,2,259127.688,310500.551,48.752
Changing station's coords

$PDAS,PREFNE **Checking new coords**
$PDAS,PREFNE,2,259127.6882,310500.5510,48.7520*38

8.41 $PDAS,QC

- **Functions**
  - Enables Quality (Integrity) Control in the receiver and simultaneously chooses the type of Quality Control used (internal or external).
  - Disables Quality Control
  - Reports the type of Quality Control currently used, if any

    Of the two types of Quality Control possible, only the external one, relying on the WAAS/EGNOS system, is operational to date.

- **Syntax**
  - Complete command:

$PDAS,QC,a,b,c[*hh]<CR><LF>

  - QUERY command:

$PDAS,QC[*hh]<CR><LF>

- **Command identification:**
  - $PDAS,QC

- **Parameters**

ref.	format	
a	x	: Internal (or autonomous) Quality Control: 0: No internal Quality Control 1: UKOOA Control
b	a	: External Quality Control: 0: No external Quality Control 1: WAAS/EGNOS Quality Control 2: RTCM-SC104Quality Control, message type 5
c	x	: Provider of external Quality Control: if **b** = 1, PRN of the GEO to be received if **b** = 2, Number of the RTCM-SC104 reference station to be received
***hh**		: Checksum (optional)
<CR><LF>		: End of command

- **Examples**

$PDAS,QC	**QUERY**
$PDAS,QC,0,0,0	**No Quality Control currently used**
$PDAS,QC,0,1,138	**Selecting External Quality Control using WAAS/EGNOS GEO PRN 138**
$PDAS,QC	**Checking new setting**
$PDAS,QC,0,1,0138	**(Reply)**

8.42 $PDAS,RAZALM

• Function

- Deletes the specified almanacs from the receiver's memory.

• Syntax

$PDAS,RAZALM,a[*hh]<CR><LF>

• Parameters

ref.	format	
a	x	: Defines the type of almanacs you want to delete: 0 (or **a** omitted): all 1: GPS almanacs only 2: WAAS/EGNOS almanacs only
***hh**		: Checksum (optional)
<CR><LF>		: End of command

• Examples

$PDAS,RAZALM	**Deletes all almanacs**

8.43 $PDAS,RESET

- **Function**

Resets the software of the unit as power-on would do.

- **Syntax**

$PDAS,RESET<CR><LF>

- **Parameters**

None

- **Example**

$PDAS,RESET

8.44 $PDAS,SELGEO

- **Function**
  - Of the coordinate systems defined with the $PDAS,GEO command, selects one to be the current coordinate system in the receiver.
- **Syntax**
  - Complete command:

$PDAS,SELGEO,a[*hh]<CR><LF>

  - QUERY command:

$PDAS,SELGEO[*hh]<CR><LF>

- **Command identification:**
  - $PDAS,SELGEO
- **Parameters**

ref.	format	
a	x	: Number of the coordinate system to be used (1 to 10) (default: 0)
***hh**		: Checksum (optional)
<CR><LF>		: End of command

- **Examples**

$PDAS,SELGEO	**QUERY**
$PDAS,SELGEO,0*21	**(Reply: coordinate system No. 1)**
$PDAS,SELGEO,2	**Selecting coord. system 2**
$PDAS,SELGEO	**QUERY**
$PDAS,SELGEO,2*23	**(Reply: coordinate system No. 2 used)**

8.45 $PDAS,SESSN

- **Functions**
  - Edits the definitions of the programmed sessions in the receiver.
  - Adds new sessions in the receiver.
- **Syntax**
  - Complete command:

$PDAS,SESSN,a,b,c,d,e[*hh]<CR><LF>

  - QUERY command:

$PDAS,SESSN[*hh]<CR><LF>

- **Command identification:**
  - $PDAS,SESSN
- **Parameters**

ref.	format	
a	x	: Line number (1 to 8)
b	hhmmss.ss	: Start time
c	hhmmss.ss	: End time
d	x.x	: Recording indicator 0: without data recording 1: with data recording on PCMCIA
e	c--c	: Session label (8 characters max.)
***hh**		: Checksum (optional)
<CR><LF>		: End of command

• Information supplement

- This command should be used in conjunction with $PDAS,EXPSES (see Par. 8.22).
- With AUTPW selected as the power control type (see $PDAS,EXPSES), the receiver will be turned on 5 minutes before the beginning of the session, and turned off 30 seconds after the end of the session.
- For a session defined with Recording indicator=1, and provided the recording firmware is present, a file will be created on the PCMCIA at the beginning of this session. The file will be named <session_label>.Dxx, where xx is a software-set order number, and all outputs performed on port P will be written into that file until the end of the session.

• Examples

$PDAS,SESSN	**QUERY**
$PDAS,SESSN,0*6E	**(Reply: no existing sessions)**

Defining three sessions:

$PDAS,SESSN,1,111500,140000,1,ESSAI1
$PDAS,SESSN,2,154500,173000,1,ESSAI2
$PDAS,SESSN,3,180000,203000,1,TEST1

Listing the programmed sessions:

$PDAS,SESSN
$PDAS,SESSN,1,111500.00,140000.00,1,ESSAI1*23
$PDAS,SESSN,2,154500.00,173000.00,1,ESSAI2*22
$PDAS,SESSN,3,180000.00,203000.00,1,TEST1*73

8.46 $PDAS,SVDSEL

• Functions

- Allows intentional rejection of satellites from the position processing in the receiver. Satellites may be GPS SVs or GEOs.
- Lists the intentionally rejected satellites
- Reads/changes the elevation threshold (minimum elevation angle) required of a non-rejected satellite to be involved in the position processing.

• Syntax

- Command relative to rejected satellites:

$PDAS,SVDSEL,a,b,c,d,...[*hh]<CR><LF>

- Command relative to elevation threshold:

$PDAS,SVDSEL,a [*hh]<CR><LF>

- QUERY command:

$PDAS,SVDSEL[*hh]<CR><LF>

• Parameters

ref.	format	
a	x.x	: Elevation threshold (in degrees)
b	x.x	: Indicates whether the PRNs that follow (**c,d,...**) are those of the only satellites you want to reject (this will be obtained by setting **b** to 0), or are added to the list of rejected satellites (in which case **b** will also designate one of these satellites). As a summary: **b** = 0 ⇒ No satellite is rejected except those specified in the next fields (**c,d,...)** **b** ≠ 0 ⇒ PRN of a satellite you want to reject (0 ≤ **b** ≤ 210)
c	x.x	: PRN of other satellite you want to reject (1 ≤ **c** ≤ 210)
d	x.x	: PRN of other satellite you want to reject (1 ≤ **d** ≤ 210) etc. (up to 12 SVs)
***hh**		: Checksum (optional)
<CR><LF>		: End of command

• Note

The elevation threshold has no effect on the position processing when the EDGPS, KART or LRK processing mode is selected.

• Examples

$PDAS,SVDSEL	**QUERY**
$PDAS,SVDSEL,20,2,6,8	**Elevation threshold is 20 °;** **SVs PRN 2, 6 ,8 are currently rejected**
$PDAS,SVDSEL,,5	**Adding SV PRN 5 to the list of rejected satellites**
$PDAS,SVDSEL	**QUERY (checking the change made)**
$PDAS,SVDSEL,20,2,6,8,5	**(Reply)**
$PDAS,SVDSEL,,0,2,7	**Clearing the list of intentionally rejected SVs. SV PRN 2 and 7 will now be the only SVs that are rejected**
$PDAS,SVDSEL	**QUERY (checking the change made)**
$PDAS,SVDSEL,20,2,7	**(Reply)**
$PDAS,SVDSEL,15	**Changing elevation threshold (15°)**
$PDAS,SVDSEL	**QUERY (checking the change made)**
$PDAS,SVDSEL,15,2,7	**(Reply)**
$PDAS,SVDSEL,,0	**Clearing the list of rejected satellites**
$PDAS,SVDSEL	**QUERY (checking the change made)**
$PDAS,SVDSEL,15	**(Reply) No satellite rejected**

8.47 $PDAS,TR

- **Function**
  - Triggers data output in RS232 mode on the specified port.
- **Syntax**

$PDAS,TR,a,b[*hh]<CR><LF>

- **Command identification:**
  - $PDAS,TR
- **Parameters**

ref.	format		
a	a	:	Output port identification (A, B, etc.) Placing a comma (,) behind this letter will delete the current user text (to be replaced by the next one (see below).
b	c--c	:	User text (60 characters max.)
***hh**		:	Checksum (optional)
<CR><LF>		:	End of command

- **Examples**

$PDAS,OUTMES,1,A,6,1 — **Validating output 1 on port A in TR mode**

$PDAS,OUTMES — **Checking output 1 definition**
$PDAS,OUTMES,1,A,6,1,1,5,7,8,9,10,20*55

$PDAS,TR — **Asking for output 1 to be sent**

Resulting data blocks (example):

$GPGGA,104849.99,4716.12353,N,00129.44097,W,0,09,1,85.99,M,0.00,M,-1.0,0000

8.48 $PDAS,UNIT

- **Function**
  - Edits the unit number, or the identification number in the case of a reference station.
- **Syntax**
  - Complete command

$PDAS,UNIT,a[*hh]<CR><LF>

  - QUERY command

$PDAS,UNIT[*hh]<CR><LF>

- **Command identification:**
  - $PDAS,UNIT
- **Parameters**

ref.	format	
a	x	: Unit number or station identification number (4 char. max.; 0 to 1023)
***hh**		: Checksum (optional)
<CR><LF>		: End of command

- **Examples**

$PDAS,UNIT	**QUERY**
$PDAS,UNIT,0*30	**(Reply: No 0000)**
$PDAS,UNIT,801	**Changing unit number**
$PDAS,UNIT	**QUERY**
$PDAS,UNIT,801*39	**(Reply: No 0004)**

8.49 $PDAS,XSOFT

- **Function**

Re-loads software into the unit.

- **Syntax**

$PDAS,XSOFT,a,b,c,d,…[*hh]<CR><LF>

- **Parameters**

Ref	Format		
,a	**c**	:	C : Read the checksum computed by sector or sector group
			E : Erase one of the programs
			P : Load an operating program into the Flash memory
***hh**		:	Checksum (optional)
<CR><LF>		:	End of command

- **Command identification**

$PDAS,XSOFT

- **Examples**

See below.

8.50 $PDAS,XSOFT,C

• Function

Edits the checksum computed by sector or by sector group for each program loaded in the Flash memory. It returns as many replies as there are programs. This command is automatically sent by the Core Module after receiving the binary data.

• Syntax

$PDAS,XSOFT,C[*hh]<CR><LF>

The reply looks like that:

$PDAS,XSOFT,C,a,b,c[*hh]<CR><LF>

• Parameters

Ref	Format		
,a	x	:	Index of the control program (0 to 2)
,b	h.h	:	Value of the checksum computed on the program specified
,c	h.h	:	Value of the checksum read on the program specified
***hh**		:	Checksum (optional)

<CR><LF> End of command

• Command identification

$PDAS,XSOFT,C

• Example

$PDAS,XSOFT,C	**Reading the checksum**
$PDAS,XSOFT,C,1,8D7,8D7	**(Reply)**

8.51 $PDAS,XSOFT,E

- ## Function

Totally erases the programs specified in the command (one or more). Erasure sets to $FFFF_h$. Any sector needs to be erased before it is re-loaded.

- ## Syntax

$PDAS,XSOFT,E[,a][,b][,c][*hh]<CR><LF>

The reply looks like that:

$PDAS,XSOFT,E,a,b,c[*hh]<CR><LF>

- ## Parameters

Ref	Format			
,a	x	:	**-1**	Do not erase program 1
			1	Erase program 1
,b	x	:	**-2**	Do not erase program 2
			2	Erase program 2
,c	x	:	**-3**	Do not erase application program
			3	Erase application program
***hh**		:	Checksum (optional)	

<CR><LF> End of command

- ## Command identification

$PDAS,XSOFT,E

- ## Examples

$PDAS,XSOFT,E	**Erase both programs.**
$PDAS,XSOFT,E,1,2	**Both programs have been erased.**
or	
$PDAS,XSOFT,E,1,-2	**Erasure of program 2 was not completed successfully.**
$PDAS,XSOFT,E,1	**Only erase program 1.**
$PDAS,XSOFT,E,1	**Program 1 has been erased**

8.52 $PDAS,XSOFT,P

• Function

Re-loads an operating program into the Flash memory.

• Syntax

$PDAS,XSOFT,P[,a][,b][,c][,d][*hh]<CR><LF>

The reply looks like that:

$PDAS,XSOFT,P,a,b,c,d[*hh]<CR><LF>

• Parameters

Ref	Format			
,a	x	:	**1**	Program P1
			2	Program P2
			3	Application program
			4	Core Module program (P1 and P2)
,b	x.x	:	Length of the program to be loaded (number of bytes, from 0 to 983040)	
,c	x		**0**	No Reset after loading
			.	Reset after loading
,d	c-c		Name of the file to be loaded	
***hh**		:	Checksum (optional).	

<CR><LF> End of command.

• Command identification

$PDAS,XSOFT,P

• Examples

$PDAS,XSOFT,P,1,21000,0 **Load program P1, with no Reset**
$PDAS,XSOFT,P,1,21000,0 **Loading completed successfully**

$PDAS,XSOFT,P,2,21000,1 **Load program P2, with Reset**

$PDAS,XSOFT,P,3,,1,E1_1b.omf **Load application program, with Reset**

$PDAS,XSOFT,P,4,,1,S1.s **Load Core Module program, with Reset**

8.53 Errors

8.53.1 Front Panel Indicators

ON/OFF indicator	**Scroll** indicator	**Meaning**
OFF	OFF	Receiver not powered. If this status is obtained after pressing the ON/OFF pushbutton, check power supply connection (cable, connectors), power source, power voltage, rear panel fuse. If this status is obtained after a period of operation, this means that the receiver has completed the programmed sessions and the receiver is now OFF due to automatic power supply control.
Flashing	OFF	Self-tests in progress (initialization phase)
ON	OFF	Operating receiver. No pending session.
ON	Flashing	Operating receiver, due to manual power supply control. Pending session.
OFF	Flashing	Receiver in standby, due to automatic power supply control. Pending session.
ON	ON	Operating receiver. Recording on PCMCIA in progress.

8.53.2 Error report

Errors are reported in two different ways:

- On the status display, on Screen No.1. Each error occupies a "subscreen" (see *AQUARIUS 5000 series User's Manual*).
- As a receiver reply to the $PDAS,DEFLT command (see Page 8-23).

8.53.3 Error families

Errors are classified into families, depending on the probable origin of error. The table below summarizes the 11 different error families

Family number	Origin	Error label
00	No errors	NONE
01	Core Module	CM
02	Application Configuration	CONFG
03	DGPS	DGPS
04	Coordinate system	GEODY
05	Input/Output	I/O
06	User Interface	IHM
07	Power supply/interface	INTRF
08	Navigation	NAVIG
09	Fix processing	POSIT
10	System	SYSTM
11	Data link	TD

8.53.4 Error classification

Errors are classified into four categories depending on gravity:

- Simple information reported to user (code 1)
- Warnings (code 2). The receiver operates correctly but might be disturbed by the reported error.
- Serious errors (code 3). The receiver operates but delivers erroneous results.
- Fatal errors (code 4). The receiver can no longer operate correctly. You should re-initialize the receiver.

8.53.5 Error list

No.	Family	Gra-vity	Meaning	Error label
01	1 - CM	4	GPS not ready	`GPS not ready`
02	1 - CM	4	RAM error	`RAM anomaly`
03	1 - CM	3	Processor error	`Processor anomaly`
04	1 - CM	3	Timing error	`Timing anomaly`
05	1 - CM	3	Program memory error	`Program memory anomaly`
06	1 - CM	3	Data memory error	`Data memory anomaly`
07	1 - CM	3	Reception circuit error	`Reception circuit anomaly`
08	1 - CM	3	Correlation circuit error	`Correlation circuit anom`
09	1 - CM	4	C/A-P/YCommunication error	`Communication C/A - P/Y`
10	1 - CM	2	Non-used output data	`Unread output datas`
11	1 - CM	2	Non-identified input data	`Unknown input datas`
12	1 - CM	2	Non-complying input data	`Bad input datas`
13	1 - CM	1	GPS data error	`GPS data anomaly`
14	1 - CM	1	DPRAM error	`DPRAM anomaly`
15	1 - CM	1	Erroneous message length	`Bad message length`
16	1 - CM	1	EEPROM error	`EEPROM anomaly`
17	1 - CM	3	Trigger time-tag error	`Datation Trigger Error`
18	2 - CONFG	4	Config integrity altered	`Bad config integrity`
19	2 - CONFG	3	Config parameter error	`Config parameter error`
20	3 - DGPS	3	No transmitting station	`No sending dtation`
21	3 - DGPS	3	CPU-DIFF overflow	`CPU-DIFF overflow`
22	4 - GEODY	3	Coordinate system error	`Geodesy error`
23	5 - I/O	2	Unknown remote command	`Unknown telecommand`
24	5 - I/O	2	Non-complying param. format	`Bad parameter format`
25	5 - I/O	2	Non-complying format block	`Bad block format`
26	5 - I/O	3	Command checksum error	`Bad telecommand checksum`
27	5 - I/O	3	DPR1 Input error	`Input error on DPR1`
30	5 - I/O	3	Non-complying LRK block	`Bad LRK block on port D`
31	5 - I/O	3	Port A Overflow	`Overflow PortA`
32	5 - I/O	3	Port B Overflow	`Overflow PortB`
33	5 - I/O	3	Port C Overflow	`Overflow PortC`
34	5 - I/O	3	Port D Overflow	`Overflow PortD`
35	5 - I/O	2	Format interpretation error	`Format interpretation`
36	5 - I/O	3	Port A Input error	`Input error PortA`
37	5 - I/O	3	Port B Input error	`Input error PortB`
38	5 - I/O	3	Port C Input error	`Input error PortC`
39	5 - I/O	3	Port D Input error	`Input error PortD`
40	6 - IHM	2	User Interface error	`IHM error`
41	7 - INTRF	4	Xilinx Load	`Xilinx Load`
42	7 - INTRF	4	Low Power Command	`Low Power Command`
43	7 - INTRF	3	PCMCIA overflow	`PCMCIA overflow`
44	7 - INTRF	3	File system full	`File system full`
45	7 - INTRF	2	PC board not recognized	`Unknown PC card`
46	7 - INTRF	4	Battery voltage too low	`Battery voltage`

47	7 - INTRF	3	Corrupted file system	`Corrupted file system`
48	7 - INTRF	4	First antenna error	`First antenna error`
52	7 - INTRF	3	File-opening error	`File open error`
53	7 - INTRF	3	File-closing error	`File close error`
54	7 - INTRF	3	File-writing error	`File write error`
55	7 - INTRF	3	File-reading error	`File read error`
56	8 - NAVIG	3	Navigation error	`Navigation error`
57	9 - POSIT	1	No differential reception	`No differential reception`
58	9 - POSIT	1	Too few Svs	`Too few Svs`
59	9 - POSIT	1	GDOP too high	`GDOP too high`
60	9 - POSIT	3	LPME too high	`LPME too high`
61	9 - POSIT	1	No fix computation	`No fix computation`
62	10 - SYSTM	2	Frozen display	`Frozen display`
63	10 - SYSTM	2	Unknown option code	`Unknown option code`
64	10 - SYSTM	4	C3 codes checksum error	`Bad checksum codes C3`
65	10 - SYSTM	2	Log checksum error	`Bad log checksum`
66	10 - SYSTM	4	Real-time clock	`Real Time Clock`
67	10 - SYSTM	4	Dual-port RAM	`Dual port RAM`
68	11 - SYSTM	4	Core module not ready	`Core module not ready`
69	10 - SYSTM	4	Program checksum error	`Bad program checksum`
70	10 - SYSTM	4	Data memory test	`Data memory test`
71	10 - SYSTM	4	Coprocessor test	`Coprocessor test`
72	10 - SYSTM	4	Serial port error	`Error on serial port`
73	10 - SYSTM	3	IDE file system mounting error	`File system IDE mount err`
74	10 - SYSTM	1	Option lending period has now elapsed	`Option no more available`
75	10 - SYSTM	4	Max number of tries reached	`Max option tries reached`
76	10 - SYSTM	1	Journal full	`Full anomalies journal`
77	10 - SYSTM	3	CMOS date failed	`CMOS date Failed`
78	11 - TD	4	Selftest error	`Autotest error`
79	11 - TD	3	Erroneous blocks	`Bad blocks`
80	11 - TD	1	Count of restarts since selftest	`Nb restart since autotest`
81	10 - SYSTM	3	Mailbox overflow	`Mailbox overflow`
82	10 - SYSTM	3	PCMCIA removed	`PCMCIA removed`
83	5 - I/O	3	DPR1 Overflow	`Overflow DPR1`
86	POSIT	3	Kinematic initialization	`Kinematic initialization`
87	10 - SYSTM	3	Line in CM file too long	`Line file CM too long`
88	10 - SYSTM	3	CM identification error	`Identification CM error`
89	10 - SYSTM	3	CM card file inconsistency	`Incoherence file card CM`

90	10 - SYSTM	3	Flash CM clear error	`Clear flash CM error`
91	10 - SYSTM	3	CM program loading error	`CM program file load error`
92	6 - IHM	3	Kinematic mode change	`Kinematic mode change`
93	6 - IHM	3	No position computed	`No computed position`
94	7 - INTRF	4	Binary file inconsistency	`Binary file incoherent`
95	10 - SYSTM		RTC send error	`RTC send error`
96	4 - GEODY		Altimetry error	`Altimetry error`
97	10 - SYSTM		Applic software Re-load error	`Appli soft reload error`
98	10 - SYSTM	4	Protected memory error	`Back memory failure`
99	10 - SYSTM	4	Stack overflow	`Stack overflow`
100	5 - I/O	2	Error on port A in reception	`Receiving error on port A`
101	5 - I/O	2	Error on port B in reception	`Receiving error on port B`
102	5 - I/O	2	Error on port C in reception	`Receiving error on port C`
103	5 - I/O	2	Error on port D in reception	`Receiving error on port D`
104	10 - SYSTM	1	Unexpected software error	`Software error`

9. RAW DATA FORMATS

9.1 GPS Raw Data in SVAR format

9.1.1 Notation rules

- **Reserved characters**

	(02_h)	<stx>	Beginning of message
!	(21_h)		Format indicator
,	$(2C_h)$		Field delimiter
@	(40_h)		Checksum delimiter
.	$(2E_h)$		Decimal separator
"	(22_h)		Beginning and end of label
	$(0D_h,0A_h)$	<eoln>	End of line
	(03_h)	<etx>	End of message

Subscript letter h at the end of a character string means that this string is in hexadecimal notation.

- **Conventions used**

field	Generic term representing one or more data
data	Numerical value or label
< >	Surrounds a field name
<stx>	Beginning of message (02_h)
<sobk>	Beginning of block: one or more characters, identifies beginning of block
<soln>	Beginning of line: one or more characters, identifies beginning of line in a block
<eoln>	End of line, 2 characters: CR,LF $(0D_h\ 0A_h)$
<etx>	End of message (03_h)

The term "block" stands for a group of data of the same nature.

The term "numerical value" encompasses all types of possible encodings : binary, decimal, hexadecimal.

The term "number" used without any further indication stands for a decimal number (base 10).

The term "label" stands for an ASCII character string.

- ## General form

<stx> <eoln>

<sobk> <,> < time tagging line > <eoln>

<soln> <,> < 1st data line > <eoln>

...

<soln> <,> < n^{th} data line > <eoln>

<etx>

The count and type of data in any given line are predefined, which means that the count of separators <,> is invariable.

Any data missing or replaced by one or more spaces means that this data is not available.

- ## Rule about numerals

A "zero" value is assumed to be valid. Spaces placed before or after numerals are not significant. There cannot be spaces within a numeral. The following formats are usable:

decimal	:	decimal separator is the "." symbol. It is always preceded by at least one figure (.25 is written 0.25) and followed by at least one figure, otherwise the integer notation is used.
integer	:	particular case of decimal notation without separator.
floating	:	exponent character is 'E' (example : 6.2512E3 = 6251.2)
signed	:	signs are placed at the beginning of the mantissa and after the exponent character. A numeral with no sign is assumed to be positive. There cannot be spaces between the sign and the first figure.

- ## Rule about labels

Labels are denoted by <"> characters surrounding them. They can take any ASCII value except <">, <stx> and <etx>.

Labels can optionally be associated with a numeral. In this case:

- They are placed just before or after the <,> field delimiter
- They are separated from the numeral by a <space> character

- ## Error check rule

An optional checksum can be placed at the end of every line (except for the <stx> and<etx> lines), between the last data in the line and <eoln>. The presence of the checksum is denoted by the @ character followed by the two end-of-line characters.

The checksum results from exclusive-OR gating all the characters in the line, excluding the @ character. The resulting 8-bit checksum is converted into 2 × 4 bits in hexadecimal notation and then the two half-bytes are ASCII-encoded. The most significant character is transferred first.

9.1.2 SVAR!D : Single-frequency Differential corrections

- **General Form**

<stx> <eoln>

<**!D**>,< time tagging > <eoln>

<soln>,< parameters > <eoln>

<soln>,< 1st line of differential corrections> <eoln>

...

<soln>,< nth line of differential corrections > <eoln>

<etx>

- **Time tagging line**

!D,< GPS week>,< GPS time><eoln>

- GPS week number and time within week, in seconds.
- Reference time is jan 6 1980 at 0hr00.

- **Parameter line**

<soln>	2 char	<**%S**> DSNP-type corrections (includes ionospheric corrections)
		<**%R**> RTCM-type corrections(does not include ionospheric corrections)
		<***#n***> message other than corrections (further use to be notified at a later date)
<Station number>		Read from the receiver configuration or from the RTCM 104 message
<Reception Quality>		0 to 10, corresponds to the ratio of the messages received correctly; 10 = 100%
<iono/tropo flag>		0 : Iono/tropo corrections are not included in differential corrections
		1 : Iono/tropo corrections are included in differential corrections
<eoln>		

• Corrections line

<soln>	3 characters: * and SV number
<C/A code correction>	PRC, in meters, at time To of message; Positive correction means it must be added to pseudorange
<correction speed>	RRC, in m/s
<correction age>	in seconds, algebraic difference between time of message and time of GPS measurements from which corrections were generated
<IOD>	Issue Of Data, for DSNP corrections, counter output modulo 256, incremented by 1 every time IOD changes state
<UDRE>	User Differential Range Error
<eoln>	

Time correction value (T) = PRC + RRC(T-To)

• Data block example

```
!D,945,410950.1
%R,710,,0
*3,-20.3,0.05,1.2,224
*19,20.7,-0.15,1.2,33
*17,-17.3,-0.06,1.2,235
*31,1.7,-0.09,1.2,181
*21,16.9,-0.10,1.2,231
*22,1.0,0.04,1.2,78
*23,-3.6,0.17,1.2,103
```

9.1.3 SVAR!R : Single-frequency GPS pseudoranges in satellite time

- **General Form**

<stx> <eoln>

<**!R**>,< time tagging > <eoln>

<soln>,< parameters > <eoln>

<soln>,< 1st line of raw data> <eoln>

...

<soln>,< nth line of raw data > <eoln>

<etx>

- **Time tagging line**

!R,< GPS week>,< GPS time><eoln>

- GPS week number and time within week, in seconds.
- Reference time is jan 6 1980 at 0hr00 (assuming the modulo 2^{10} ambiguity is removed).

- **Parameter line**

<soln> 1st char	<&> (data type 2)
2nd char	<C> L1 phase measurement, C/A code
<filter. time constant>	in seconds (code smoothed by carrier)
<eoln>	

• Raw data lines

<soln>	2 characters: * and channel No. (in hexadecimal)
<SV No.>	
<C/A code pseudorange>	in 10^{-10} s, modulo 10 s
<$L1_{C/A}$ carrier phase>	in 10^{-3} cycles, modulo 10^{8} cycles
<$L1_{C/A}$ carrier speed>	in 10^{-3} cycle/s
<C/A L1 C/No>	in dBHz
<L1 channel status>	encoded on a 4-bit ASCII character [0 to F] bit 0 = 0 *(not used)* bit 1 = 0 (*reserved*) bit 2 = 1 if invalid L1 phase measurement bit 3 = 0 (*reserved*)
<L1 carrier quality indicator>	encoded on 2 ASCII characters [0 to F], 8 bits, MSB first bits 0 to 4 : "cumulative loss of continuity indicator", (complies with RTCM message No. 18, counter modulo 32 incremented every time the continuity of the carrier phase measurement is lost) bits 5 to 7 : "data quality indicator", (complies with RTCM message No. 18): "000": phase error ≤ 0.00391 cycle "001": phase error ≤ 0.00696 cycle "010": phase error ≤ 0.01239 cycle "011": phase error ≤ 0.02208 cycle "100": phase error ≤ 0.03933 cycle "101": phase error ≤ 0.07006 cycle "110": phase error ≤ 0.12480 cycle "111": phase error > 0.12480 cycle
<C/A code quality indicator>	encoded on 2 ASCII characters [0 to F], 8 bits, MSB first bits 0 to 3: "pseudo-range multipath error indicator", (complies with RTCM message No. 19): "1111":multipath error not determined

bits 4 to 7 : "pseudo-range data quality indicator",
(complies with RTCM message No. 19):
"0000": pseudorange error ≤ 0.020
"0001": pseudorange error ≤ 0.030
"0010": pseudorange error ≤ 0.045
"0011": pseudorange error ≤ 0.066
"0100":pseudorange error ≤ 0.099
"0101": pseudorange error ≤ 0.148
"0110": pseudorange error ≤ 0.220
"0111": pseudorange error > 0.329
"1000": pseudorange error ≤ 0.491
"1001": pseudorange error ≤ 0.732
"1010": pseudorange error ≤ 1.092
"1011": pseudorange error ≤ 1.629
"1100": pseudorange error ≤ 2.430
"1101": pseudorange error ≤ 3.625
"1110": pseudorange error ≤ 5.409
"1111": pseudorange error > 5.409

<eoln>

- **Data block example**

!R,945,409178.0

&C,30

***0,3,1642748611,1336643,-3745940,50,0,21,8F**

***1,6,1770768785,6159605,1173036,38,0,20,BF**

***2,17,1653042024,2007234,66112,49,0,2A,8F**

***3,19,1765372780,2787887,-4030232,38,0,54,BF**

***4,21,1750942628,5177540,-5179588,46,0,02,9F**

***5,22,1622832882,903573,-850340,51,0,0B,7F**

***6,23,1707824729,5132206,-3991356,45,0,0C,9F**

***7,25,1786374004,4350534,1642228,37,0,40,BF**

***9,31,1756457738,8208042,-5146444,37,0,62,CF**

9.1.4 SVAR!R : Dual frequency GPS pseudoranges in satellite time

- **General Form**

<stx> <eoln>

<**!R**>,< time tagging > <eoln>

<soln>,< parameters > <eoln>

<soln>,< 1st line of raw data> <eoln>

...

<soln>,< nth line of raw data > <eoln>

<etx>

- **Time tagging line**

!R,< GPS week>,< GPS time><eoln>

- GPS week number and time within week, in seconds.
- Reference time is jan 6 1980 at 0hr00 (assuming the modulo 2^{10} ambiguity is removed).

- **Parameter line**

<soln> 1st char	<&>
	<P> L1 and L2 phase measurements, C/A, P/Y codes
<filter. time constant>	in seconds (C/A code smoothed by carrier)
<eoln>	

- ## Dual-frequency raw data lines

<soln>	2 characters: * and channel No. (in hexadecimal)
<SV No.>	
<C/A code pseudorange>	in 10^{-10} s, modulo 10 s
<$L1_{C/A}$ carrier phase>	in 10^{-3} cycles, modulo 10^{8} cycles
<$L1_{C/A}$ carrier speed>	in 10^{-3} cycle/s
<C/A L1 C/No>	in dBHz
<L1, L2 channel status>	encoded on a 4-bit ASCII character [0 to F] bit 0 = 0 *(not used)* bit 1 = 0 if code P; 1 if code Y (antispoofing) bit 2 = 1 if $L1_{C/A}$ phase measurement not valid bit 3 = 1 if $L2_{P/Y}$ phase measurement not valid
<L1 carrier quality indicator>	encoded on 2 ASCII characters [0 to F], 8 bits, MSB first bits 0 to 4 : "cumulative loss of continuity indicator", (complies with RTCM message No. 18, counter modulo 32 incremented every time the continuity of the carrier phase measurement is lost) bits 5 to 7 : "data quality indicator", (complies with RTCM message No. 18): "000": phase error ≤ 0.00391 cycle "001": phase error ≤ 0.00696 cycle "010": phase error ≤ 0.01239 cycle "011": phase error ≤ 0.02208 cycle "100": phase error ≤ 0.03933 cycle "101": phase error ≤ 0.07006 cycle "110": phase error ≤ 0.12480 cycle "111": phase error > 0.12480 cycle
<C/A code quality indicator>	encoded on 2 ASCII characters [0 to F], 8 bits, MSB first bits 0 to 3: "pseudo-range multipath error indicator", (complies with RTCM message No. 19): "1111": multipath error not determined bits 4 to 7: "pseudo-range data quality indicator", (complies with RTCM message No. 19): "0000": pseudorange error ≤ 0.020 "0001": pseudorange error ≤ 0.030 "0010": pseudorange error ≤ 0.045 "0011": pseudorange error ≤ 0.066 "0100": pseudorange error ≤ 0.099 "0101": pseudorange error ≤ 0.148 "0110": pseudorange error ≤ 0.220 "0111": pseudorange error > 0.329

"1000": pseudorange error ≤ 0.491
"1001": pseudorange error ≤ 0.732
"1010": pseudorange error ≤ 1.092
"1011": pseudorange error ≤ 1.629
"1100": pseudorange error ≤ 2.430
"1101": pseudorange error ≤ 3.625
"1110": pseudorange error ≤ 5.409
"1111": pseudorange error > 5.409

<$L1_{P/Y}$ - $L1_{C/A}$ carrier phase deviation> — in 10^{-3} cycles, modulo 1 cycle, centered around zero

<P_{L1} - C/A_{L1}code deviation> — in 10^{-10} s

<P_{L2} - C/A_{L1}code deviation> — in 10^{-10} s

<$L2_P$ carrier phase> — in 10^{-3} cycles, modulo 10^8 cycles of L2

<$L2_P$ carrier speed> — in 10^{-3} cycles

<L2 carrier quality indicator> — encoded on 2 ASCII characters [0 to F], 8 bits, MSB first

bits 0 to 4 : "cumulative loss of continuity indicator", (complies with RTCM message No. 18, counter modulo 32 incremented every time the continuity of the carrier phase measurement is lost)

bits 5 to 7 : "data quality indicator", (complies with RTCM message No. 18):

"000": phase error ≤ 0.00391 cycle
"001": phase error ≤ 0.00696 cycle
"010": phase error ≤ 0.01239 cycle
"011": phase error ≤ 0.02208 cycle
"100": phase error ≤ 0.03933 cycle
"101": phase error ≤ 0.07006 cycle
"110": phase error ≤ 0.12480 cycle
"111": phase error > 0.12480 cycle

<P/Y code quality indicator> — encoded on 2 ASCII characters [0 to F], 8 bits, MSB first

bits 0 to 3: "pseudo-range multipath error indicator", (complies with RTCM message No. 19):
"1111": multipath error not determined

bits 4 to 7: "pseudo-range data quality indicator", (complies with RTCM message No. 19):

"0000": pseudorange error ≤ 0.020
"0001": pseudorange error ≤ 0.030
"0010": pseudorange error ≤ 0.045
"0011": pseudorange error ≤ 0.066
"0100": pseudorange error ≤ 0.099
"0101": pseudorange error ≤ 0.148
"0110": pseudorange error ≤ 0.220
"0111": pseudorange error > 0.329
"1000": pseudorange error ≤ 0.491
"1001": pseudorange error ≤ 0.732
"1010": pseudorange error ≤ 1.092
"1011": pseudorange error ≤ 1.629
"1100": pseudorange error ≤ 2.430
"1101": pseudorange error ≤ 3.625
"1110": pseudorange error ≤ 5.409
"1111": pseudorange error > 5.409

<eoln>

• Data block example

```
!R,945,409517.0
&P,30
*0,3,2137408867,7051638,-1159380,51,2,0B,8F,-23,50,-45,50D76954,-903432,01,6F
*1,6,2275926394,9438843,3673120,39,2,60,BF,-43,17,-18,5496814,2862292,81,DF
*2,19,2259497283,5974953,-13A74584,39,A,43,BF,0,-208,,0,,A1,EF
*3,17,2155976904,3988834,2716264,48,2,21,8F,-23,-143,-211,1373394,2116524,01,7F
*4,21,2242445140,6696450,-2660704,47,2,46,9F,-20,64,28,5048311,-2073184,21,8F
*5,22,212381893S3,1570001,1821372,51,2,42,7F,-12,-158,-234,1893847,1419264,01,5F
*6,23,2202008192,7741120,-1358284,45,2,01,9F,12,-106,-130,6822254,-1058392,21,9F
*7,25,2292481156,6441213,4086108,37,A,54,BF,-500,,,0,,00,00
*9,31,2248027302,5125919,-2635232,39,2,5B,BF,12,-212,-243,3302338,-2053544,62,CF
```

9.1.5 SVAR!A : Almanac data

• General Form

<stx> <eoln>

<**!A**>**,**< time tagging > <eoln>

< parameters > <eoln>

< Almanac line > <eoln>

<etx>

• Time tagging line

!A,< GPS week>**,**< GPS time><eoln>

- GPS week number and time within week, in seconds.
- Reference time is Jan 6 1980 at 0hr00 (assuming the modulo 2^{10} ambiguity is removed).

• Parameter line

- <Number of the SV corresponding to the transmitted almanac>
- <Almanac reference week number> (assuming the modulo 2^{10} ambiguity is removed)
- <eoln>

• Almanac data lines

Bits 1 to 24 from words 3 to 10 in subframes 4 or 5 (depending on SV number).

Each GPS word (bits 1 to 24) is split into six 4-bit strings which are hex-encoded to form 6 bytes (0 to 1, A to F), with the first byte corresponding to bits 1 to 4.

The almanac line is organized as follows:

<word 3>,<word 4>,<word 5>,<word 6>,<word 7>,<word 8>,<word 9>,<word 10>,<eoln>

- ## Message example

!A,945,414504.2

4,945

4426B6,901606,FD3F00,A10D2F,AAA009,DDC8B3,ECF6F5,01003B

9.1.6 SVAR!E : Ephemeris data

- **General Form**

  <stx> <eoln>

  <**!E**>,< time tagging > <eoln>

  < parameters > <eoln>

  < 1st line of ephemeris data> <eoln>

  < 2nd line of ephemeris data> <eoln>

  < 3rd line of ephemeris data> <eoln>

  <etx>

- **Time tagging line**

  !E,< GPS week>,< GPS time><eoln>

  - GPS week number and time within week, in seconds.
  - Reference time is Jan 6 1980 at 0hr00 (assuming the modulo 2^{10} ambiguity is removed).

- **Parameter line**

  - <Number of the SV corresponding to the transmitted ephemeris>
  - <eoln>

- **Ephemeris data line**

  Line 1: bits 1 to 24 from words 3 to 10 in subframe 1

  Line 2: bits 1 to 24 from words 3 to 10 in subframe 2

  Line 3: bits 1 to 24 from words 3 to 10 in subframe 3.

  Each GPS word (bits 1 to 24) is split into six 4-bit strings which are hex-encoded to form 6 bytes (0 to 1, A to F), with the first byte corresponding to bits 1 to 4.

  Each ephemeris data line is organized as follows:

  <word 3>,<word 4>,<word 5>,<word 6>,<word 7>,<word 8>,<word 9>,<word 10>,<eoln>

- ## Data block example

!E,945,414347.7

10

EC5701,73336D,D49E97,A3469F,FEEBFC,346432,000004,027605

34FBF4,2FAA69,5E1FFF,FCA201,5BF1EC,11BCA1,0D90EA,64327C

0006D4,97F2C8,002527,577D88,1B60F3,6B16D7,FFA8CD,340D02

9.1.7 SVAR!U : Iono/UTC data

- **General Form**

<stx> <eoln>

<**!U**>,< time tagging > <eoln>

< Iono/UTC data line> <eoln>

<etx>

- **Time tagging line**

!U,< GPS week>,< GPS time><eoln>

- GPS week number and time within week (Z count in seconds), when the receiver generates the message.
- Reference time is Jan 6 1980 at 0hr00 (assuming the modulo 2^{10} ambiguity is removed).

- **Iono/UTC data line**

Bits 1 to 24 from words 3 to 10 in subframe 4, page 2-13.

Each GPS word (bits 1 to 24) is split into six 4-bit strings which are hex-encoded to form 6 bytes (0 to 1, A to F), with the first byte corresponding to bits 1 to 4.

The Iono/UTC data line is organized as follows:

<word 3>,<word 4>,<word 5>,<word 6>,<word 7>,<word 8>,<word 9>,<word 10>,<eoln>

- **Data block example**

!U,945,414740.3

780F00,FF0136,FEFC03,000032,000000,0F90B1,0C9002,0CAAAA

9.1.8 SVAR!S : Health & A/S data

- **General Form**

<stx> <eoln>

<**!S**>,< time tagging > <eoln>

< Health & A/S data line> <eoln>

<etx>

- **Time tagging line**

!S,< GPS week>,< GPS time><eoln>

- GPS week number and time within week (Z count in seconds), when the receiver generates the message.
- Reference time is Jan 6 1980 at 0hr00 (assuming the modulo 2^{10} ambiguity is removed).

- **Health & A/S data line**

A/S & Health: Bits 1 to 24 from words 3 to 10 in subframe 4, page 2-13

- Health: Bits 1 to 24 from words 3 to 10 in subframe 5, page2-13.

Each GPS word (bits 1 to 24) is split into six 4-bit strings which are hex-encoded to form 6 bytes (0 to 1, A to F), with the first byte corresponding to bits 1 to 4.

The Health & A/S data line is organized as follows:

<word 3>,<word 4>,<word 5>,<word 6>,<word 7>,<word 8>,<word 9>,<word 10>,<eoln>

- **Data block example**

!S,945,414740.3

7F9999,999999,009999,999099,999990,999080,000FC0,000FE9

7390B1,000000,000000,000FFF,F00000,00003F,000000,AAAAAB

9.1.9 SVAR!W: WAAS/EGNOS Data

- ### General Form

<stx> <eoln>

<**!W**>,<time tagging > <eoln>

<soln><parameters> <eoln>

<soln><Data from 1st GEO> <eoln>

...

<soln><Data from nth GEO> <eoln>

<etx>

- ### Time tagging line

!W,< GPS week>**,**< GPS time><eoln>

  - GPS week number and time within week, in seconds, when generating the message.
  - Reference time is Jan 6 1980 at 0hr00.

- ### Parameter line

%C,<message counter>,<count of GEOs in the message> <eoln>

  - The counter is modulo 16, incremented by 1 on arrival of a new message.
  - GEO count: from 1 to 4

- ### Pre-decoded WAAS data line

<soln>	2 characters: * and channel No. (in hexadecimal)
<GEO Number>	PRN of geostationary satellite (≥ 100)
<CRC validity flag>	0: Good; 1: Bad
<WAAS message No.>	From 0 to 63 (same as WAAS encoding)
<Preamble identifier>	From 1 to 3 (byte number in preamble)
<WAAS word>	occupies 212 bits in 53 ASCII/HEX- encoded characters (preamble and parity excluded)
<Checksum>	Optional , but recommended, checksum word
<eoln>	

• Data block example

!W,980,209274.0

%C,14,2

***D,120,0,9,1,F471A0418A0F158CD50A1B178034D586AF55127E070B10E144F82@48**

***E,132,0,9,1,8AC442C6AF0F16AF558A0F471A0410ECD500418A15837AF89A0B4@62**

9.2 GPS Raw Data in SBIN format

9.2.1 Notation Rules

• Reserved characters

By principle, all possible binary values in a byte are allowed. However three ASCII characters are used for message identification :

ASCII byte **FE_h** : denotes beginning of binary block

ASCII byte **FF_h** : denotes end of binary block

ASCII byte **FD_h** : denotes intentionally altered character

If between the beginning and the end of a block, the binary string initially includes such characters, then the following modifications are made to the string to avoid misinterpretation of the data at a further stage :

FD_h is converted into **FD_h 00_h**

FE_h is converted into **FD_h 01_h**

FF_h is converted into **FD_h 02_h**

NOTE *: When counting bytes in a message, remember that all the "doubled" characters (i.e. FD_h 00_h FD_h 01_h and FD_h 02_h) resulting from the transcoding described above must be counted as single characters.*

• Conventions used

- The term "field" stands for one or more parameters.
- The term "data" stands for a binary value occupying a byte.
- In a byte, bit "**0**" stands for the least significant bit, bit "**7**" for the most significant bit. The most significant bit is always placed ahead.

• Symbols used

< >	:	denotes a field
<stb>	:	beginning of block : ASCII character **FE_h**
<blid>	:	block type: 1 ASCII character allowing identification of the data type
<long>	:	2 bytes in binary notation specifying the count of bytes in the block, from <stb> excluded up to <checksum> excluded
<checksum>	:	2 bytes (for transmission error check)
<etb>	:	end of block: ASCII character **FF_h**

- ## General form

<stb>	:	1 byte (FE_h)
<blid>	:	1 byte
<long>	:	2 bytes
<data>	:	1 to 1023 bytes
<checksum>	:	2 bytes
<etb>	:	1 byte (FE_h)

The meaning of the data in each block type is predefined

- ## Error check rule

The message content is checked for transmission error through two "checksum" bytes the values of which result from the sum of all bytes, modulo 2^{16} , from <stb> excluded to <checksum> excluded.

- ## Rule about numerals

Unless otherwise specified:

- Numerals are expressed in binary, with fixed decimal point
- The notation of signed numbers meets the rule of the 2' s complement.

9.2.2 SBIN@R : Single-frequency GPS pseudoranges in satellite time

- **General form**

<stb><**R**>	2 bytes
<long>	2 bytes
<time tagging>	5 bytes
<parameters>	1 byte
<Raw Data, 1stSV>	14 bytes
...	
<Raw Data, last SV>	14 bytes
<checksum>	2 bytes
<etb>	1 byte

- **Time tagging**

First 2 bytes : GPS week number (assuming the modulo 2^{10} ambiguity is removed)

Last 3 bytes : GPS time in week (unit: 1/10 s). The reference time is Jan 6 1980 at 0hr00.

- **Parameters**

A single byte:

Bits 0 and 1 : Code smoothed by carrier according to RTCM message No. 19

Code	Smoothing Interval
00	0 to 1 minute
01	1 to 5 minutes
10	5 to 15 minutes
11	Indefinite

Bit 2 =1

Bit 3 : =0

Bits 4 to 6 : (*reserved*)

Bit 7 : =0 (single-frequency measurements)

- ## Satellite Raw Data

1st byte	:	SV number
Next 4 bytes	:	C/A code pseudorange (unit=10^{-10} s; modulo 400 ms)
Next byte	:	bits 0 to 4: Level indicator (C/No-26 dB.Hz) bits 5 to 6 not used bit 7=1 if phase measurement not valid
Next 3 bytes	:	$L1_{C/A}$ carrier phase (unit: 10^{-3} cycle, modulo 10^4 cycles)
Next 3 bytes	:	$L1_{C/A}$ carrier phase (unit 4×10^{-3} cycle/s, field ~ 32 Hz; MSB= sign; 800000_h=measurement not valid)
Next byte	:	$L1_{C/A}$ carrier quality indicator Bits 0 to 4: "cumulative loss of continuity indicator", complies with RTCM message No. 18, counter modulo 32 incremented every time the continuity of the carrier phase measurement is lost Bits 5 to 7: "data quality indicator", complies with RTCM message No. 18 "000": phase error ≤ 0.00391 cycle "001": phase error ≤ 0.00696 cycle "010": phase error ≤ 0.01239 cycle "011": phase error ≤ 0.02208 cycle "100": phase error ≤ 0.03933 cycle "101": phase error ≤ 0.07006 cycle "110": phase error ≤ 0.12480 cycle "111": phase error > 0.12480 cycle
Last byte	:	C/A code quality indicator Bits 0 to 3: "pseudorange multipath error indicator", complies with RTCM message No. 19 "1111": multipath error not determined

Bits 4 to 7: "pseudorange data quality indicator", complies with RTCM message No. 19

"0000": pseudorange error ≤ 0.020
"0001": pseudorange error ≤ 0.030
"0010": pseudorange error ≤ 0.045
"0011": pseudorange error ≤ 0.066
"0100": pseudorange error ≤ 0.099
"0101": pseudorange error ≤ 0.148
"0110": pseudorange error ≤ 0.220
"0111": pseudorange error > 0.329
"1000": pseudorange error ≤ 0.491
"1001": pseudorange error ≤ 0.732
"1010": pseudorange error ≤ 1.092
"1011": pseudorange error ≤ 1.629
"1100": pseudorange error ≤ 2.430
"1101": pseudorange error ≤ 3.625
"1110": pseudorange error ≤ 5.409
"1111": pseudorange error > 5.409

9.2.3 SBIN@R : Dual-frequency GPS pseudoranges in satellite time

- **General form**

<stb><**R**>	2 bytes
<long>	2 bytes
<time tagging>	5 bytes
<parameters>	1 byte
<Raw Data, 1stSV>	14 bytes
...	
<Raw Data, last SV>	14 bytes
<checksum>	2 bytes
<etb>	1 byte

- **Time tagging**

First 2 bytes : GPS week number (assuming the modulo 2^{10} ambiguity is removed)

Last 3 bytes : GPS time in week (unit: 1/10 s). The reference time is Jan 6 1980 at 0hr00.

- **Parameters**

A single byte:

bits 0 and 1 : C/A code smoothed by carrier, complies with RTCM message No. 19

Code	Smoothing Interval
00	0 to 1 minute
01	1 to 5 minutes
10	5 to 15 minutes
11	Indefinite

Bit 2=Bit 3 : =1

Bits 4 to 6 : =0 (*reserved*)

Bit 7 : =1 (dual-frequency measurements)

• Satellite Raw Data

1st byte	:	SV number
Next 4 bytes	:	C/A code pseudorange (unit: 10^{-10} s modulo:0.4 s)
Next byte	:	bits 0 to 4: Level indicator (C/No – 26), in dB.Hz bits 5, 6 and 7: channel status bit 5=0 if P code; =1 if Y code bit 6=1 if $L2_{P/Y}$ phase measurement not valid bit 7=1 if $L1_{C/A}$ phase measurement not valid
Next 3 bytes	:	$L1_{C/A}$ carrier phase (unit=10^{-3} cycle, modulo 10^4 cycles)
Next 3 bytes	:	$L1_{C/A}$ carrier phase (unit=4×10^{-3} cycles/s; field~32 kHz; MSB= sign; 800000_h=measurement not valid)
Next byte	:	$L1_{C/A}$ carrier quality indicator Bits 0 to 4: "cumulative loss of continuity indicator", complies with RTCM message No. 18, counter modulo 32 incremented every time the continuity of the carrier phase measurement is lost Bits 5 to 7: "data quality indicator", complies with RTCM message No. 18 "000": phase error ≤ 0.00391 cycle "001": phase error ≤ 0.00696 cycle "010": phase error ≤ 0.01239 cycle "011": phase error ≤ 0.02208 cycle "100": phase error ≤ 0.03933 cycle "101": phase error ≤ 0.07006 cycle "110": phase error ≤ 0.12480 cycle "111": phase error > 0.12480 cycle
Next byte	:	C/A code quality indicator Bits 0 to 3: "pseudorange multipath error indicator", complies with RTCM message No. 19 "1111": multipath error not determined

Bits 4 to 7: "pseudorange data quality indicator", complies with RTCM message No. 19

"0000": pseudorange error ≤ 0.020
"0001": pseudorange error ≤ 0.030
"0010": pseudorange error ≤ 0.045
"0011": pseudorange error ≤ 0.066
"0100": pseudorange error ≤ 0.099
"0101": pseudorange error ≤ 0.148
"0110": pseudorange error ≤ 0.220
"0111": pseudorange error > 0.329
"1000": pseudorange error ≤ 0.491
"1001": pseudorange error ≤ 0.732
"1010": pseudorange error ≤ 1.092
"1011": pseudorange error ≤ 1.629
"1100": pseudorange error ≤ 2.430
"1101": pseudorange error ≤ 3.625
"1110": pseudorange error ≤ 5.409
"1111": pseudorange error > 5.409

Next byte : $L1_{P/Y} - L1_{C/A}$ carrier phase deviation, centred around zero (unit=1/256th cycle; MSB= sign; 80_h=measurement not valid)

Next 2 bytes : $P_{L1} - C/A_{L1}$ code deviation (unit=10^{-10} s; field~3.2 µs; MSB= sign; 8000_h=measurement not valid)

Next 2 bytes : $P_{L2} - C/A_{L1}$ code deviation (unit=10^{-10} s; field~3.2 µs; MSB= sign; 8000_h=measurement not valid)

Next 3 bytes : $L2_{P/Y}$ carrier phase (unit=10^{-3} cycles modulo 10^4 cycles of L2)

Next 3 bytes : $L2_{P/Y}$ carrier speed (unit=4×10^{-3} cycles/s; field~32 kHz; MSB= sign; 800000_h=measurement not valid)

Next byte : L2 carrier quality indicator

Bits 0 to 4: "cumulative loss of continuity indicator", complies with RTCM message No. 18, counter modulo 32 incremented every time the continuity of the carrier phase measurement is lost

Bits 5 to 7: "data quality indicator", complies with RTCM message No. 18

"000": phase error ≤ 0.00391 cycle
"001": phase error ≤ 0.00696 cycle
"010": phase error ≤ 0.01239 cycle
"011": phase error ≤ 0.02208 cycle
"100": phase error ≤ 0.03933 cycle
"101": phase error ≤ 0.07006 cycle
"110": phase error ≤ 0.12480 cycle
"111": phase error > 0.12480 cycle

Last byte : P/Y code quality indicator

Bits 0 to 3: "pseudorange multipath error indicator", complies with RTCM message No. 19
"1111": multipath error not determined

Bits 4 to 7: "pseudorange data quality indicator", complies with RTCM message No. 19

"0000": pseudorange error ≤ 0.020
"0001": pseudorange error ≤ 0.030
"0010": pseudorange error ≤ 0.045
"0011": pseudorange error ≤ 0.066
"0100": pseudorange error ≤ 0.099
"0101": pseudorange error ≤ 0.148
"0110": pseudorange error ≤ 0.220
"0111": pseudorange error > 0.329
"1000": pseudorange error ≤ 0.491
"1001": pseudorange error ≤ 0.732
"1010": pseudorange error ≤ 1.092
"1011": pseudorange error ≤ 1.629
"1100": pseudorange error ≤ 2.430
"1101": pseudorange error ≤ 3.625
"1110": pseudorange error ≤ 5.409
"1111": pseudorange error > 5.409

9.2.4 SBIN@A: Almanac data

• General form

<stb><**A**>	2 bytes
<long>	2 bytes
<almanac ident.>	3 bytes
<SV almanac>	24 bytes
<checksum>	2 bytes
<etb>	1 byte

• Almanac identification

First byte	:	Number of the GPS satellite corresponding to the transmitted almanac (binary)
Last 2 bytes	:	Almanac *reference* week number (modulo 2^{10} ambiguity removed)

• Almanac data

- Bits 1 to 24 from words 3 to 10 in subframes 4 or 5 (depending on SV number)

9.2.5 SBIN@E: Ephemeris data

- **General form**

<stb><**E**>	2 bytes
<long>	2 bytes
<ephemeris ident.>	1 byte
<SV almanac>	24 bytes
<words 3 to 10, subfr 1>	24 bytes
<words 3 to 10, subfr 2>	24 bytes
<words 3 to 10, subfr 3>	24 bytes
<checksum>	2 bytes
<etb>	1 byte

- **Ephemeris identification**

A single byte : Number of the GPS satellite corresponding to the transmitted ephemeris (binary)

- **Ephemeris data**
  - Bits 1 to 24 from words 3 to 10 in subframe 1
  - Bits 1 to 24 from words 3 to 10 in subframe 2
  - Bits 1 to 24 from words 3 to 10 in subframe 3

9.2.6 SBIN@U: Iono/UTC data

• General form

<stb><**U**>	2 bytes
<long>	2 bytes
<Iono/UTC data>	24 bytes
<checksum>	2 bytes
<etb>	1 byte

• Iono/UTC Data

- Bits 1 to 24 from words 3 to 10 in subframe 4, page 2-13, declared valid by the GPS sensor

9.2.7 SBIN@S: Health & A/S data

- **General form**

<stb><**S**>	2 bytes
<long>	2 bytes
<A/S & Health data>	24 bytes
<Health data>	24 bytes
<checksum>	2 bytes
<etb>	1 byte

- **Health & A/S Data**

A/S & Health	:	Bits 1 to 24 from words 3 to 10 in subframe 4, page 2-13, declared valid by the GPS sensor
Health	:	Bits 1 to 24 from words 3 to 10 in subframe 5, page 2-13, declared valid by the GPS sensor

9.2.8 SBIN!W: WAAS/EGNOS Data

- **General Form**

<stb><**W**>	2 bytes
<long>	2 bytes
<Parameters>	1 byte
<Data from 1st GEO>	29 bytes
...	
<Data from nth GEO>	29 bytes
<checksum>	2 bytes
<etb>	1 byte

- **Parameters line**

A single byte:

bits 7 to 4 : Message counter (modulo 16, incremented by 1 whenever a new message is received)

bits 3 and 2 : =0 (no particular meaning)

bits 1 and 0 : Count of GEOs in the message:

Bit 1	Bit 0	GEO count
0	1	1
1	0	2
1	1	3
0	0	4

• GEO data line

First byte	:	GEO PRN
2nd byte	:	Message type: Bit 7: CRC validity flag (0: Good; 1: Bad) Bit 6: =0 (no particular meaning) Bits 5 to 0: message type (0 to 63, same as WAAS encoding)
3rd byte	:	Bits 7 and 6: Identifies preamble (8 bits out of 24 totally) as follows: "1": 1st byte from preamble "2": 2nd byte from preamble "3": 3rd byte from preamble Bits 5 and 4: = 0 (no particular meaning) Bits 3 to 0: first 4 bytes (MSB) from the 212-bit WAAS word
Next 26 bytes		The last 208 bits from the 212-bit WAAS word (excluding preamble, message number and parity)

10. TECHNICAL SPECIFICATIONS

10.1

CE certification.

Prior to putting any transmitting station into operation the user has to take any necessary steps for compliance with applicable regulations.

Below are the major specifications of the UHF transmission circuitry that may be required in applying for any certification or transmission authorization.

Output power	:	4 W (36 dBm)
		2 W (33 dBm)
		0.5 W (27 dBm)
		0.1 W (36dBm).
Frequency band	:	from 410 MHz to 470 MHz in 12.5-KHz steps.
Modulation	:	DQPSK 1200Bits/s, BELL 212A standard.
		or GMSK 4800 Bits/s, base band BT=0.5.
EMI specifications	:	ETS 300 279.
RF specifications	:	ETS 300 113.
Protection index	:	IP65.
Operating temperature	:	-20° à +55°C.
CCIR Code	:	3K-F1D / 5K6-F1D.

DSNP

16 rue de Bel Air B.P. 433
44474 CARQUEFOU Cedex

☎ +33 (0)2 40 30 59 00

Fax +33 (0)2 40 30 58 92

Web site: www.dsnp.com

S.A. à Directoire et Conseil de surveillance
au capital de 5 000 000 F

321 391 237 RCS Nantes