LUND UNIVERSITY

Simulation of Adaptive Ship Steering with Penalty on the Rudder Motion

Källström, Claes

1976

Document Version:
Publisher's PDF, also known as Version of record

Citation for published version (APA):
Källström, C. (1976). *Simulation of Adaptive Ship Steering with Penalty on the Rudder Motion*. (Research Reports TFRT-3133). Department of Automatic Control, Lund Institute of Technology (LTH).

LUND UNIVERSITY

PO Box 117
221 00 Lund
+46 46-222 00 00

ISRN LUTFD2/TFRT--3133--SE/1976

SIMULATION OF ADAPTIVE SHIP STEERING WITH PENALTY ON THE RUDDER MOTION

Claes Källström

This work has been supported by the Swedish Board for Technical Development under Contracts 734187 and 744127.

Table of Contents

Page

1. INTRODUCTION

Simulations of straight course keeping by different autopilots are presented in this report. The simulations are performed on the computer PDP 15 by use of the interactive program SIMNON (see Elmqvist (1975)). The ship model used describes a 255 000 tdw tanker. The model is a slightly simplified version of the one described in Aspernäs and Foisack (1975).

Three different autopilots are considered, viz. a fixed gain PID-regulator, a simple self-tuning regulator based on least squares identification and minimum variance control and a self-tuning regulator which contains the solving of a Riccati equation. It is possible to include penalty on the rudder motions in the two self-tuning regulators.

Simulations of ship steering are also discussed in Aspernäs and Foisack (1975) and Aspernäs and Källström (1975). Full-scale experiments on 255 000 tdw tankers are described in Källström (1974) and (1975).

Listings of the program used are given in the appendix.

2. SHIP STEERING DYNAMICS

A slightly simplified model of a 255 000 tdw tanker compared to the model described in Aspernäs and Foisack (1975) is used in the simulations. The number of propeller revolutions n and the forward speed u are assumed constant, and the actual values during all the simulations are 1.283 rps (77 rpm) and 8.21 m/s (16 knots), resp. The model, which is discussed in Norrbin (1970), then reads:

$$\dot{\delta} = -\frac{1}{T_s}\delta + \frac{1}{T_s \cdot CRG}\delta_s$$

$$|\dot{\delta}| \le \frac{1}{CRG}\delta_{lim}$$

$$\begin{aligned}(1-Y''_{\dot{v}})\dot{v} + (x''_G-Y''_{\dot{r}})L\dot{r} = {} & Y''_{uv}\frac{1}{L}uv + (Y''_{ur}-1)ur + \\ & + Y''_{|v|v}\frac{1}{L}|v|v + Y''_{c|c|\delta}\frac{1}{L}c|c|\delta + k_{TY}(T^P/m) + \\ & + \frac{1}{2}Y''_{nn}Ln^2 - K\sin\left(\frac{\alpha}{CRG}-\psi\right) + w_1\end{aligned}$$

$$\begin{aligned}(x''_G-N''_{\dot{v}})\frac{1}{L}\dot{v} + (k''_{zz}-N''_{\dot{r}})\dot{r} = {} & N''_{uv}\frac{1}{L^2}uv + (N''_{ur}-x''_G)\frac{1}{L}ur + \\ & + N''_{|v|r}\frac{1}{L}|v|r + N''_{c|c|\delta}\frac{1}{L^2}c|c|\delta + \\ & + k_{TN}(T^P/m)\frac{1}{L} + \frac{1}{2}N''_{nn}n^2 + K\ell_v\frac{1}{L^2}\sin\left(\frac{\alpha}{CRG}-\psi\right) + w_2\end{aligned}$$

$$\dot{\psi} = r$$

$$(T^P/m) = \left(\frac{1}{2}T^{P''}_{uu}\right)\frac{1}{L}u^2 + (T^{P''}_{un})un + (T^{P''}_{|n|n})L|n|n$$

$$c|c| = c_1u^2 + c_2\frac{n}{|n|}u^2 + c_3un + c_4n^2 \qquad (2.1)$$

Input signal:

rudder servo position	δ_s [deg]

States:

rudder deviation	δ [rad]
sway velocity	v [m/s]
yaw rate	r [rad/s]
heading	Ψ [rad]

Disturbances:

sway acceleration disturbance	w_1 [m/s^2]
disturbance of yaw angle acceleration	w_2 [rad/s^2]

Other notations:

propeller thrust per mass unit	T^p/m [m/s^2]
flow velocity past rudder	c [m/s]
number of propeller revolutions	n [1/s]
forward velocity	u [m/s]
length of ship	L [m]
wind force per mass unit	K [m/s^2]
lever arm of wind force	ℓ_v [m]
angle of relative wind direction	α [deg]
conversion factor rad - deg	CRG [deg]

The following parameter values are used:

$n = 1.283$ 1/s
$u = 8.21$ m/s
$L = 329.18$ m
$\ell_v = 25$ m
CRG = 57.2958 deg

The parameter values of the complete model are given in Aspernäs and Foisack (1975). Notice that it is possible to simulate two load conditions, corresponding to the mean draught $T = 20$ m (full load) and $T = 10.5$ (ballast). The ship is stable during these two conditions, but it is possible to obtain an artificial unstable ship by putting T = 25 m. The sign of the rudder angle in the model is chosen in such a way that a positive rudder angle (port rudder) gives a negative yaw rate (port yaw).

The disturbance signals w_1 and w_2 are obtained as white, gaussian noise filtered through a low pass filter. Two wind speeds, roughly 6-8 m/s (moderate breeze) and 17-20 m/s (fresh gale) are simulated. The following values are then used for the moderate breeze case

$$K = 0.002 \text{ m/s}^2$$

$$R_w = \begin{bmatrix} 10^{-9} & 0 \\ 0 & 10^{-11} \end{bmatrix}$$

and the following for the fresh gale case

$$K = 0.004 \text{ m/s}^2$$

$$R_w = \begin{bmatrix} 4\cdot 10^{-9} & 0 \\ 0 & 4\cdot 10^{-11} \end{bmatrix}$$

where R_w is the covariance matrix of the white noise vector, which generates w_1 and w_2.

The measured outputs from the model (2.1) are:

$$r_m = \bar{r} + e_1, \qquad \bar{r} = CRG \cdot r$$

$$\Psi_m = \bar{\Psi} + e_2, \qquad \bar{\Psi} = CRG \cdot \Psi$$

where e_1 and e_2 are white, gaussian measurement noise with covariance matrix

$$R_e = \begin{bmatrix} 0.0004 & 0 \\ 0 & 0.0025 \end{bmatrix}$$

The measured yaw rate r_m [deg/s] and the measured heading Ψ_m [deg] are used by the different autopilots.

The program of the ship model, SHIP1, is given in the appendix.

3. AUTOPILOTS

Three different autopilots are considered in the simulations, one fixed gain PID-regulator and two self-tuning regulators.

3.1. PID-regulator.

The following discrete PID-regulator is used

$$\delta_s(nT_s) = k_P[\Psi_m(nT_s) - \Psi_{ref}] + k_D r_m(nT_s) + $$

$$+ k_I T_s \sum_{i=0}^{n} [\Psi_m(iT_s) - \Psi_{ref}]$$

$$n = 0, 1, 2, \ldots \qquad (3.1)$$

where the sampling interval T_s always is equal to 15 s and where Ψ_{ref} denotes the reference course. Suitable values of k_P, k_D and k_I for different load and wind conditions are given in Aspernäs and Foisack (1975), Table 6.3, where the sampling interval T_s, however, is equal to 10 s.

3.2. Simple Self-Tuning Regulator (STURE).

A simple self-tuning regulator based on least squares identification and minimum variance control is used in the simulations. The basic self-tuning regulator is described in Wittenmark (1973).

The following model of the ship is used by the simple self-tuning regulator:

$$\begin{aligned}
&\left(\Psi_m(t) - \Psi_{ref}\right) + a_1\left(\Psi_m(t-k-1) - \Psi_{ref}\right) + \ldots + \\
&\quad + a_{NA}\left(\Psi_m(t-k-NA) - \Psi_{ref}\right) = \\
&\quad = b_0[\nabla\delta_s(t-k-1) + b_1\nabla\delta_s(t-k-2) + \ldots + \\
&\quad + b_{NB}\nabla\delta_s(t-k-NB-1)] + \\
&\quad + c_1\nabla r_m(t-k-1) + c_2\nabla r_m(t-k-2) + \ldots + \\
&\quad + c_{NC}\nabla r_m(t-k-NC) + \varepsilon(t)
\end{aligned} \tag{3.2}$$

Then the minimum variance control is given by

$$\begin{aligned}
\nabla_s\delta_s(t) = {} & a_1\left(\Psi_m(t) - \Psi_{ref}\right) + \ldots + \\
& + a_{NA}\left(\Psi_m(t-NA+1) - \Psi_{ref}\right) - \\
& - b_1\nabla_s\delta_s(t-1) - \ldots - b_{NB}\nabla_s\delta_s(t-NB) - \\
& - c_1\nabla r_m(t) - \ldots - c_{NC}\nabla r_m(t-NC+1)
\end{aligned} \tag{3.3}$$

where

$$\nabla_s\delta_s(t) = b_0\nabla\delta_s(t) = b_0\left(\delta_s(t) - \delta_s(t-1)\right)$$

$$\nabla r_m(t) = r_m(t) - r_m(t-1)$$

The value of NC may be zero, which means that no feedforward signal is used.

In the sequel the parameters NA and NB are fixed to 3 and 2, resp. Furthermore, the sampling interval T_s is equal to 15 s and the exponential forgetting factor λ_f is equal to 0.99.

By use of the minimum variance control (3.3) the following criterion is minimized:

$$J_1 = \sum_{n=k+1}^{N} \left(\Psi_m(nT_s) - \Psi_{ref}\right)^2 \tag{3.4}$$

If the criterion

$$J_2 = \sum_{n=k+1}^{N} \left[\left(\Psi_m(nT_s) - \Psi_{ref}\right)^2 + q_2\left(\nabla\delta_s\left((n-k-1)T_s\right)\right)^2\right] \tag{3.5}$$

is minimized instead, a penalty on the rudder motions is introduced by the parameter q_2. However, a proper solution of this problem requires the solving of a Riccati equation. A self-tuning regulator, which performs this, is described in Sec. 3.3.

If the criterion (3.5) is modified to read

$$J_3(n) = \left(\Psi_m\left((n+k+1)T_s\right) - \Psi_{ref}\right)^2 + q_2\left(\nabla\delta_s(nT_s)\right)^2$$

$$n = 0, 1, \ldots, N-k-1 \tag{3.6}$$

and if (3.6) is minimized at every sample event, then a simpler regulator is obtained. By inserting (3.2) into (3.6) and then performing the minimization the following control is obtained

$$\nabla_s\bar{\delta}_s(t) = \frac{b_0^2}{b_0^2 + q_2}\,\nabla_s\delta_s(t) \tag{3.7}$$

where $\nabla_s\delta_s(t)$ is the minimum variance control given by (3.3). If $q_2 = 0$, then minimization of (3.6) gives the same result as minimization of (3.4) and consequently the controls (3.7) and (3.3) are equivalent. Notice that (3.7) only is a very small modification of (3.3) and that the identification part of the self-tuning regulator is unchanged. However, the control (3.7) with positive q_2 has the serious disadvantage that no guarantee of closed loop stability is obtained in the general case.

The simple self-tuning regulator, which is described in this section, is implemented as a FORTRAN subroutine STURE, which is shown in the Appendix. A modification of STURE, where the parameter b_0 is estimated instead of assumed known, is also given in the Appendix as subroutine STURB.

3.3. Self-tuning Regulator which contains the Solving of a Riccati Equation (STURE3).

This regulator is based on least squares identification and linear quadratic control, i.e. the criterion (3.5) is properly minimized by the solving of a Riccati equation. The regulator is described in Åström (1974) and Åström-Wittenmark (1974).

The following model of the ship is used by the self-tuning regulator:

$$\begin{aligned}\left(\Psi_m(t) - \Psi_{ref}\right) + a_1\left(\Psi_m(t-1) - \Psi_{ref}\right) + \ldots +\\ + a_{NA}\left(\Psi_m(t-NA) - \Psi_{ref}\right) =\\ = b_1\nabla\delta_s(t-k-1) + \ldots + b_{NB}\nabla\delta_s(t-k-NB) + \varepsilon(t) \quad (3.8)\end{aligned}$$

The control law, which is given in Åström (1974), guarantees that all poles of the closed loop system are within a circle with radius r_0. In the sequel the parameter r_0 is fixed to 1. Furthermore, the sampling interval T_s is equal to 15 s and the exponential forgetting factor λ_f is equal to 0.99.

The FORTRAN subroutine STURE3, which is an implementation of the self-tuning regulator described in this section, is given in the Appendix together with the called subroutines RTLS1 and CORI.

4. SIMULATIONS

To make it possible to compare the steering quality of different regulators, three loss functions are now introduced:

$$V_1 = \frac{1}{\tau}\int_0^\tau\left[\left(\bar{\Psi}(t) - \Psi_{ref}\right)^2 + \lambda\delta_s^2(t)\right]dt$$

$$V_2 = \frac{1}{\tau}\int_0^\tau\left[\left(\bar{\Psi}(t) - \Psi_{ref}\right)^2 + \lambda\left(\delta_s(t) - m_{\delta_s}(t)\right)^2\right]dt$$

$$V_3 = \frac{1}{\tau}\int_0^\tau\left[\left(\bar{\Psi}(t) - \Psi_{ref}\right)^2 + \lambda\left(\nabla\delta_s(t)\right)^2\right]dt \tag{4.1}$$

where $m_{\delta_s}(t)$ is the mean value of $\delta_s(t)$ and the weighting factor λ always is assigned the value 0.1. The duration of the simulation is denoted τ and is always equal to 30 min. The three loss functions are approximated by:

$$V_1 = \frac{1}{N}\sum_{n=0}^{N-1}\left[\left(\bar{\Psi}(nT_s) - \Psi_{ref}\right)^2 + \lambda\delta_s^2(nT_s)\right]$$

$$V_2 = \frac{1}{N}\sum_{n=0}^{N-1}\left[\left(\bar{\Psi}(nT_s) - \Psi_{ref}\right)^2 + \lambda\left(\delta_s(nT_s) - m_{\delta_s}(nT_s)\right)^2\right]$$

$$V_3 = \frac{1}{N}\sum_{n=0}^{N-1}\left[\left(\bar{\Psi}(nT_s) - \Psi_{ref}\right)^2 + \lambda\left(\nabla\delta_s(nT_s)\right)^2\right] \tag{4.2}$$

where $NT_s = \tau$ and the sampling interval T_s always is equal to 15 s.

In the sequel the mean values m_{δ_s} and $m_{\bar{\Psi}}$ and the standard deviations σ_{δ_s} and $\sigma_{\bar{\Psi}}$ of the rudder servo position and the heading, respectively, will be presented as well as the standard deviation of the rudder servo changes $\sigma_{\nabla\delta_s}$. Notice that the yaw rate $\bar{r}$ and the heading $\bar{\Psi}$ without measurement noise are plotted and that $\bar{\Psi}$ is used in the loss functions instead of Ψ_m. The value of Ψ_{ref} is zero during all the simulations. The rudder servo position δ_s and the rudder servo change $\nabla\delta_s$ are both limited to ±20 deg if nothing else is remarked.

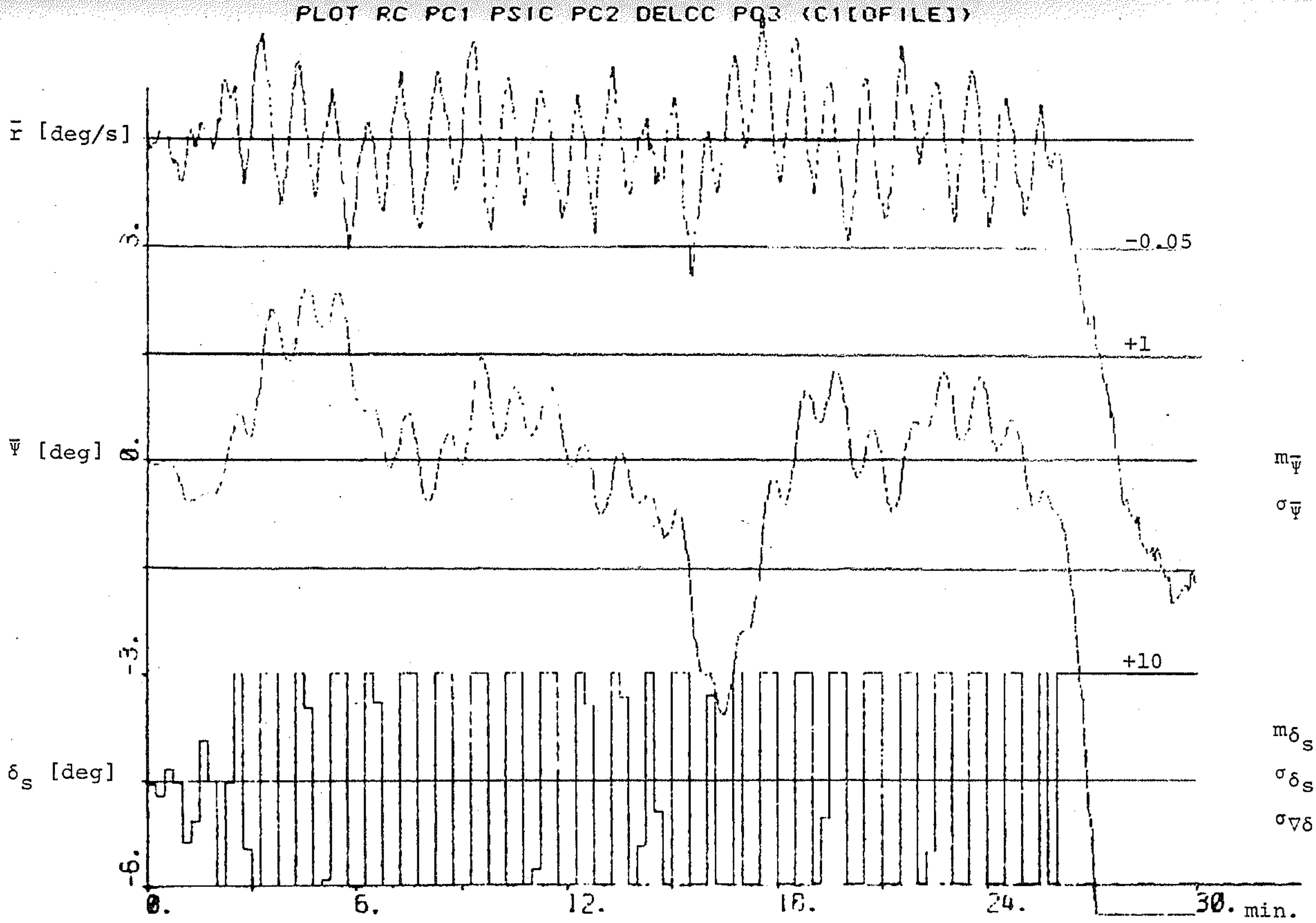
4.1. Full Load and Moderate Breeze.

The simulations presented in this section are all performed with a full loaded tanker (T = 20 m) and during moderate breeze conditions. The angle of relative wind direction α is equal to 0 deg. The parameters of the self-tuning regulators are always tuned beforehand during 60 min and all the plots and results shown in this section are related to the next 30 min.

The only simulation with regulator STURB is shown in Fig. 4.1. Although the initial value of b_0 was -1, the sign was changed and an abnormal behaviour was obtained. Two different fixed values of b_0 in the regulator STURE, when $q_2 = 0$, are compared in Figs. 4.2 and 4.3. The same comparisons when $q_2 \neq 0$ are shown in Figs. 4.4 and 4.5. Notice that $b_0 = -1$ and $q_2 = 0.1$ gives the same value of $b_0^2/(b_0^2+q_2)$ (cf. (3.7)) as $b_0 = -0.1$ and $q_2 = 0.001$. A summary of the 5 simulations is given in Table 4.1. As a conclusion the regulator STURE with $b_0 = -1$ will be considered in the sequel. No attempts with $b_0 = -10$ were performed since the magnitude of the a-parameters already was large when $b_0 = -1$.

Regulator	NA	NB	NC	k	q_2	b_0	$\sigma_{\overline{\Psi}}$ [deg]	σ_{δ_s} [deg]	$\sigma_{\nabla\delta_s}$ [deg]	V_1	V_2	V_3	Fig.
STURB	3	3	0	5	0	–	6.35	9.32	12.57	52.46	52.35	59.46	4.1
STURE	3	2	0	5	0	-1	0.29	2.36	3.09	0.66	0.65	1.04	4.2
STURE	3	2	0	5	0	-0.1	1.39	10.97	14.52	13.97	13.96	23.00	4.3
STURE	3	2	0	2	0.1	-1	0.39	2.95	2.16	1.04	1.02	0.62	4.4
STURE	3	2	0	2	0.001	-0.1	0.23	4.74	6.70	2.32	2.31	4.55	4.5

Table 4.1 - Comparisons between regulator STURB, where the parameter b_0 is estimated, and regulator STURE with different fixed values of b_0 and q_2.



$m_{\bar{\Psi}} = -1.84$

$\sigma_{\bar{\Psi}} = 6.35$

$m_{\delta_s} = 1.06$

$\sigma_{\delta_s} = 9.32$

$\sigma_{\nabla\delta_s} = 12.57$

Fig. 4.1 - Regulator STURB (NA=3 NB=3 NC=0 k=5 q_2=0): $V_1 = 52.46$ $V_2 = 52.35$ $V_3 = 59.46$. δ_s is limited to ±10 deg. The abnormal behaviour after 27 min is caused by incorrect sign of the parameter b_0.

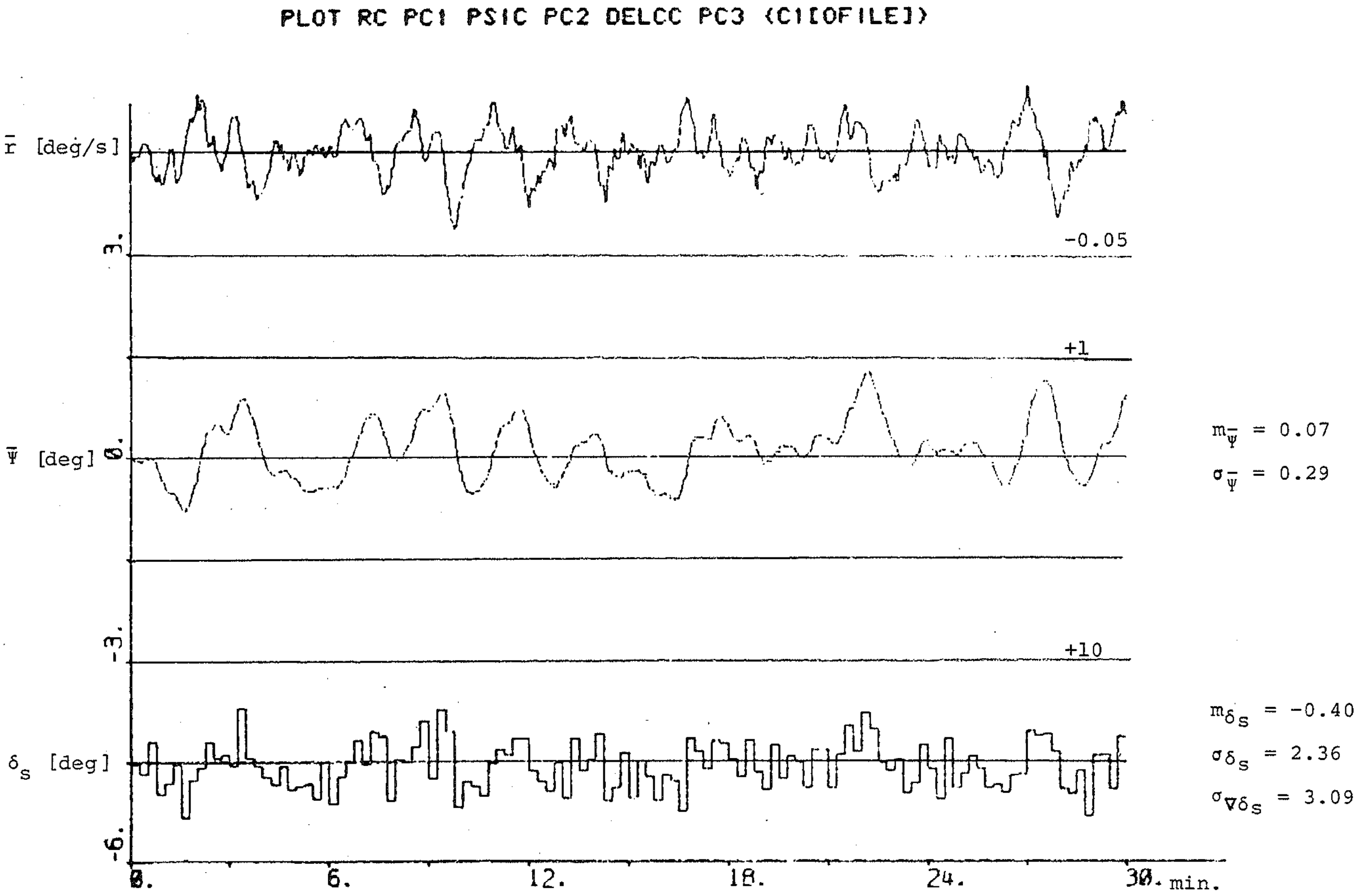


Fig. 4.2 - Regulator STURE (NA=3 NB=2 NC=0 k=5 $q_2=0$ $b_0=-1$): $V_1 = 0.66$ $V_2 = 0.65$ $V_3 = 1.04$

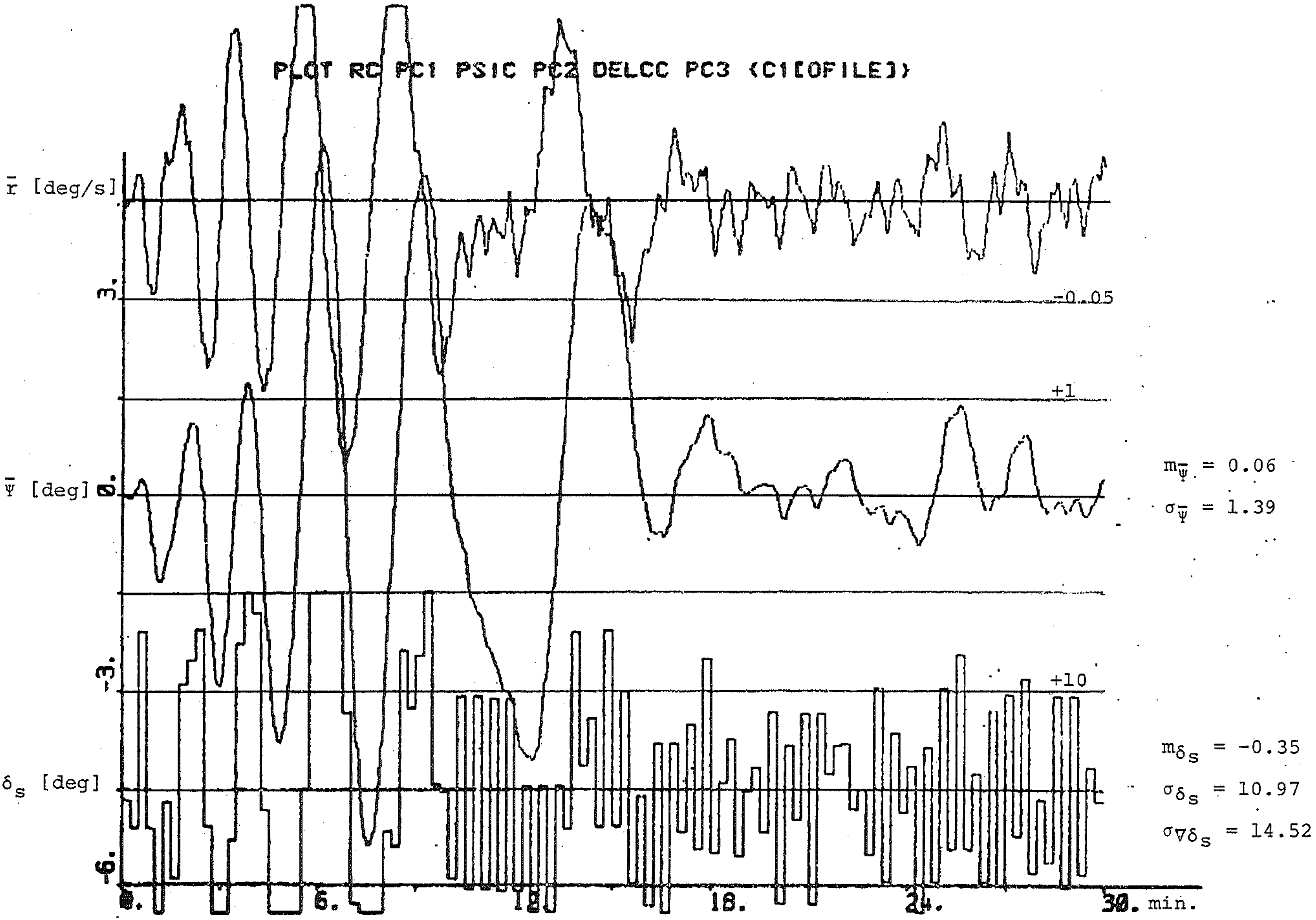


Fig. 4.3 - Regulator STURE (NA=3 NB=2 NC=0 k=5 $q_2=0$ $b_0=-0.1$): $V_1 = 13.97$ $V_2 = 13.96$ $V_3 = 23.00$.

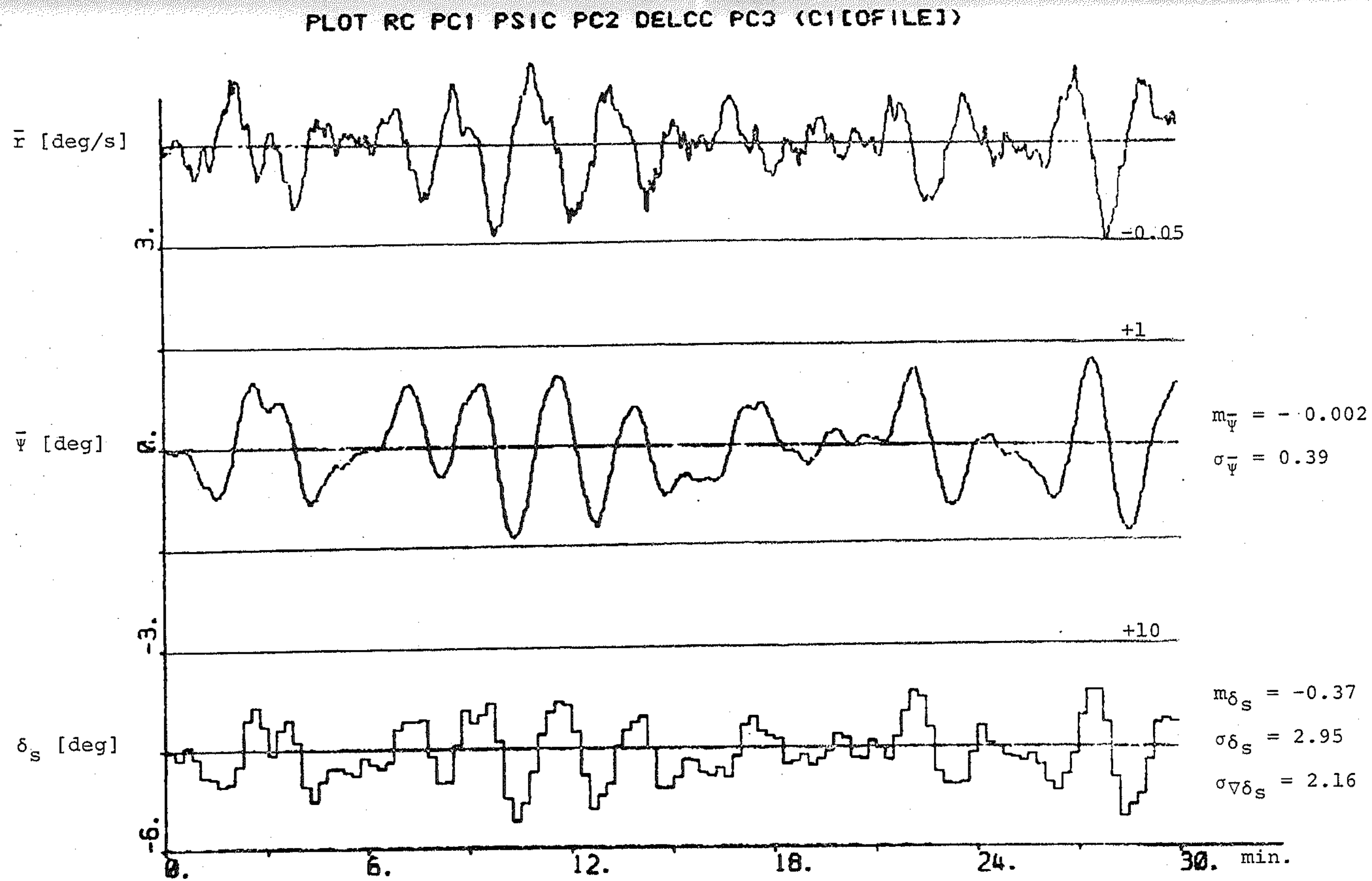
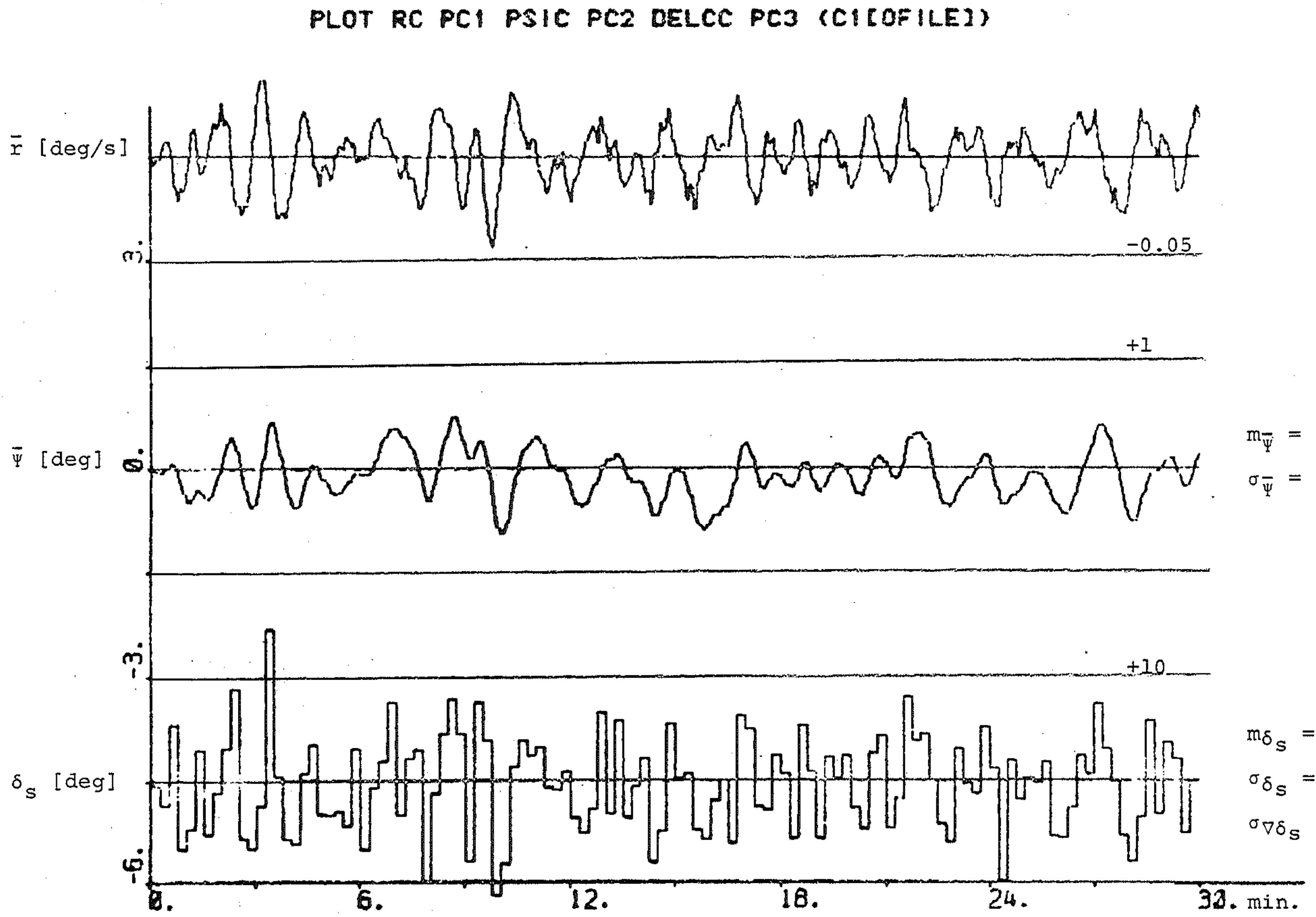


Fig. 4.4 - Regulator STURE (NA=3 NB=2 NC=0 k=2 q_2=0.1 b_0=-1): V_1 = 1.04 V_2 = 1.02 V_3 = 0.62.



$m_{\bar{\Psi}} = -0.06$

$\sigma_{\bar{\Psi}} = 0.23$

$m_{\delta_s} = -\ 0.37$

$\sigma_{\delta_s} = 4.74$

$\sigma_{\nabla\delta_s} = 6.70$

<u>Fig. 4.5</u> - Regulator STURE (NA=3 NB=2 NC=0 k=2 q_2=0.001 b_0=-0.1): V_1 = 2.32 V_2 = 2.31 V_3 = 4.55.

Several simulations of the regulator STURE with NA = 3, NB = 2 and b_0 = -1 and with varying values of NC, k and q_2 are shown in Fig. 4.6 - 4.26. A summary is given in Table 4.2. It is concluded that the combinations (NC = 0, k = 4, q_2 = 0.1), (NC = 0, k = 4, q_2 = 0.2), (NC = 0, k = 5, q_2 = 0.2) and (NC = 1, k = 4, q_2 = 0.2) give the best steering quality.

NC	k	q_2	$\sigma_{\bar{\Psi}}$ [deg]	σ_{δ_s} [deg]	$\sigma_{\nabla\delta_s}$ [deg]	V_1	V_2	V_3	Fig.
0	2	0	0.36	3.43	3.11	1.37	1.36	1.15	4.6
0	2	0.05	0.41	3.05	2.35	1.16	1.15	0.77	4.7
0	2	0.1	0.39	2.95	2.16	1.04	1.02	0.62	4.4
0	2	0.2	0.44	2.88	1.94	1.04	1.03	0.57	4.8
0	2	1	3.87	11.40	3.76	28.03	27.97	16.38	4.9
0	3	0	0.30	2.70	2.64	0.85	0.84	0.81	4.10
0	3	0.05	0.32	2.23	1.90	0.63	0.62	0.48	4.11
0	3	0.1	0.33	2.07	1.63	0.57	0.56	0.39	4.12
0	3	0.2	0.36	1.92	1.36	0.51	0.50	0.31	4.13
0	3	0.5	0.50	1.96	0.91	0.66	0.64	0.34	4.14
0	3	1	0.99	2.84	0.99	1.80	1.78	1.07	4.15
0	4	0	0.34	1.93	2.19	0.55	0.53	0.64	4.16
0	4	0.05	0.34	1.81	1.76	0.50	0.48	0.46	4.17
0	4	0.1	0.33	1.70	1.49	0.43	0.41	0.35	4.18
0	4	0.2	0.37	1.67	1.32	0.44	0.42	0.32	4.19
0	4	0.5	0.47	1.57	0.87	0.50	0.48	0.31	4.20
0	4	1	0.72	1.67	0.68	0.84	0.83	0.59	4.21
0	5	0	0.29	2.36	3.09	0.66	0.65	1.04	4.2
0	5	0.1	0.28	2.01	2.32	0.50	0.49	0.62	4.22
0	5	0.2	0.35	1.66	1.56	0.42	0.40	0.37	4.23
0	5	0.5	0.48	1.40	0.90	0.46	0.44	0.32	4.24
1	4	0.1	0.36	1.74	1.57	0.48	0.46	0.40	4.25
1	4	0.2	0.38	1.56	1.17	0.43	0.41	0.30	4.26

Table 4.2 - Regulator STURE with NA = 3, NB = 2 and b_0 = -1 and with varying values of NC, k and q_2.

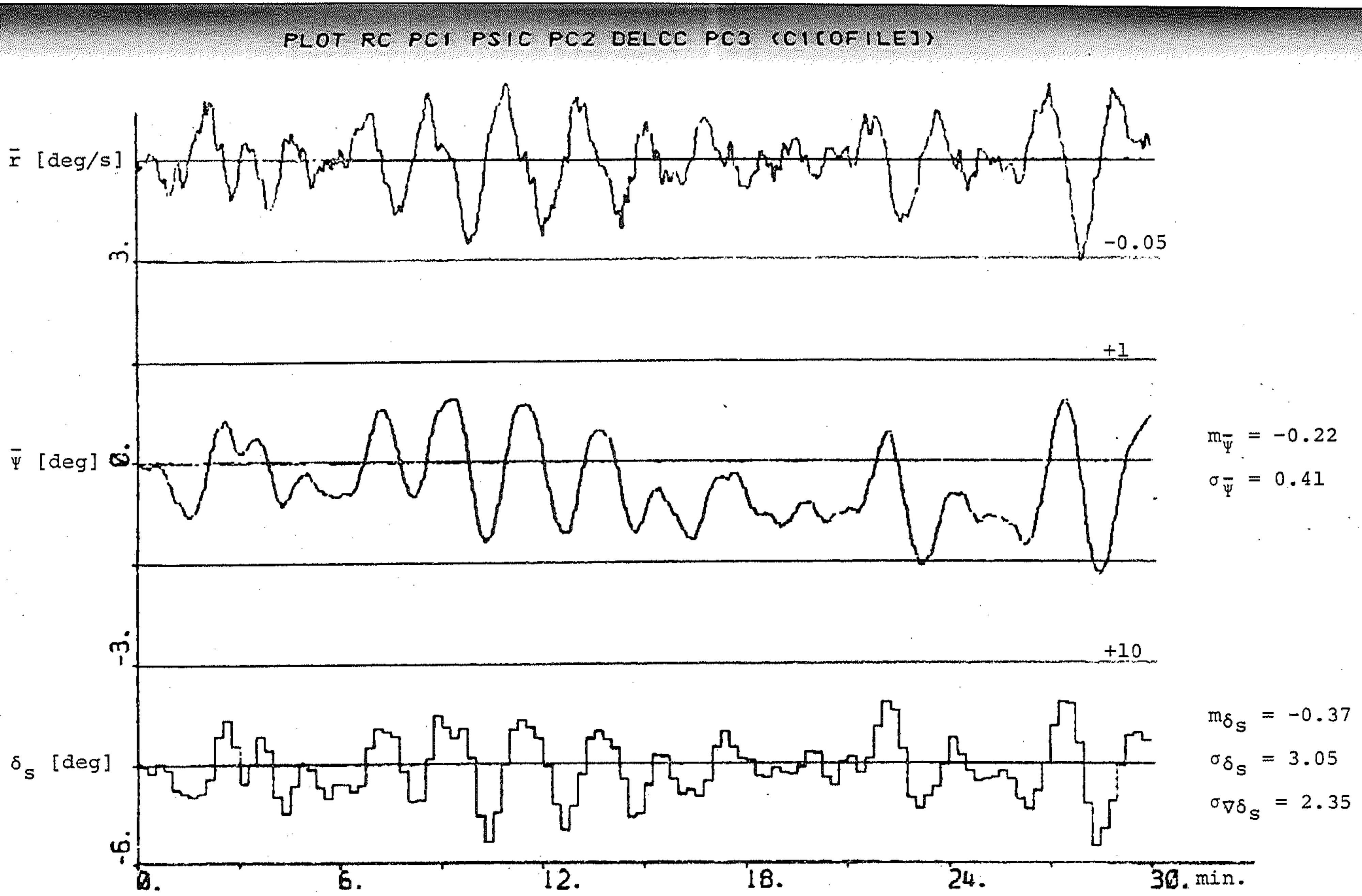


Fig. 4.7 - Regulator STURE (NA=3 NB=2 NC=0 k=2 q_2=0.05): V_1 = 1.16 V_2 =1.15 V_3 = 0.77

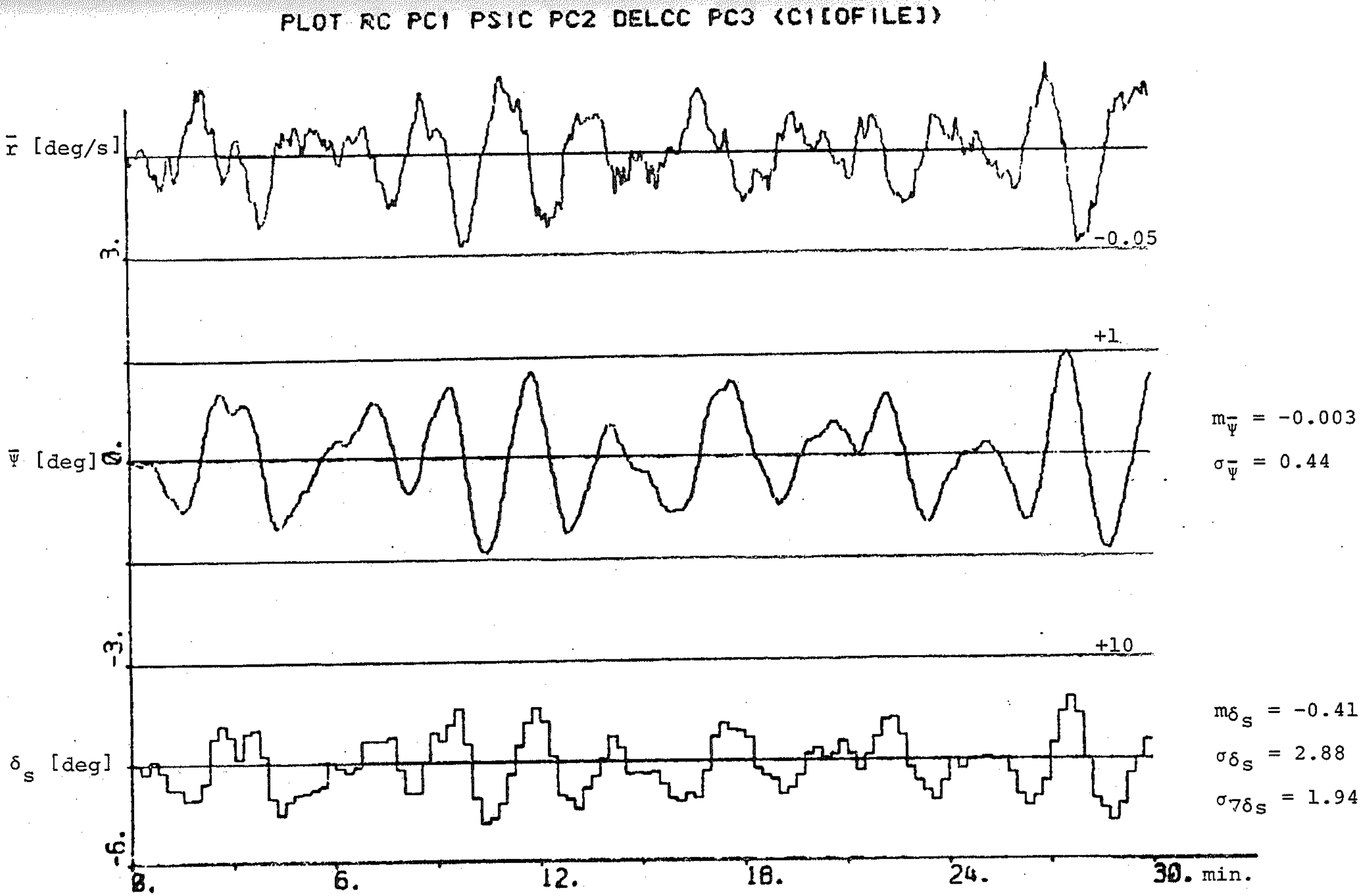


Fig. 4.8 - Regulator STURE (NA=3 NB=2 NC=0 k=2 q_2=0.2): V_1 = 1.04 V_2 = 1.03 V_3 = 0.57.

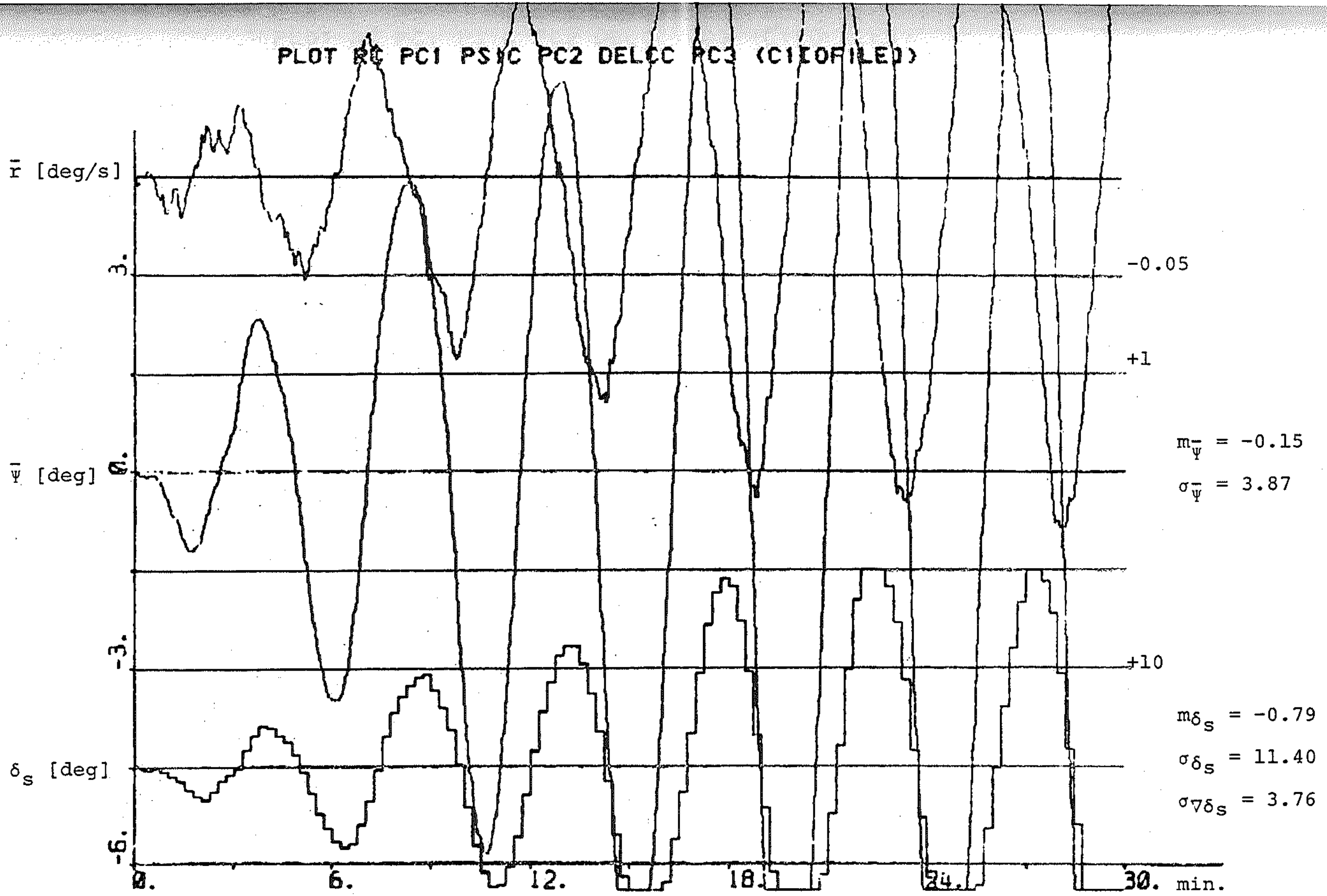


Fig. 4.9 - Regulator STURE (NA=3 NB=2 NC=0 k=2 q_2=1): V_1 = 28.03 V_2 = 27.97 V_3 = 16.38

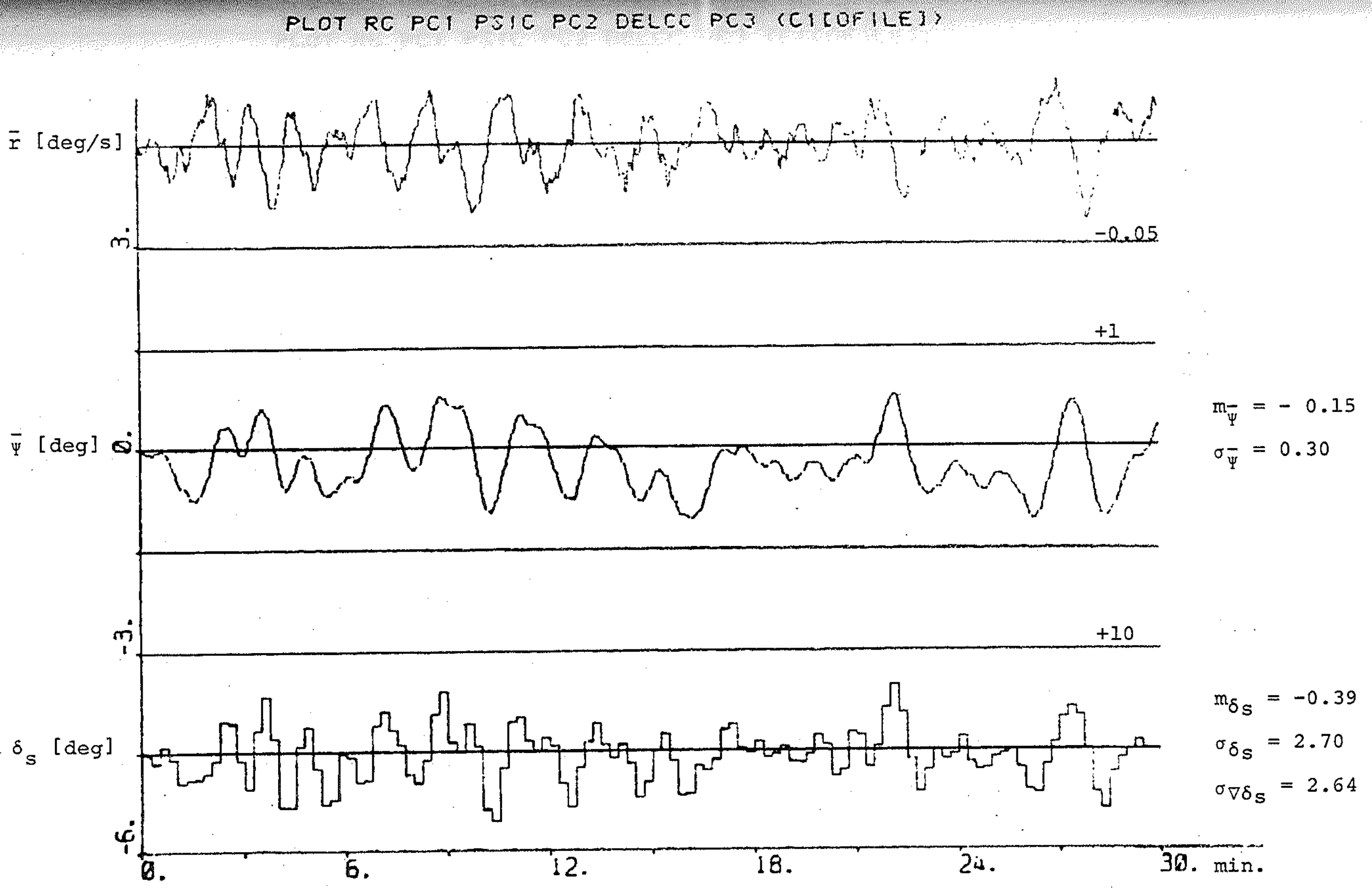


Fig. 4.10 - Regulator STURE (NA=3 NB=2 NC=0 k=3 q_2=0): $V_1 = 0.85$ $V_2 = 0.84$ $V_3 = 0.81$.

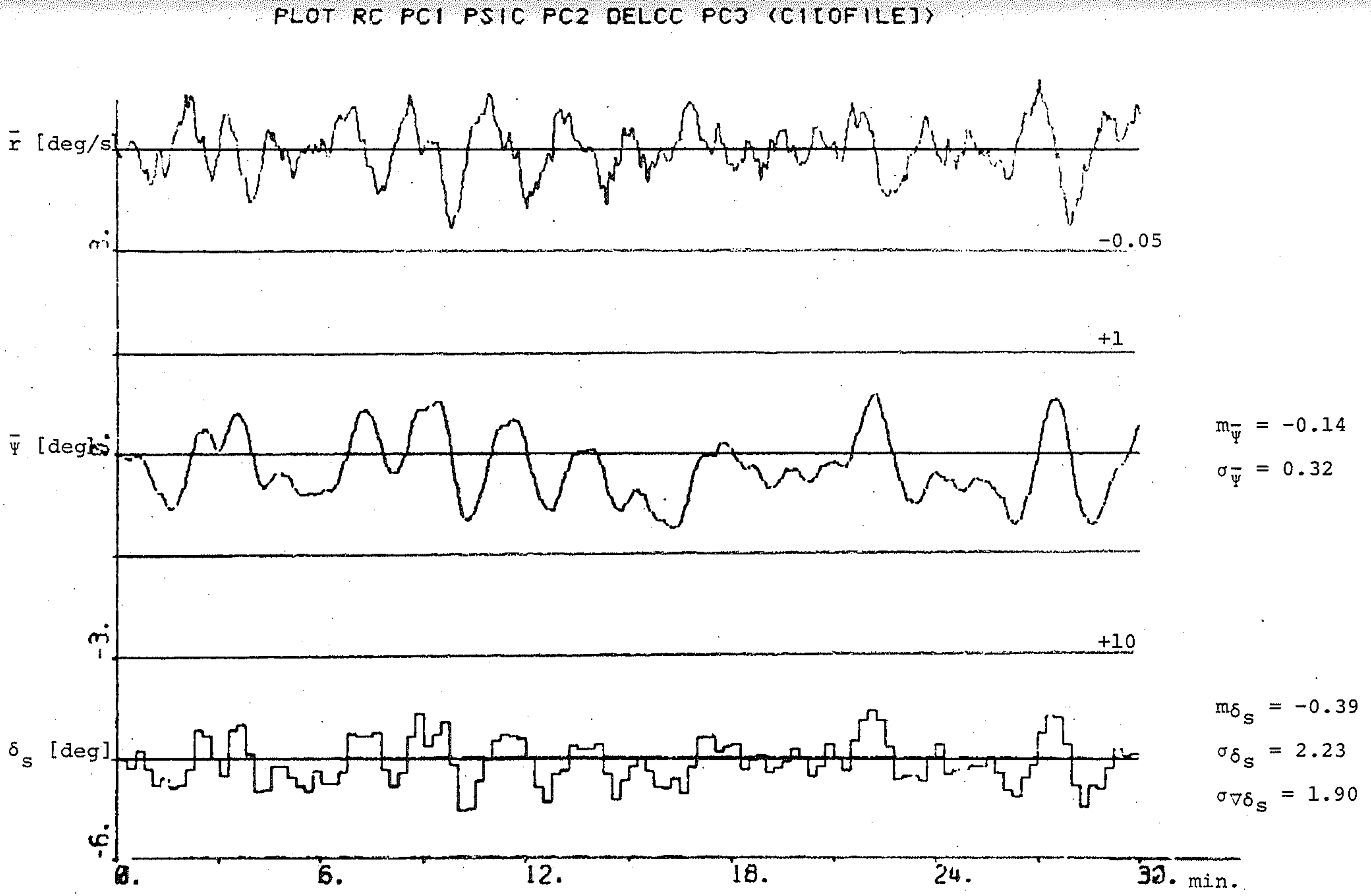


Fig. 4.11 - Regulator STURE (NA=3 NB=2 NC=0 k=3 q_2=0.05): V_1 = 0.63 V_2 = 0.62 V_3 = 0.48.

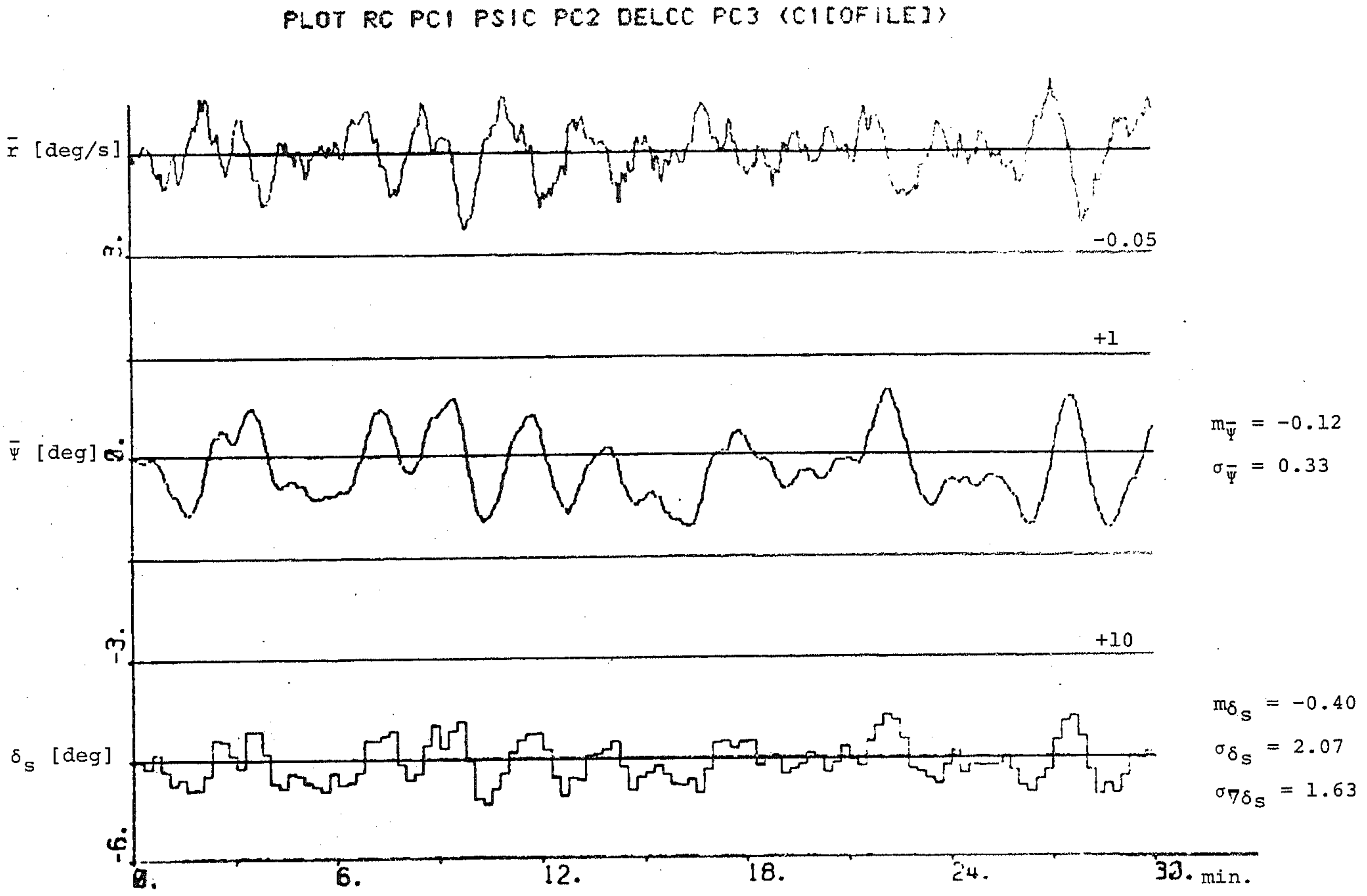


Fig. 4.12 - Regulator STURE (NA=3 NB=2 NC=0 k=3 q_2=0.1): $V_1 = 0.57$ $V_2 = 0.56$ $V_3 = 0.39$.

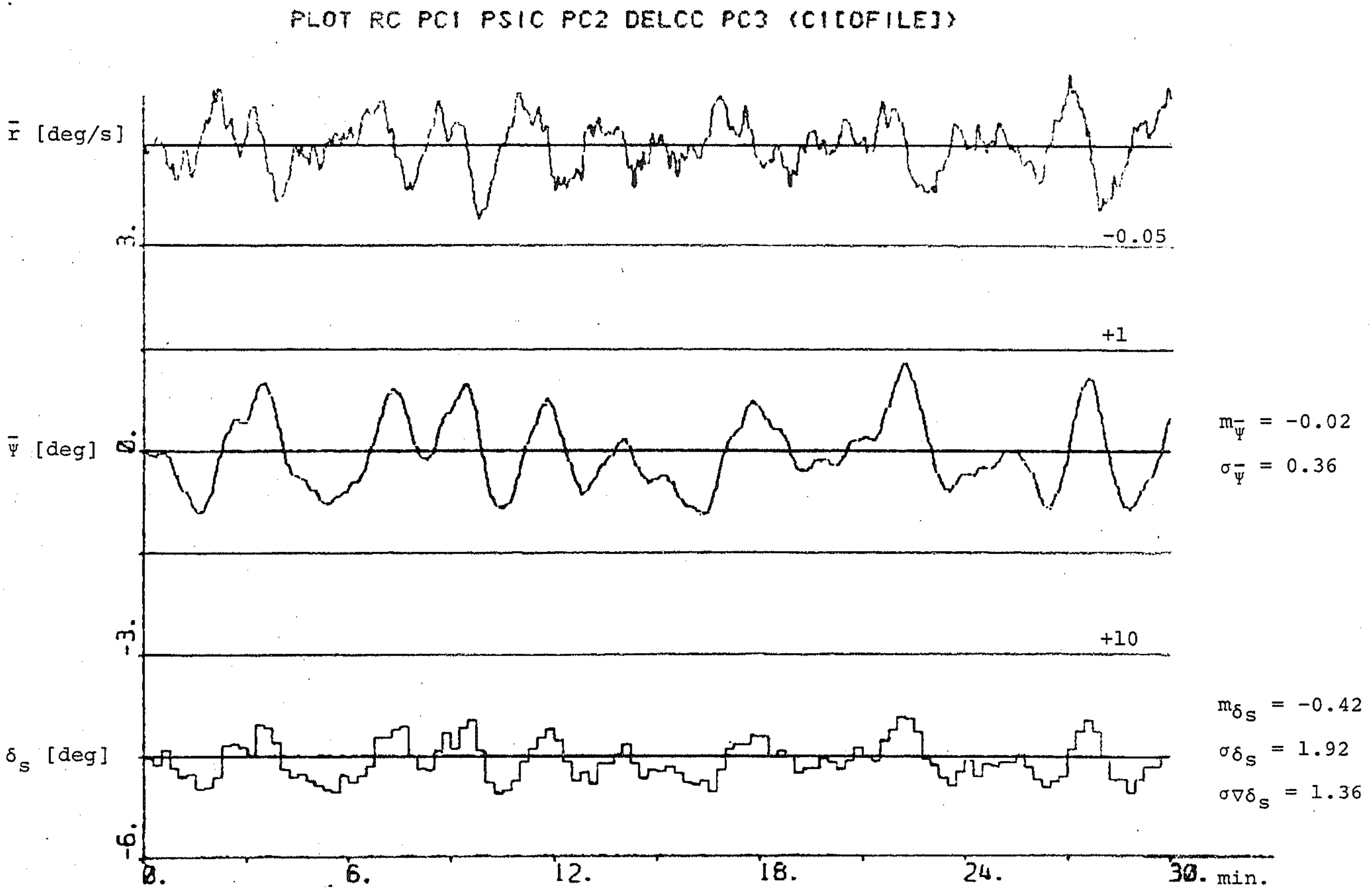


Fig. 4.13 - Regulator STURE (NA=3 NB=2 NC=0 k=3 q_2=0.2): V_1 = 0.51 V_2 = 0.50 V_3 = 0.31.

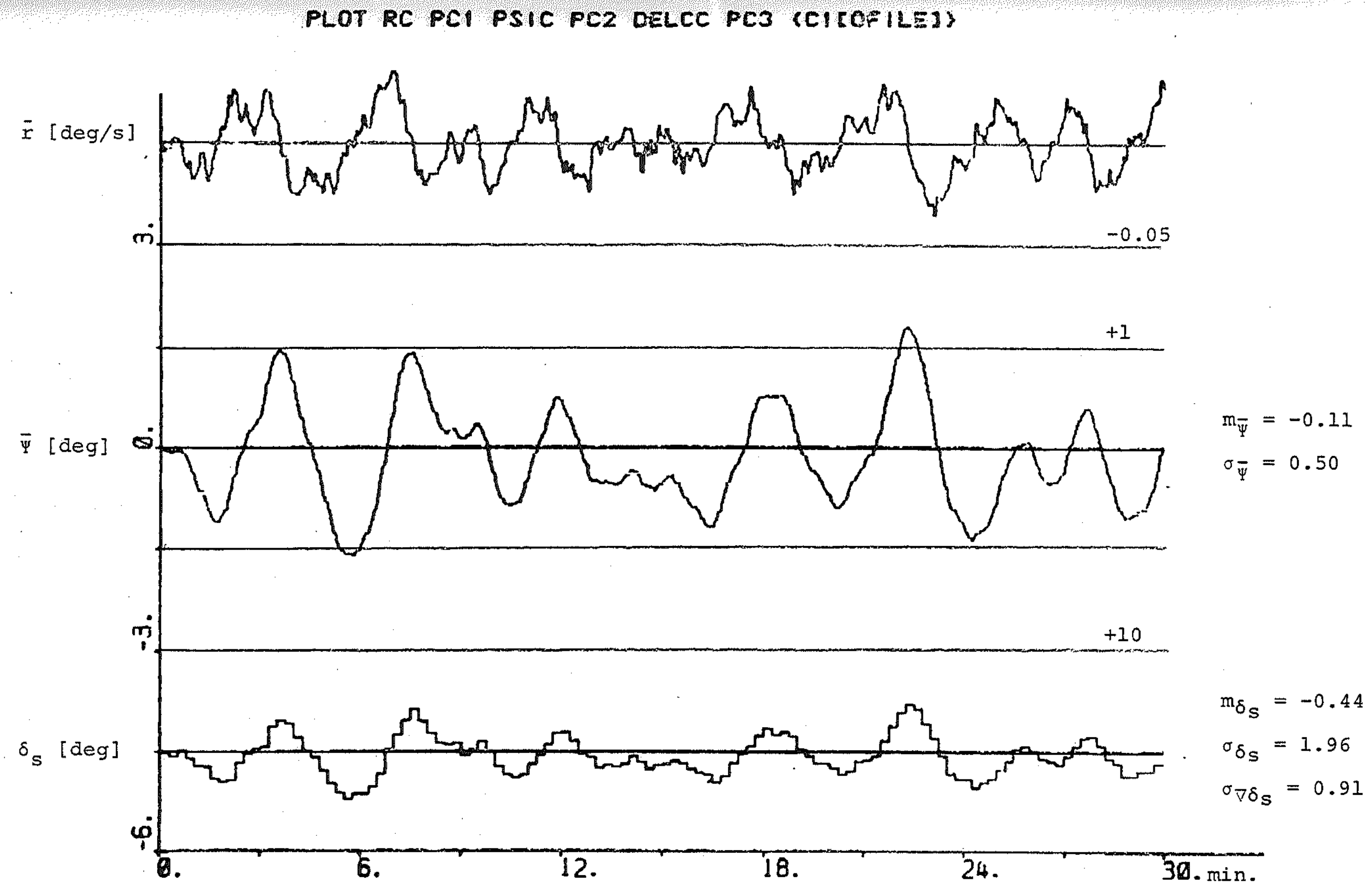


Fig. 4.14 - Regulator STURE (NA=3 NB=2 NC=0 k=3 q_2=0.5): V_1 = 0.66 V_2 = 0.64 V_3 = 0.34.

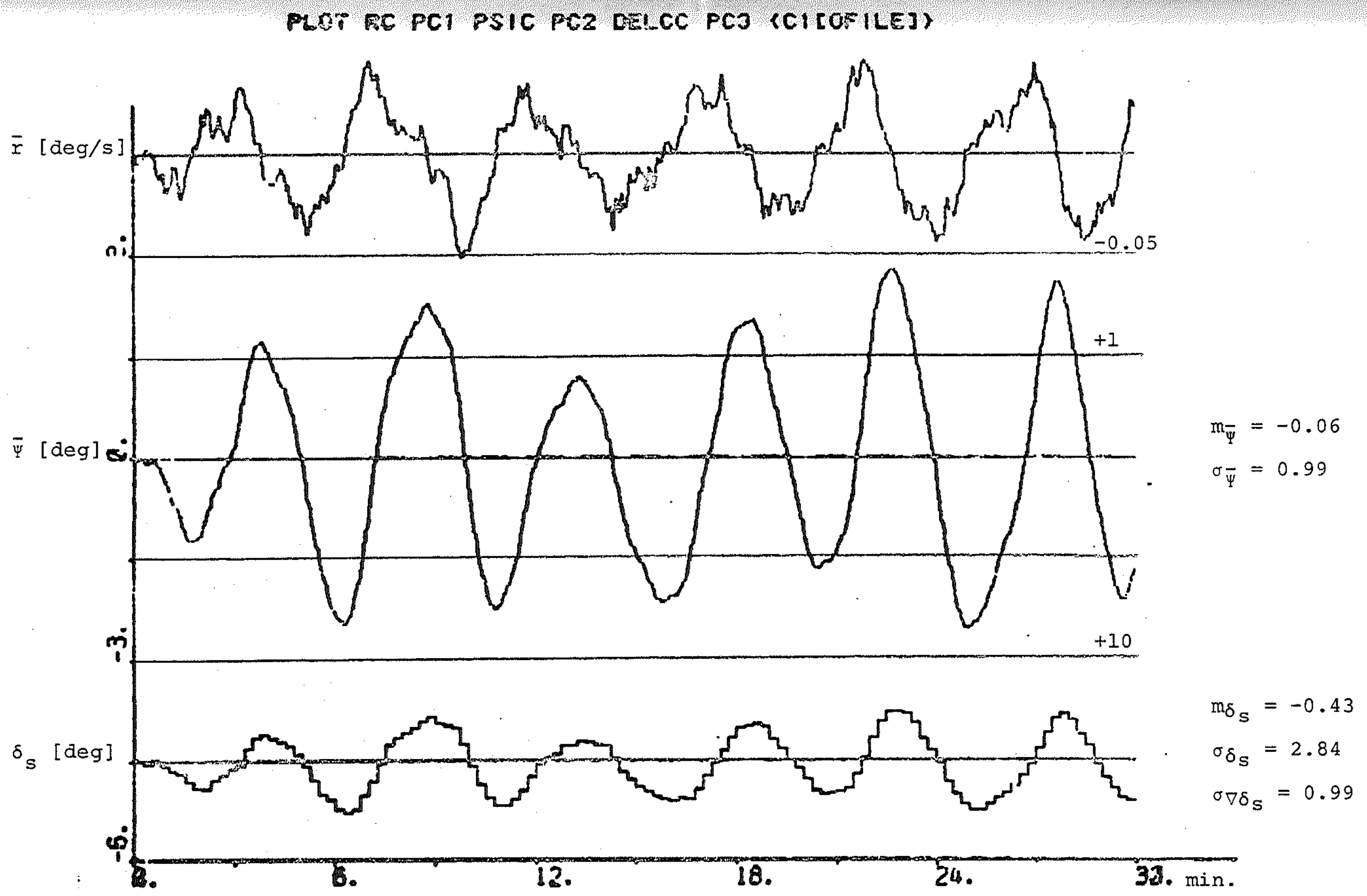


Fig. 4.15 - Regulator STURE (NA=3 NB=2 NC=0 k=3 q_2=1): V_1 = 1.80 V_2 = 1.78 V_3 = 1.07

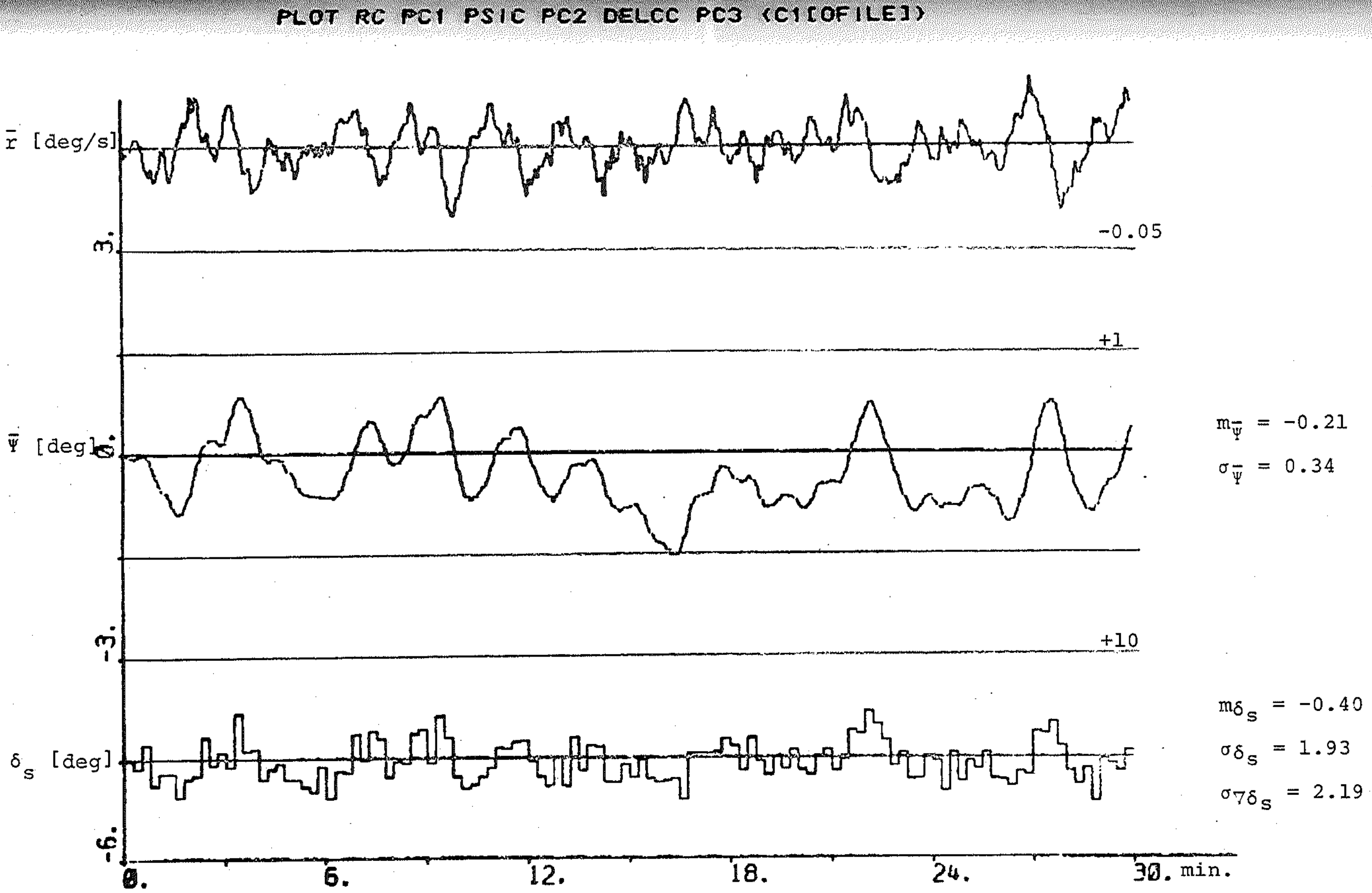


Fig. 4.16 - Regulator STURE (NA=3 NB=2 NC=0 k=4 q_2=0): $V_1 = 0.55$ $V_2 = 0.53$ $V_3 = 0.64$.

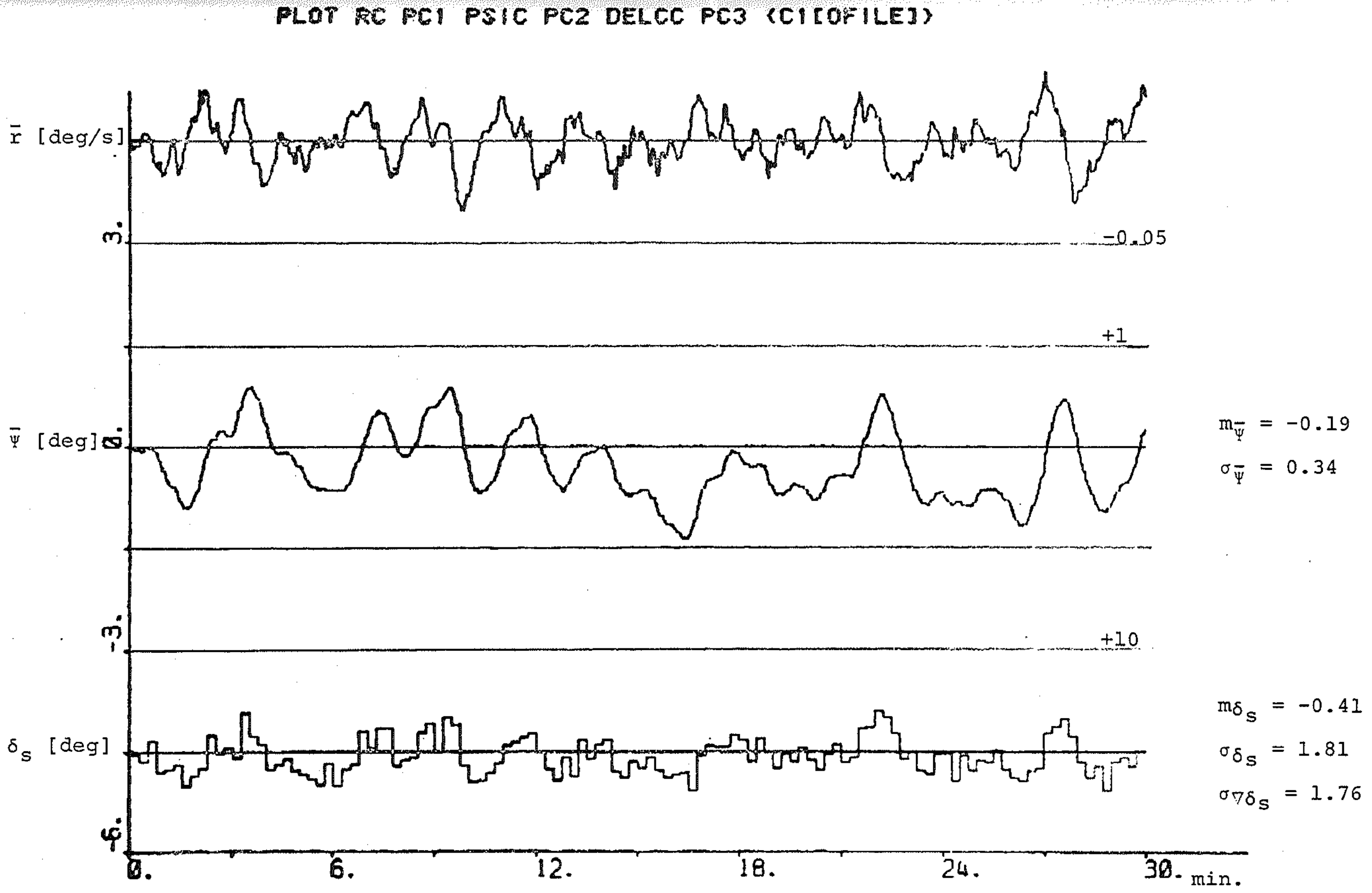


Fig. 4.17 - Regulator STURE (NA=3 NB=2 NC=0 k=4 q_2=0.05): V_1 = 0.50 V_2 = 0.48 V_3 = 0.46

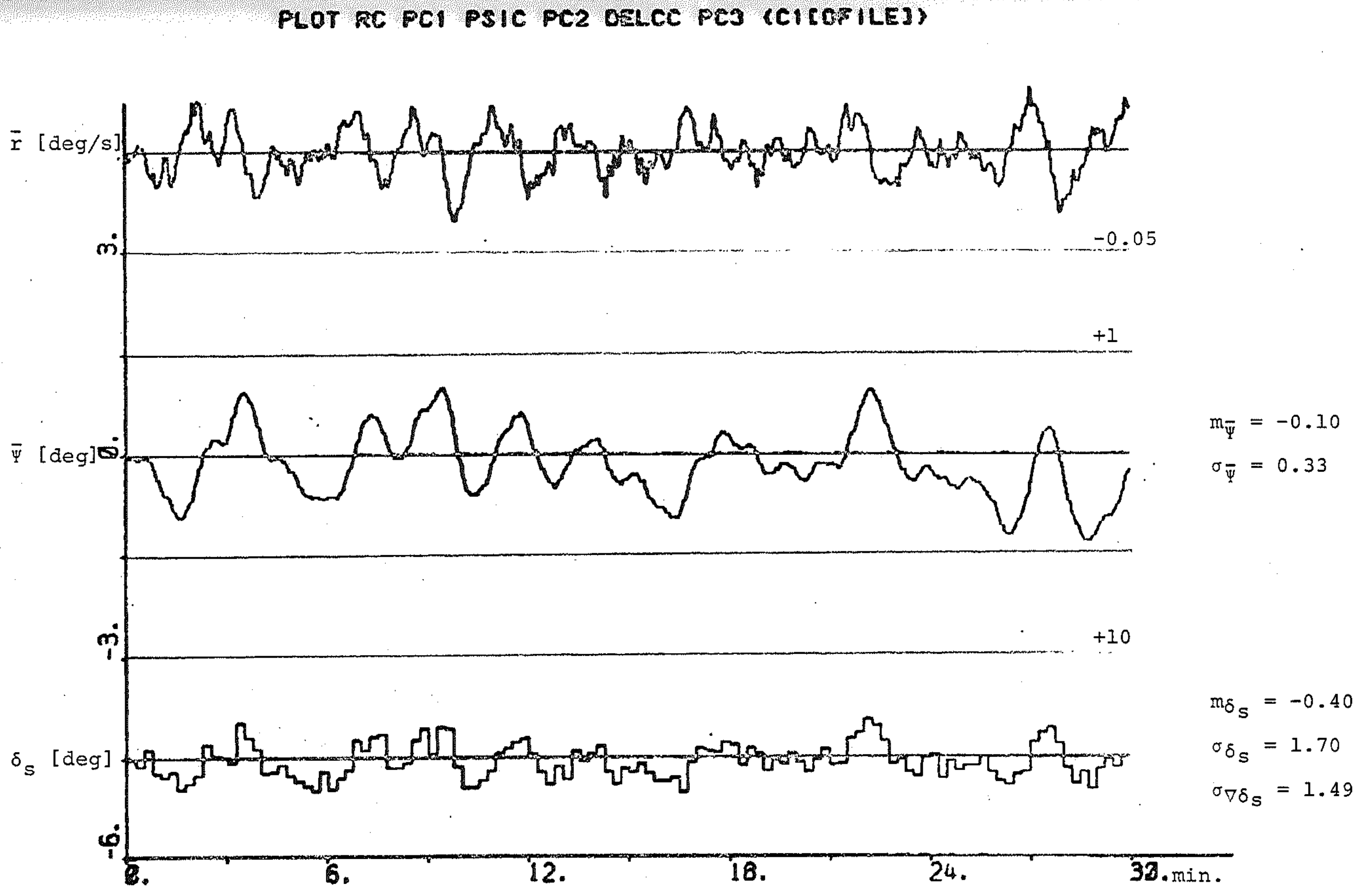


Fig. 4.18 - Regulator STURE (NA=3 NB=2 NC=0 k=4 q_2=0.1): $V_1 = 0.43$ $V_2 = 0.41$ $V_3 = 0.35$

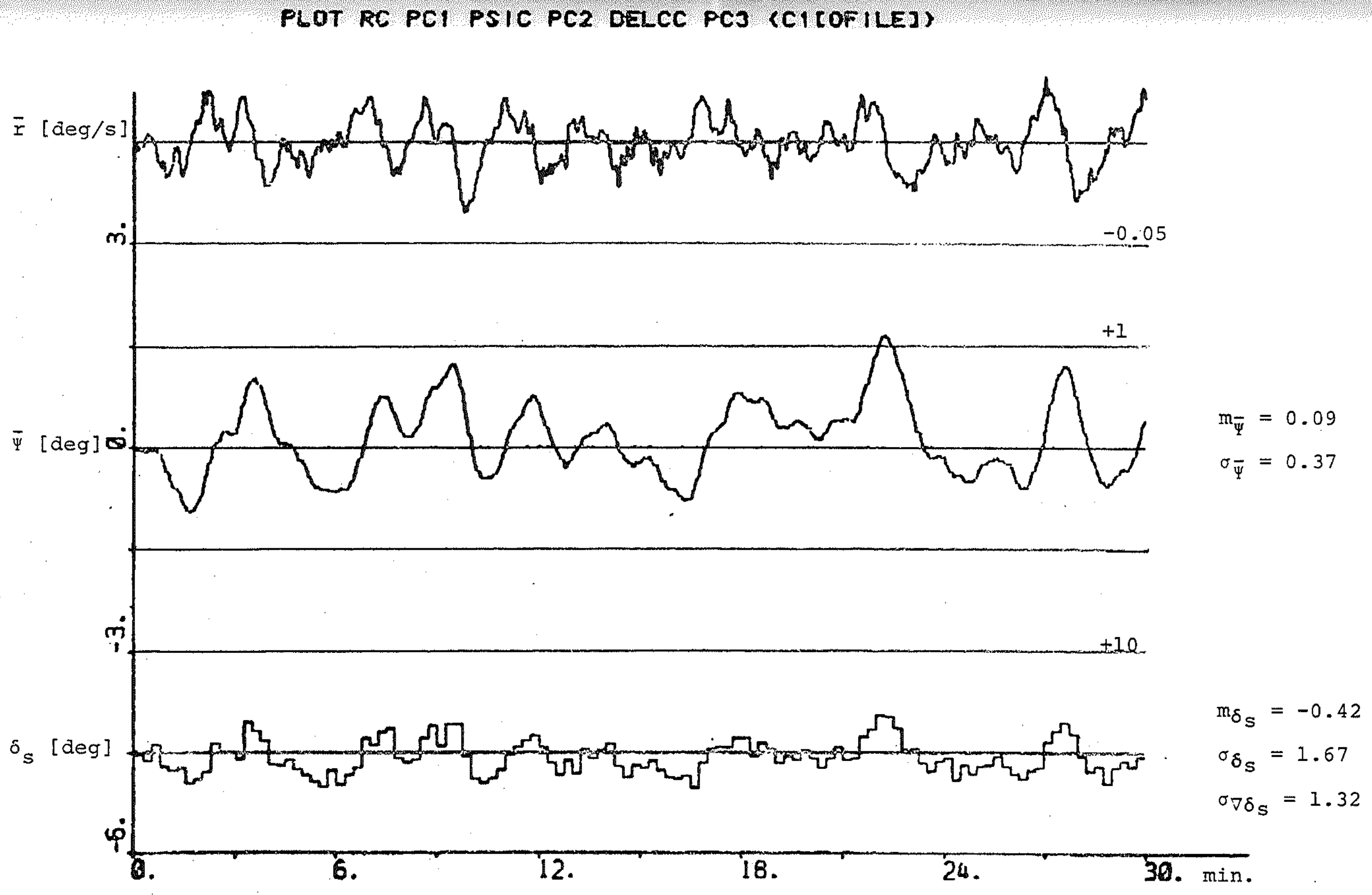


Fig. 4.19 - Regulator STURE (NA=3 NB=2 NC=0 k=4 q_2=0.2): $V_1 = 0.44$ $V_2 = 0.42$ $V_3 = 0.32$.

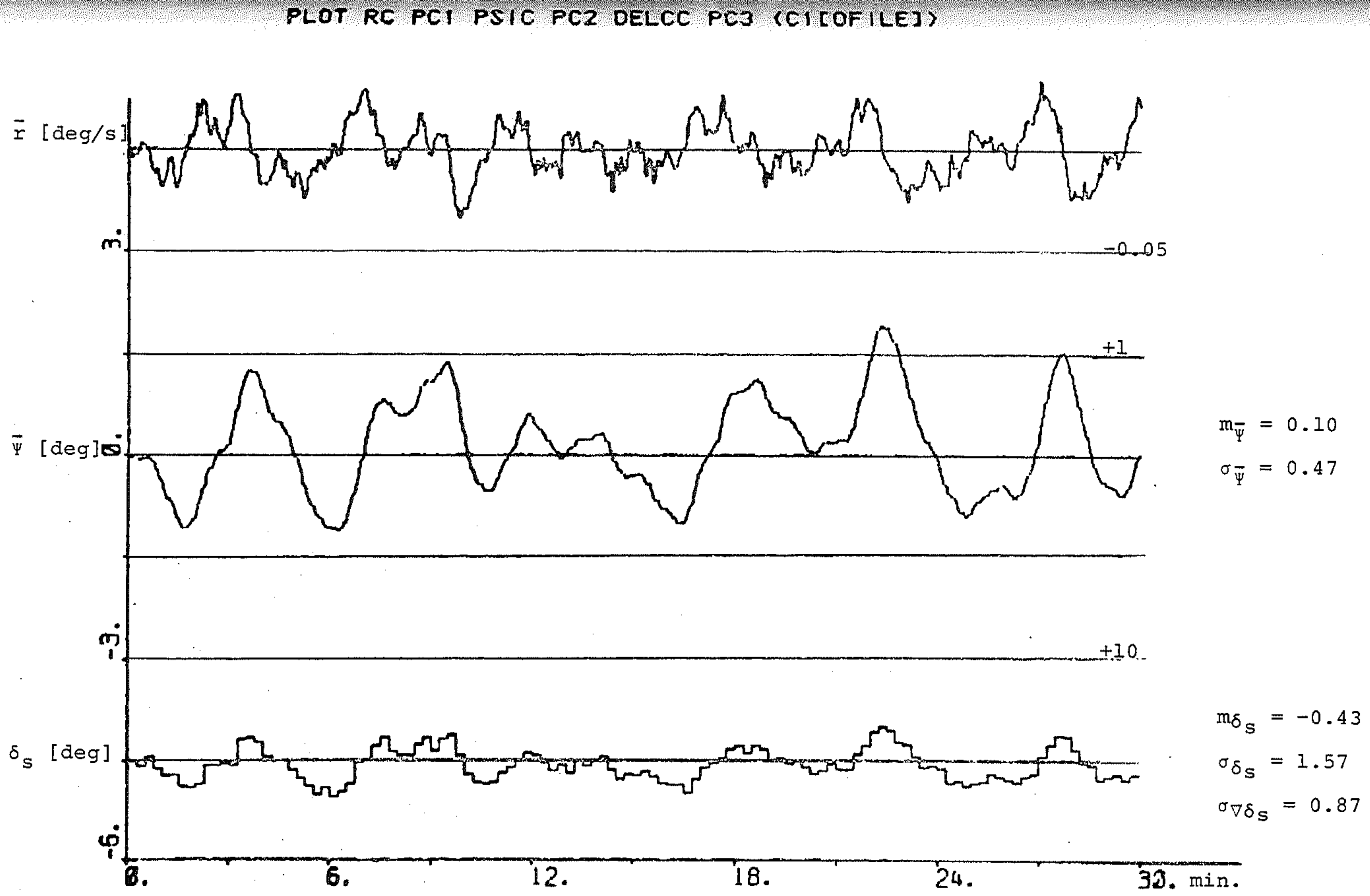


Fig. 4.20 - Regulator STURE (NA=3 NB=2 NC=0 k=4 q_2=0.5): V_1 = 0.50 V_2 = 0.48 V_3 = 0.31.

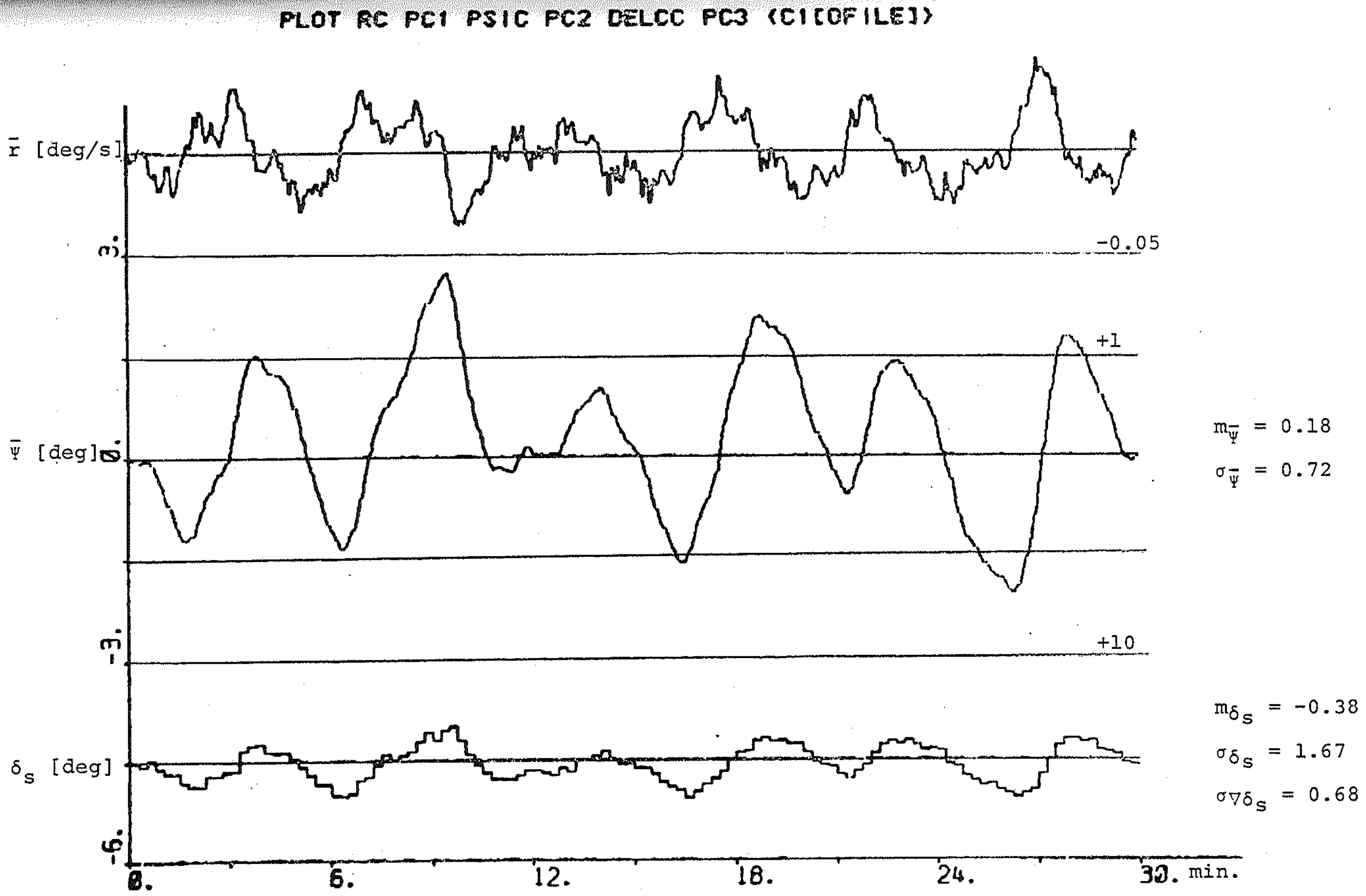
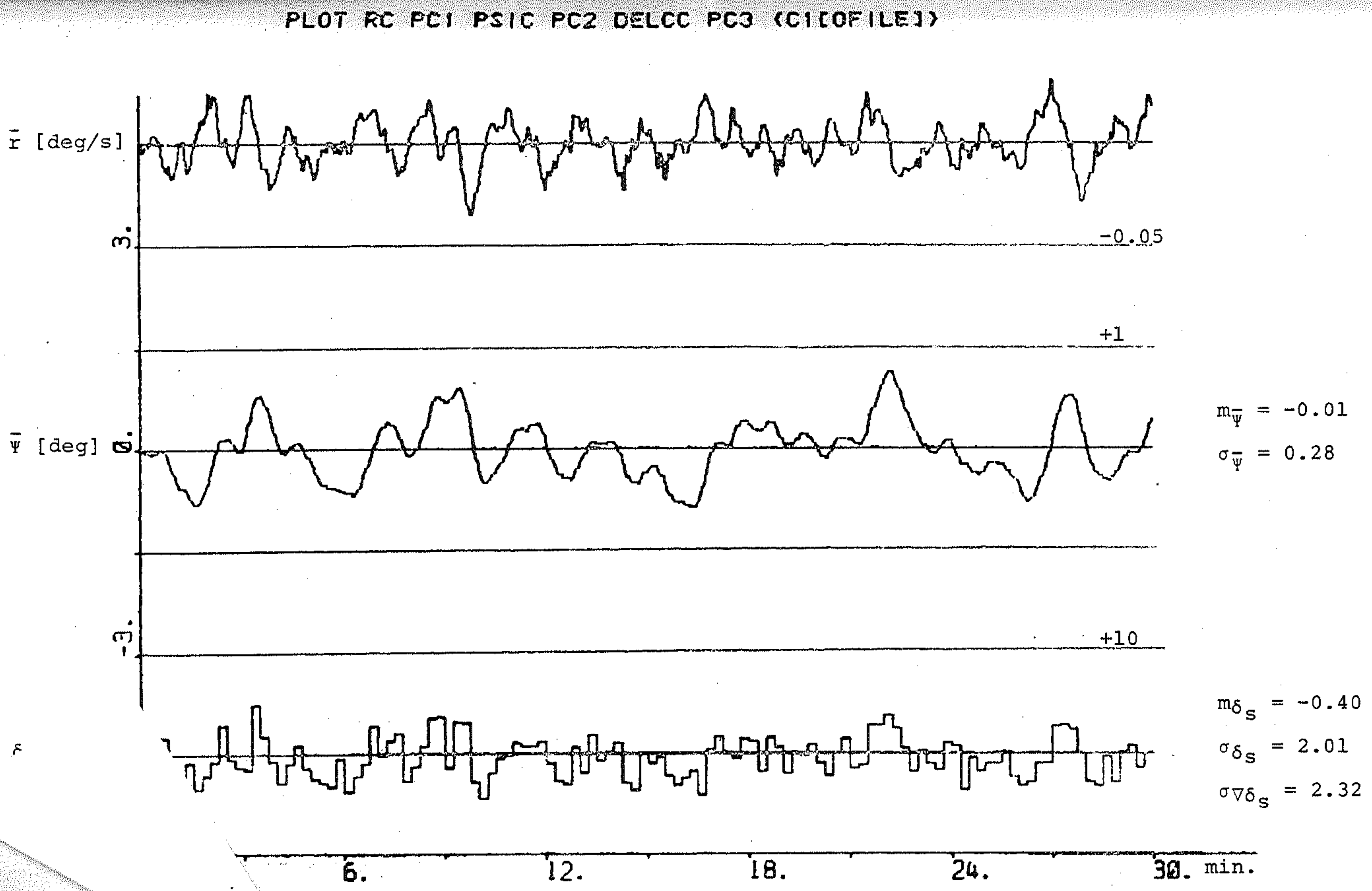
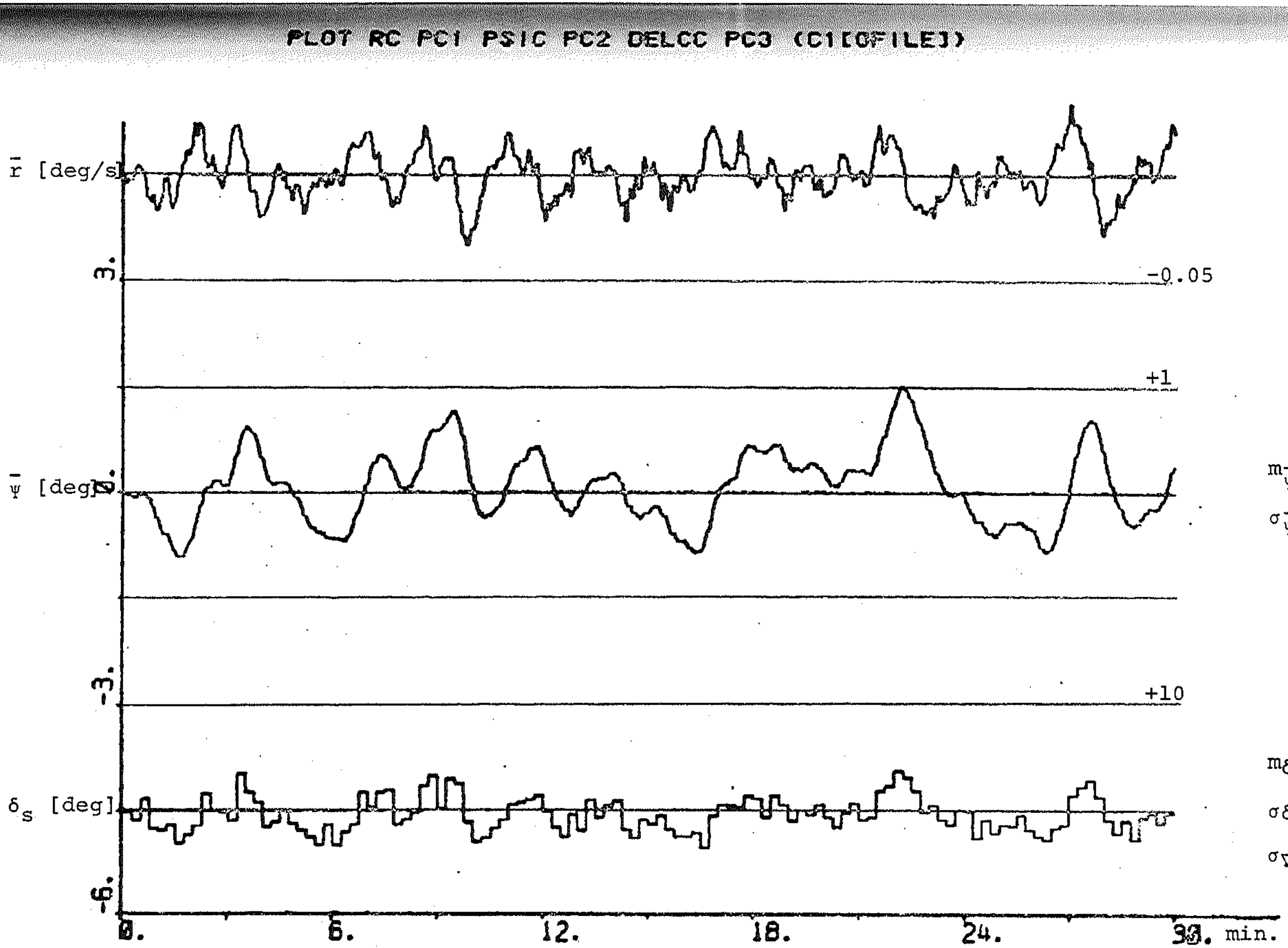


Fig. 4.21 - Regulator STURE (NA=3 NB=2 NC=0 k=4 $q_2=1$): $V_1 = 0.84$ $V_2 = 0.83$ $V_3 = 0.59$.



<u>Fig. 4.23</u> - STURE (NA=3 NB=2 NC=0 k=5 q_2=0.1): V_1 = 0.50 V_2 = 0.49 V_3 = 0.62.



$m_{\bar{\psi}} = 0.06$

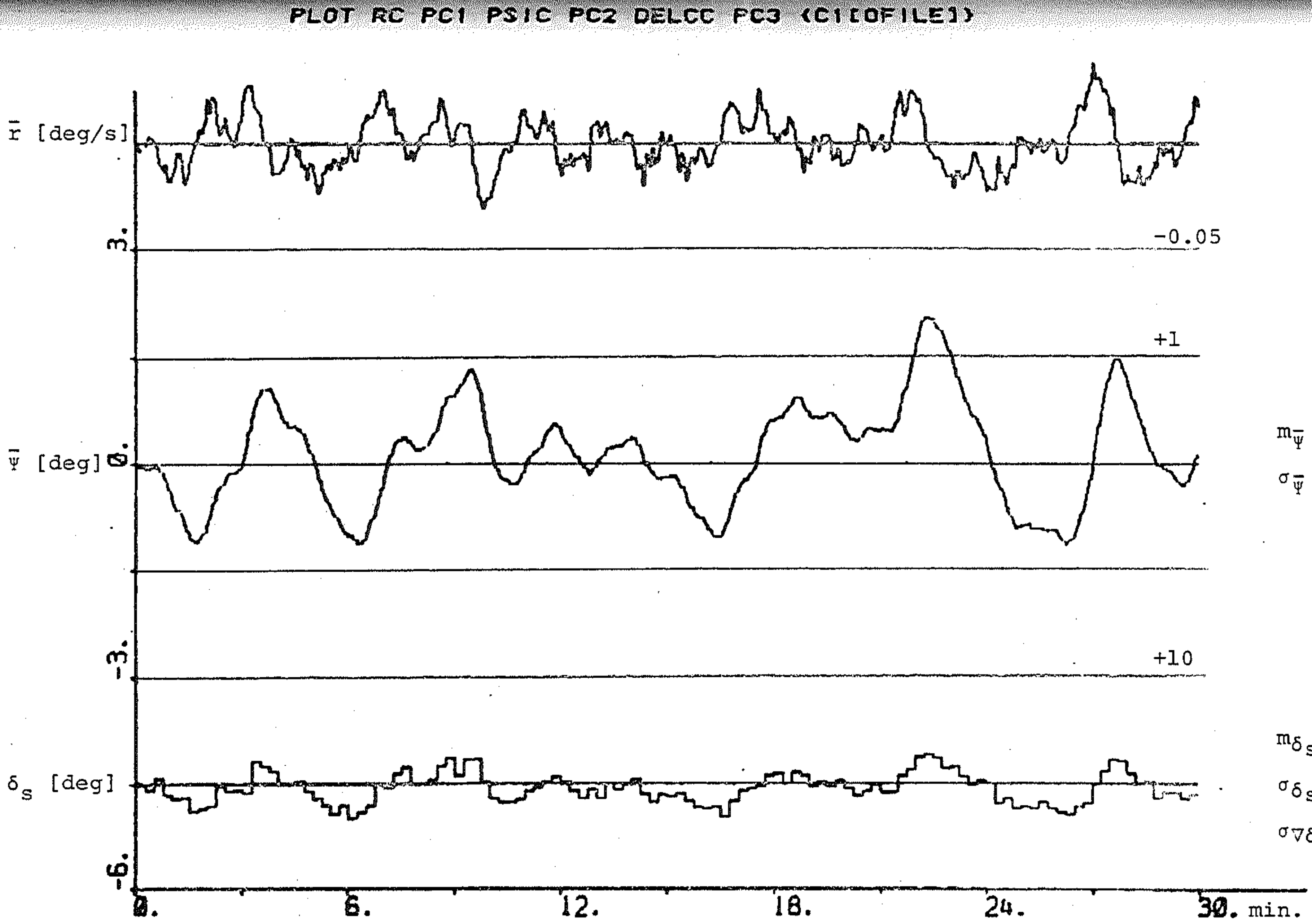
$\sigma_{\bar{\psi}} = 0.35$

$m\delta_s = -0.41$

$\sigma\delta_s = 1.66$

$\sigma\nabla\delta_s = 1.56$

Fig. 4.23 - Regulator STURE (NA=3 NB=2 NC=0 k=5 q_2=0.2): V_1 = 0.42 V_2 = 0.40 V_3 = 0.37.



$m_{\bar{\Psi}} = 0.10$

$\sigma_{\bar{\Psi}} = 0.48$

$m_{\delta_s} = -0.41$

$\sigma_{\delta_s} = 1.40$

$\sigma_{\nabla\delta_s} = 0.90$

Fig. 4.24 - Regulator STURE (NA=3 NB=2 NC=0 k=5 q_2=0.5): $V_1 = 0.46$ $V_2 = 0.44$ $V_3 = 0.32$.

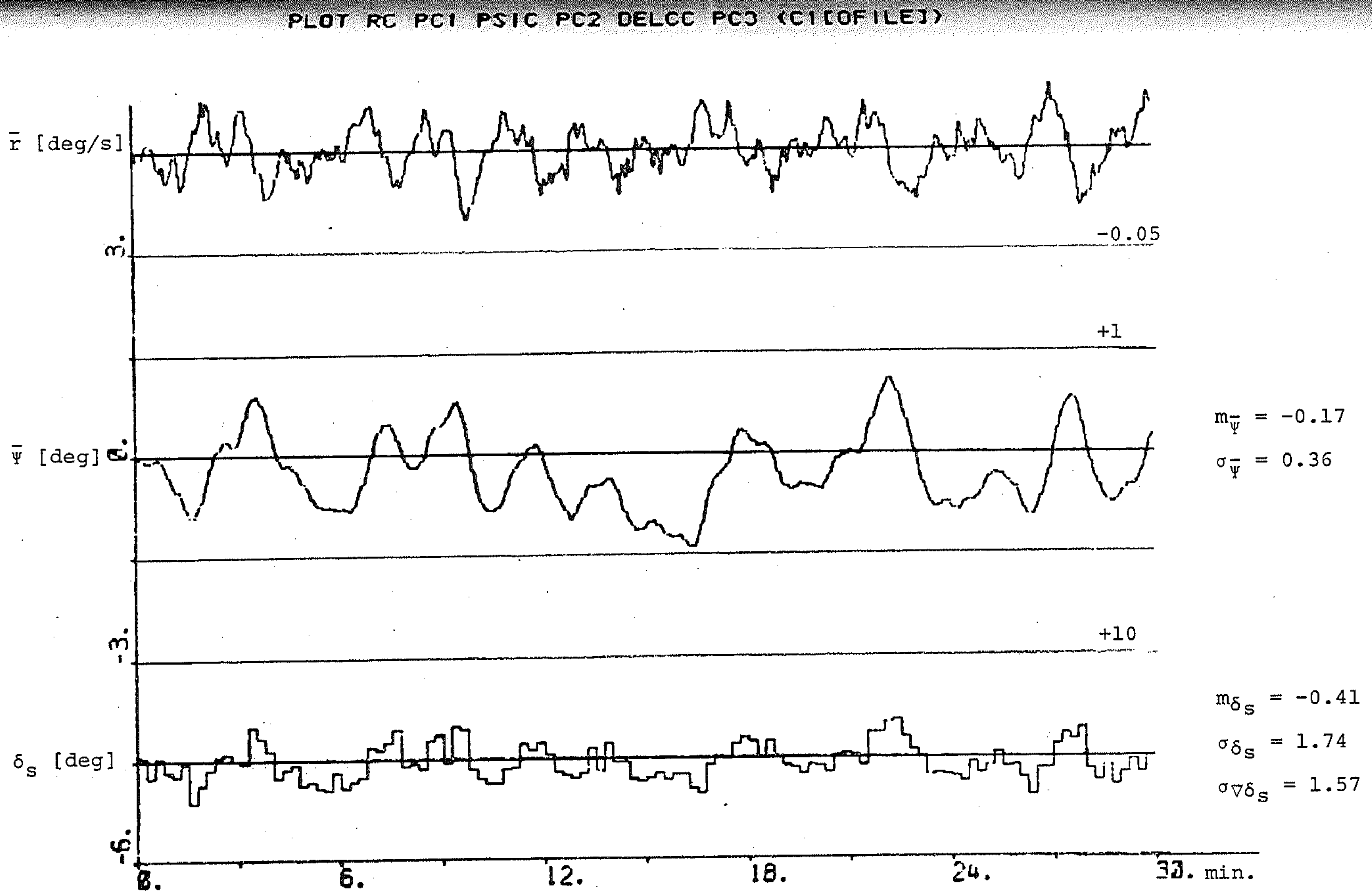


Fig. 4.25 - Regulator STURE (NA=3 NB=2 NC=1 k=4 q_2=0.1): $V_1 = 0.48$ $V_2 = 0.46$ $V_3 = 0.40$.

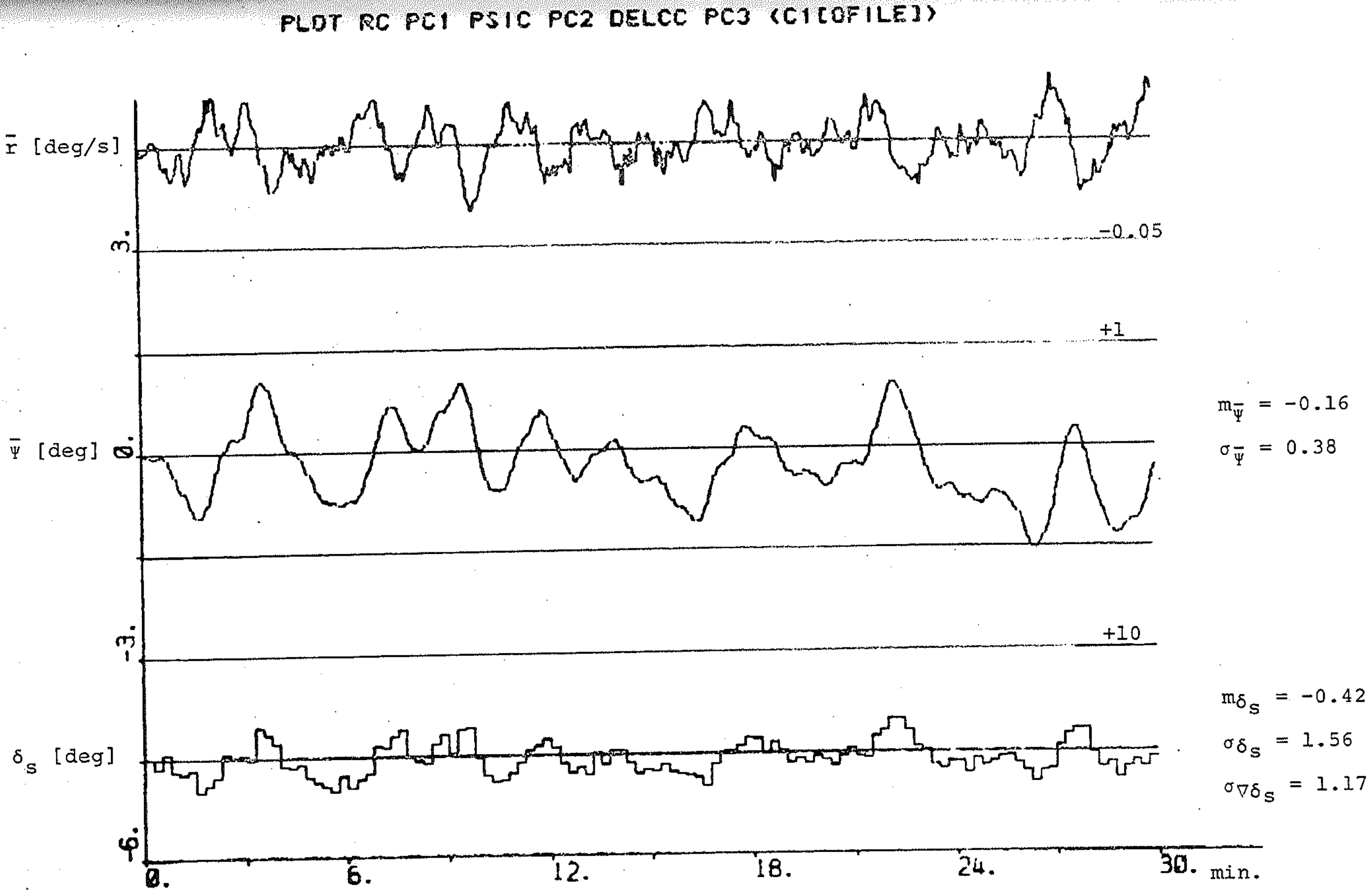


Fig. 4.26 - Regulator STURE (NA=3 NB=2 NC=1 k=4 q_2=0.2): V_1 = 0.43 V_2 = 0.41 V_3 = 0.30.

Regulator STURE3 is used with varying values of NA, NB, k and q_2 in the simulations of Figs. 4.27-4.41. A summary is given in Table 4.3. The results should be considered with some caution, because the parameters of some experiments have not converged properly before the simulation. However, it can be concluded that the steering quality is increased by changing NA = 3, NB = 3 to NA = 4, NB = 4. The least squares identification method, which is used in STURE3, gives in general biased parameter estimates when the disturbances are coloured noise. By increasing NA and NB it is possible to decrease the effects of the coloured noise. The proper way, however, is to replace the least squares method with the maximum likelihood method. When the simple self-tuning regulator STURE is used, the biased parameters obtained from the least squares identification and the special minimum variance controller together will give the optimal performance, although the disturbances are coloured noise.

NA	NB	k	q_2	$\sigma_{\bar{\Psi}}$ [deg]	σ_{δ_s} [deg]	$\sigma_{V\delta_s}$ [deg]	V_1	V_2	V_3	Fig.
3	3	0	0.05	0.51	4.50	4.22	2.44	2.42	2.17	4.27
3	3	0	0.1	0.84	4.46	3.08	2.78	2.77	1.72	4.28
3	3	0	0.2	0.53	2.84	2.04	1.16	1.14	0.75	4.29
3	3	1	0.02	0.43	2.54	1.97	0.86	0.84	0.59	4.30
3	3	1	0.05	0.52	2.51	1.79	0.94	0.92	0.61	4.31
3	3	1	0.1	0.54	2.72	1.89	1.14	1.12	0.74	4.32
3	3	1	0.2	0.55	2.30	1.36	0.93	0.91	0.57	4.33
3	3	1	0.5	0.64	2.25	1.20	1.07	1.05	0.69	4.34
3	3	2	0.05	2.51	6.23	5.68	11.25	11.24	10.60	4.35
3	3	2	0.1	0.78	2.83	2.02	1.45	1.43	1.04	4.36
4	4	1	0.02	0.40	2.62	2.24	0.89	0.87	0.69	4.37
4	4	1	0.05	0.41	2.33	1.79	0.77	0.75	0.53	4.38
4	4	1	0.1	0.44	2.20	1.55	0.75	0.73	0.48	4.39
4	4	1	0.2	0.48	2.10	1.35	0.74	0.72	0.47	4.40
4	4	1	0.5	0.60	2.09	1.21	0.90	0.88	0.59	4.41

Table 4.3 - Regulator STURE3 with varying values of NA, NB, k and q_2.

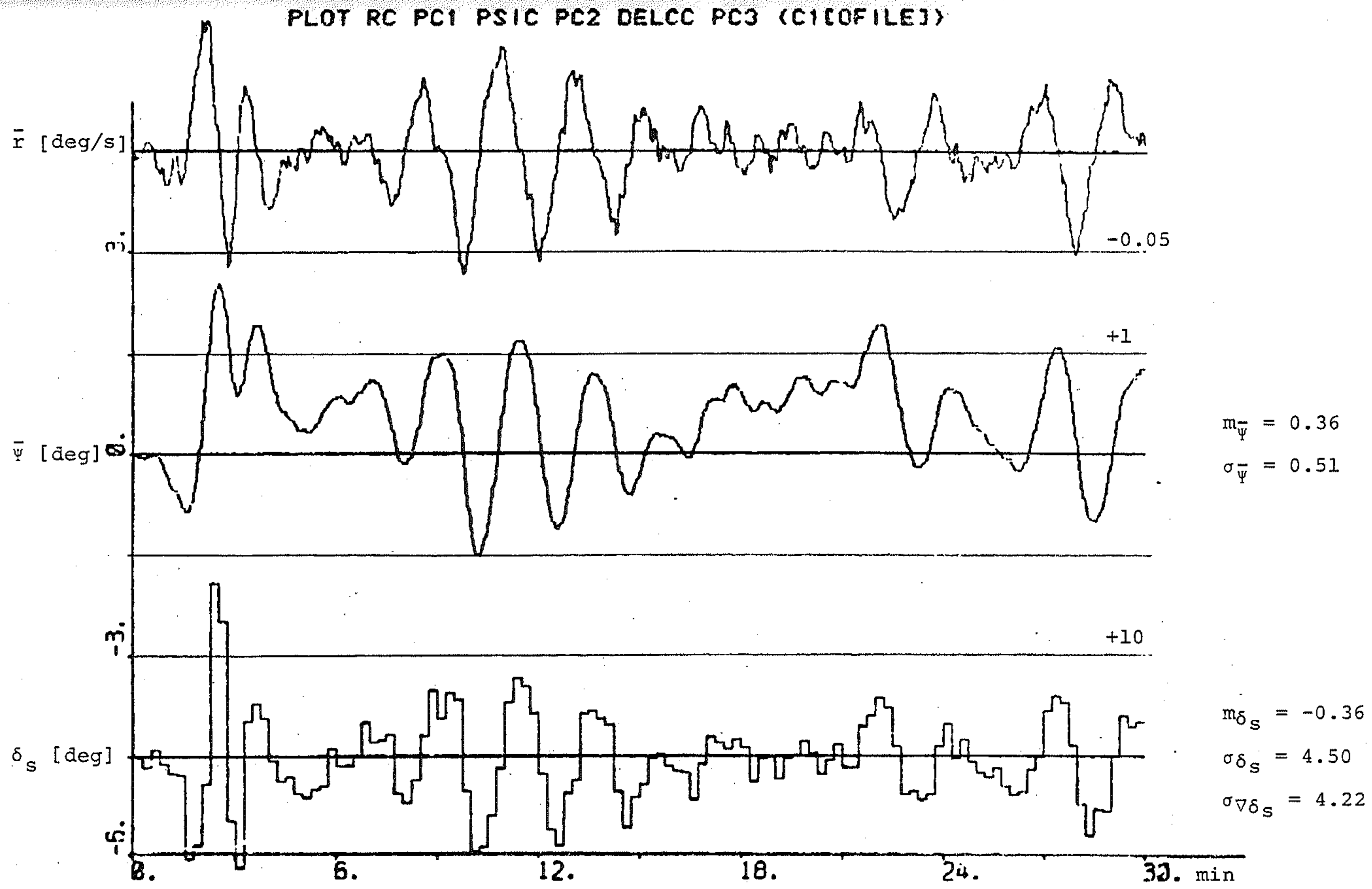


Fig. 4.27 - Regulator STURE3 (NA=3 NB=3 k=0 q_2=0.05): V_1 = 2.44 V_2 = 2.42 V_3 = 2.17.

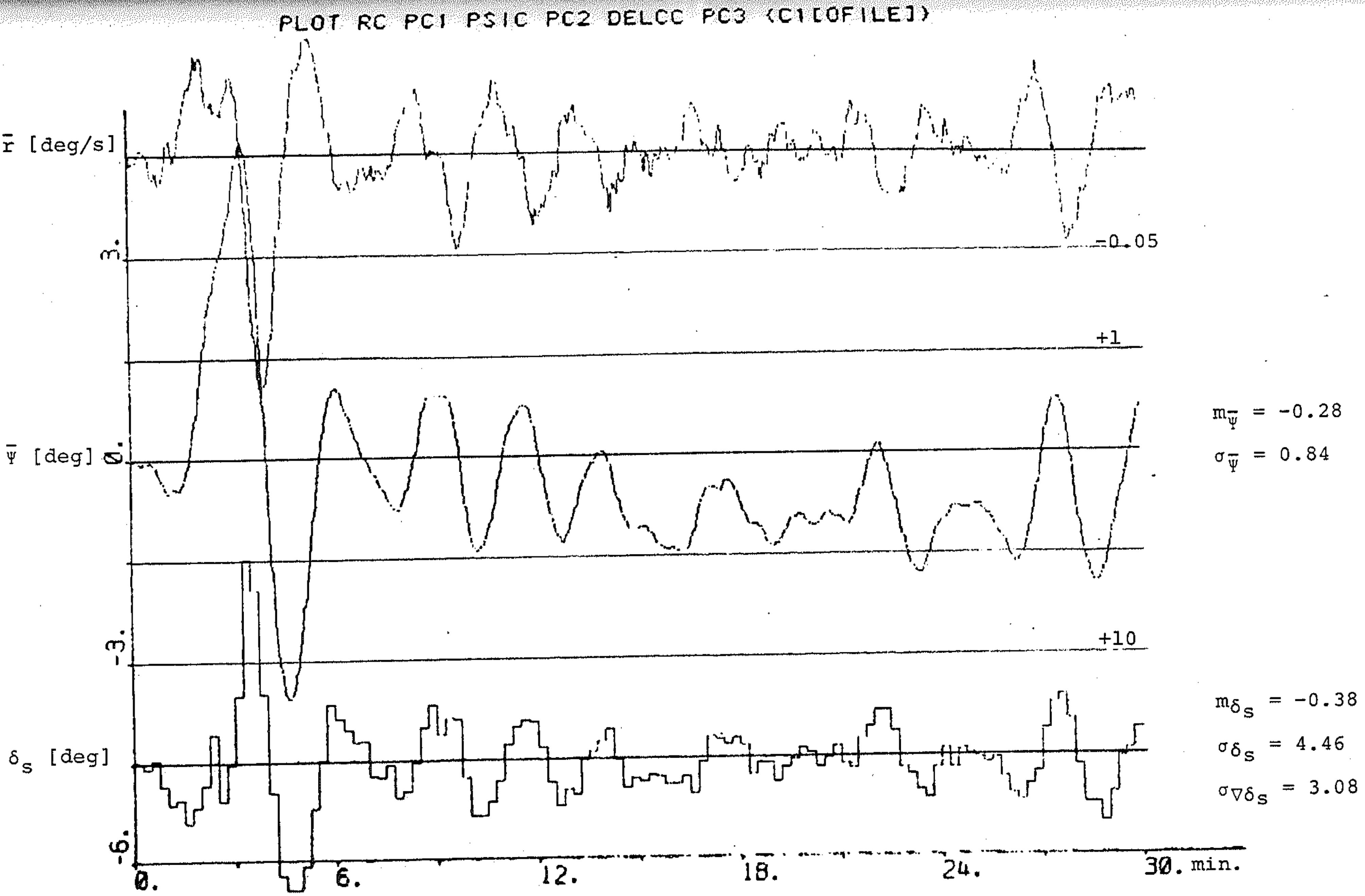


Fig. 4.28 - Regulator STURE3 (NA=3 NB=3 k=0 q_2=0.1): V_1 = 2.78 V_2 = 2.77 V_3 = 1.72.

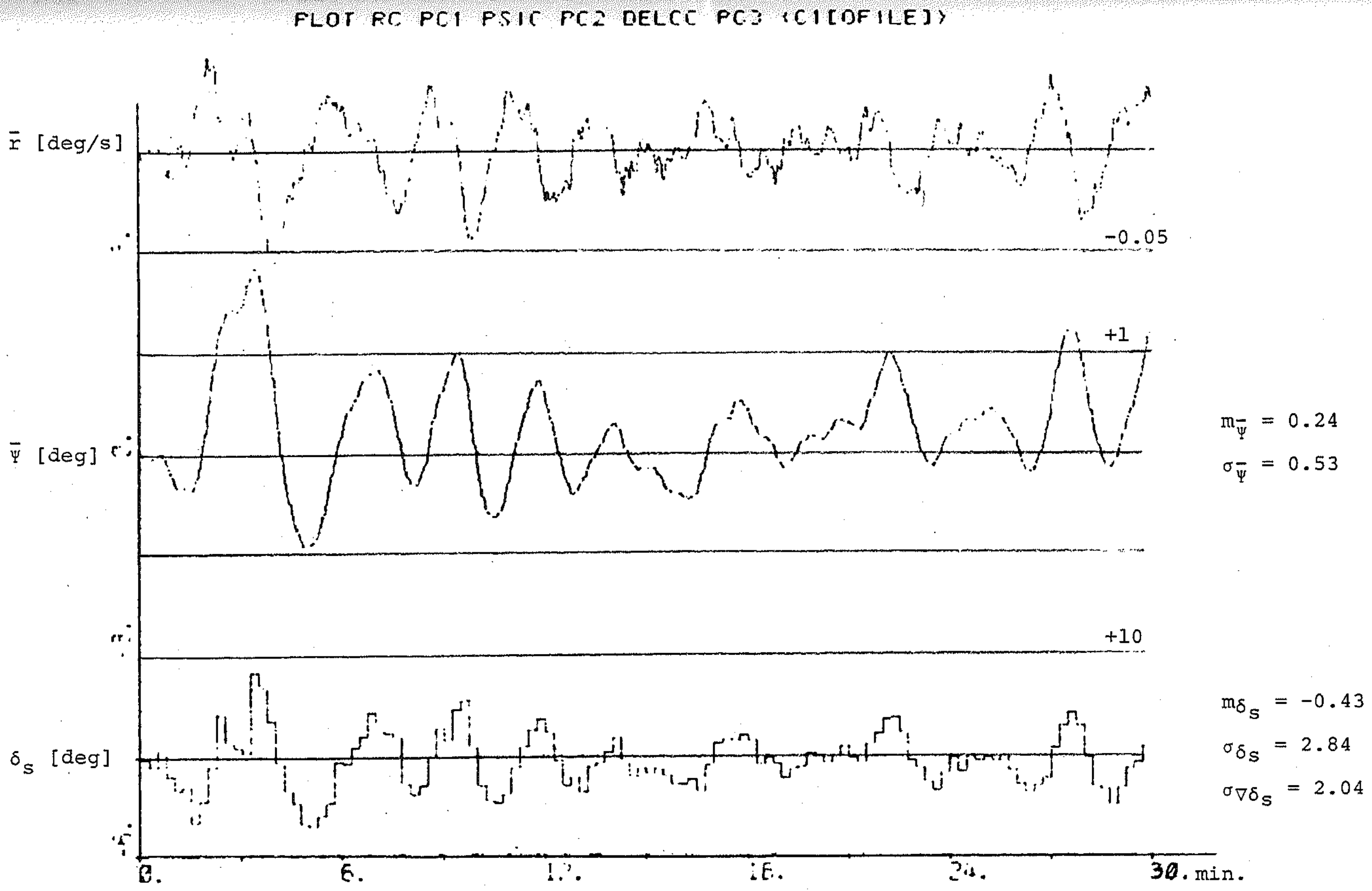


Fig. 4.29 - Regulator STURE3 (NA=3 NB=3 k=0 q_2=0.2): $V_1 = 1.16$ $V_2 = 1.14$ $V_3 = 0.75$.

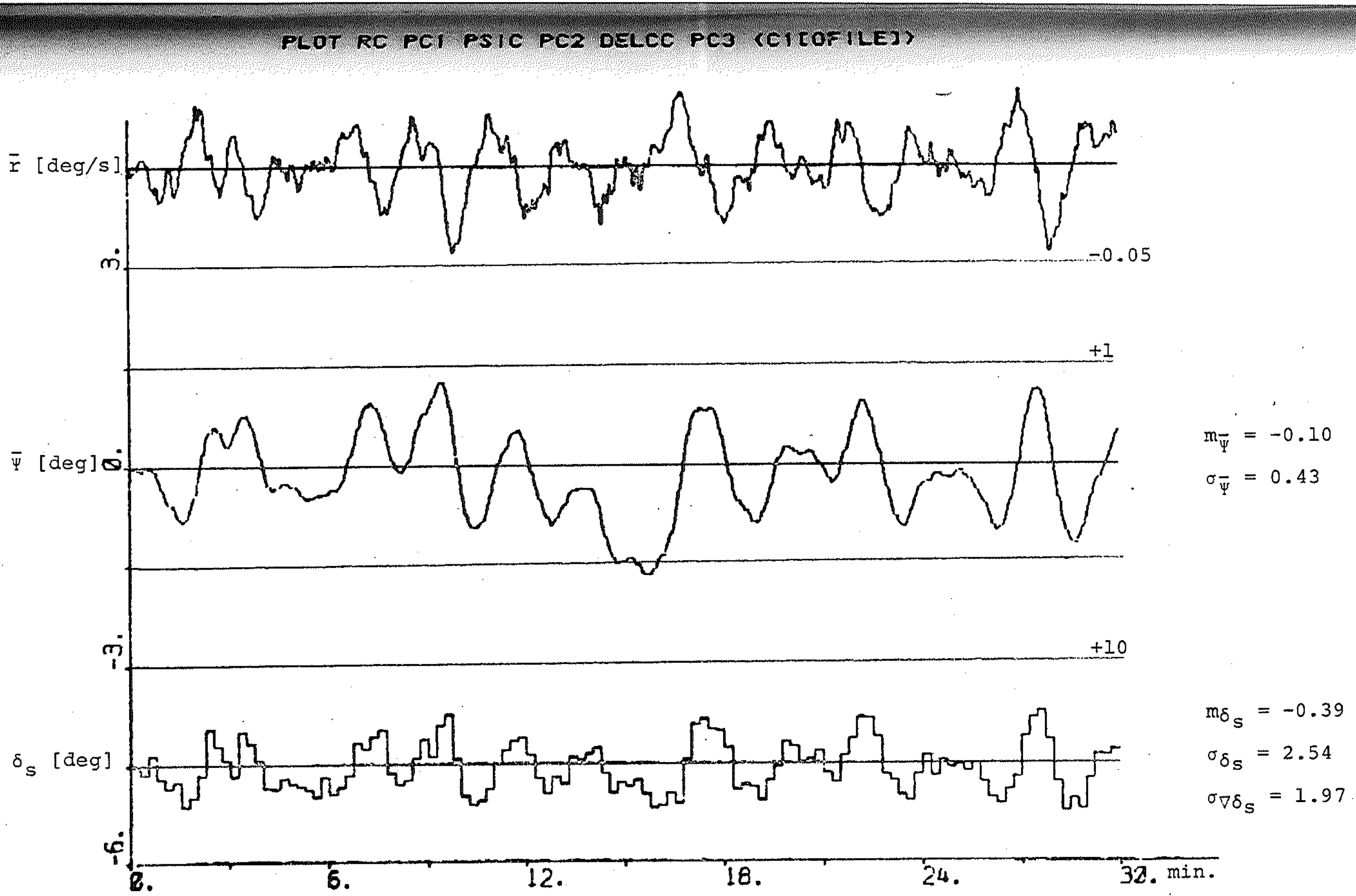


Fig. 4.30 - Regulator STURE3 (NA=3 NB=3 k=1 q_2=0.02): V_1 = 0.86 V_2 = 0.84 V_3 = 0.59.

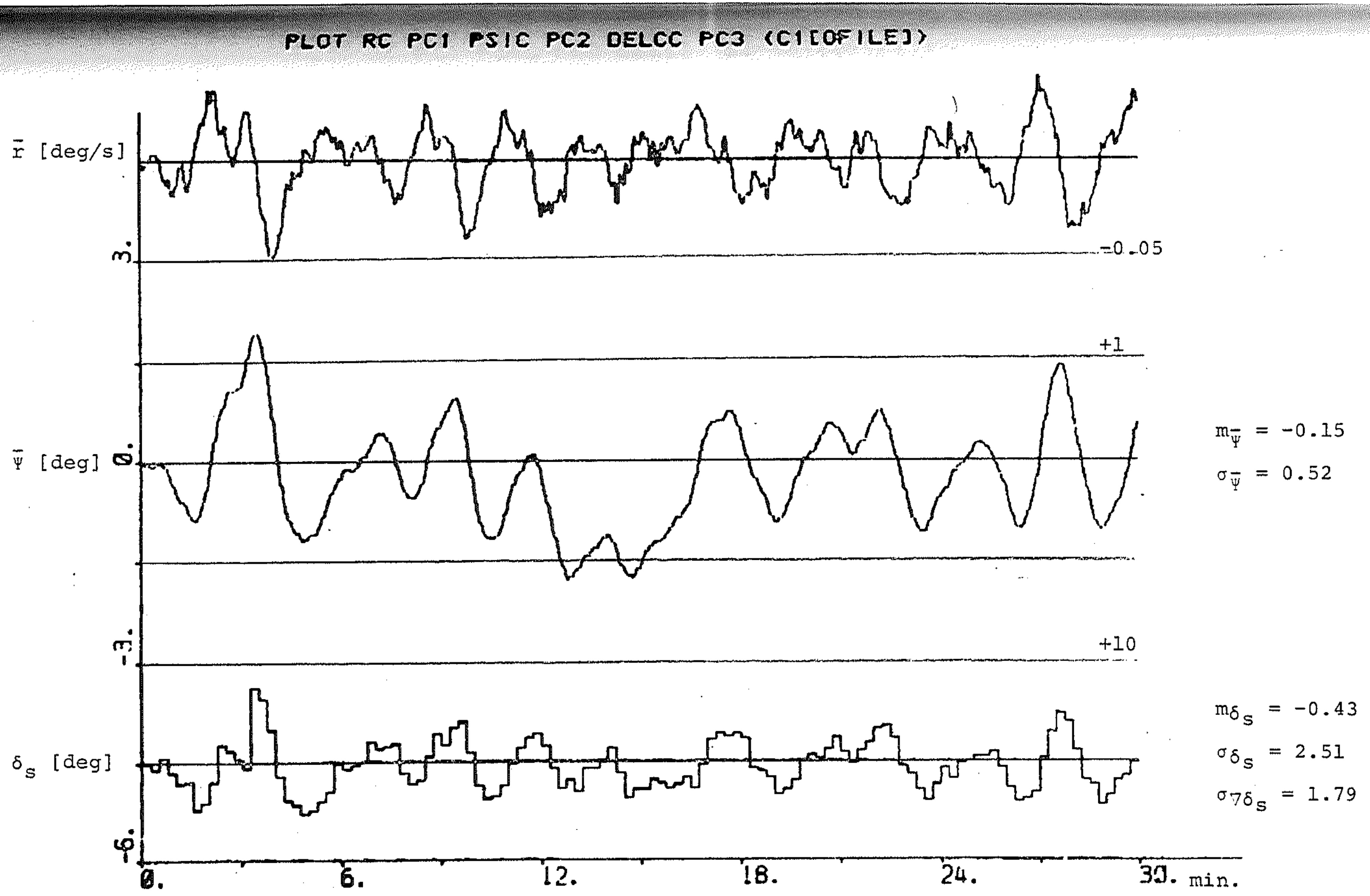


Fig. 4.31 - Regulator STURE3 (NA=3 NB=3 k=1 q_2=0.05): V_1 = 0.94 V_2 = 0.92 V_3 = 0.61.

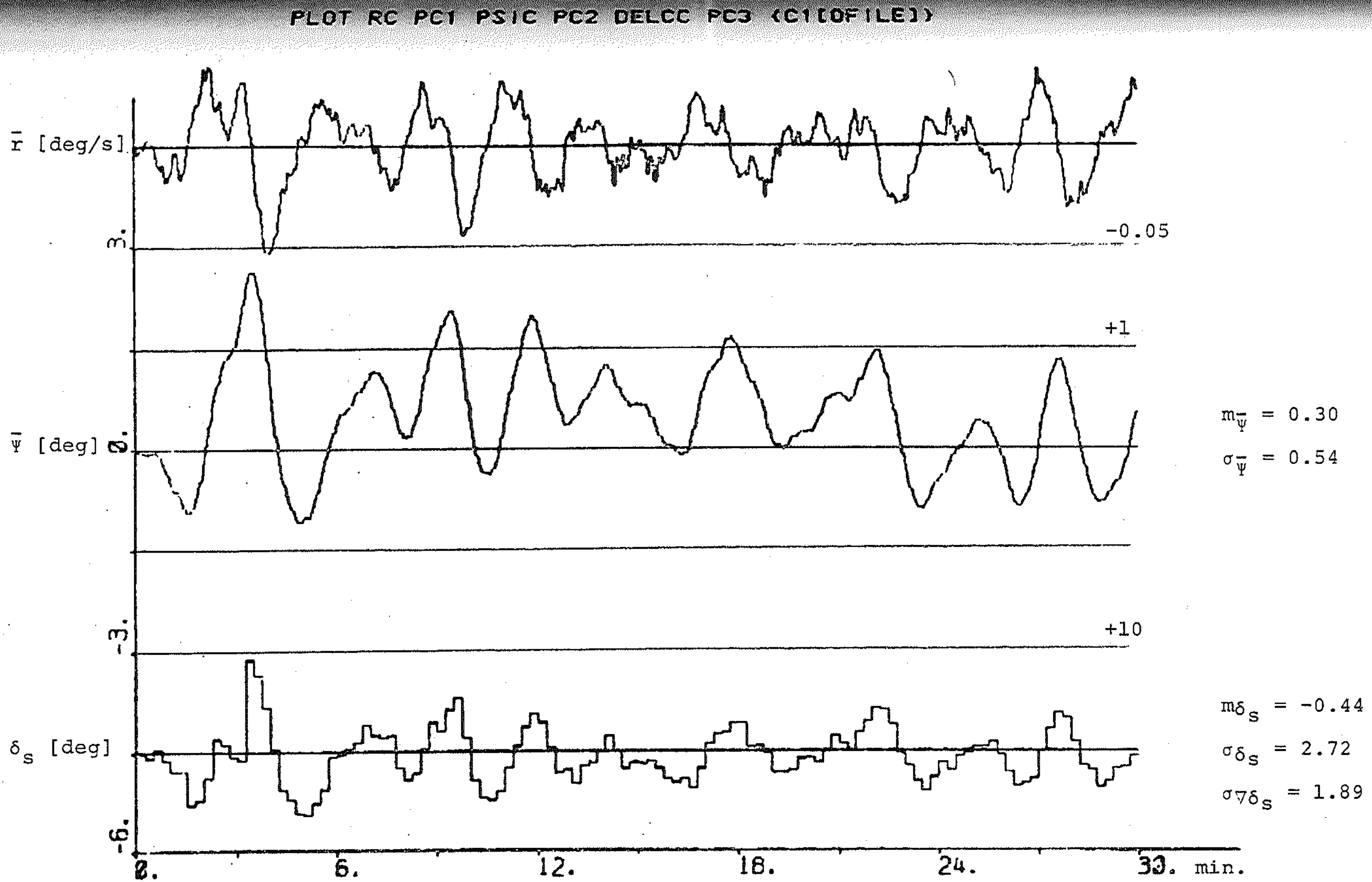


Fig. 4.32 - Regulator STURE3 (NA=3 NB=3 k=1 q_2=0.1): V_1 = 1.14 V_2 = 1.12 V_3 = 0.74.

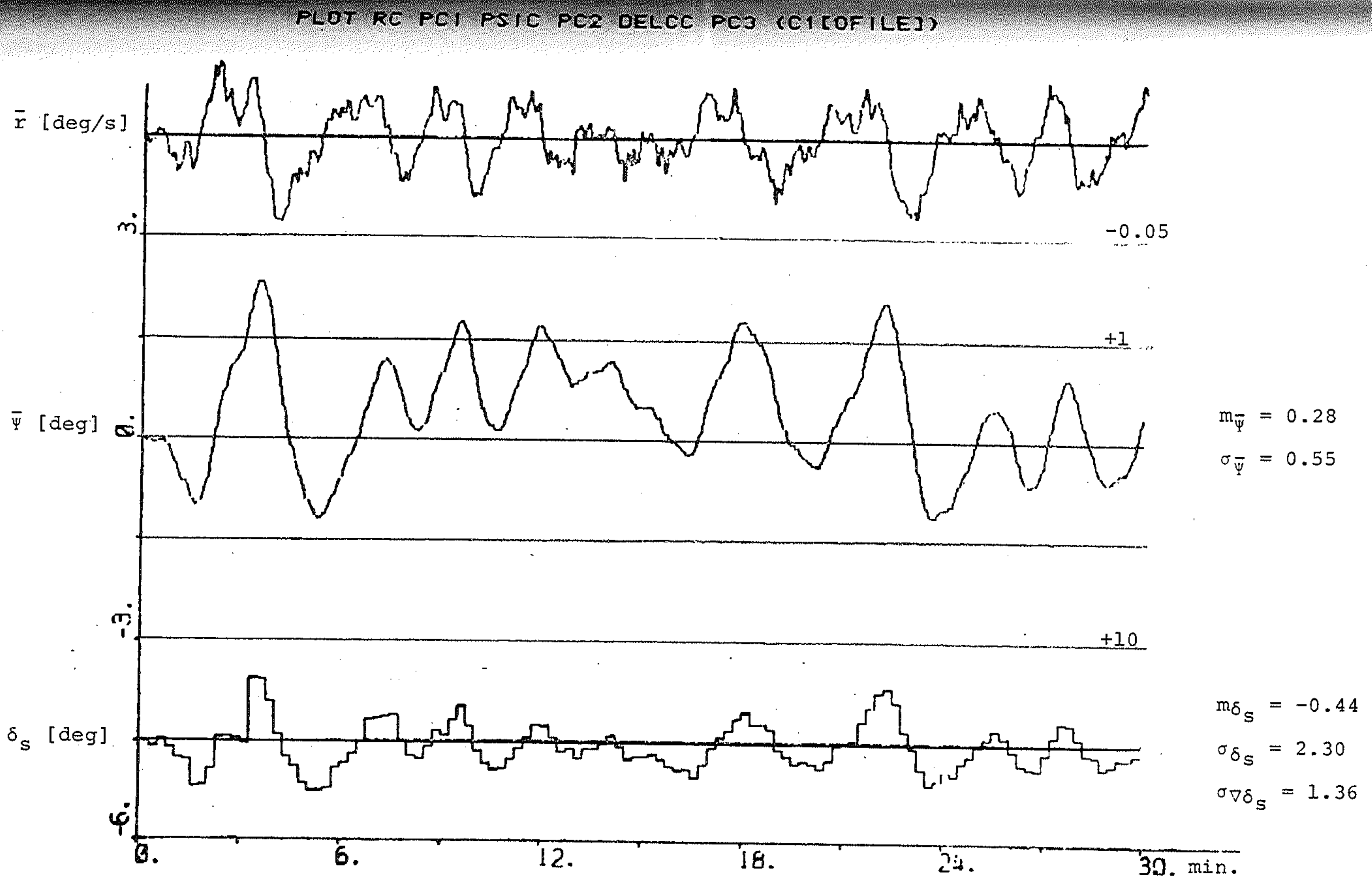


Fig. 4.33 - Regulator STURE3 (NA=3 NB=3 k=1 q_2=0.2): $V_1 = 0.93$ $V_2 = 0.91$ $V_3 = 0.57$.

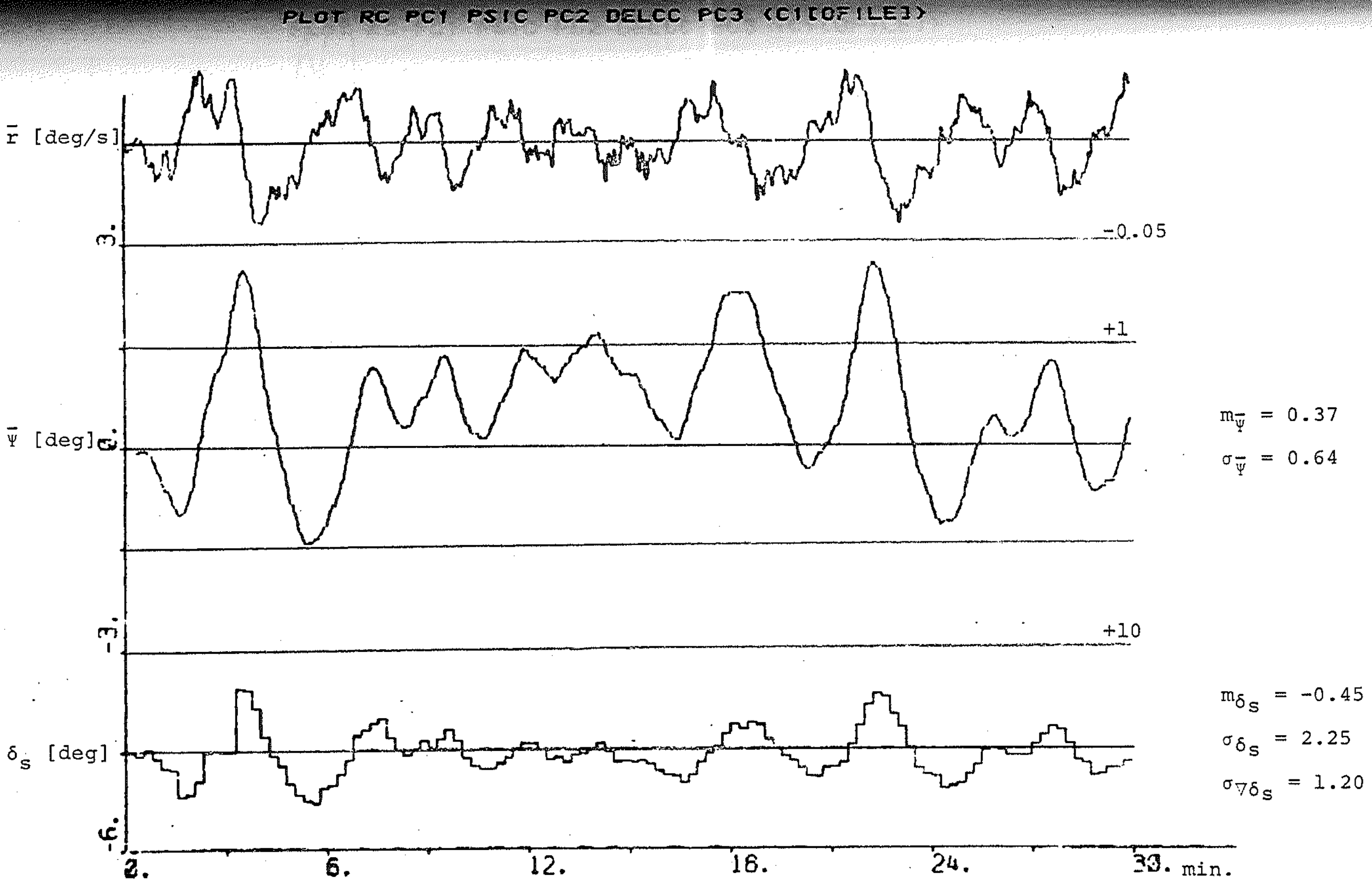
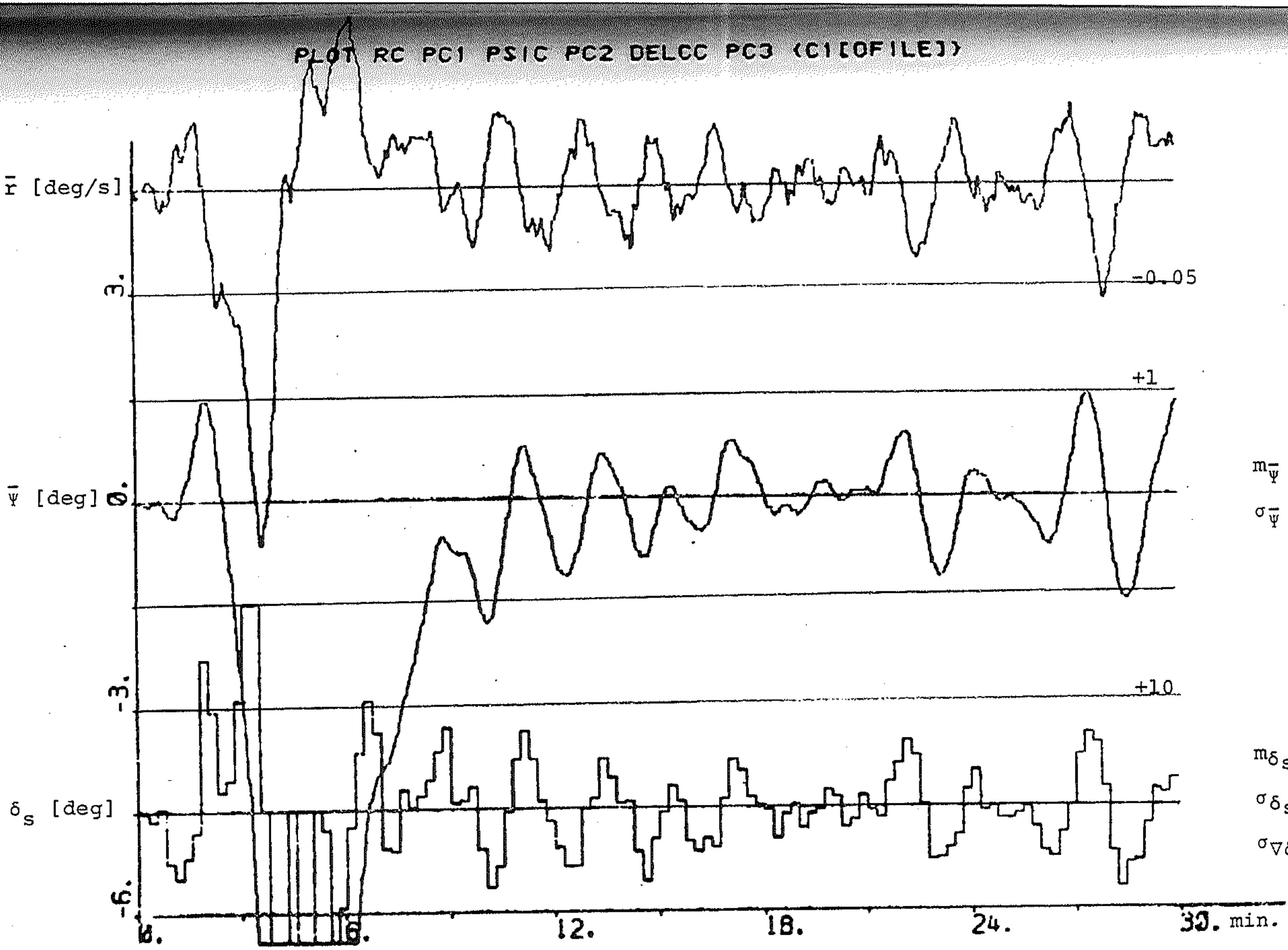


Fig. 4.34 - Regulator STURE3 (NA=3 NB=3 k=1 q_2=0.5): V_1 = 1.07 V_2 = 1.05 V_3 = 0.69.



$m_{\bar{\Psi}} = -1.03$

$\sigma_{\bar{\Psi}} = 2.51$

$m_{\delta_s} = -0.34$

$\sigma_{\delta_s} = 6.23$

$\sigma_{\nabla\delta_s} = 5.68$

Fig. 4.35 - Regulator STURE3 (NA=3 NB=3 k=2 q_2=0.05): V_1 = 11.25 V_2 = 11.24 V_3 = 10.60.

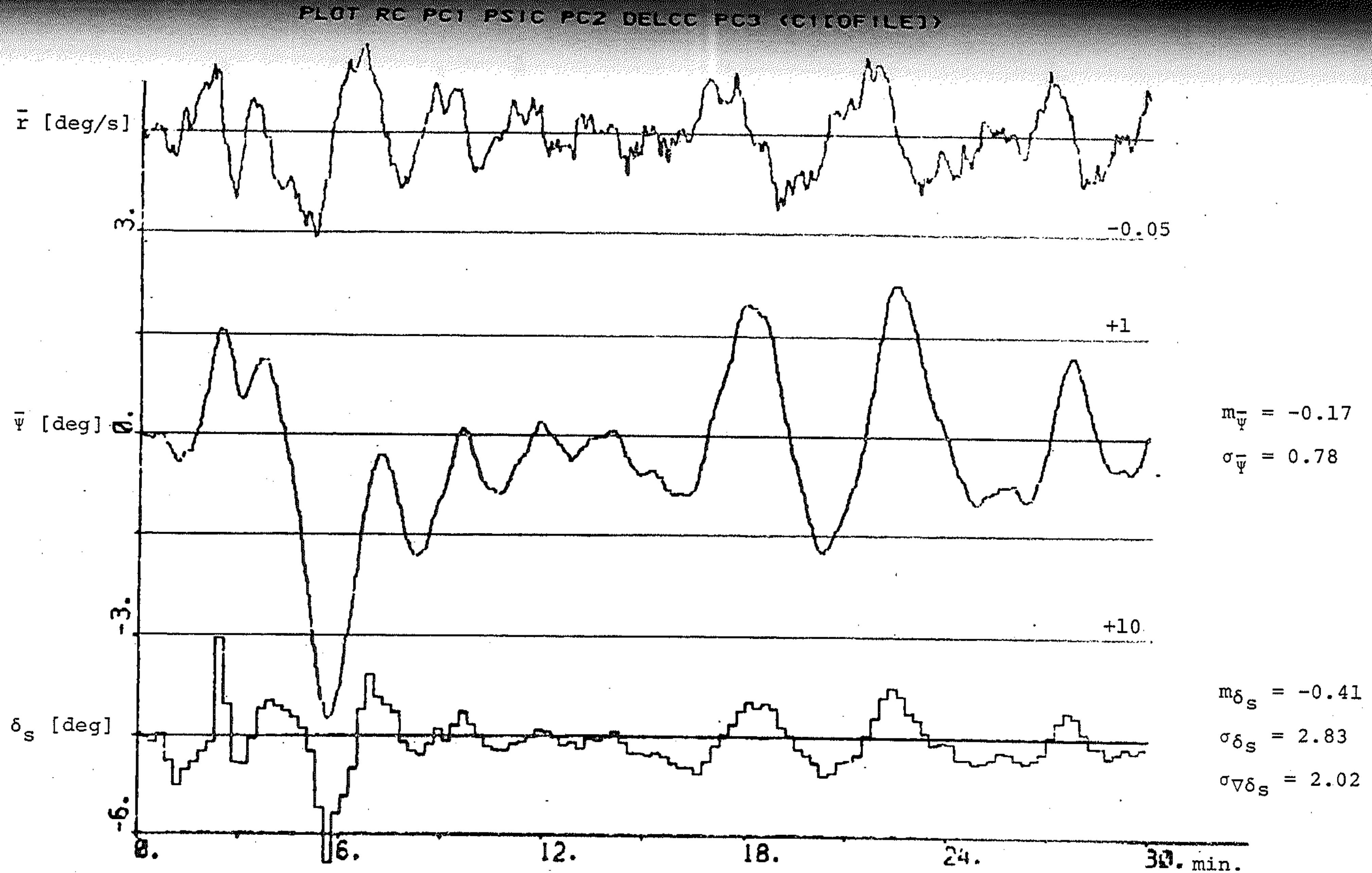
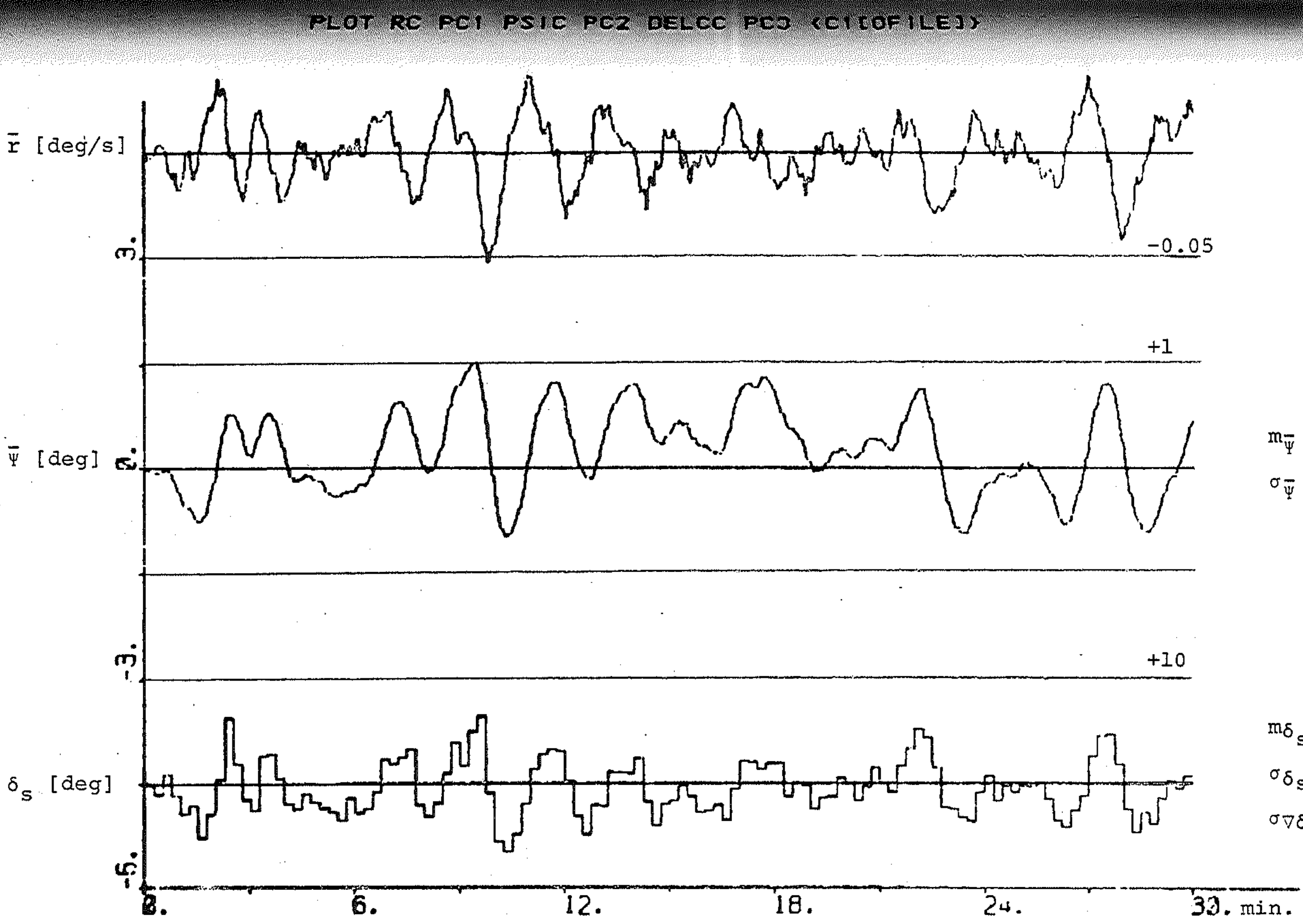


Fig. 4.36 - Regulator STURE3 (NA=3 NB=3 k=2 q_2=0.1): V_1 = 1.45 V_2=1.43 V_3 = 1.04.



$m_{\bar{\Psi}} = 0.17$

$\sigma_{\bar{\Psi}} = 0.40$

$m\delta_s = -0.41$

$\sigma\delta_s = 2.62$

$\sigma\nabla\delta_s = 2.24$

Fig. 4.37 - Regulator STURE3 (NA=4 NB=4 k=1 q_2=0.02): V_1 = 0.89 V_2 = 0.87 V_3 = 0.69.

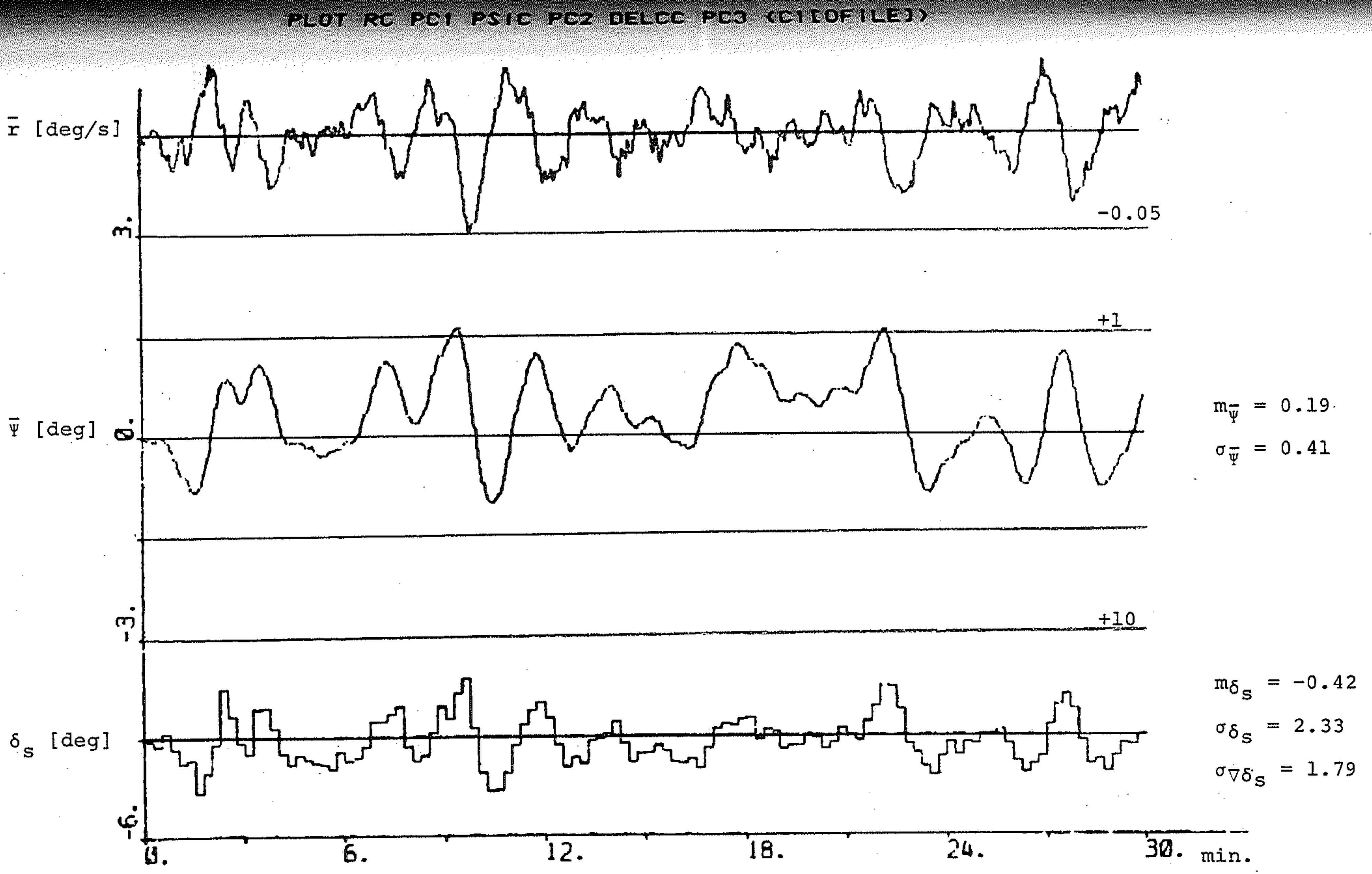


Fig. 4.38 - Regulator STURE3 (NA=4 NB=4 k=1 q_2=0.05): V_1 = 0.77 V_2 = 0.75 V_3 = 0.53.

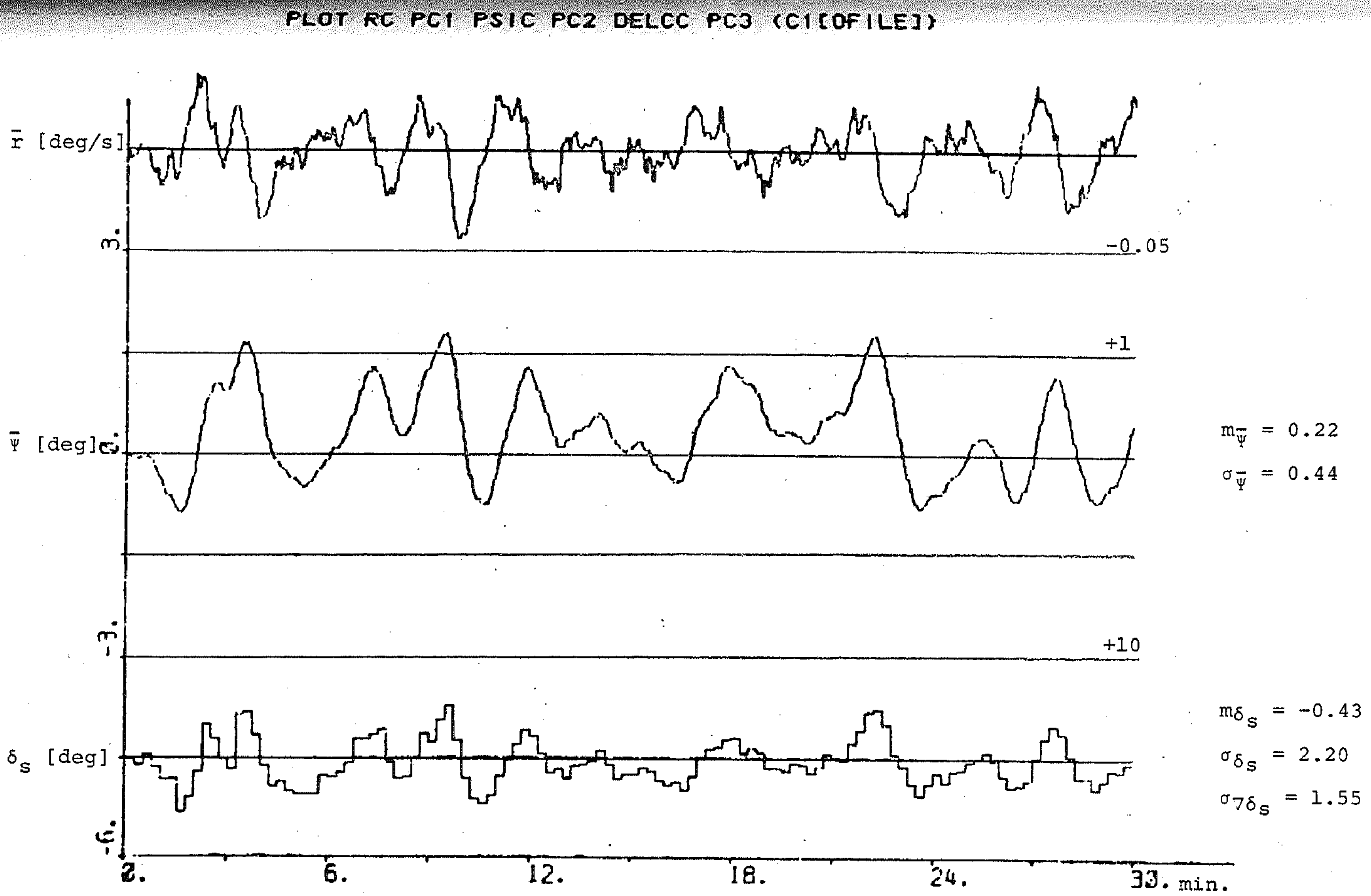


Fig. 4.39 - Regulator STURE3 (NA=4 NB=4 k=1 q_2=0.1): V_1 = 0.75 V_2 = 0.73 V_3 = 0.48.

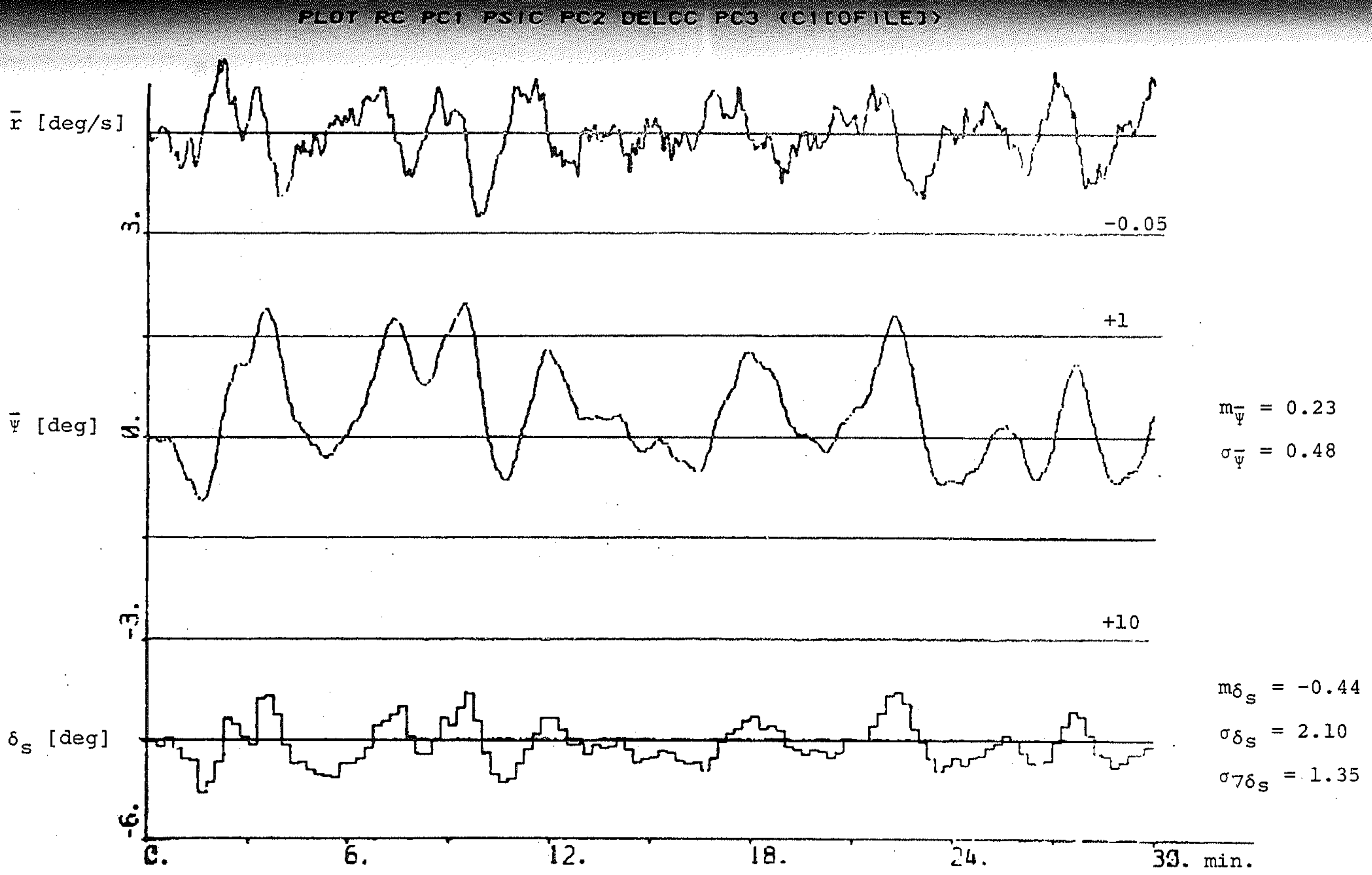


Fig. 4.40 - Regulator STURE3 (NA=4 NB=4 k=1 q_2=0.2): V_1 = 0.74 V_2 = 0.72 V_3 = 0.47.

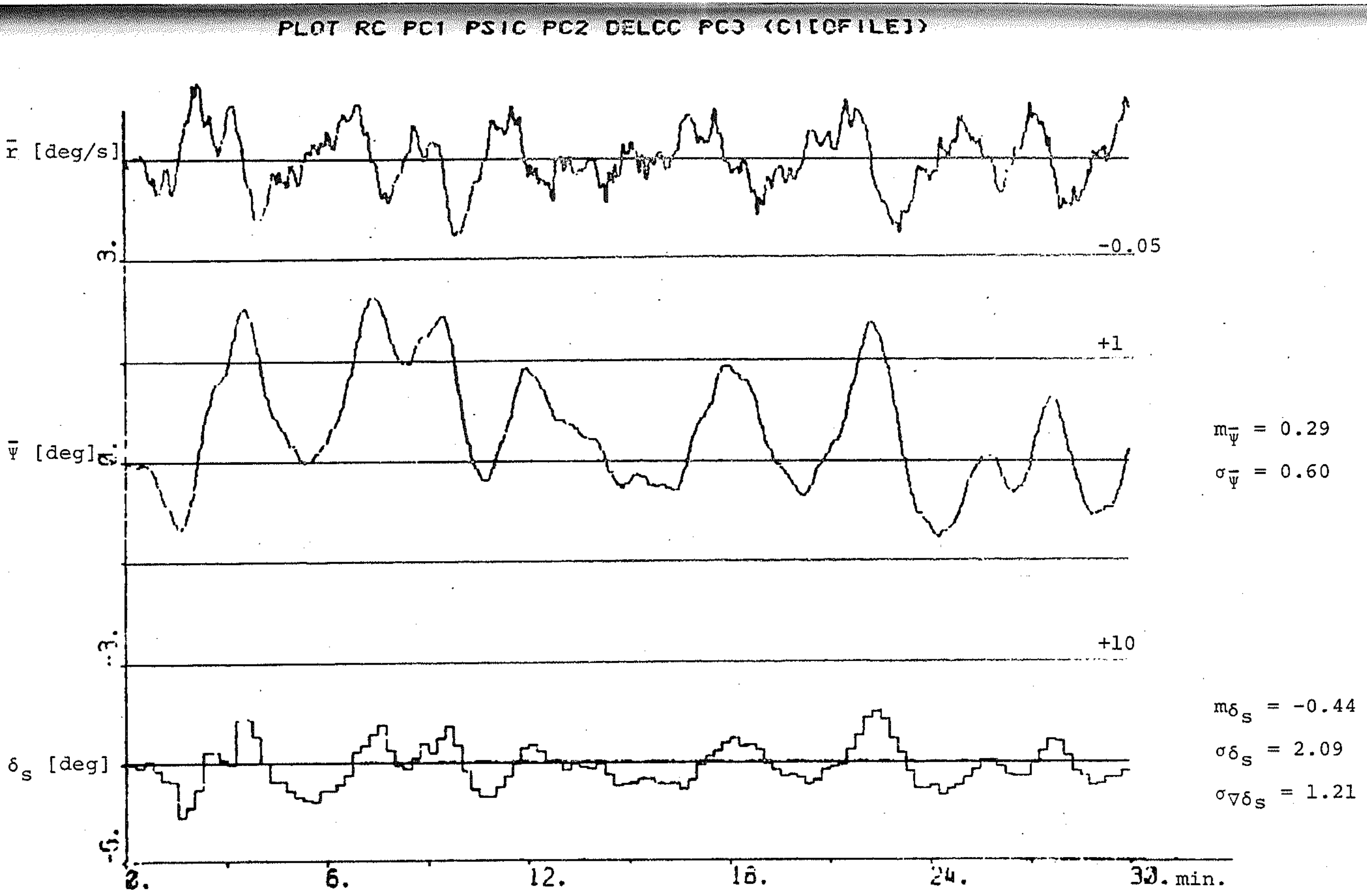


Fig. 4.41 - Regulator STURE3 (NA=4 NB=4 k=1 q_2=0.5): V_1 = 0.90 V_2 = 0.88 V_3 = 0.59.

One simulation with the PID-regulator, where the parameters have been manually tuned, is shown in Fig. 4.42. A summary of the good steering quality simulations of the different autopilots is given in Table 4.4, where also the final parameter values are shown. It is concluded that the regulator STURE gives the best steering and that the PID-regulator and STURE3 are of about the same quality.

Regulator	NA	NB	NC	k	q_2	a_1	a_2	a_3	a_4	b_1	b_2	b_3	b_4	c_1
PID	-	-	-	-	-	-	-	-	-	-	-	-	-	-
STURE	3	2	0	4	0.1	-9.80	13.82	-3.96	-	0.28	0.13	-	-	-
STURE	3	2	0	4	0.2	-9.23	13.69	-4.60	-	0.13	0.04	-	-	-
STURE	3	2	0	5	0.2	-10.63	16.59	-6.15	-	0.22	0.01	-	-	-
STURE	3	2	1	4	0.1	-8.53	9.87	-1.58	-	0.44	0.15	-	-	23.45
STURE	3	2	1	4	0.2	-8.22	10.46	-2.45	-	0.30	0.10	-	-	8.31
STURE3	4	4	0	1	0.1	-1.54	0.27	0.31	0.08	-0.05	-0.04	-0.02	-0.02	-
STURE3	4	4	0	1	0.2	-1.54	0.24	0.27	0.15	-0.06	-0.05	-0.02	-0.03	-

Table 4.4a - A summary of the good steering quality simulations of the different autopilots with the final parameter values. The parameters of the PID-regulator were: $k_P = 4$, $k_D = 80$, $k_I = 0.02$. Notice that the parameters b_1, b_2 and b_3 of STURE3 are comparable to b_0, $b_0 b_1$ and $b_0 b_2$ of STURE, where $b_0 = -1$.

Regulator	$\sigma_{\bar{\Psi}}$ [deg]	$\sigma_{\delta s}$ [deg]	$\sigma_{\nabla\delta s}$ [deg]	V_1	V_2	V_3	Fig.
PID	0.35	2.39	2.92	0.71	0.69	0.98	4.42
STURE	0.33	1.70	1.49	0.43	0.41	0.35	4.18
STURE	0.37	1.67	1.32	0.44	0.42	0.32	4.19
STURE	0.35	1.66	1.56	0.42	0.40	0.37	4.23
STURE	0.36	1.74	1.57	0.48	0.46	0.40	4.25
STURE	0.38	1.56	1.17	0.43	0.41	0.30	4.26
STURE3	0.44	2.20	1.55	0.75	0.73	0.48	4.39
STURE3	0.48	2.10	1.35	0.74	0.72	0.47	4.40

<ins>Table 4.4b</ins>

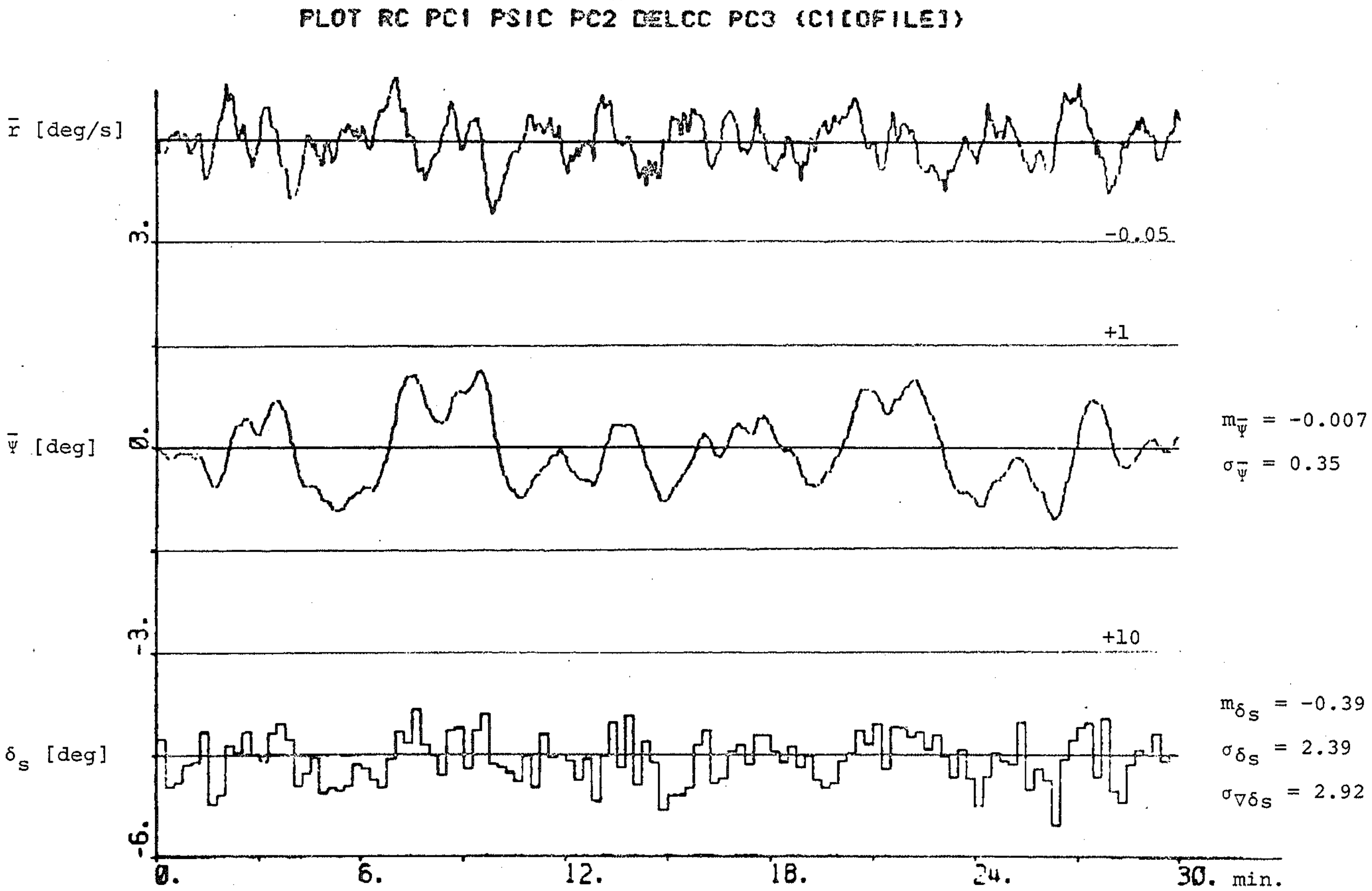


Fig. 4.42 - PID-regulator (k_P=4 k_D=80 k_I=0.02): V_1 = 0.71 V_2 = 0.69 V_3 = 0.98.

4.2. Varying Load and Wind Conditions.

The PID-regulator, with well tuned parameters, and the simple, self-tuning regulator STURE with different structures are used as autopilots in this section. Three load conditions, T = 10.5, 20 and 25 m, are simulated during the two wind conditions moderate breeze and fresh gale. The angle of relative wind direction α is equal to 135 deg. The initial values of the parameters of STURE are obtained from the simulations presented in Sec. 4.1. The parameters are always tuned beforehand during 30 min and all the plots and results shown in this section are related to the next 30 min.

Simulations with the mean draught T equal to 10.5 m are shown in Figs. 4.43 - 4.54. A summary is given in Table 4.5, where also the final parameter values of STURE are shown. Notice that an example of the parameter variations is shown in Fig. 4.45b.

The corresponding simulations with T = 20 m and T = 25 m are shown in Figs. 4.55 - 4.66 and Figs. 4.67 - 4.78, respectively, and the summaries are given in Tables 4.6 and 4.7.

Wind	Regulator	NC	k	q_2	a_1	a_2	a_3	b_1	b_2	c_1
Moderate breeze	PID	-	-	-	-	-	-	-	-	-
	STURE	0	5	0	-14.60	16.45	-2.17	1.02	0.60	-
	STURE	0	4	0.1	-8.33	11.67	-3.75	0.15	0.09	-
	STURE	0	4	0.2	-6.99	10.25	-3.67	-0.02	-0.03	-
	STURE	1	4	0.1	-7.05	8.30	-1.65	0.26	0.11	19.00
	STURE	1	4	0.2	-6.17	7.91	-2.13	0.10	0.04	6.34
Fresh gale	PID	-	-	-	-	-	-	-	-	-
	STURE	0	5	0	-14.66	16.61	-2.26	0.98	0.59	-
	STURE	0	4	0.1	-7.91	11.00	-3.49	0.15	0.06	-
	STURE	0	4	0.2	-6.44	8.68	-2.63	0.11	-0.03	-
	STURE	1	4	0.1	-6.63	7.77	-1.52	0.22	0.08	20.09
	STURE	1	4	0.2	-5.68	6.53	-1.25	0.20	0.02	8.12

Table 4.5a - A summary of the simulations with the mean draught T equal to 10.5 m. The final parameter values of STURE (NA=3 NB=2) are also shown. The parameters of the PID-regulator were $k_P = 4$, $k_D = 20$, $k_I = 0.02$ for the moderate breeze case and $k_P = 4$, $k_D = 30$, $k_I = 0.02$ for the fresh gale case.

Wind	Regulator	$\sigma_{\bar{\Psi}}$ [deg]	σ_{δ_s} [deg]	$\sigma_{\nabla\delta_s}$ [deg]	V_1	V_2	V_3	Fig.
Moderate breeze	PID	0.35	1.56	1.11	0.39	0.36	0.24	4.43
	STURE	0.26	2.92	4.53	0.96	0.93	2.13	4.44
	STURE	0.34	1.61	1.50	0.40	0.38	0.34	4.45
	STURE	0.39	1.45	1.13	0.39	0.37	0.28	4.46
	STURE	0.33	1.56	1.46	0.38	0.35	0.32	4.47
	STURE	0.39	1.40	1.02	0.37	0.35	0.26	4.48
Fresh gale	PID	0.66	2.98	2.07	1.46	1.32	0.87	4.49
	STURE	0.51	5.08	7.38	3.00	2.86	5.73	4.50
	STURE	0.70	3.08	2.49	1.57	1.44	1.11	4.51
	STURE	0.83	2.80	1.84	1.61	1.49	1.04	4.52
	STURE	0.66	2.97	2.35	1.46	1.33	1.00	4.53
	STURE	0.84	2.80	1.72	1.63	1.50	1.01	4.54

Table 4.5b

Wind	Regulator	NC	k	q_2	a_1	a_2	a_3	b_1	b_2	c_1
Moderate breeze	PID	-	-	-	-	-	-	-	-	-
	STURE	0	5	0	-14.52	16.62	-2.36	1.02	0.60	-
	STURE	0	4	0.1	-9.17	12.41	-3.45	0.18	0.14	-
	STURE	0	4	0.2	-8.01	11.11	-3.27	0.02	0.04	-
	STURE	1	4	0.1	-7.81	9.24	-1.60	0.27	0.13	19.77
	STURE	1	4	0.2	-7.27	8.73	-1.65	0.14	0.11	7.28
Fresh gale	PID	-	-	-	-	-	-	-	-	-
	STURE	0	5	0	-14.89	17.30	-2.69	0.98	0.59	-
	STURE	0	4	0.1	-9.46	12.30	-3.12	0.27	0.17	-
	STURE	0	4	0.2	-8.30	10.31	-2.29	0.24	0.10	-
	STURE	1	4	0.1	-7.93	9.14	-1.45	0.29	0.12	22.32
	STURE	1	4	0.2	-7.46	8.16	-1.01	0.28	0.13	10.36

Table 4.6a - A summary of the simulations with the mean draught T equal to 20 m. The final parameter values of STURE (NA=3 NB=2) are also shown. The parameters of the PID-regulator were $k_P = 4$, $k_D = 80$, $k_I = 0.02$ for the moderate breeze case and $k_P = 4$, $k_D = 100$, $k_I = 0.04$ for the fresh gale case.

Wind	Regulator	$\sigma_{\bar{\Psi}}$ [deg]	σ_{δ_s} [deg]	$\sigma_{\nabla\delta_s}$ [deg]	V_1	V_2	V_3	Fig.
Moderate breeze	PID	0.36	2.40	2.92	0.78	0.70	0.98	4.55
	STURE	0.30	2.35	3.19	0.74	0.67	1.13	4.56
	STURE	0.41	1.85	1.38	0.60	0.53	0.38	4.57
	STURE	0.48	1.87	1.15	0.66	0.59	0.38	4.58
	STURE	0.37	1.75	1.36	0.53	0.46	0.34	4.59
	STURE	0.48	1.85	1.08	0.65	0.59	0.36	4.60
Fresh gale	PID	0.66	4.12	4.30	2.47	2.13	2.29	4.61
	STURE	0.30	4.10	4.80	2.47	2.13	2.74	4.62
	STURE	0.84	3.63	2.38	2.39	2.06	1.32	4.63
	STURE	1.00	3.74	2.04	2.76	2.45	1.47	4.64
	STURE	0.78	3.50	2.29	2.20	1.88	1.18	4.65
	STURE	0.97	3.66	1.94	2.65	2.34	1.38	4.66

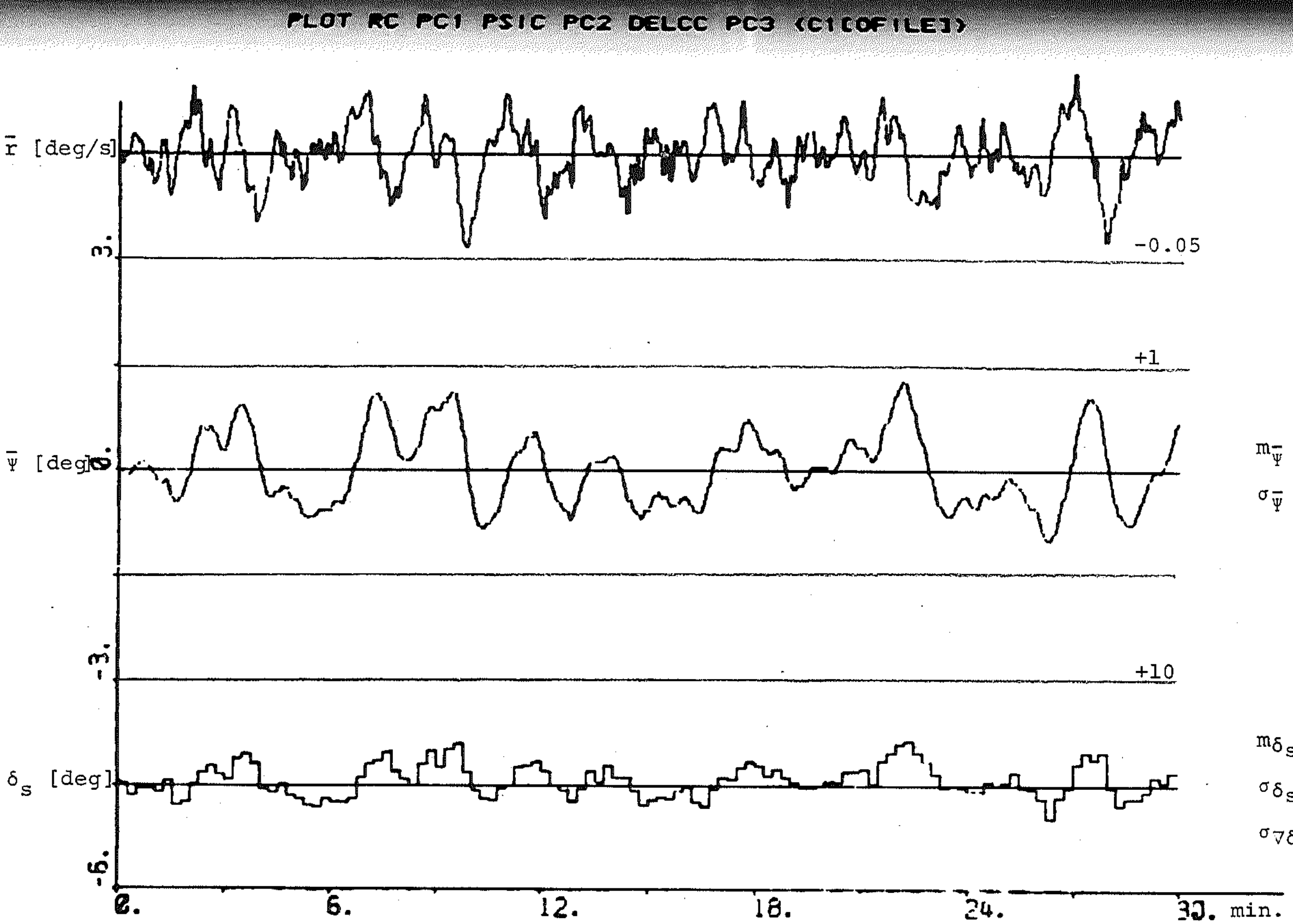
Table 4.6b

Wind	Regulator	NC	k	q_2	a_1	a_2	a_3	b_1	b_2	c_1
Moderate breeze	PID	-	-	-	-	-	-	-	-	-
	STURE	0	5	0	-15.03	17.06	-2.33	1.02	0.60	-
	STURE	0	4	0.1	-9.80	13.02	-3.38	0.15	0.16	-
	STURE	0	4	0.2	-8.60	11.80	-3.31	-0.03	0.08	-
	STURE	1	4	0.1	-8.53	9.87	-1.48	0.25	0.14	20.60
	STURE	1	4	0.2	-8.12	9.45	-1.51	0.10	0.17	8.29
Fresh gale	PID	-	-	-	-	-	-	-	-	-
	STURE	0	5	0	-15.69	18.53	-3.15	0.95	0.58	-
	STURE	0	4	0.1	-10.46	13.79	-3.58	0.20	0.19	-
	STURE	0	4	0.2	-9.34	11.97	-2.88	0.12	0.17	-
	STURE	1	4	0.1	-8.89	10.31	-1.65	0.26	0.14	23.72
	STURE	1	4	0.2	-8.46	9.70	-1.52	0.17	0.20	11.50

Table 4.7a - A summary of the simulations with the mean draught T equal to 25 m. The final parameter values of STURE (NA=3 NB=2) are also shown. The parameters of the PID-regulator were $k_P = 4$, $k_D = 100$, $k_I = 0.02$ for the moderate breeze case and $k_P = 4$, $k_D = 120$, $k_I = 0.02$ for the fresh gale case.

Wind	Regulator	$\sigma_{\bar{\Psi}}$ [deg]	σ_{δ_s} [deg]	$\sigma_{\nabla\delta_s}$ [deg]	V_1	V_2	V_3	Fig.
Moderate breeze	PID	0.40	2.87	3.58	1.08	0.99	1.45	4.67
	STURE	0.34	2.38	3.01	0.83	0.74	1.08	4.68
	STURE	0.45	2.14	1.42	0.77	0.68	0.43	4.69
	STURE	0.53	2.22	1.24	0.87	0.78	0.45	4.70
	STURE	0.41	2.05	1.42	0.72	0.62	0.40	4.71
	STURE	0.53	2.27	1.22	0.93	0.83	0.47	4.72
Fresh gale	PID	0.67	4.37	4.88	2.82	2.35	2.83	4.73
	STURE	0.65	4.18	4.41	2.78	2.33	2.53	4.74
	STURE	0.88	4.11	2.50	2.97	2.54	1.47	4.75
	STURE	1.08	4.42	2.28	3.62	3.20	1.77	4.76
	STURE	0.82	3.96	2.40	2.76	2.33	1.34	4.77
	STURE	1.00	4.16	2.12	3.26	2.84	1.56	4.78

Table 4.7b



$m_{\bar{\Psi}} = 0.02$

$\sigma_{\bar{\Psi}} = 0.35$

$m_{\delta_s} = 0.52$

$\sigma_{\delta_s} = 1.56$

$\sigma_{\nabla\delta_s} = 1.11$

Fig. 4.43 - T = 10.5 m, moderate breeze, PID-regulator (k_P=4 k_D=20 k_I=0.02):
$V_1 = 0.39$ $V_2 = 0.36$ $V_3 = 0.24$

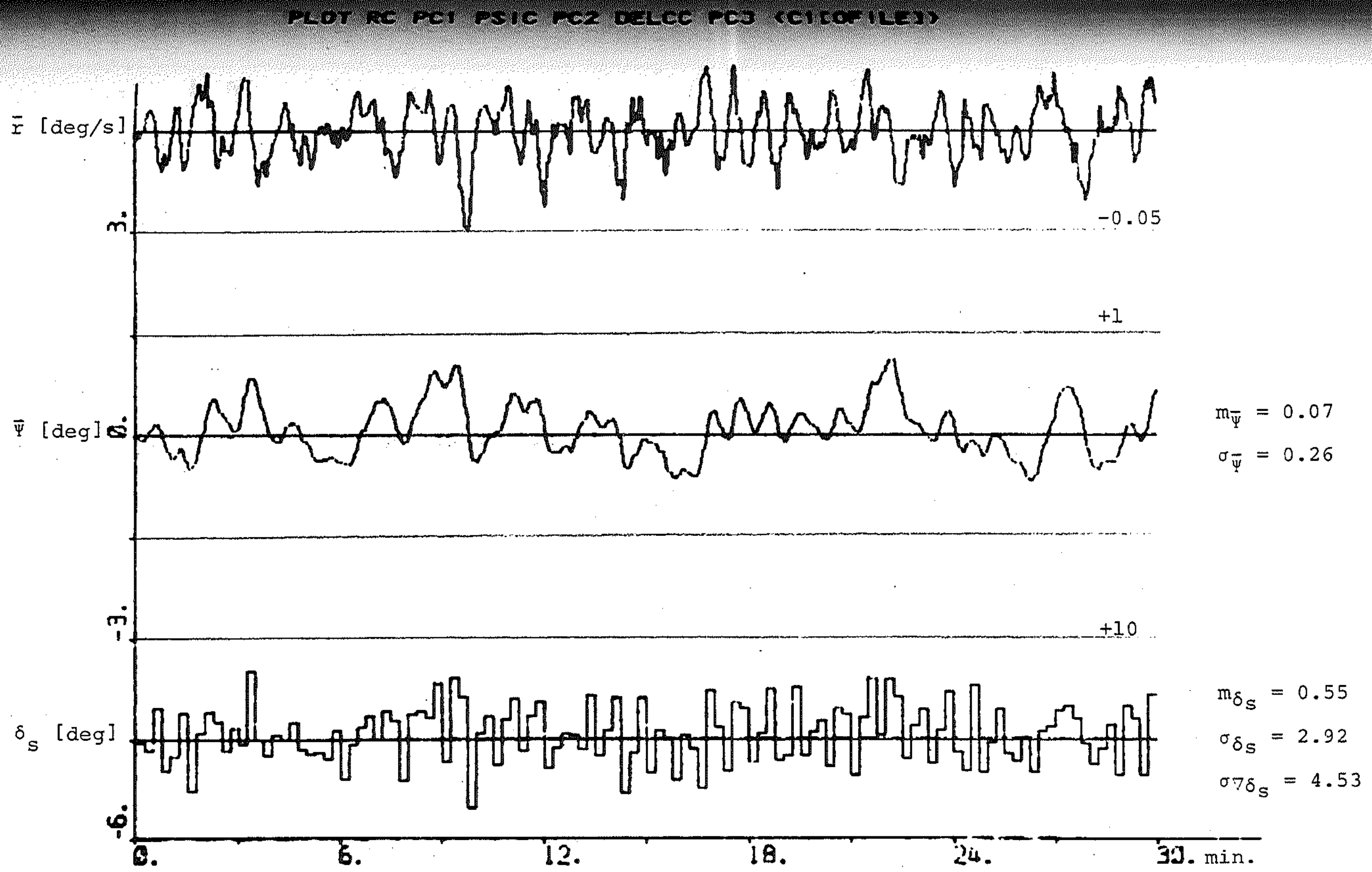
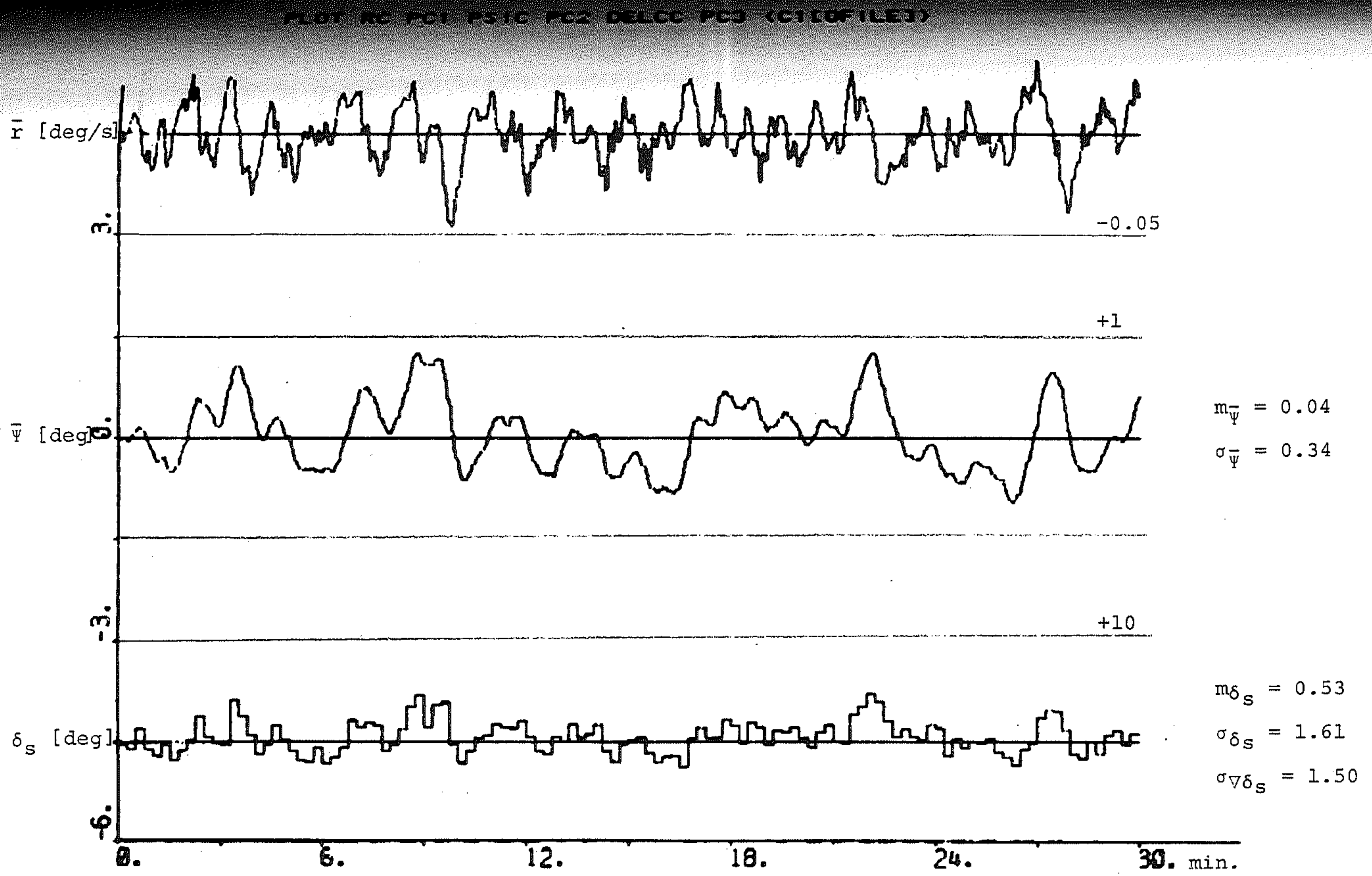


Fig. 4.44 - T = 10.5 m, moderate breeze, regulator STURE (NA=3 NB=2 NC=0 k=5 q_2=0): $V_1 = 0.96$ $V_2 = 0.93$ $V_3 = 2.13$



<u>Fig. 4.45a</u> - T = 10.5 m, moderate breeze, regulator STURE (NA=3 NB=2 NC=0 k=4 q_2=0.1):
V_1 = 0.40 V_2 = 0.38 V_3 = 0.34

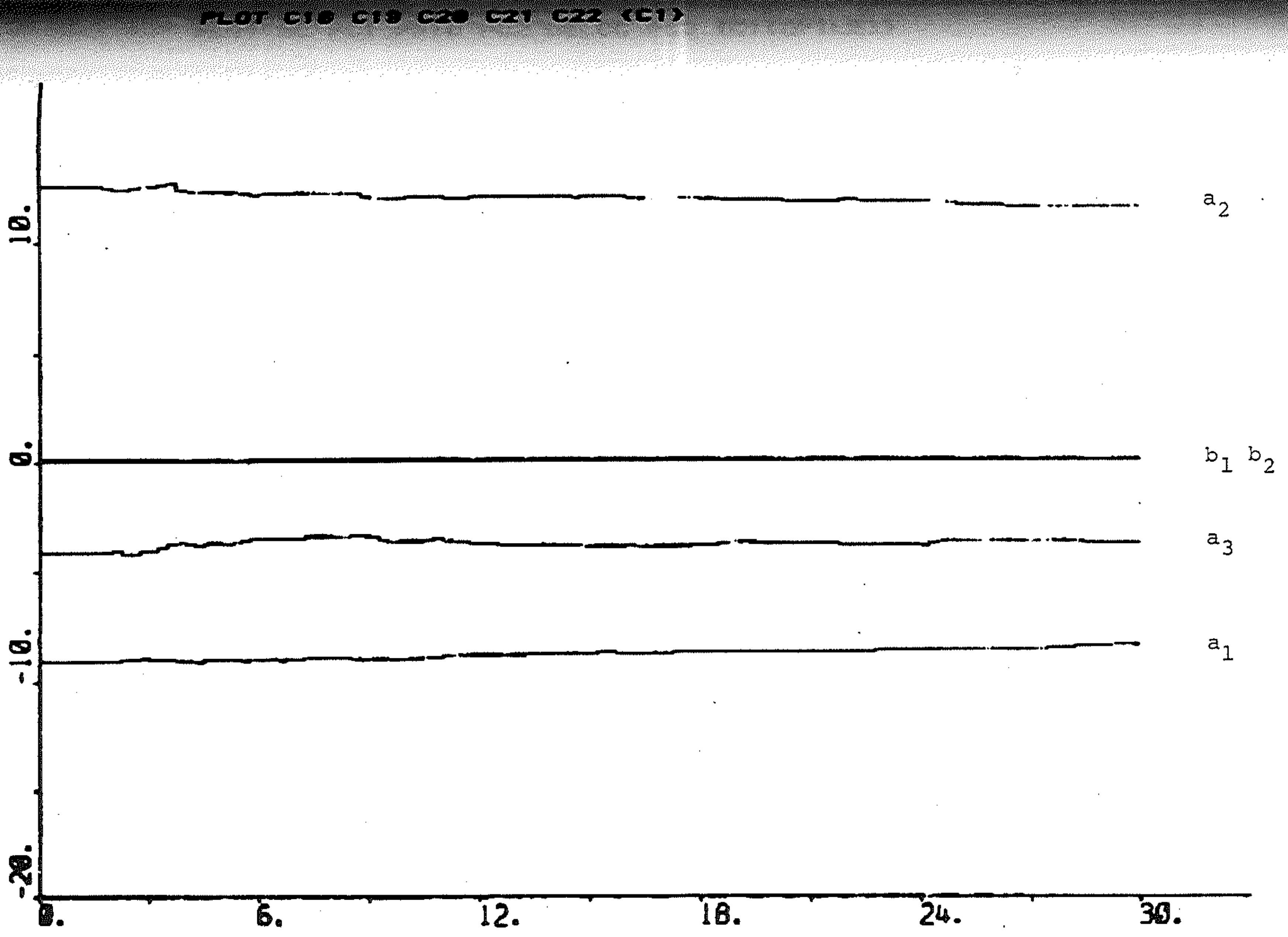
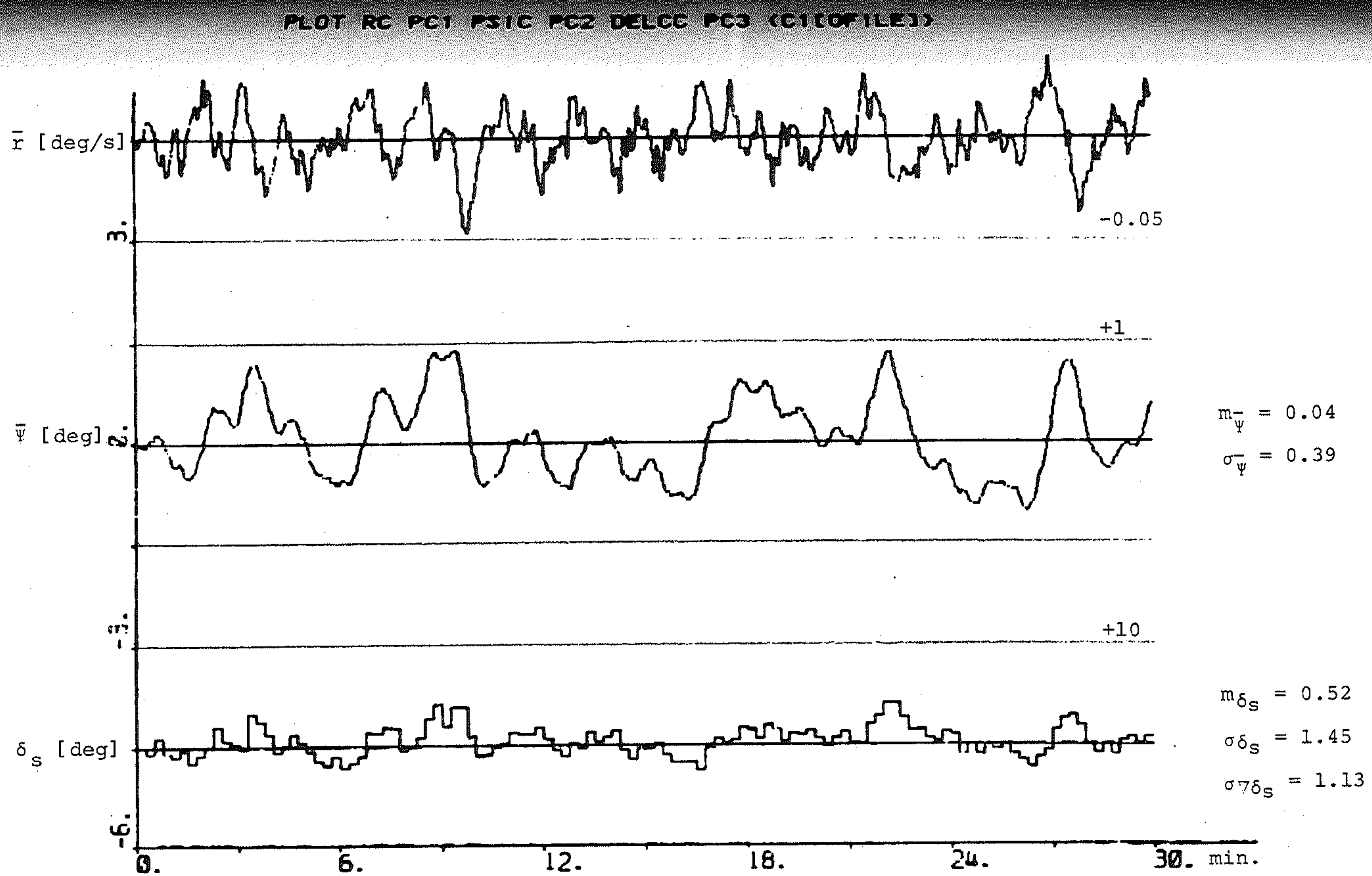


Fig. 4.45b - Parameter values of STURE.



<u>Fig. 4.46</u> - T = 10.5 m, moderate breeze, regulator STURE (NA=3 NB=2 NC=0 k=4 q_2=0.2):

$V_1 = 0.39$ $V_2 = 0.37$ $V_3 = 0.28$

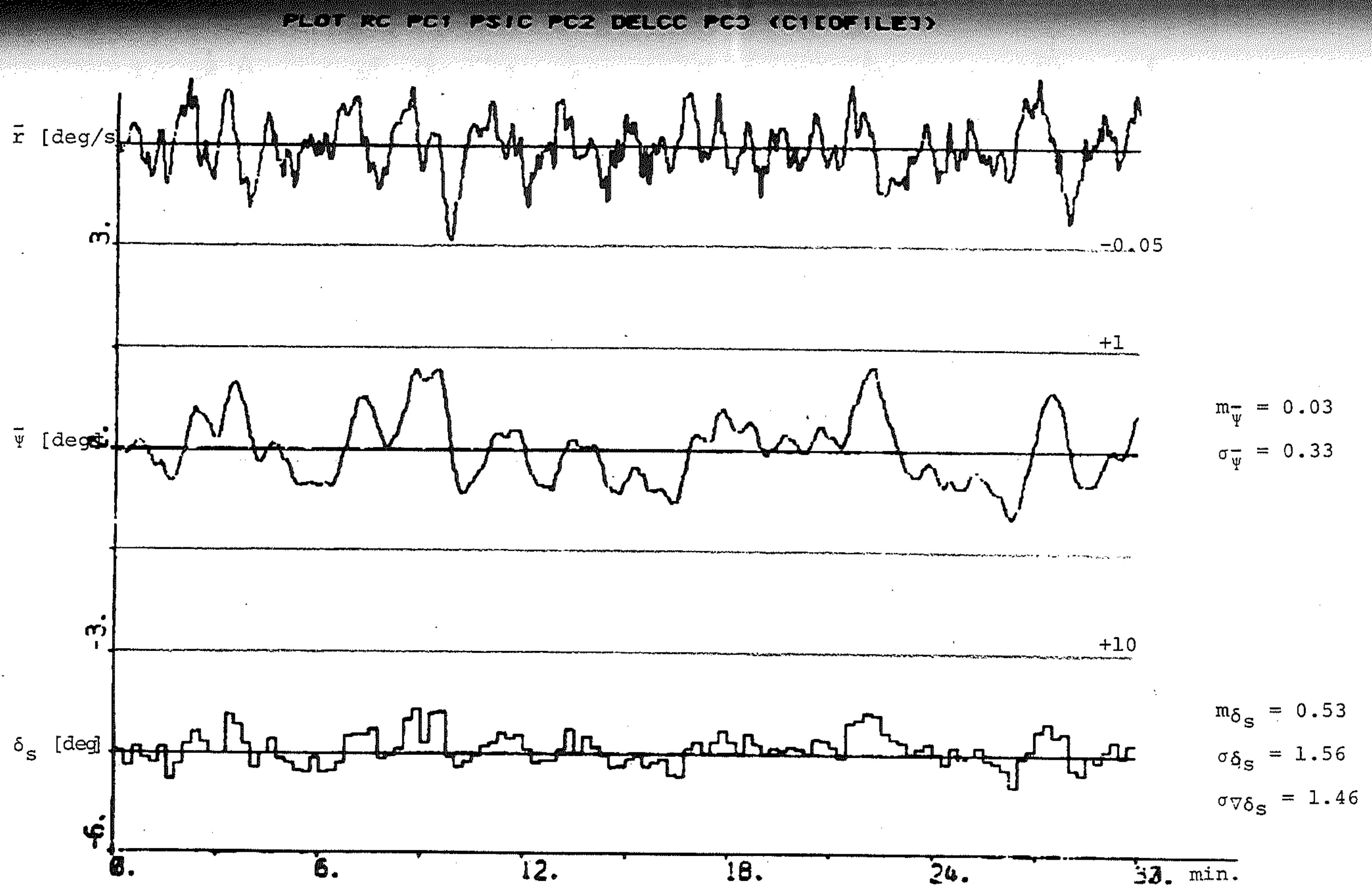


Fig. 4.47 - T = 10.5 m, moderate breeze, regulator STURE (NA=3 NB=2 NC=1 k=4 q_2=0.1):

$V_1 = 0.38$ $V_2 = 0.35$ $V_3 = 0.32$

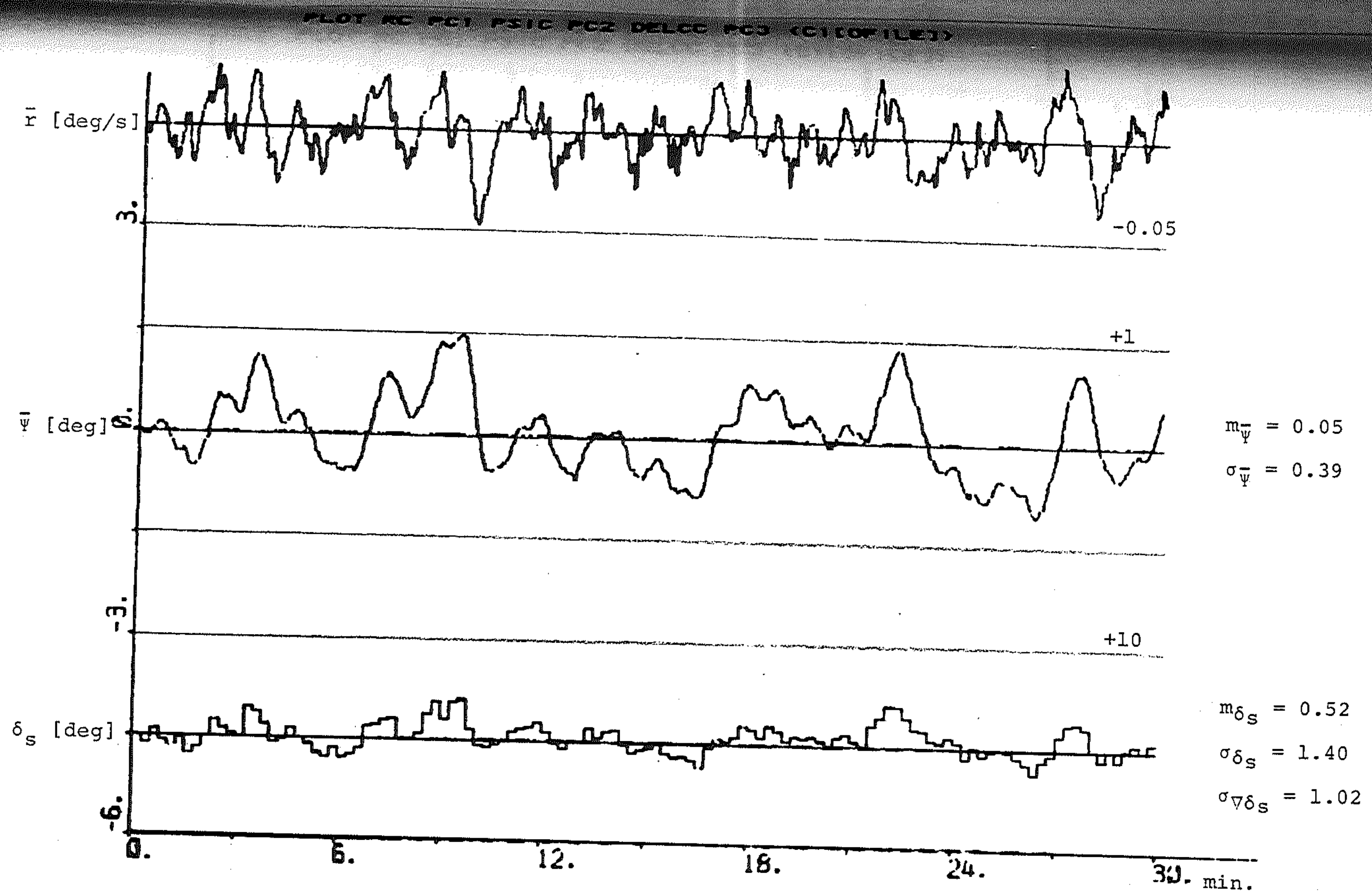


Fig. 4.48 - T = 10.5 m, moderate breeze, regulator STURE (NA=3 NB=2 NC=1 k=4 q_2=0.2): $V_1 = 0.37$ $V_2 = 0.35$ $V_3 = 0.26$.

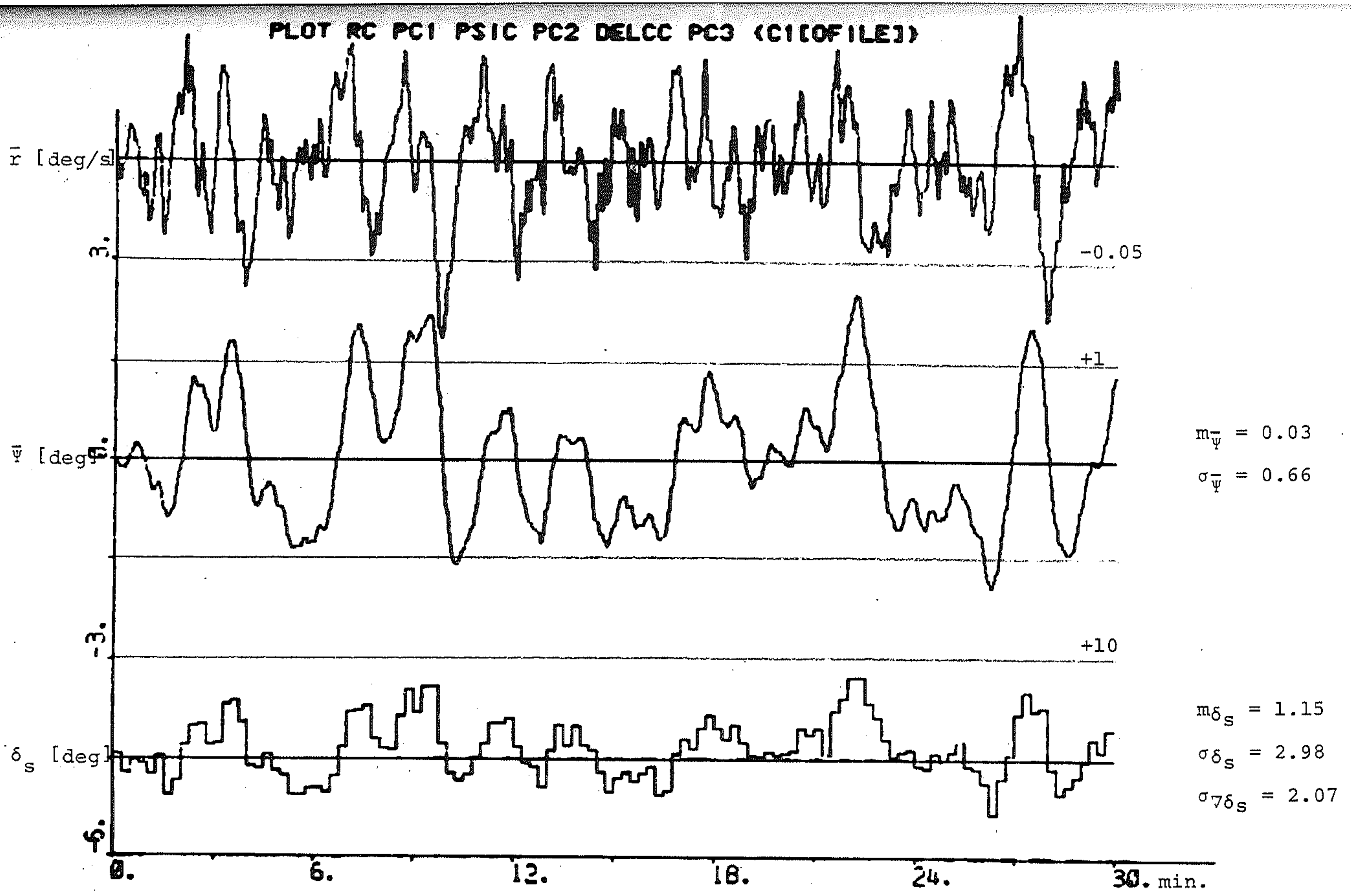


Fig. 4.49 - T = 10.5m, fresh gale, PID-regulator ($k_P=4$ $k_D=30$ $k_I=0.02$):
$V_1 = 1.46$ $V_2 = 1.32$ $V_3 = 0.87$

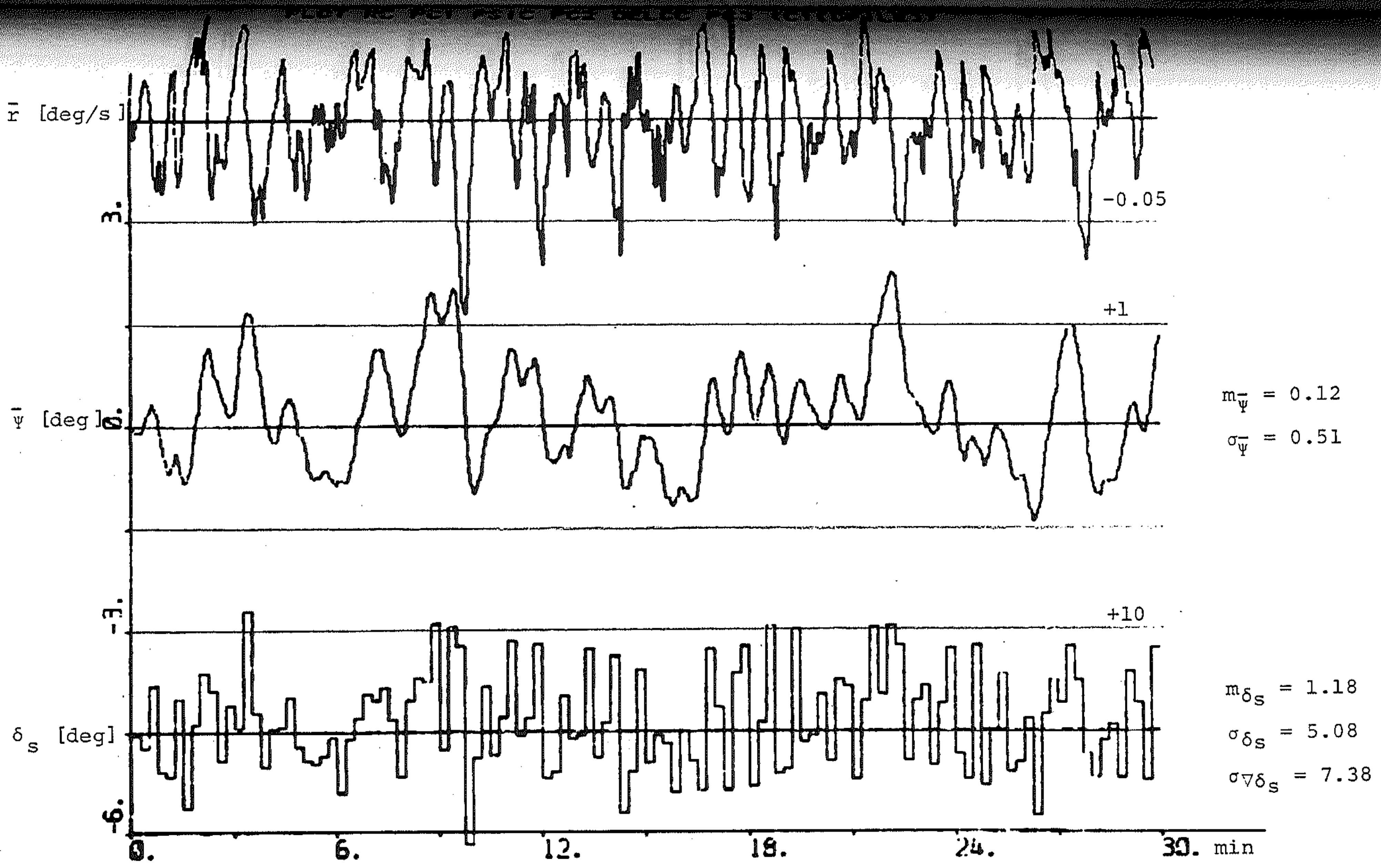


Fig. 4.50 - T = 10.5 m, fresh gale, regulator STURE (NA=3 NB=2 NC=0 k=5 q_2=0):
V_1 = 3.00 V_2 = 2.86 V_3 = 5.73

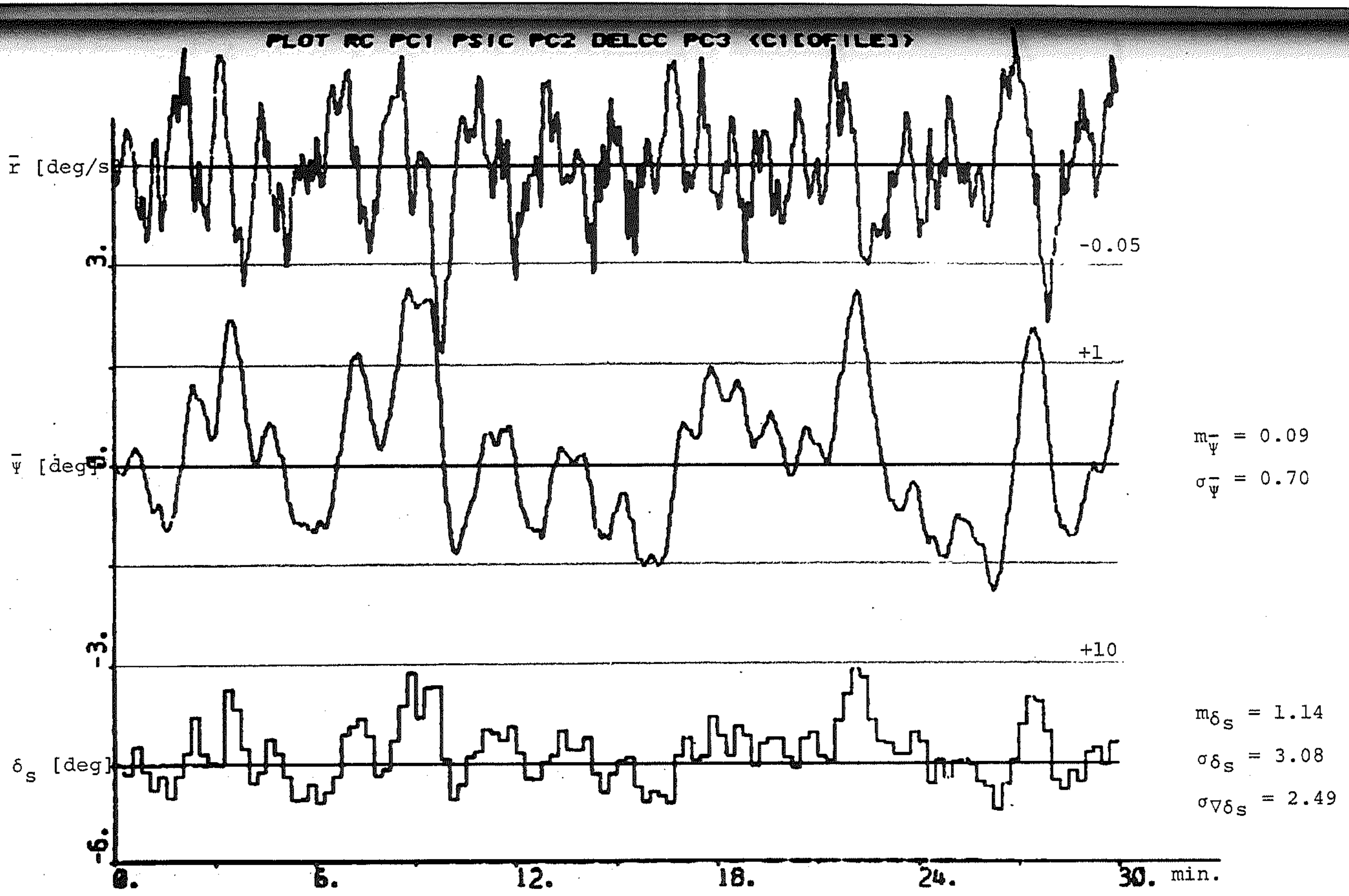


Fig. 4.51 - T = 10.5 m, fresh gale, regulator STURE (NA=3 NB=2 NC=0 k=4 q_2=0.1): $V_1 = 1.57$ $V_2 = 1.44$ $V_3 = 1.11$

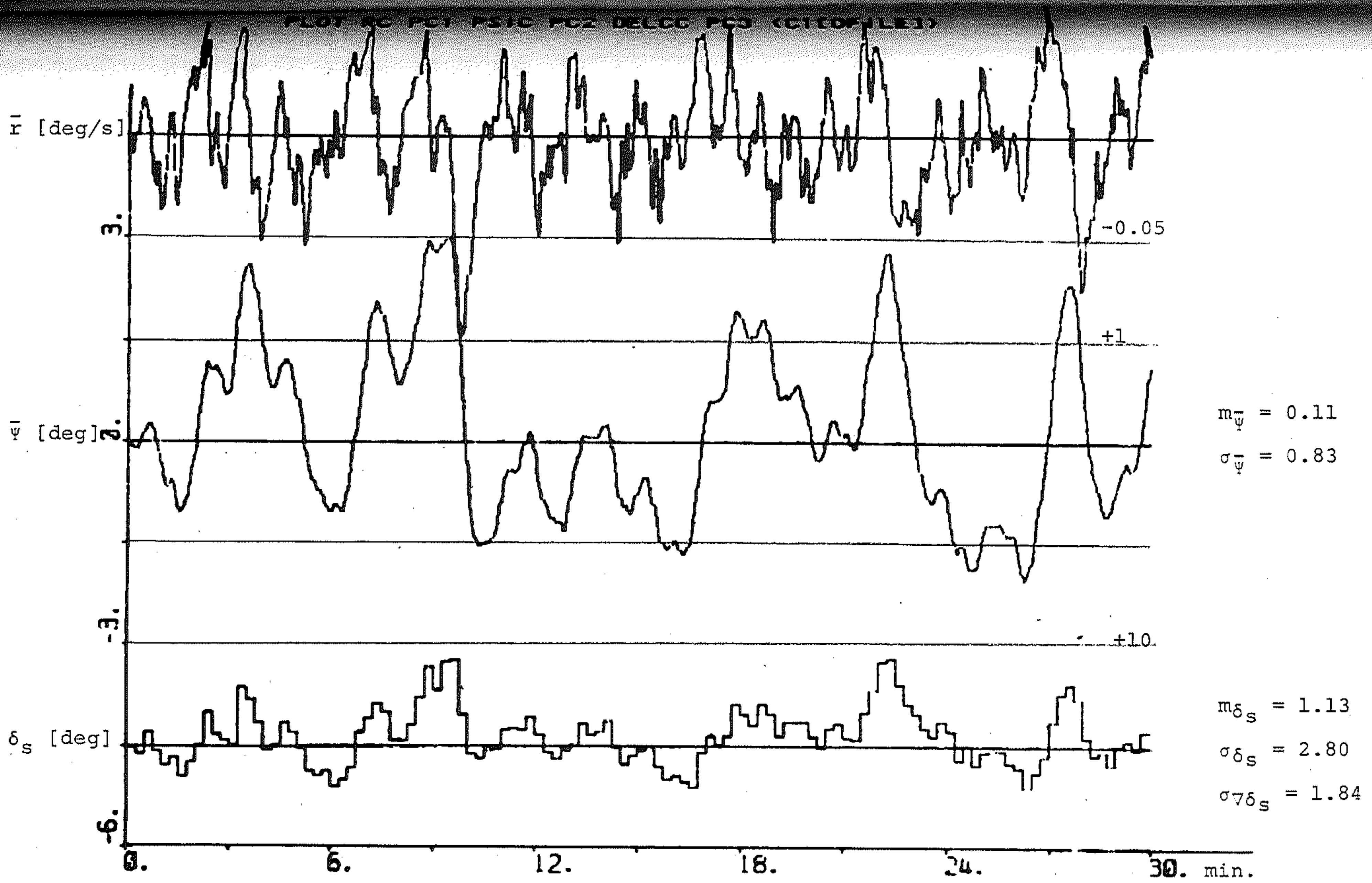


Fig. 4.52 - T = 10.5 m, fresh gale, regulator STURE (NA=3 NB=2 NC=0 k=4 q_2=0.2): V_1 = 1.61 V_2 = 1.49 V_3 = 1.04.

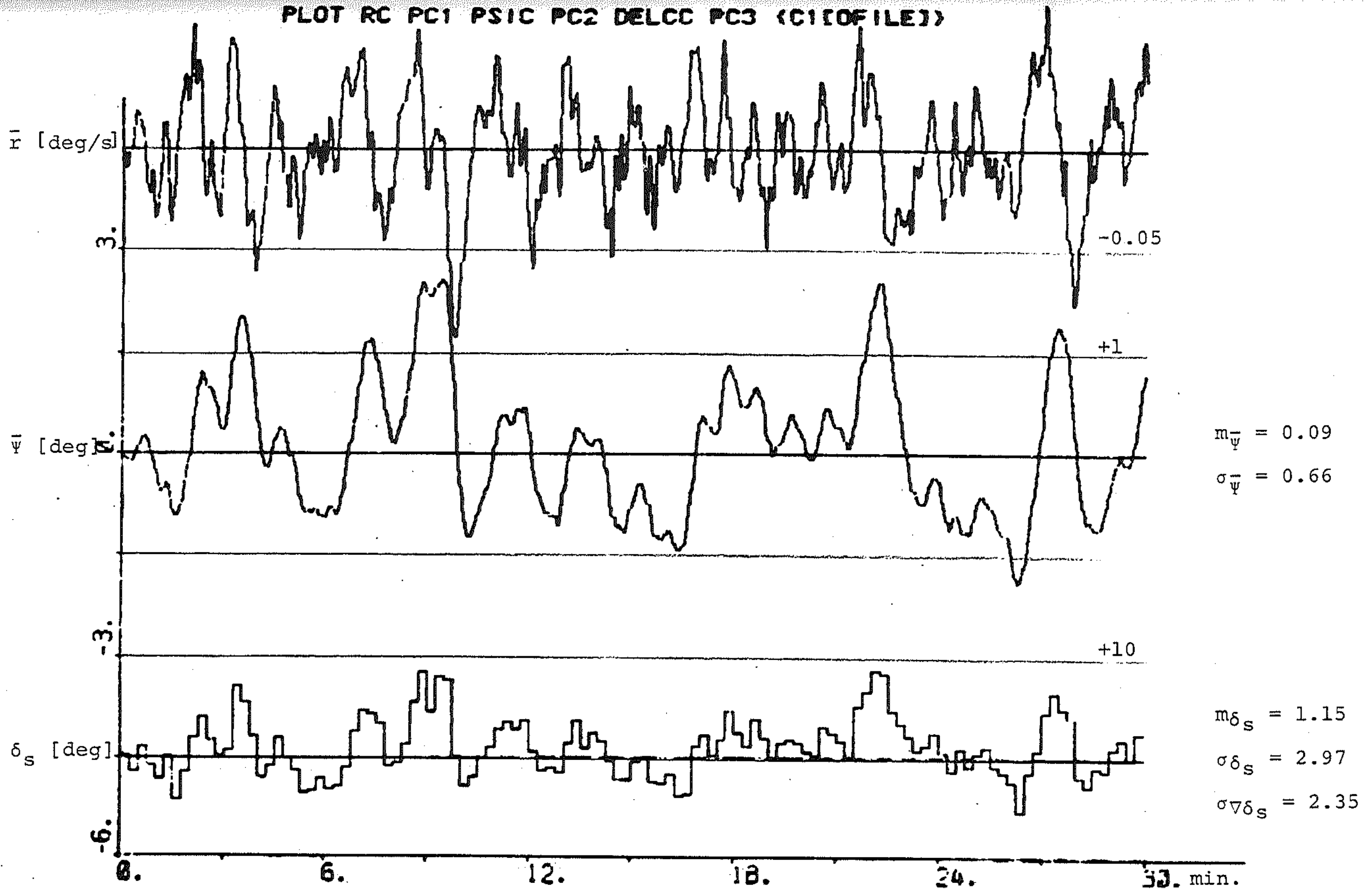
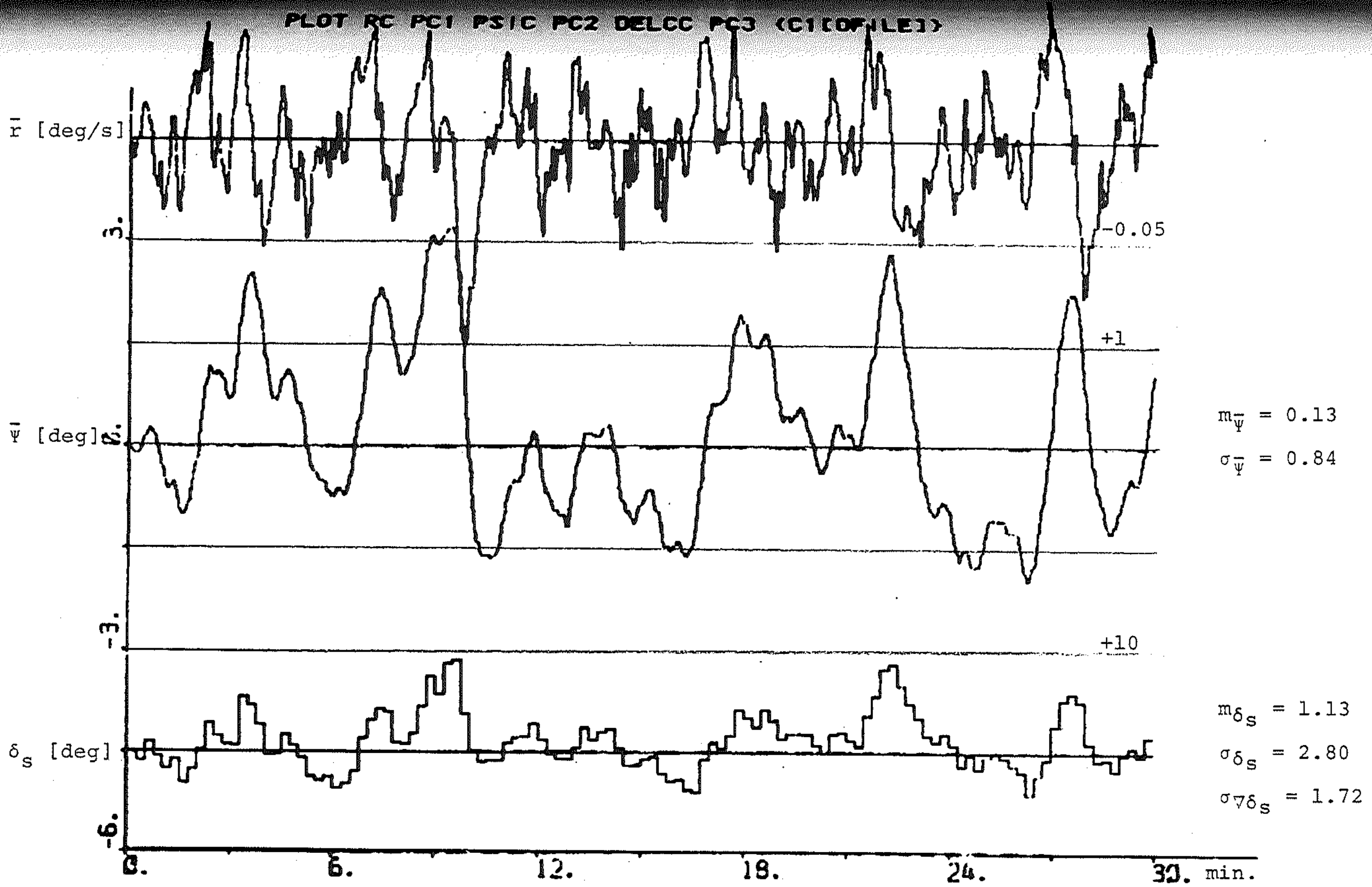
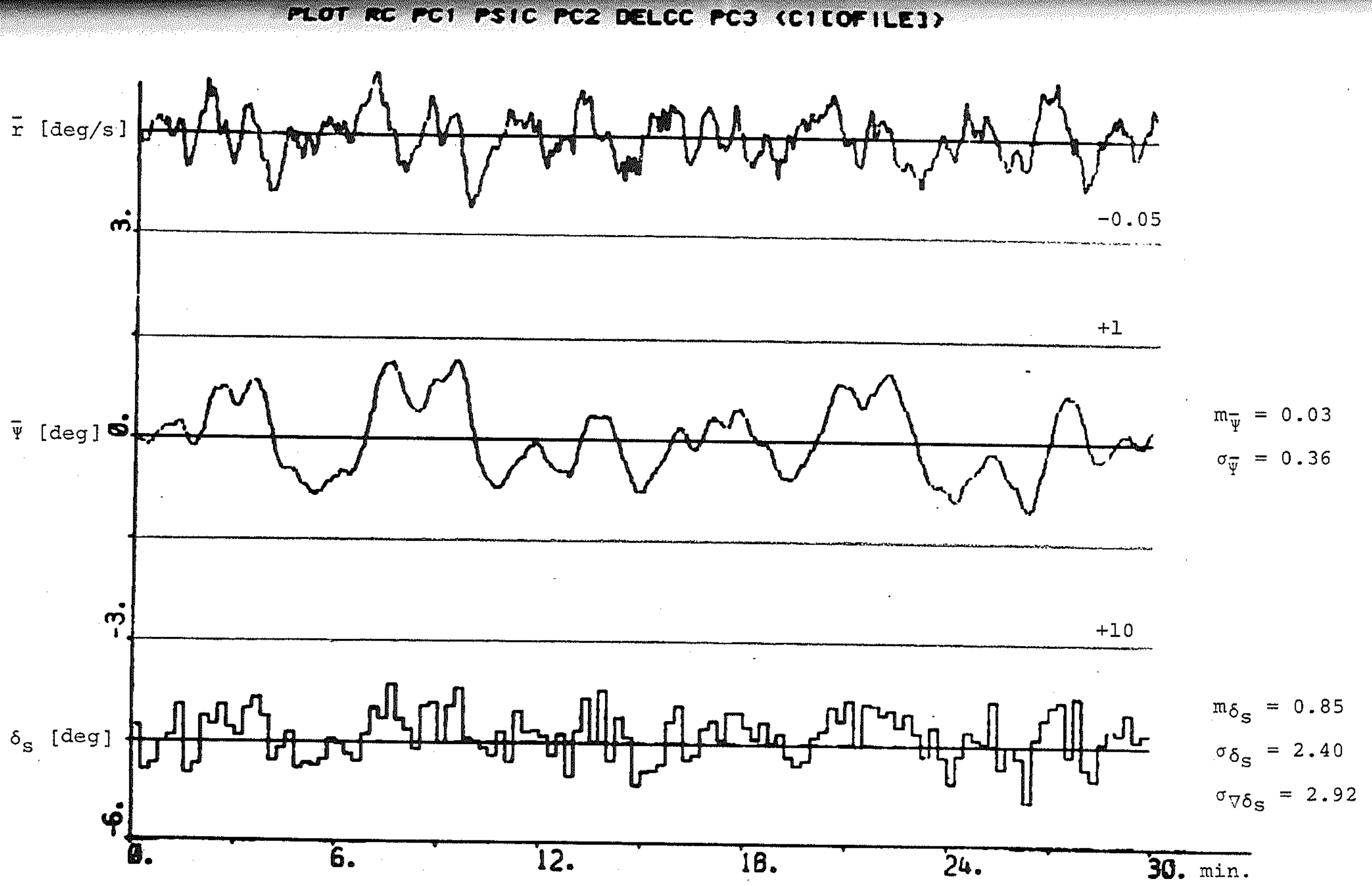


Fig. 4.53 - T = 10.5 m, fresh gale, regulator STURE (NA=3 NB=2 NC=1 k=4 q_2=0.1):
$V_1 = 1.46$ $V_2 = 1.33$ $V_3 = 1.00$



<u>Fig. 4.54</u> - T = 10.5 m, fresh gale, regulator STURE (NA=3 NB=2 NC=1 k=4 q_2=0.2):
$V_1 = 1.63$ $V_2 = 1.50$ $V_3 = 1.01$



<u>Fig. 4.55</u> - T = 20 m, moderate breeze, PID-regulator (k_P=4 k_D=80 k_I=0.02):
V_1 = 0.78 V_2 = 0.70 V_3 = 0.98.

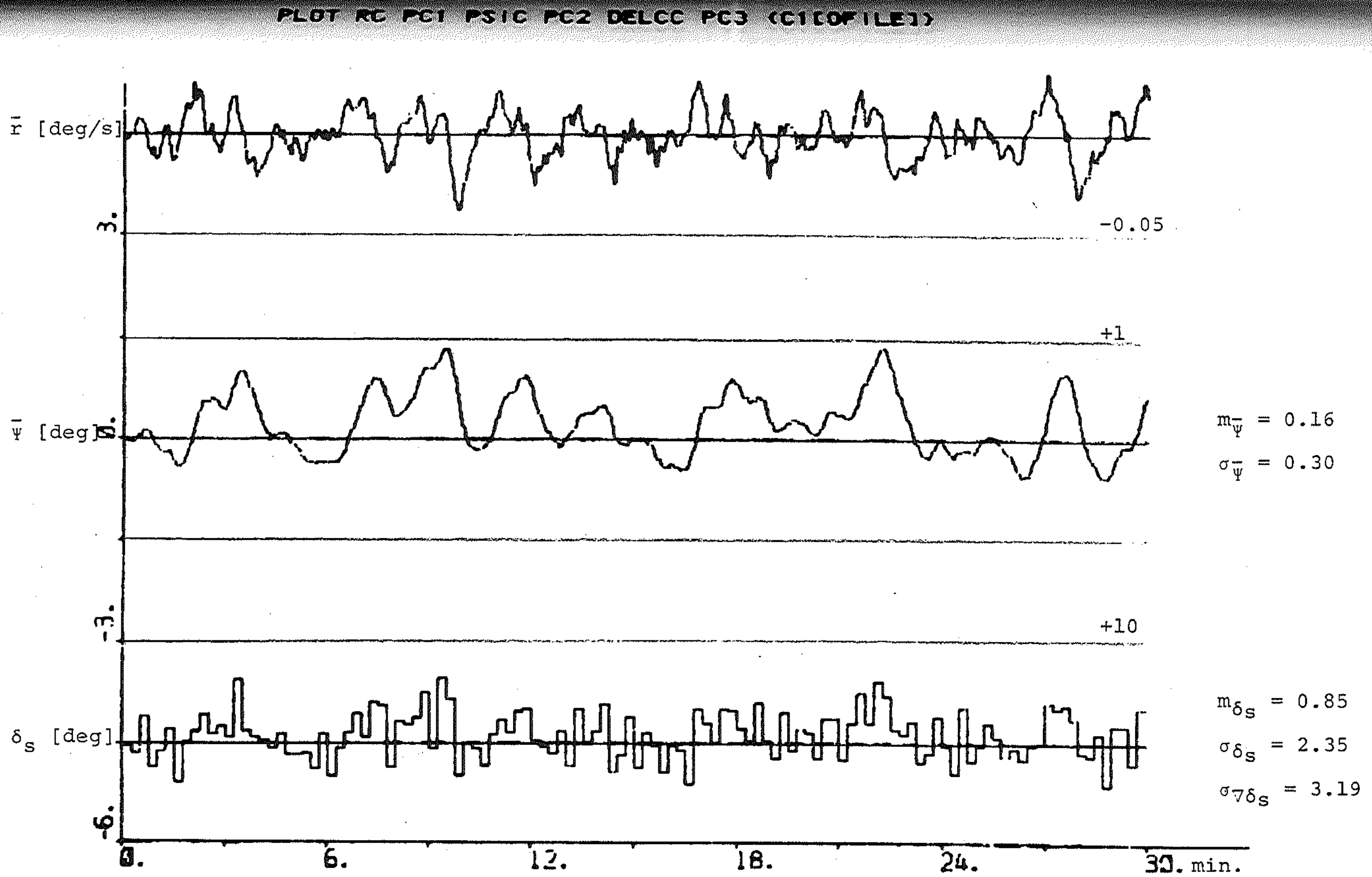


Fig. 4.56 - T = 20 m, moderate breeze, regulator STURE (NA=3 NB=2 NC=0 k=5 q_2=0): $V_1 = 0.74$ $V_2 = 0.67$ $V_3 = 1.13$.

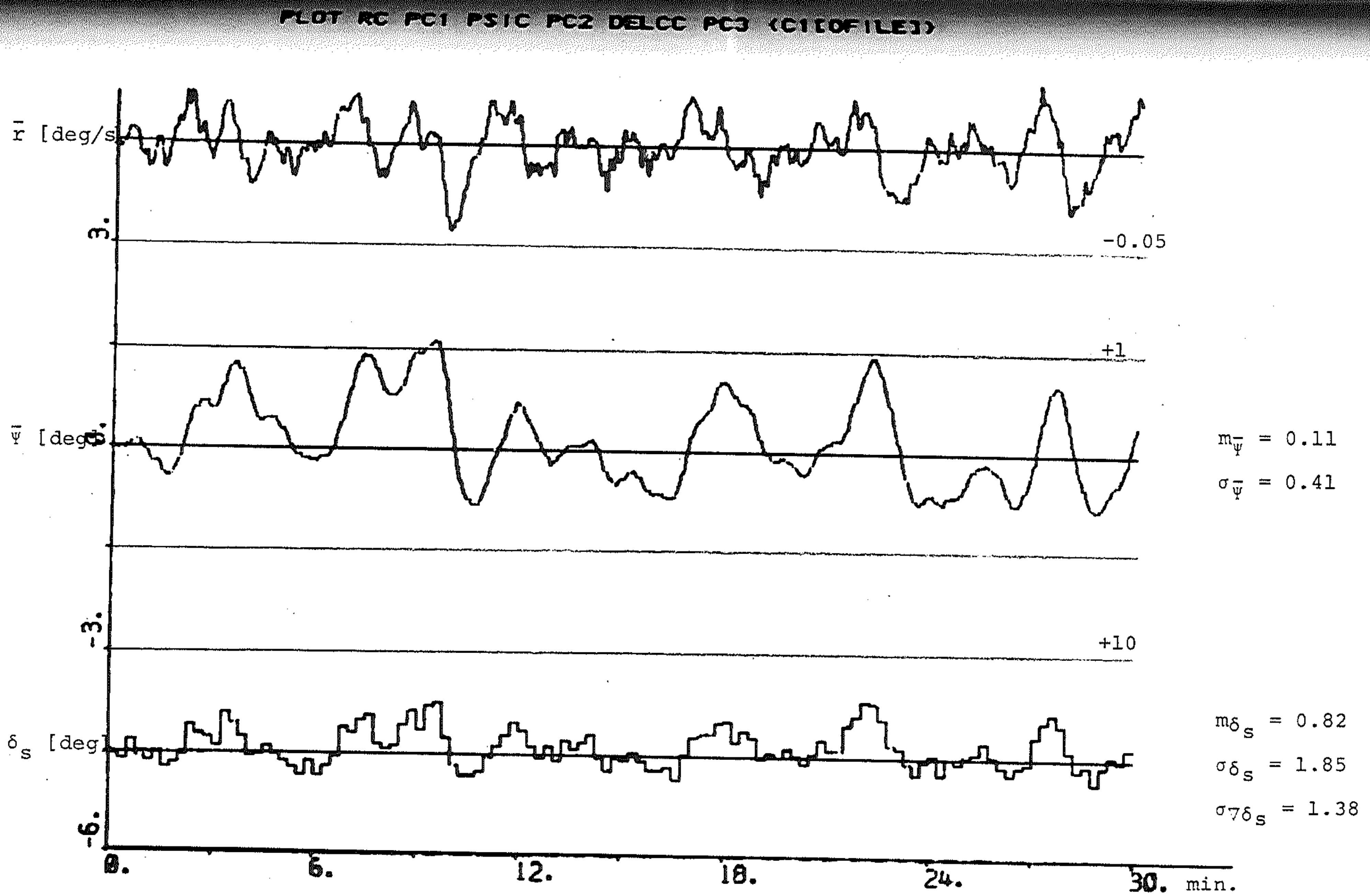


Fig. 4.57 - T = 20 m, moderate breeze, regulator STURE (NA=3 NB=2 NC=0 k=4 q_2=0.1): $V_1 = 0.60$ $V_2 = 0.53$ $V_3 = 0.38$.

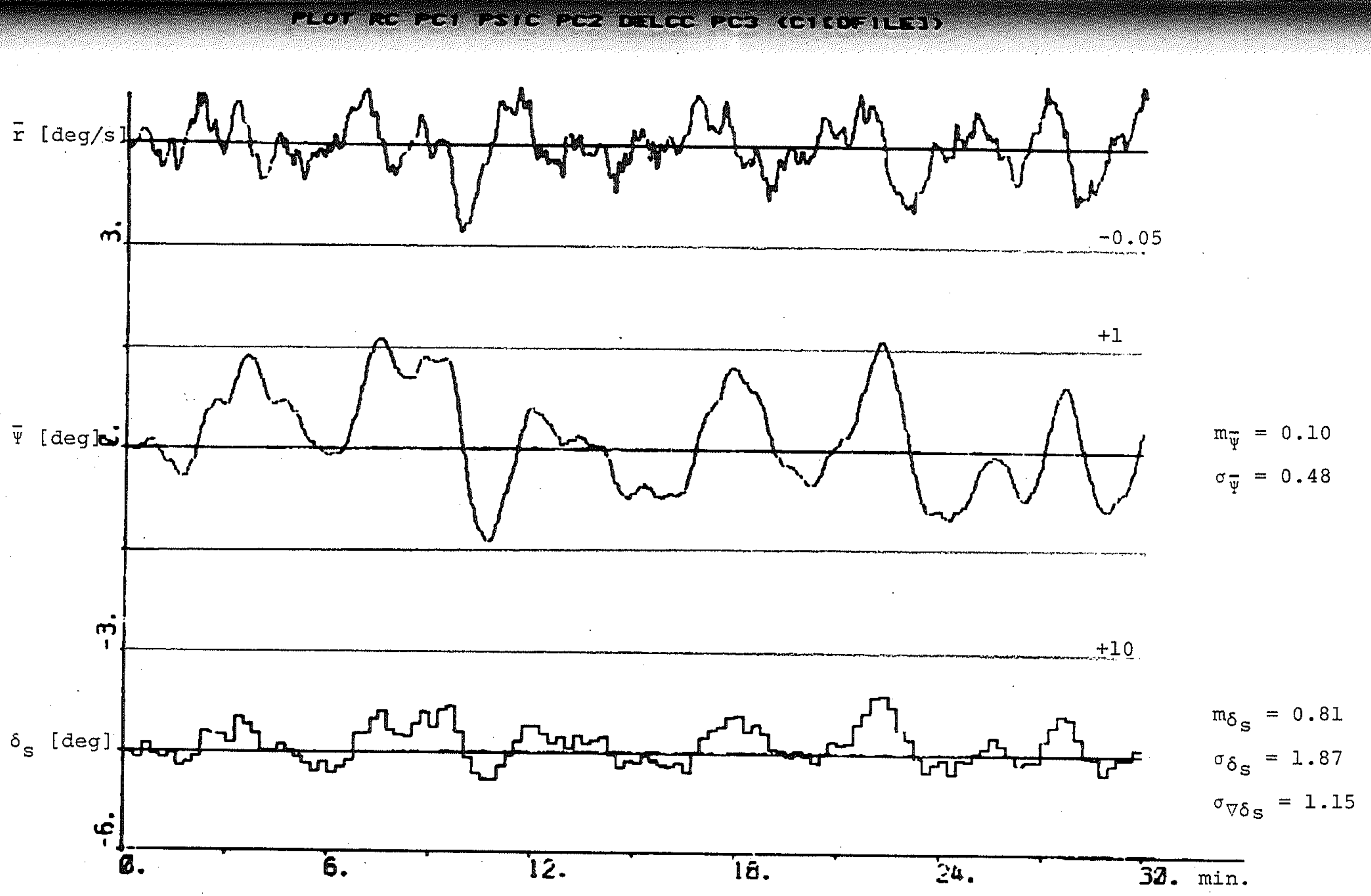


Fig. 4.58 - T = 20 m, moderate breeze, regulator STURE (NA=3 NB=2 NC=0 k=4 q_2=0.2): V_1 = 0.66 V_2 = 0.59 V_3 = 0.38.

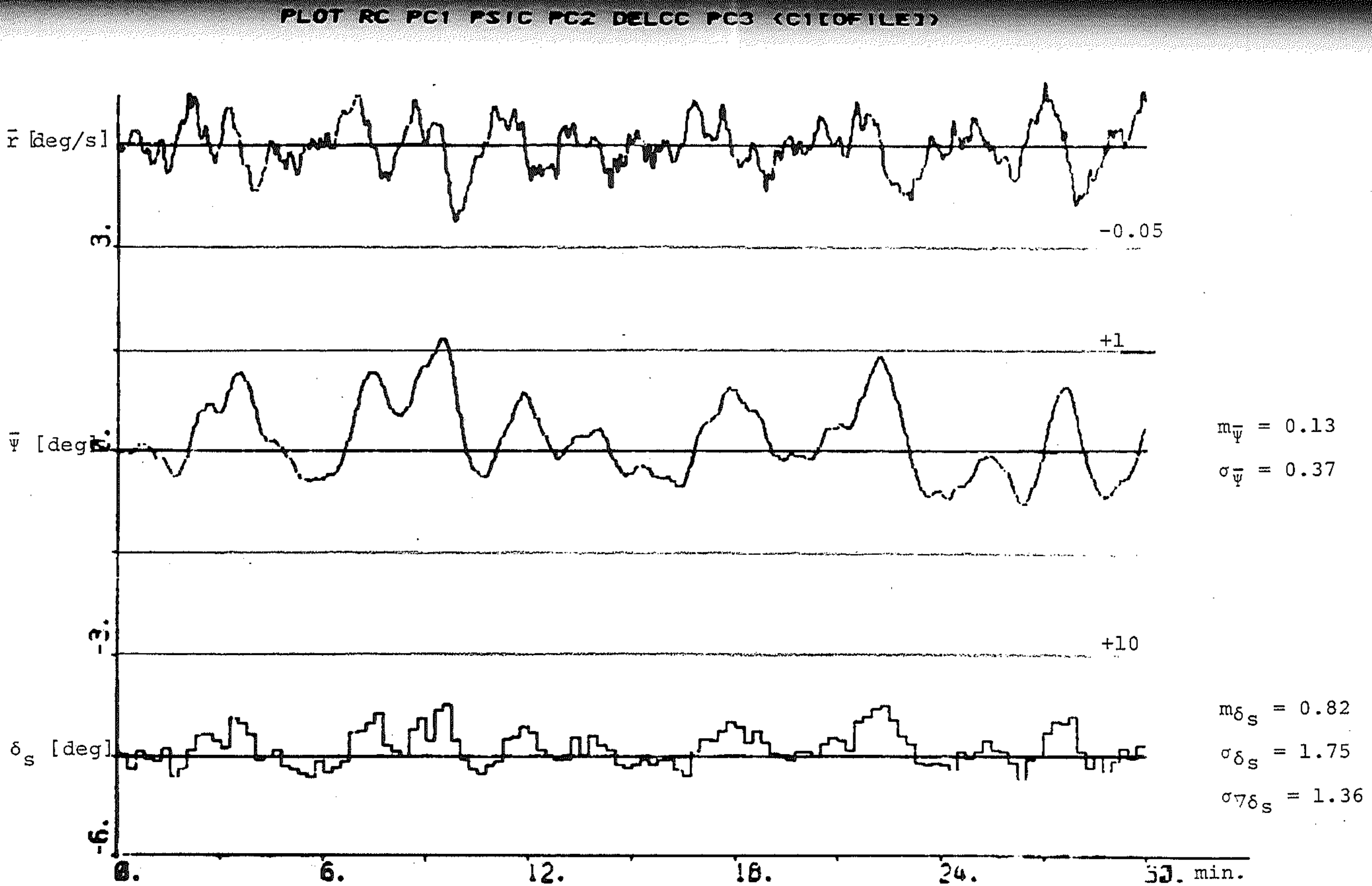


Fig. 4.59 - T = 20 m, moderate breeze, regulator STURE (NA=3 NB=2 NC=1 k=4 q_2=0.1): $V_1 = 0.53$ $V_2 = 0.46$ $V_3 = 0.34$.

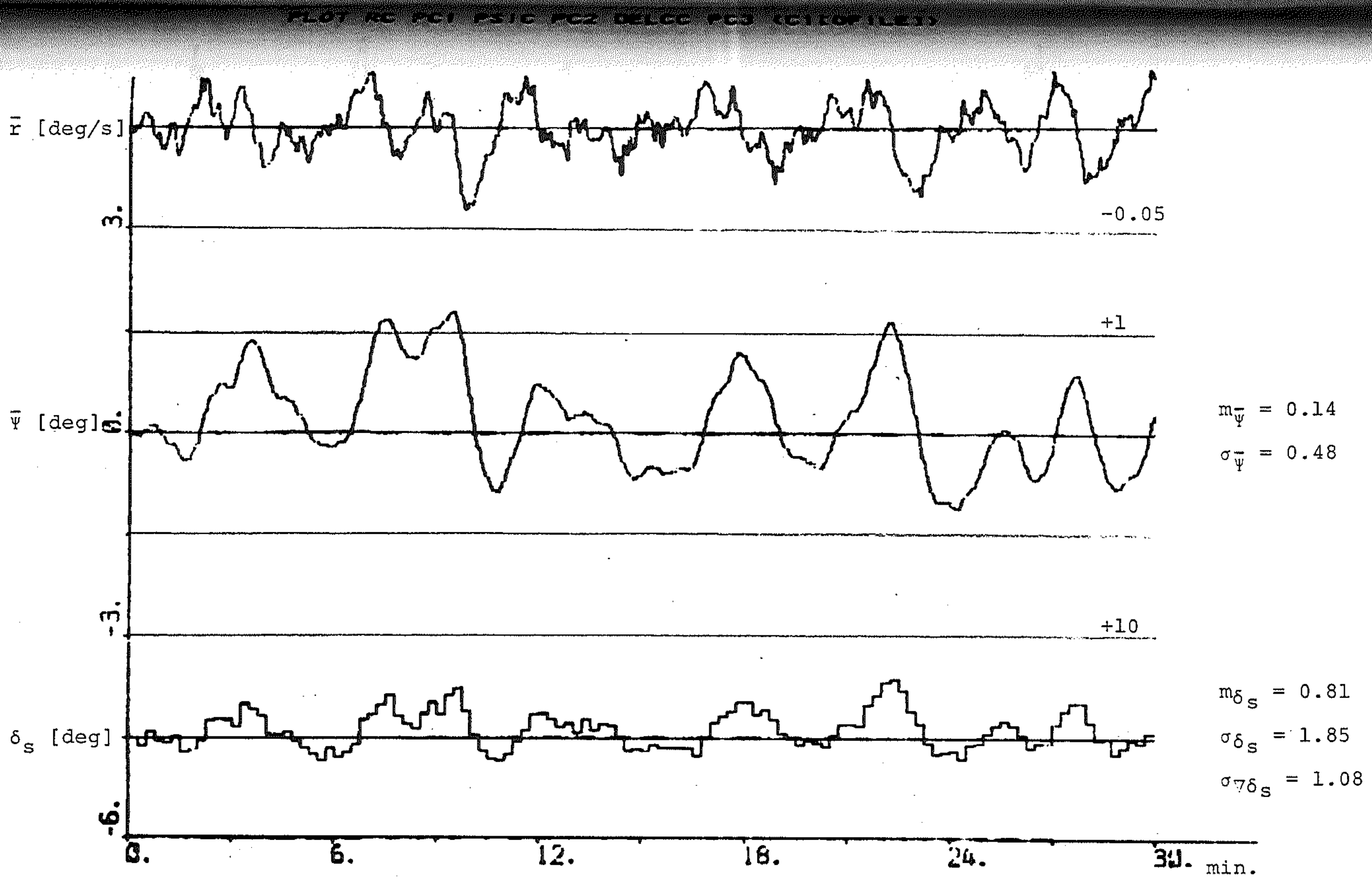


Fig. 4.60 - T = 20 m, moderate breeze, regulator STURE (NA=3 NB=2 NC=1 k=4 $q_2=0.2$): $V_1 = 0.65$ $V_2 = 0.59$ $V_3 = 0.36$

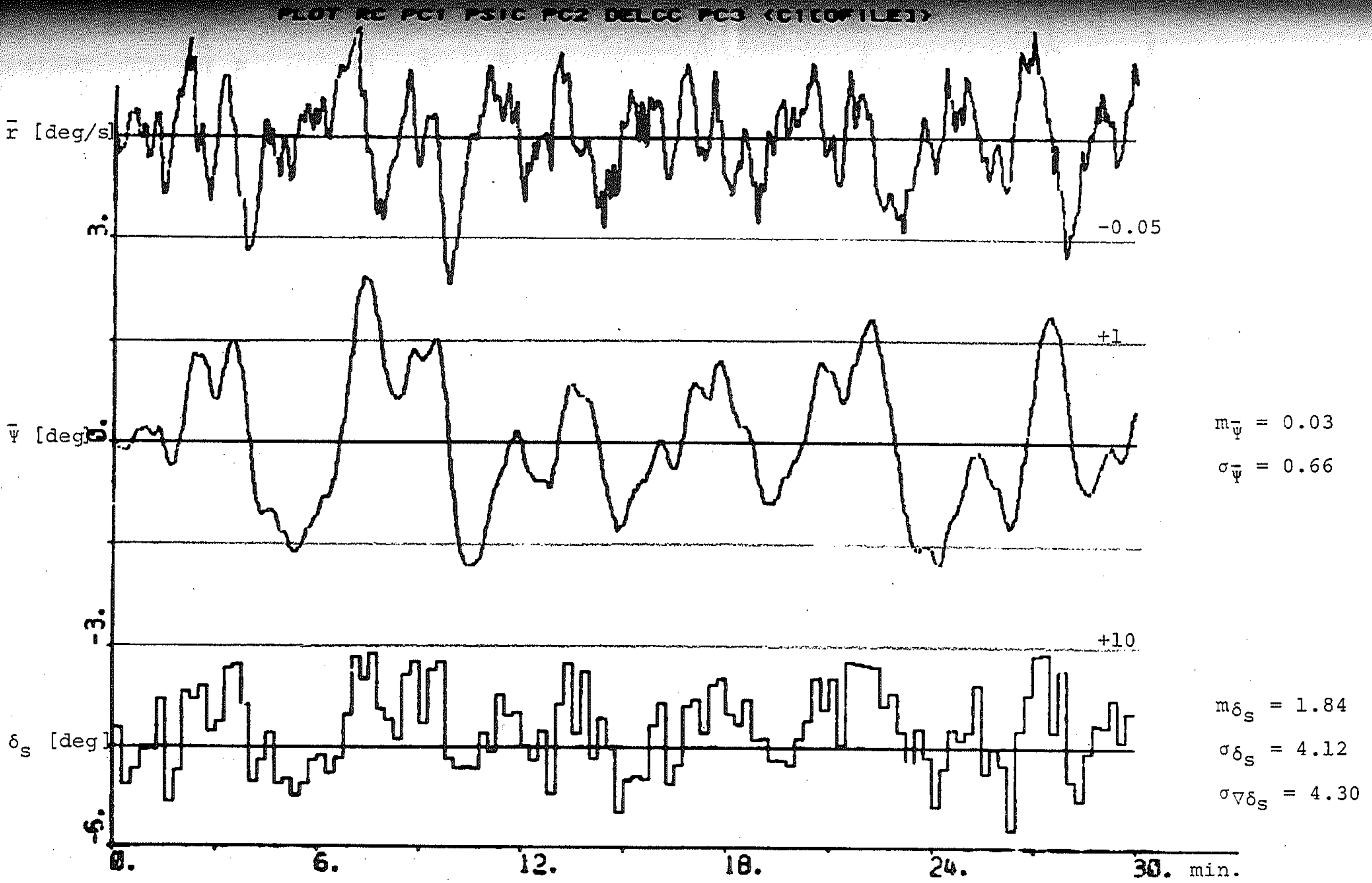


Fig. 4.61 - T = 20 m, fresh gale, PID-regulator ($k_P=4$ $k_D=100$ $k_I=0.04$):
$V_1 = 2.47$ $V_2 = 2.13$ $V_3 = 2.29$

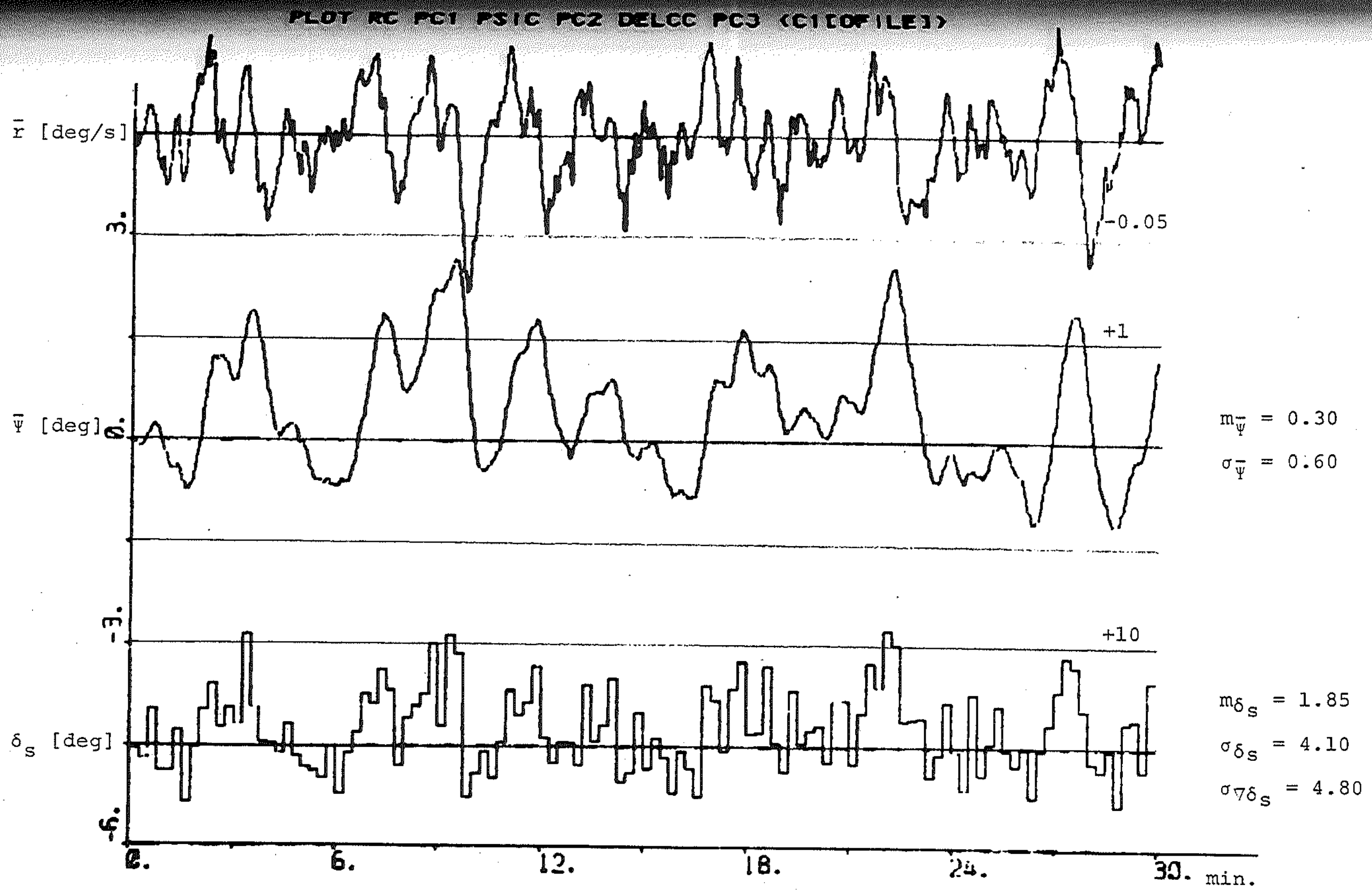


Fig. 4.62 - T = 20 m, fresh gale, regulator STURE (NA=3 NB=2 NC=0 k=5 q_2=0):
$V_1 = 2.47$ $V_2 = 2.13$ $V_3 = 2.74$

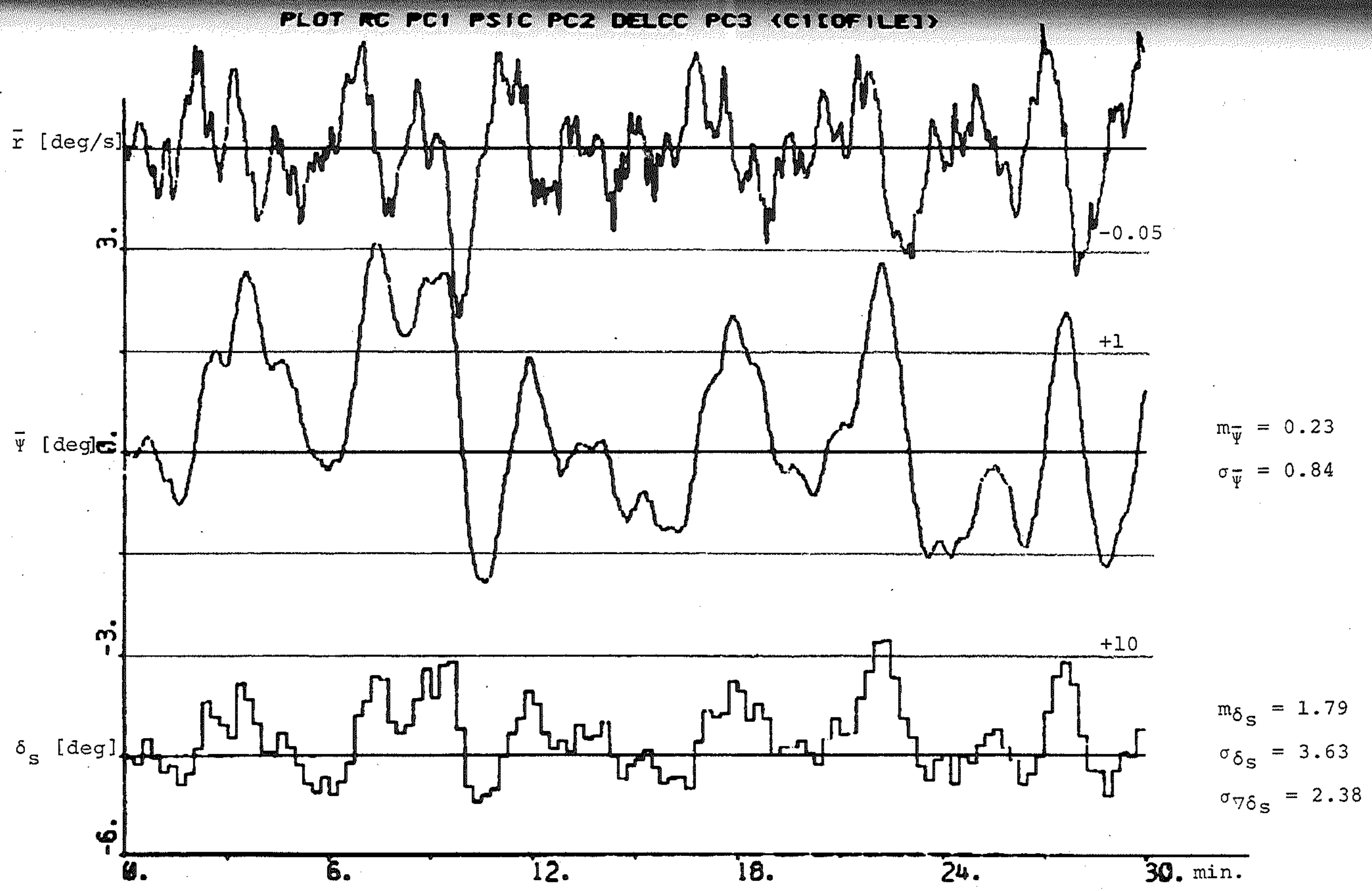


Fig. 4.63 - T = 20 m, fresh gale, regulator STURE (NA=3 NB=2 NC=0 k=4 q_2=0.1):

$V_1 = 2.39$ $V_2 = 2.06$ $V_3 = 1.32$

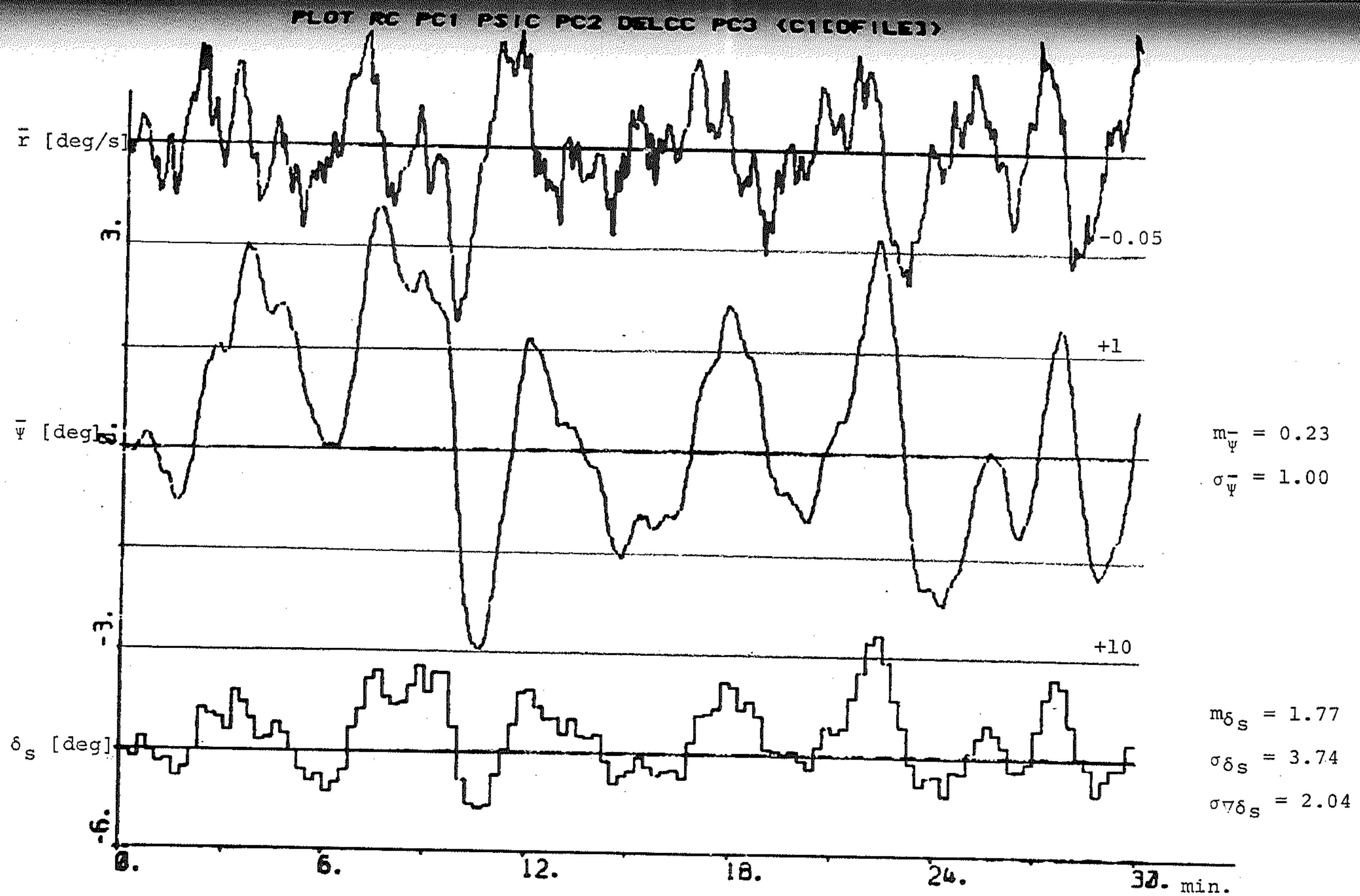


Fig. 4.64 - T = 20 m, fresh gale, regulator STURE (NA=3 NB=2 NC=0 k=4 q_2=0.2):
$V_1 = 2.76$ $V_2 = 2.45$ $V_3 = 1.47$

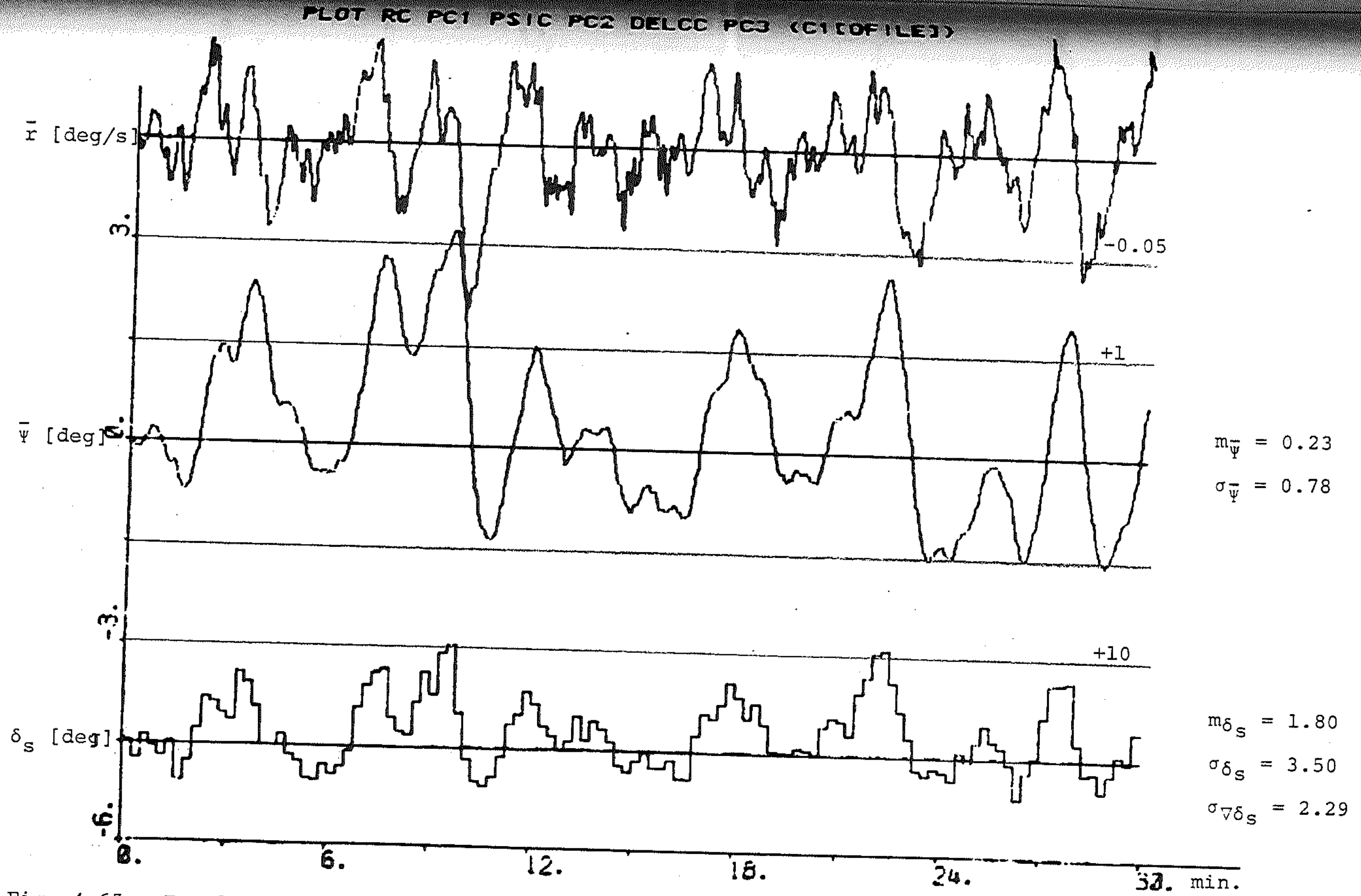


Fig. 4.65 - T = 20 m, fresh gale, regulator STURE (NA=3 NB=2 NC=1 k=4 q_2=0.1):
$V_1 = 2.20$ $V_2 = 1.88$ $V_3 = 1.18$

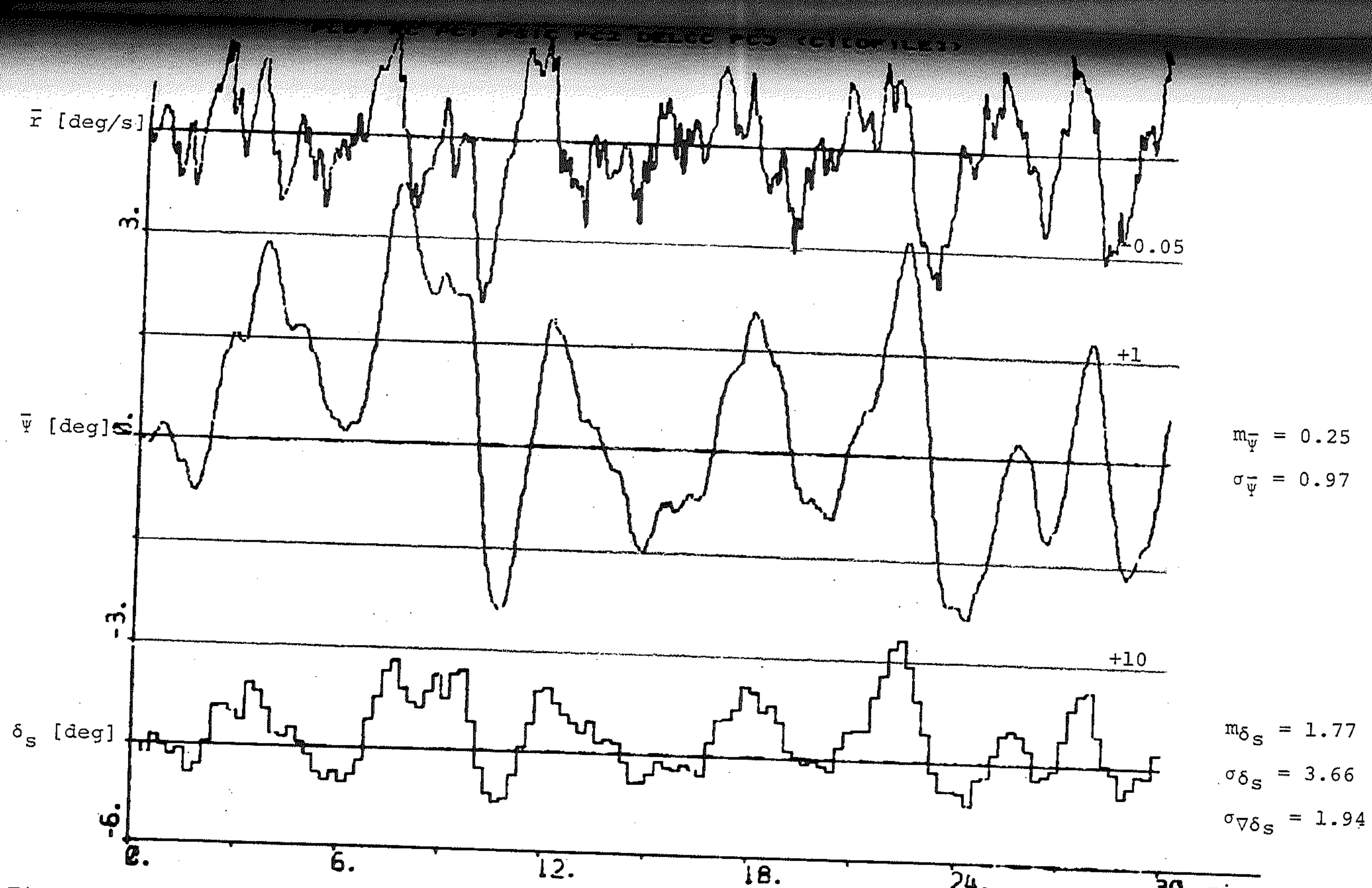


Fig. 4.66 - T = 20 m, fresh gale, regulator STURE (NA=3 NB=2 NC=1 k=4 q_2=0.2):
$V_1 = 2.65$ $V_2 = 2.34$ $V_3 = 1.38$

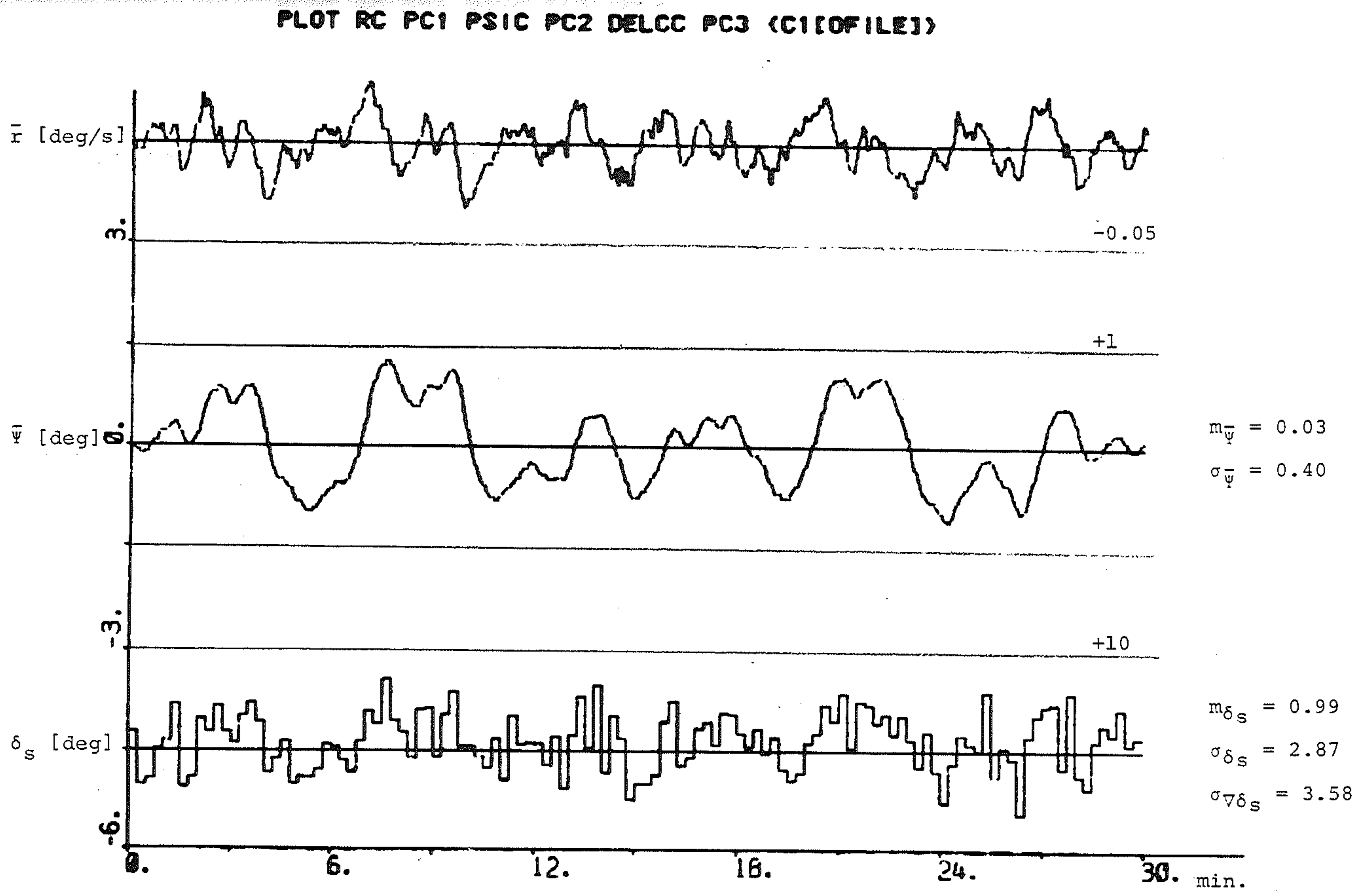


Fig. 4.67 - T = 25 m, moderate breeze, PID-regulator ($k_p=4$ $k_D=100$ $k_I=0.02$):
$V_1 = 1.08$ $V_2 = 0.99$ $V_3 = 1.45$

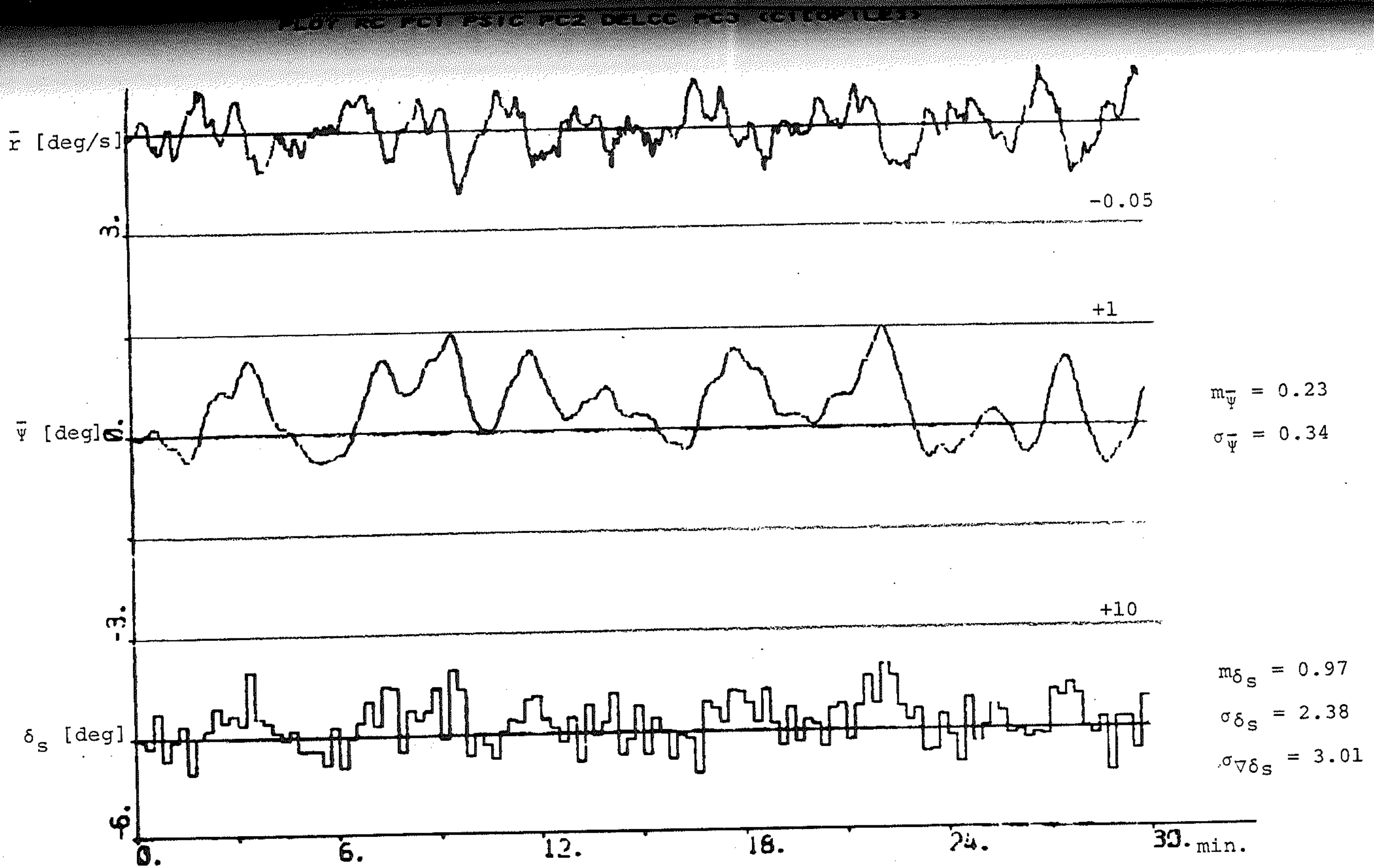


Fig. 4.68 - T = 25 m, moderate breeze, regulator STURE (NA=3 NB=2 NC=0 k=5 q_2=0):
$V_1 = 0.83$ $V_2 = 0.74$ $V_3 = 1.08$

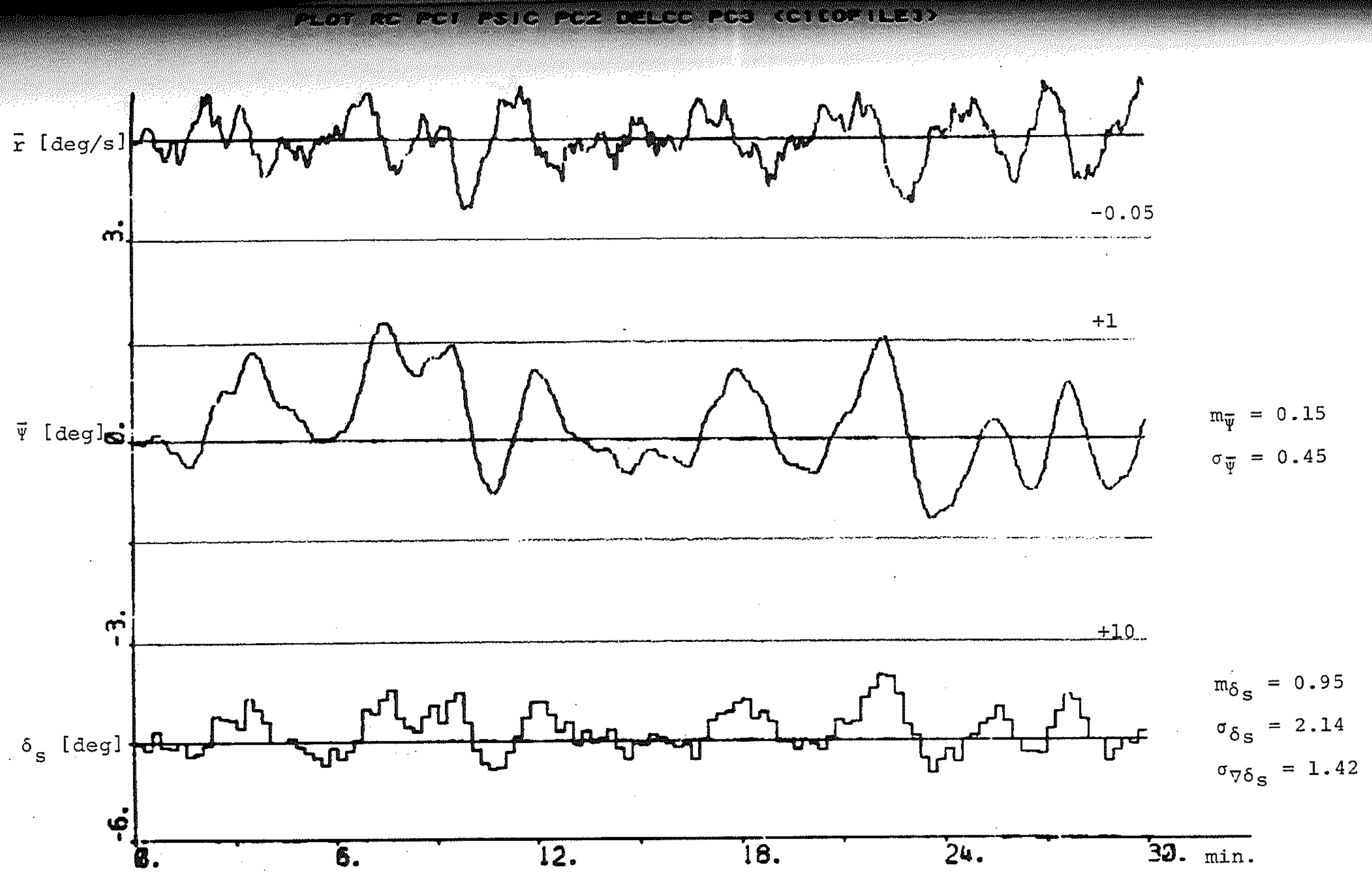


Fig. 4.69 - $T = 25$ m, moderate breeze, regulator STURE ($NA=3$ $NB=2$ $NC=0$ $k=4$ $q_2=0.1$): $V_1 = 0.77$ $V_2 = 0.68$ $V_3 = 0.43$.

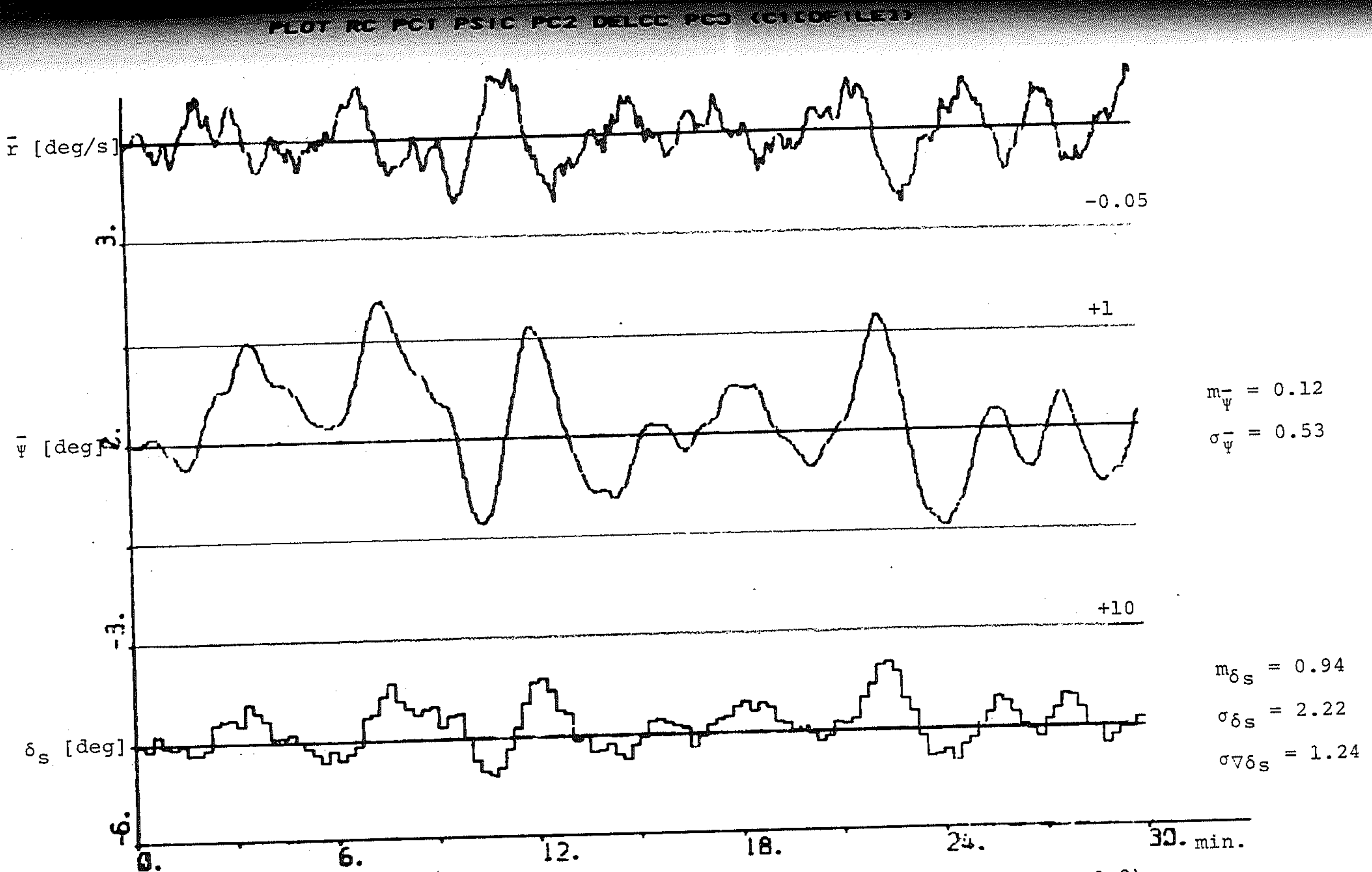


Fig. 4.70 - T = 25 m, moderate breeze, regulator STURE (NA=3 NB=2 NC=0 k=4 q_2=0.2): $V_1 = 0.87$ $V_2 = 0.78$ $V_3 = 0.45$.

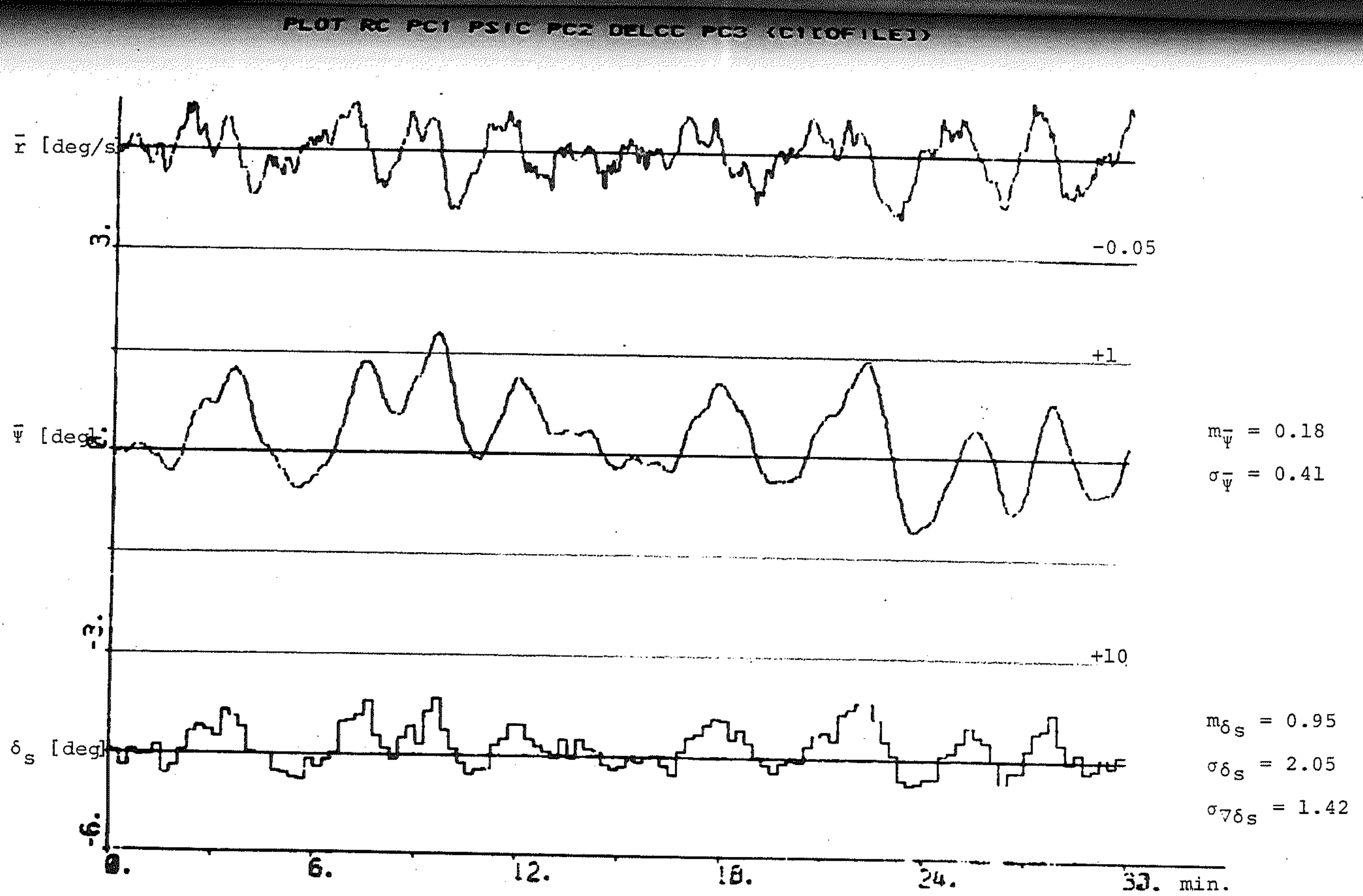


Fig. 4.71 - T = 25 m, moderate breeze, regulator STURE (NA=3 NB=2 NC=1 k=4 q_2=0.1): $V_1 = 0.72$ $V_2 = 0.62$ $V_3 = 0.40$

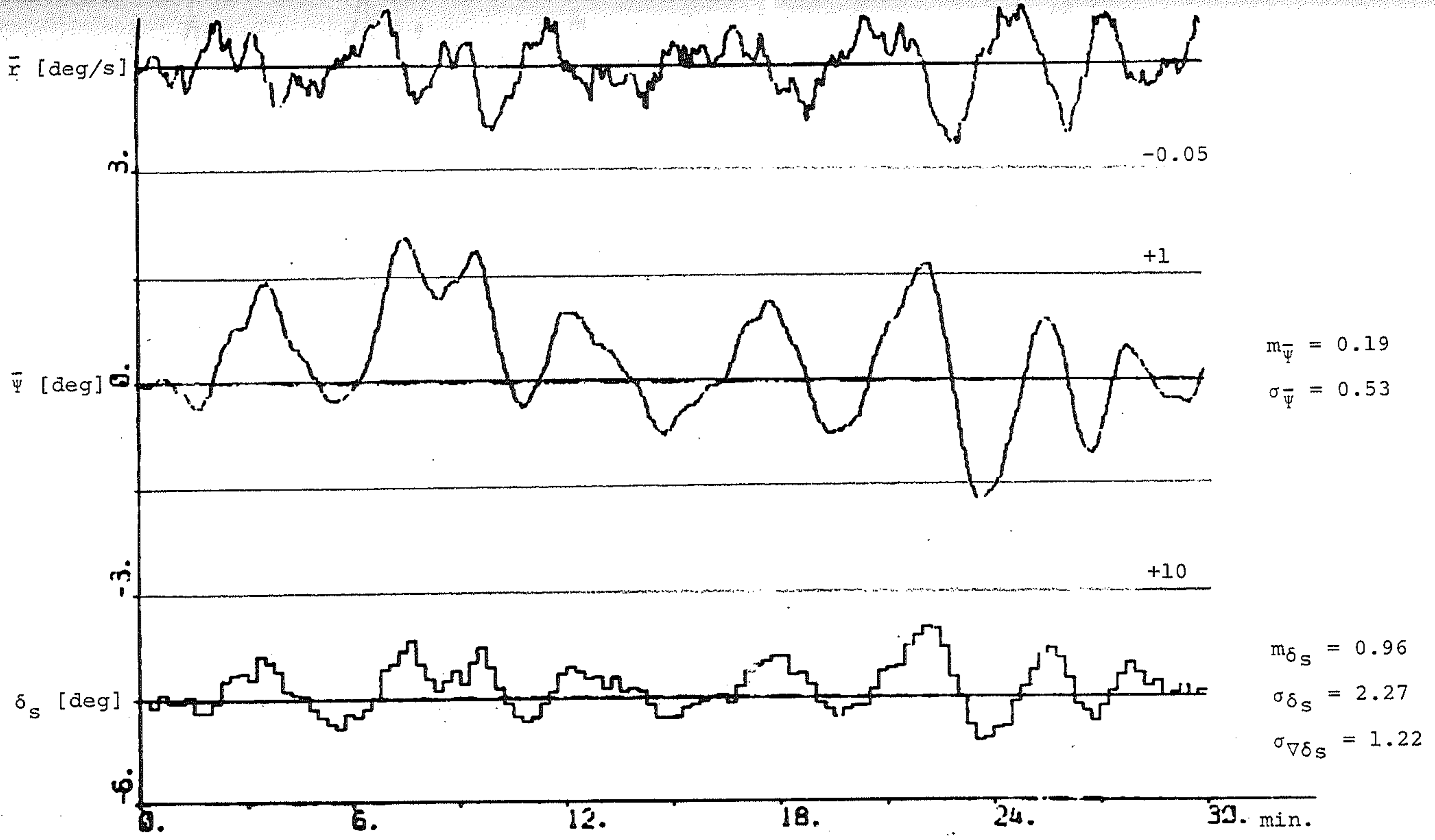


Fig. 4.72 - T = 25 m, moderate breeze, regulator STURE (NA=3 NB=2 NC=1 k=4 q_2=0.2): $V_1 = 0.93$ $V_2 = 0.83$ $V_3 = 0.47$

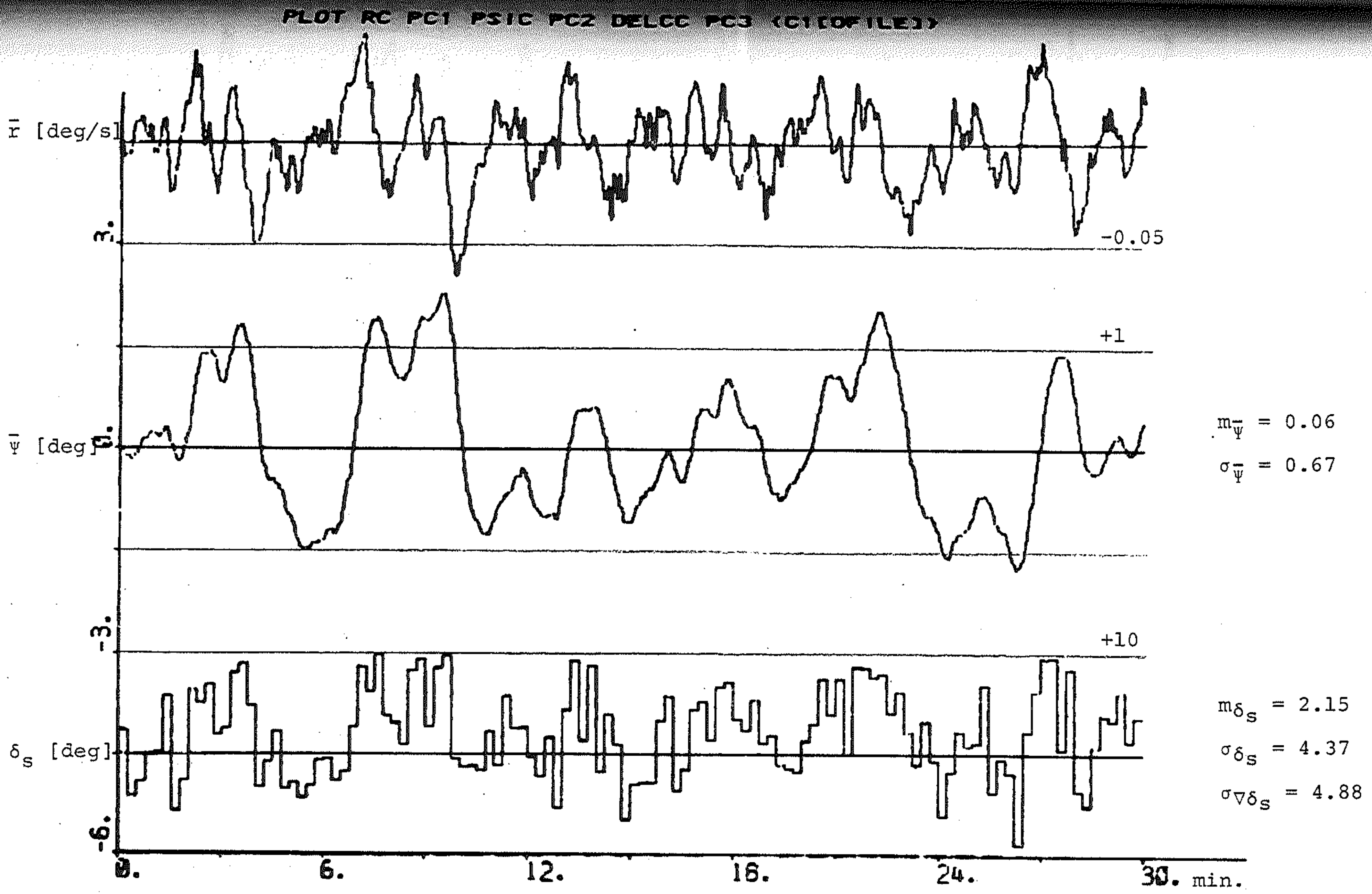


Fig. 4.73 - T = 25 m, fresh gale, PID-regulator (k_P=4 k_D=120 k_I=0.02):

$V_1 = 2.82$ $V_2 = 2.35$ $V_3 = 2.83$

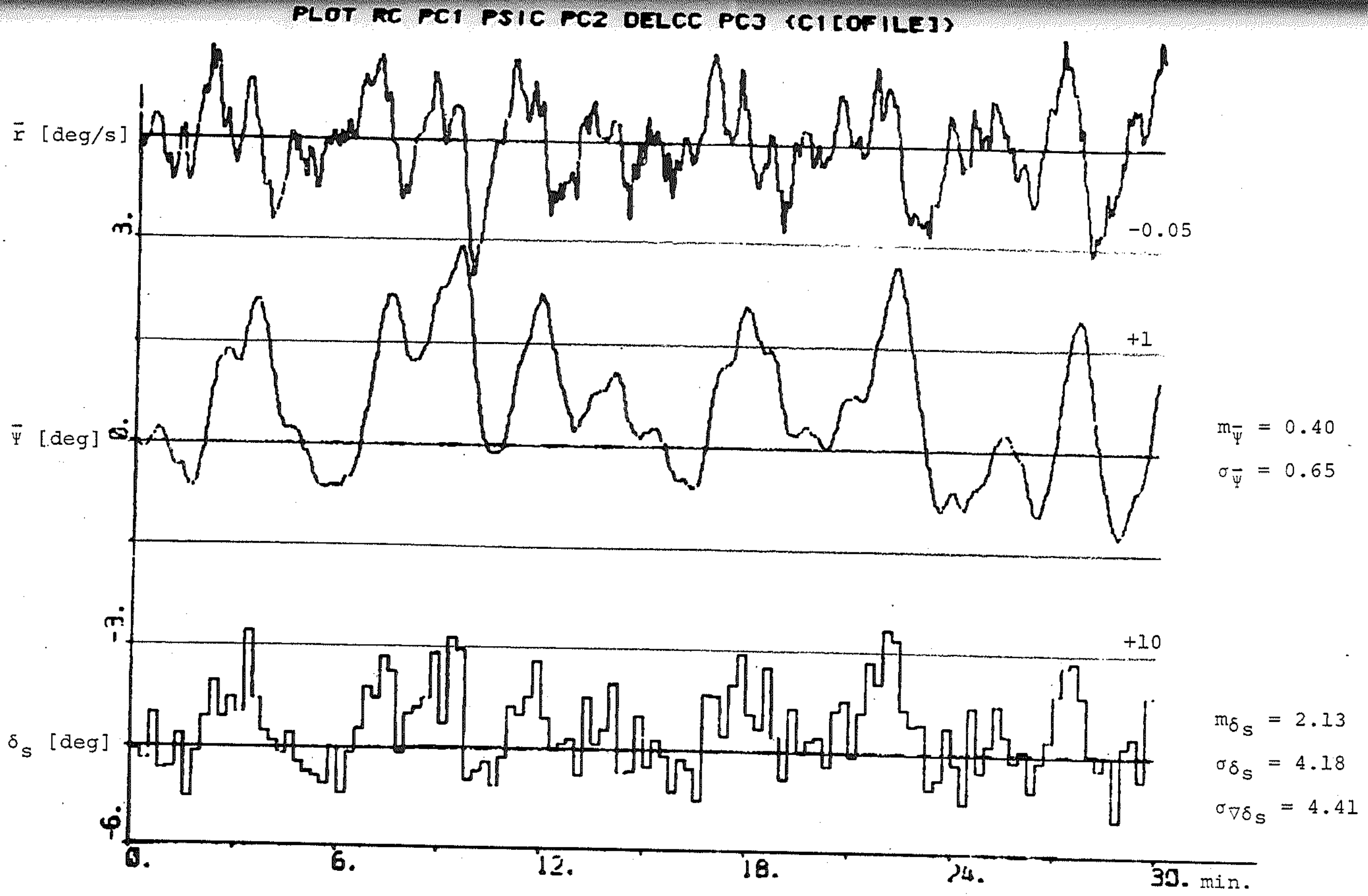


Fig. 4.74 - T = 25 m, fresh gale, regulator STURE (NA=3 NB=2 NC=0 k=5 q_2=0):
$V_1 = 2.78$ $V_2 = 2.33$ $V_3 = 2.53$

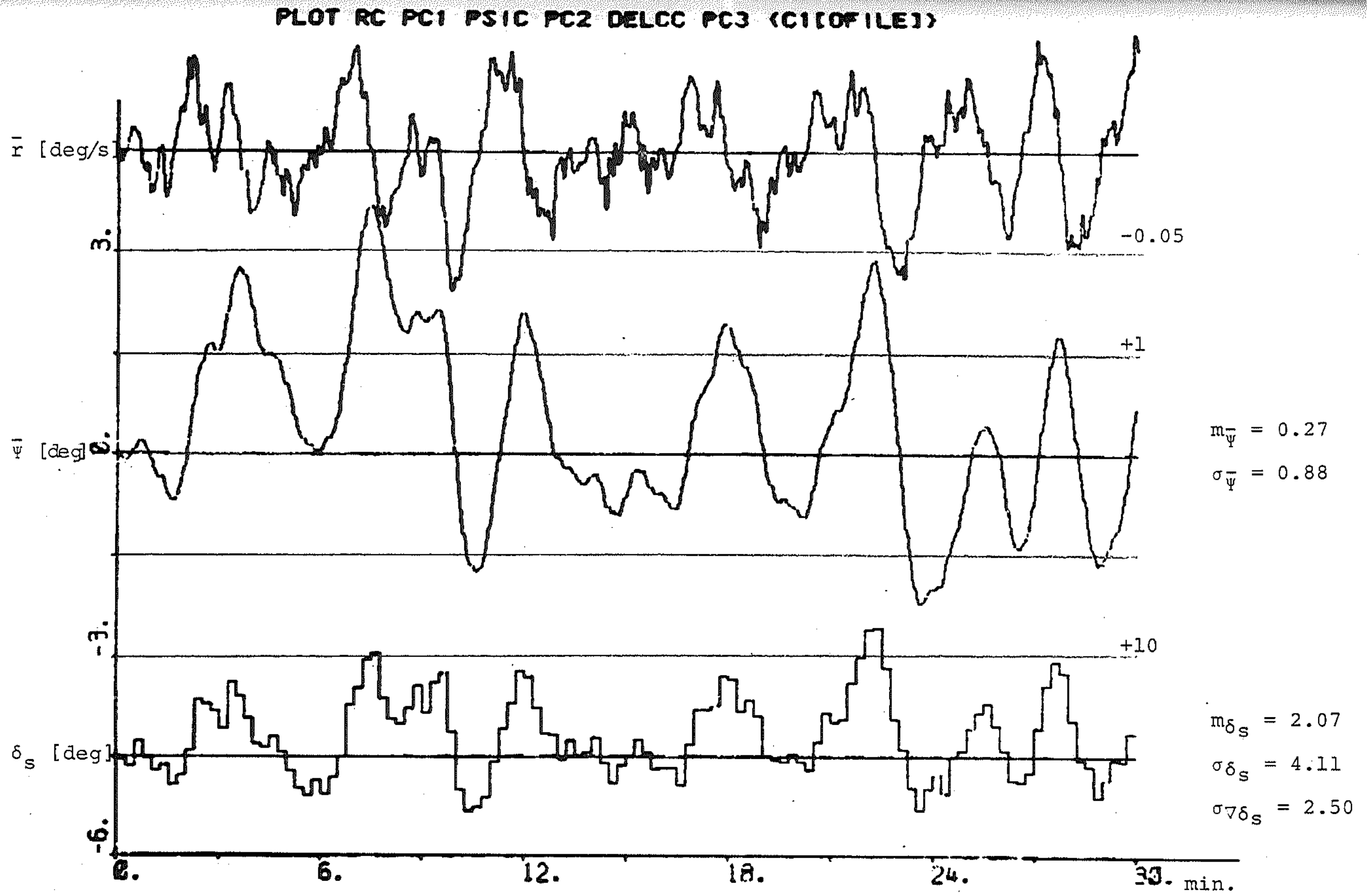


Fig. 4.75 - T = 25 m, fresh gale, regulator STURE (NA=3 NB=2 NC=0 k=4 q_2=0.1):
$V_1 = 2.97$ $V_2 = 2.54$ $V_3 = 1.47$.

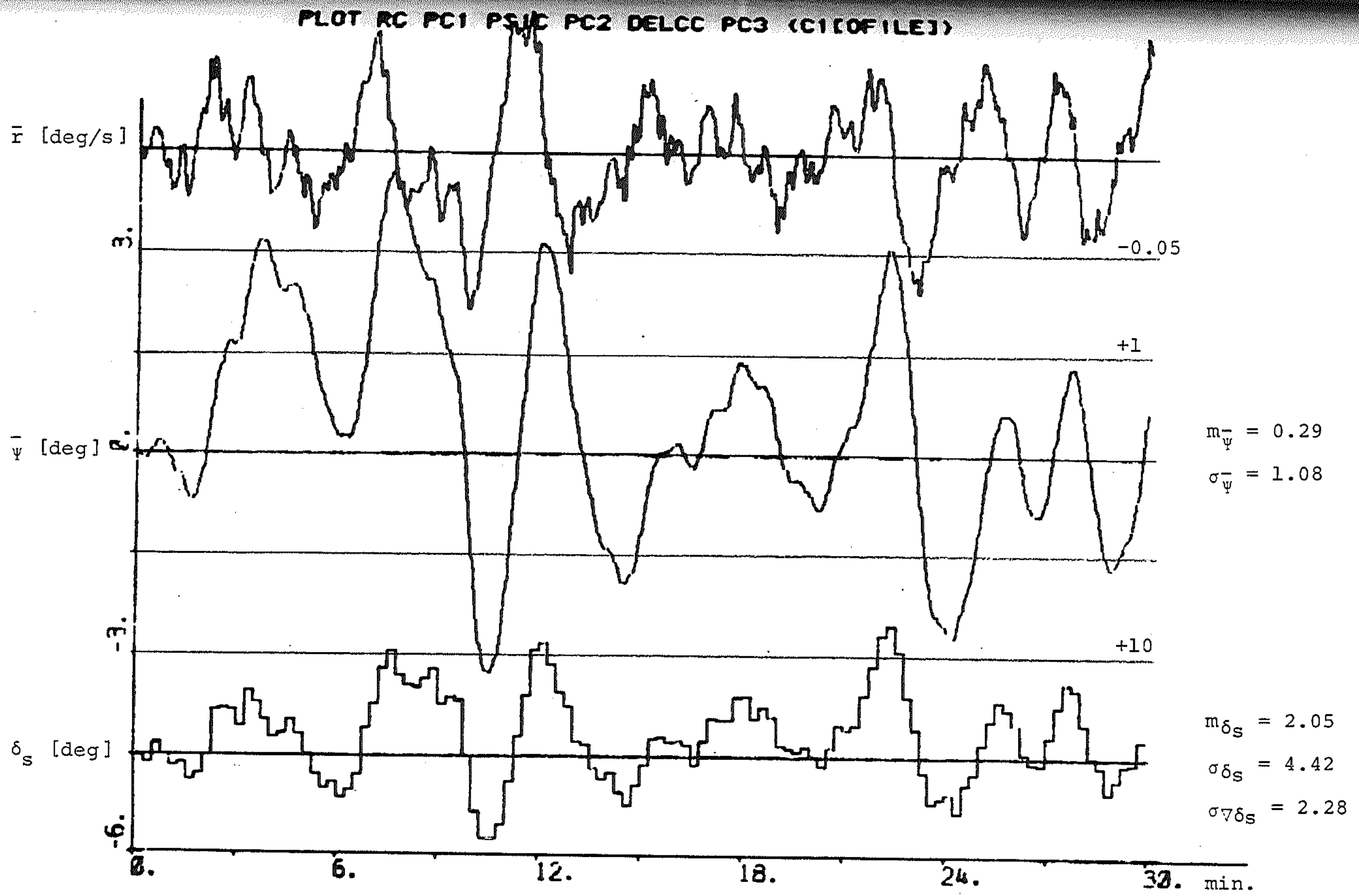


Fig. 4.76 - T = 25 m, fresh gale, regulator STURE (NA=3 NB=2 NC=0 k=4 q_2=0.2):
$V_1 = 3.62$ $V_2 = 3.20$ $V_3 = 1.77$.

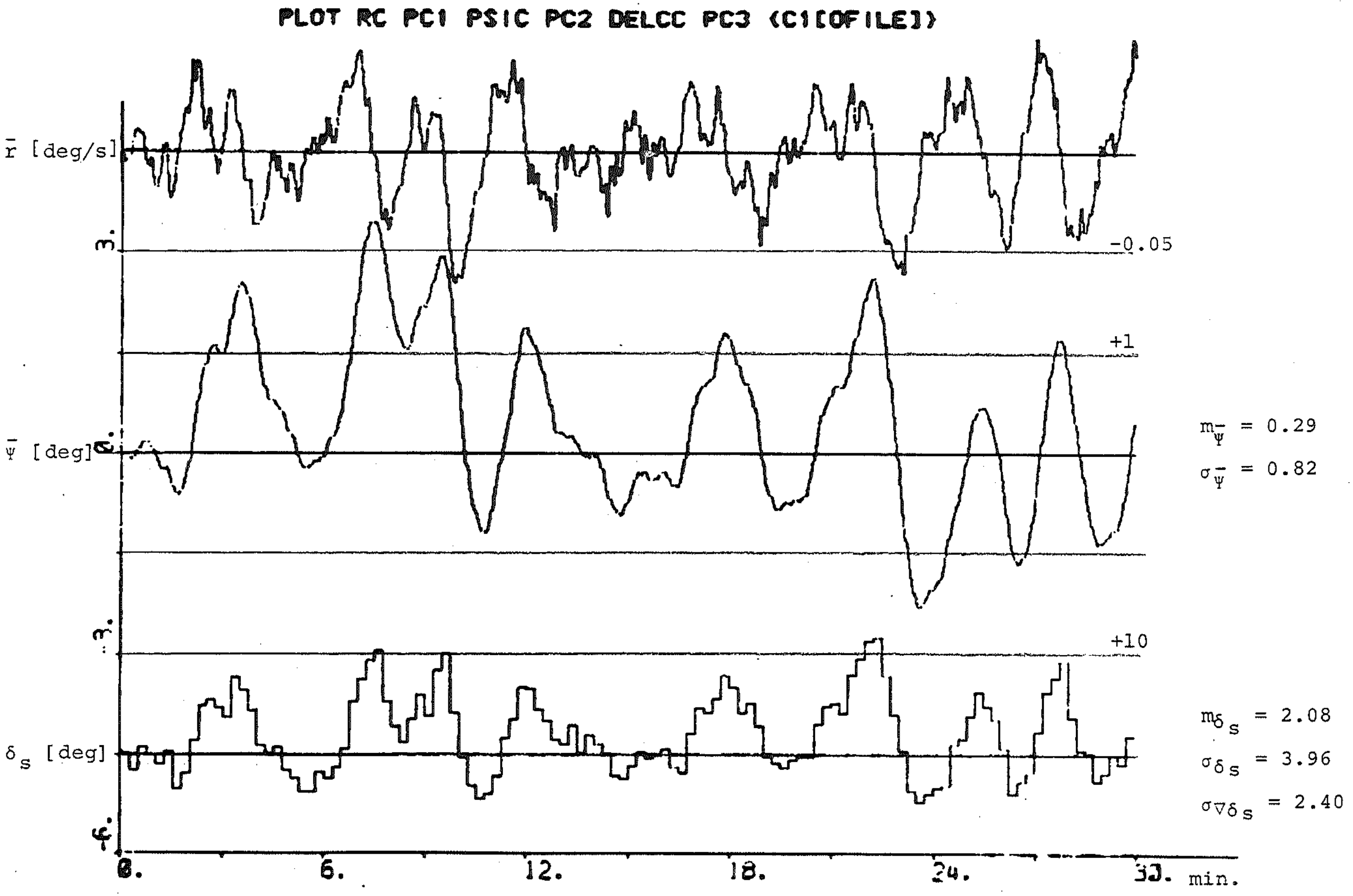


Fig. 4.77 - T = 25 m, fresh gale, regulator STURE (NA=3 NB=2 NC=1 k=4 q_2=0.1):
$V_1 = 2.76$ $V_2 = 2.33$ $V_3 = 1.34$

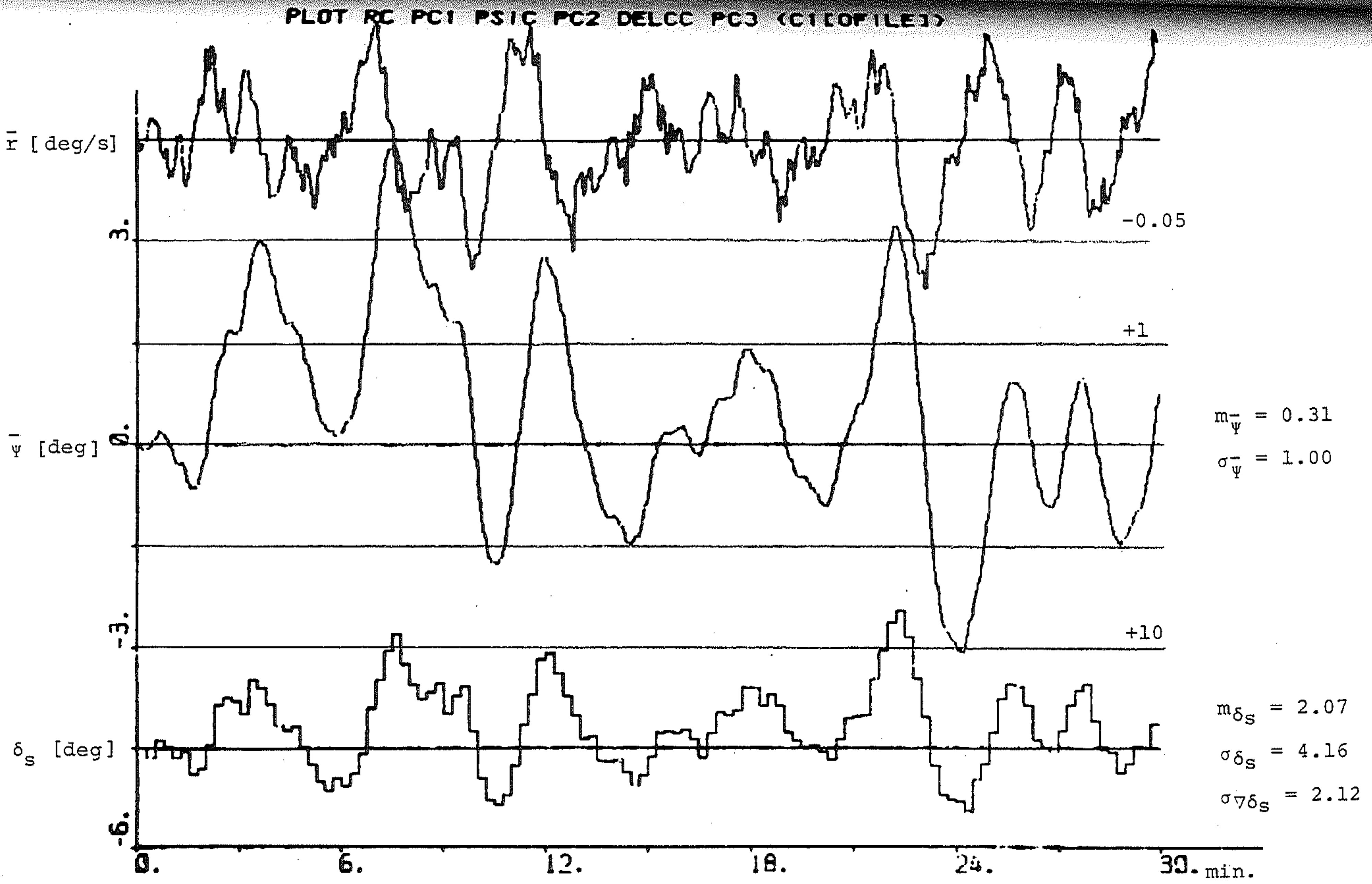


Fig. 4.78 - T = 25 m, fresh gale, regulator STURE (NA=3 NB=2 NC=1 k=4 q_2=0.2):
$V_1 = 3.26$ $V_2 = 2.84$ $V_3 = 1.56$

From Tables 4.5 - 4.7 it is concluded that STURE with NC = 1 in most cases is a little bit better than STURE with NC = 0. The structure NA = 3, NB = 2, NC = 1, k = 4, q_2 = 0.1 of STURE gives the best steering quality in average, and this regulator is to prefer in front of both a well tuned PID-regulator and STURE with the structure NA = 3, NB = 2, NC = 0, k = 5, q_2 = 0.

5. CONCLUSIONS

The simulations have shown that the simple, self-tuning regulator STURE with the structure NA = 3, NB = 2, NC = 1, k = 4, $q_2 = 0.1$, $b_0 = -1$, $\lambda_f = 0.99$, $T_s = 15$ s in average gives the best steering quality in varying load and wind conditions. Neither a well tuned PID-regulator nor the self-tuning regulator STURE3, which contains the solving of a Riccati equation, gives as good steering quality as STURE. If the optimal structure of STURE is changed in such a way that NC = 0, the steering quality is slightly, but not significantly, decreased. An attempt to estimate b_0 of STURE failed because the sign of the parameter b_0 was changed and an unstable behaviour was obtained. The fact that STURE3 gives not as good steering quality as STURE is probably due to the biased parameter estimates obtained from the least squares method when the disturbances are coloured noise.

6. REFERENCES

Aspernäs, B., and Foisack, P. (1975), "Simulering av styrsystem för tankfartyg", Report RE-154, Department of Automatic Control, Lund Institute of Technology.

Aspernäs, B., and Källström, C. (1975), "Simulering av adaptiv fartygsstyrning med Kalmanfilter", Report 7517(C), Department of Automatic Control, Lund Institute of Technology.

Åström, K.J. (1974), "A Self-tuning Regulator for Nonminimum Phase Systems", Report 7411(C), Department of Automatic Control, Lund Institute of Technology.

Åström, K.J., and Wittenmark, B. (1974), "Analysis of a Self-tuning Regulator for Nonminimum Phase Systems", IFAC Symp. on Stochastic Control, Budapest, Hungary.

Elmqvist, H. (1975), "SIMNON - An Interactive Simulation Program for Nonlinear Systems, User's Manual", Report 7502, Department of Automatic Control, Lund Institute of Technology.

Källström, C. (1974), "The Sea Scout Experiments, October 1973", Report 7407(C), Department of Automatic Control, Lund Institute of Technology.

Källström, C. (1975), "The Sea Swift Experiments, October 1974", Report 7516(C), Department of Automatic Control, Lund Institute of Technology.

Norrbin, N.H. (1970), "Theory and Observations on the Use of a Mathematical Model for Ship Manoeuvring in Deep and Confined Waters", Proc. 8th Symp. on Naval Hydrodynamics, Pasadena, California, USA. Also available as Publ. No. 68, The Swedish State Shipbuilding Experimental Tank, Gothenburg, Sweden.

Wittenmark, B. (1973), "A Self-tuning Regulator", Report 7311, Department of Automatic Control, Lund Institute of Technology.

```
C         MAIN PROGRAM FOR SIMNON
C
C         AUTHOR  HILDING ELMQVIST  1973-12-18
C         REVISED C. KALLSTROM 1975-03-11.
C
C
      COMMON /PSCODE/ IDUM1(1600)
      COMMON /VARTAB/ IDUM2(1000),DUM2(500)
      COMMON /VALUES/ DUM3(250)
      COMMON /SYSINF/ IDUM4(151),DUM4(25)
      COMMON /EXTCOM/ IDUM5(2),DUM5(2)
      COMMON /ENTRYS/ IDUM6(3)
      COMMON /ENTRY/  IDUM7
      COMMON /PNTS/   IDUM8(177)
      COMMON /COMINF/ IDUM9(33),DUM9(41)
      COMMON /MACINF/ IDUM10(73),DUM10(107)
      COMMON /MESSS/  IDUM11
      COMMON /SIMN/   IDUM12(5)
      COMMON /PLT/    IDUM13(12),DUM13(16)
      COMMON /AX/     DUM14(4)
      COMMON /ERRWEI/ DUM15(51)
      COMMON /ALG/    IDUM16
      COMMON /MARKS/  IDUM17(2),DUM17(2)
      COMMON /USER/   IDUM18(4)
      COMMON /DESTIN/ IDUM19(2)
      COMMON /TIME/   DUM20
      COMMON /STATE/  DUM21(50)
      COMMON /USRCOM/ IDUM22(10)
      COMMON /CMPVAR/ IDUM23(12)
      COMMON /NXPNT/  IDUM24(50)
      COMMON /COND/   IDUM25(26)
      COMMON /LIMITS/ MPSC,IDM261,MVAL,IDM262
C
      COMMON/DATA/ IDD1(487)
C
      COMMON/STUR/ IDD2(512)
C
      COMMON/DAT1/ IDD3(46)
C
C
      MPSC=1600
      MVAL=250
C
      CALL ISIMN
C
      CALL ESIMN
C
      STOP
C
      END
```

```
      SUBROUTINE SYSTS
C
C       SERVES A LINK BETWEEN SIMNON AND USER SUBROUTINE.
C
C       AUTHOR  HILDING ELMQVIST  1974-02-05
C       REVISED, C.KALLSTROM 1975-02-04.
C
C       CHANGE THE CONTINUE STATEMENTS TO CALLS TO
C       YOUR SUBROUTINES.
C
C
      LOGICAL ISTOP,LUSER
      DIMENSION U(40)
      COMMON /DESTIN/ ISYST
      COMMON /USER/ ISTOP,IDUM,LUSER
      COMMON /USRCOM/ IERR1,ISYERR
      COMMON /SETVAR/ NIN,IDUM1,FILEI
     : ,NOUT,IDUM2,FTYPE,FILEO
     : ,NNOI1,IDUM3,NODD1,IDUM8
     : ,NNOI,IDUM4,NODD,IDUM5
     : ,NDEL,IDUM6,METH,IDUM7
      DATA MY /40/, MU /40/
C
C
      GO TO(1,2,3,4,5,6,7,8,9,10),ISYST
C
 1    CALL IFILE(NIN,FILEI,IERR,U,MY)
      GO TO 100
C
 2    CALL OFILE(NOUT,FTYPE,FILEO,IERR,U,MU)
      GO TO 100
C
 3    CALL AUT1
      GO TO 100
C
 4    CALL NOISE(NNOI,NODD,IERR)
      GO TO 100
C
 5    CALL NOIS1(NNOI1,NODD1,IERR)
      GO TO 100
C
 6    CONTINUE
      RETURN
C
 7    CONTINUE
      RETURN
C
 8    CONTINUE
      RETURN
C
 9    CONTINUE
      RETURN
C
 10   CONTINUE
      RETURN
C
C
 100  IF(IERR.EQ.0) RETURN
      IERR1=IERR
      ISYERR=ISYST
      ISTOP=.TRUE.
      LUSER=.TRUE.
      RETURN
```

```
C
      END
```

```
      SUBROUTINE AUT1
C
C     SYSTEM DEFINITION OF AN AUTOPILOT FOR SHIP.
C
C     AUTHOR, C.KALLSTROM 1974-07-12.
C     REVISED, C.KALLSTROM 1975-02-04.
C
C     SUBROUTINE REQUIRED
C        ATIN1
C             IDENT
C             INPUT
C             OUTPUT
C             OUTPUV
C             PAR
C             PARV
C             VAR
C             VARV
C             TSAMP
C        ATS1
C             STURE
C             STURB
C             STURE3
C                     RTLS1
C                     SCAPRO
C                     MOVE
C                     CORI
C                     NORM
C
C
      COMMON/DESTIN/ IDUM,IPART
C
      IF(IPART-4) 1,1,2
C
1     CALL ATIN1
      RETURN
C
2     CALL ATS1
      RETURN
C
      END
```

```
      SUBROUTINE ATIN1
C
C     AUTHOR, C.KALLSTROM 1975-02-04,
C     REVISED, C.KALLSTROM 1975-03-11.
C
C     SUBROUTINE REQUIRED
C             IDENT
C             INPUT
C             OUTPUT
C             OUTPUV
C             PAR
C             PARV
C             VAR
C             VARV
C             TSAMP
C
      COMMON/TIME/ T
C
      COMMON/DESTIN/ IDUM,IPART
C
      COMMON/DATA/ ICON,NA,NB,NC,K,NAB,NP,K1,NDAT,NDAT1,
      1   NU1,N1,NN,NA1,ITER,
      2   TETA(8),P(36),YSC(9),U(9),PP(8,8),S(8,8),TH(8),
      3   PSI,R,ROLD,RL,R0,Q2,S0,B00,CIND,DLIMD,DLIM,DELCO,
      4   DELOLD,FLL,SUM1,SUM2,SUM3,SUM4,SUM5,SUM6,EPSIM,
      5   VPSIM,EDELC,VDELC,EDELL,VDELL,VLOS1,VLOS2,VLOS3,
      6   AL1,AL3,AK1,AK2,AK3,PID,PSIM,TS,DT
C
      COMMON/STUR/ DAT(26),DUM(8),DUMMY(222)
C
      COMMON/DAT1/ AICON,ANA,ANB,ANC,AK,AITER,B0,TH0(8),P0(8)
C
      GO TO (100,200,300,400),IPART
C
C
100   CALL IDENT('DISCR','AUT1')
C
      RETURN
C
C
200   CALL INPUT(R,'R')
      CALL INPUT(PSI,'PSI')
      CALL INPUT(PSIM,'PSIM')
C
      CALL OUTPUT(DELCO,'DELCO')
      CALL OUTPUT(VLOS1,'VLOS1')
      CALL OUTPUT(VLOS2,'VLOS2')
      CALL OUTPUT(VLOS3,'VLOS3')
C
      CALL OUTPUV(TH,8,'TH')
C
      CALL PAR(DT,'DT')
      CALL PAR(AICON,'ICON')
      CALL PAR(ANA,'NA')
      CALL PAR(ANB,'NB')
      CALL PAR(ANC,'NC')
      CALL PAR(AK,'K')
      CALL PAR(AITER,'ITER')
      CALL PAR(RL,'RL')
      CALL PAR(Q2,'Q2')
      CALL PAR(S0,'S0')
      CALL PAR(R0,'R0')
      CALL PAR(DLIMD,'DLIMD')
```

```
      CALL PAR(DLIM,'DLIM')
      CALL PAR(AL1,'AL1')
      CALL PAR(AL3,'AL3')
      CALL PAR(B0,'B0')
      CALL PAR(AK1,'AK1')
      CALL PAR(AK2,'AK2')
      CALL PAR(AK3,'AK3')
C
      CALL PARV(TH0,8,'TH0')
      CALL PARV(P0,8,'P0')
C
      CALL VAR(CIND,'CIND')
      CALL VAR(FLL,'FLL')
      CALL VAR(EPSIM,'EPSIM')
      CALL VAR(VPSIM,'VPSIM')
      CALL VAR(EDELC,'EDELC')
      CALL VAR(VDELC,'VDELC')
      CALL VAR(EDELL,'EDELL')
      CALL VAR(VDELL,'VDELL')
      CALL VAR(PID,'PID')
C
      CALL VARV(P,36,'P')
C
      CALL TSAMP(TS,'TS')
C
      RETURN
C
C
300   DT=15.
      AICON=1.
      ANA=3.
      ANB=2.
      ANC=0.
      AK=5.
      AITER=30.
      RL=0.99
      Q2=0.
      S0=1.
      R0=1.
      DLIMD=20.
      DLIM=20.
      AL1=0.1
      AL3=0.1
      B0=-1.
      AK1=4.
      AK2=80.
      AK3=0.02
C
      TH0(1)=0.
      TH0(2)=0.
      TH0(3)=0.
      TH0(4)=0.
      TH0(5)=0.
      TH0(6)=0.
      TH0(7)=0.
      TH0(8)=0.
      P0(1)=1000.
      P0(2)=1000.
      P0(3)=1000.
      P0(4)=1000.
      P0(5)=1000.
      P0(6)=0.
      P0(7)=0.
      P0(8)=0.
```

```
C
      RETURN
C
C
400   TS=T
      ICON=AICON+0.1
      NA=ANA+0.1
      NB=ANB+0.1
      NC=ANC+0.1
      K=AK+0.1
      IITER=AIITER+0.1
C
      NAB=NA+NB
      NP=NAB+NC
      K1=K+1
      NU1=NA+K+2
      N1=NU1+K
      NA1=NA+1
      DO 402 I=1,8
402   TETA(I)=THO(I)
C
      IF(ICON-2) 404,406,420
404   B00=B0
      NDAT=NP+3*K+3
      NDAT1=NDAT+1
      NN=NAB+2*K+3
      GO TO 408
C
406   B00=1.
      NDAT=NP+3*K+2
      NDAT1=NDAT+1
      NN=NAB+2*K+2
C
408   DO 410 I=1,NA
410   TETA(I)=-TETA(I)
C
      DO 416 I=1,8
      DO 416 J=1,I
      L=I*(I-1)/2+J
      IF(I-J) 414,412,414
412   P(L)=P0(I)
      GO TO 416
414   P(L)=0.
416   CONTINUE
C
      DO 418 I=1,26
418   DAT(I)=0.
C
      GO TO 430
C
420   IF(ICON-4) 421,430,430
C
421   DO 426 I=1,8
      DO 426 J=1,8
      IF(I-J) 424,422,424
422   PP(I,I)=P0(I)
      GO TO 426
424   PP(I,J)=0.
426   CONTINUE
C
      DO 428 I=1,9
      U(I)=0.
428   YSC(I)=0.
C
```

```
430     ROLD=0.
        CIND=0.
        DELOLD=0.
        FLL=0.
        SUM1=0.
        SUM2=0.
        SUM3=0.
        SUM4=0.
        SUM5=0.
        SUM6=0.
        PID=0.
C
        RETURN
C
C
        END
```

```
      SUBROUTINE ATS1
C
C     AUTHOR, C.KALLSTROM 1975-02-04.
C     REVISED, C.KALLSTROM 1975-03-11.
C
C     SUBROUTINE REQUIRED
C             STURE
C             STURB
C             STURE3
C                     RTLS1
C                     SCAPRO
C                     MOVE
C                     CORI
C                     NORM
C
      COMMON/TIME/ T
C
      COMMON/DESTIN/ IDUM,IPART
C
      COMMON/DATA/ ICON,NA,NB,NC,K,NAB,NP,K1,NDAT,NDAT1,
     1   NU1,N1,NN,NA1,ITER,
     2   TETA(8),P(36),YSC(9),U(9),PP(8,8),S(8,8),TH(8),
     3   PSI,R,ROLD,RL,R0,Q2,S0,B00,CIND,DLIMD,DLIM,DELCO,
     4   DELOLD,FLL,SUM1,SUM2,SUM3,SUM4,SUM5,SUM6,EPSIM,
     5   VPSIM,EDELC,VDELC,EDELL,VDELL,VLOS1,VLOS2,VLOS3,
     6   AL1,AL3,AK1,AK2,AK3,PID,PSIM,TS,DT
C
      COMMON/STUR/ DAT(26),DUM(8),DUMMY(222)
C
      GO TO(999,999,999,999,500,600,700,800),IPART
C
500   IF(ICON-2) 502,502,512
C
502   DAT(1)=PSI
      IF(NC) 506,506,504
504   DAT(NN)=R-ROLD
     ROLD=R
C
506   IF(ICON-2) 508,510,510
C
508   CALL STURE(DAT,TETA,P,DUM,RL,NA,NAB,NP,K1,NDAT,NDAT1,NU1,N1)
C
      DELL=B00/(B00*B00+Q2)*DAT(NU1)
      GO TO 516
C
510   CALL STURB(DAT,TETA,P,DUM,RL,NA,NAB,NP,K1,NDAT,NDAT1,NU1)
C
      B1=TETA(NA1)
      DELL=B1*B1/(B1*B1+Q2)*DAT(NU1)
      GO TO 516
C
512   IF(ICON-4) 513,515,515
513   YSC(1)=PSI
      DO 514 I=1,8
      DO 514 J=1,8
      S(I,J)=0.
      IF(I .EQ. J) S(I,I)=S0
514   CONTINUE
C
      CALL STURE3(YSC,U,TETA,PP,S,RL,R0,Q2,NA,NB,K,ITER,IND,8,8)
C
      IF(IND .GT. 0) CIND=CIND+1.
      DELL=U(1)
```

```
      GO TO 516
C
515   PID=PID+PSI
      DELL=AK1*PSI+AK2*R+AK3*DT*PID-DELOLD
C
516   IF(ABS(DELL)-DLIMD) 520,520,518
518   DELL=ABS(DELL)/DELL*DLIMD
C
520   DELCO=DELL+DELOLD
      IF(ABS(DELCO)-DLIM) 524,524,522
522   DELCO=ABS(DELCO)/DELCO*DLIM
524   DELL=DELCO-DELOLD
C
      IF(ICON-3) 526,528,530
526   DAT(NU1)=DELL*B00
      GO TO 530
528   U(1)=DELL
530   DELOLD=DELCO
C
C     COMPUTE MEAN VALUES,VARIANCES AND LOSS FUNCTIONS.
C
      FLL=FLL+1.
      SUM1=SUM1+PSIM
      SUM2=SUM2+PSIM*PSIM
      SUM3=SUM3+DELCO
      SUM4=SUM4+DELCO*DELCO
      SUM5=SUM5+DELL
      SUM6=SUM6+DELL*DELL
      EPSIM=SUM1/FLL
      VPSIM=SUM2/FLL-EPSIM*EPSIM
      EDELC=SUM3/FLL
      VDELC=SUM4/FLL-EDELC*EDELC
      EDELL=SUM5/FLL
      VDELL=SUM6/FLL-EDELL*EDELL
      SL=VPSIM+EPSIM*EPSIM
      VLOS2=SL+AL1*VDELC
      VLOS1=VLOS2+AL1*EDELC*EDELC
      VLOS3=SL+AL3*(VDELL+EDELL*EDELL)
C
      IF(ICON-4) 538,550,550
C
538   DO 540 I=1,8
540   TH(I)=TETA(I)
      IF(ICON-2) 542,542,546
542   DO 544 I=1,NA
544   TH(I)=-TH(I)
      GO TO 550
C
546   DO 548 I=1,8
      DO 548 J=1,I
      L=I*(I-1)/2+J
      SL=(PP(I,J)+PP(J,I))*0.5
      PP(I,J)=SL
      PP(J,I)=SL
548   P(L)=SL
C
550   RETURN
C
600   TS=I+DT
      RETURN
C
700   RETURN
C
800   RETURN
```

```
C
999   RETURN
C
C
      END
```

```
      SUBROUTINE STURE(DAT,TH,P,DUM,RL,NA,NAB,NP,K1,NDAT,NDAT1,NU1,N1)

C     SELFTUNING REGULATOR BASED ON LEAST SQUARES IDENTIFICATION
C     AND MINIMUM VARIANCE CONTROL, ADMITS FEEDFORWARD AND
C     EXPLOITS SYMMETRY OF P.
C
C     AUTHOR, C.KALLSTROM 1974-07-04.
C     REVISED, C.KALLSTROM 1974-09-23.
C
C     THE ALGORITHM IS BASED ON THE MODEL
C
C     Y(T)+A(1)*Y(T-K-1)+...+A(NA)*Y(T-K-NA)=
C          B0*(U(T-K-1)+B(1)*U(T-K-2)+...+B(NB)*U(T-K-NB-1))+
C          C(1)*V(T-K-1)+C(2)*V(T-K-2)+...+C(NC)*V(T-K-NC)+EPS(T)
C
C     AT EACH STEP THE LEAST SQUARES ESTIMATES OF THE PARAMETERS
C     OF THE MODEL ARE COMPUTED. THE CONTROL VARIABLE U(T) TO
C     BE APPLIED AT TIME T IS THEN COMPUTED FROM
C
C     US(T)= AE(1)*Y(T)+...+AE(NA)*Y(T-NA+1)
C             -BE(1)*US(T-1)-...-BE(NB)*US(T-NB)
C             -CE(1)*V(T)-...-CE(NC)*V(T-NC+1)
C
C     WHERE AE,BE AND CE ARE THE PARAMETER ESTIMATES
C     AND US THE SCALED CONTROL SIGNAL I.E. US=B0*U
C
C     WHEN USING THE ALGORITHM THE PROCESS OUTPUT Y(T) AND THE
C     FEEDFORWARD SIGNAL V(T) ARE READ AT TIME T AND THE CONTROL
C     SIGNAL U(T) TO BE APPLIED AT TIME T IS THEN COMPUTED
C
C     DAT- VECTOR OF DIMENSION NA+NB+NC+3*K+3 CONTAINING
C         PROCESS OUTPUTS Y, SCALED CONTROL VARIABLES U
C         AND FEED FORWARD SIGNALS V ORGANIZED AS FOLLOWS
C     DAT(1)=Y(T)                             RETURNED AS Y(T)
C     DAT(2)=Y(T-1)                           RETURNED AS Y(T)
C     DAT(3)=Y(T-2)                           RETURNED AS Y(T-1)
C     .
C     DAT(NA+K+1)=Y(T-K-NA)                   RETURNED AS Y(T-K-NA+1)
C     DAT(NA+K+2)=US(T-1)                     RETURNED AS US(T)
C     DAT(NA+K+3)=US(T-2)                     RETURNED AS US(T-1)
C     .
C     DAT(NA+NB+2*K+2)=US(T-K-NB-1)           RETURNED AS US(T-K-NB)
C     DAT(NA+NB+2*K+3)=V(T)                   RETURNED AS US(T-K-NB-1)
C     DAT(NA+NB+2*K+4)=V(T-1)                 RETURNED AS V(T)
C     .
C     DAT(NA+NB+NC+3*K+3)=V(T-K-NC)           RETURNED AS V(T-K-NC+1)
C
C
C     TH- VECTOR OF DIMENSION NP=NA+NB+NC CONTAINING THE PARAMETER
C        ESTIMATES ORGANIZED AS FOLLOWS
C     TH(1)=-AE(1)
C     TH(2)=-AE(2)
C     .
C     TH(NA)=-AE(NA)
C     TH(NA+1)=BE(1)
C     TH(NA+2)=BE(2)
C     .
C     TH(NA+NB)=BE(NB)
C     TH(NA+NB+1)=CE(1)
C     TH(NA+NB+2)=CE(2)
C     .
C     TH(NA+NB+NC)=CE(NC)
C
```

```
C     P- COVARIANCE MATRIX STORED AS FOLLOWS
C     P(1)=P(1,1)
C     P(2)=P(2,1)
C     P(3)=P(2,2)
C     .
C     P(I*(I-1)/2+J)=P(I,J)
C     .
C     P(NP*(NP+1)/2)=P(NP,NP)
C
C
C     DUM- DUMMY VECTOR OF DIMENSION NP
C     RL- BASE OF EXPONENTIAL WEIGHTING FACTOR
C     NA- NUMBER OF A-PARAMETERS  (MAX 8, MIN 0)
C        NB- NUMBER OF B-PARAMETERS  (MAX 8, MIN 0)
C        NC- NUMBER OF C-PARAMETERS  (MAX 8, MIN 0)
C        K -NUMBER OF TIME DELAYS IN THE MODEL  (MAX (32-NA-NB-NC)/3,
C             MIN 0)
C     NAB- NA+NB
C     NP- NA+NB+NC (MAX 8, MIN 1)
C     K1- K+1
C     NDAT- NP+3*K+3
C     NDAT1- NDAT+1
C     NU1- NA+K+2
C     N1- NU1+K
C
C     SUBROUTINE REQUIRED
C          NONE
C
      DIMENSION DAT(35),TH(8),P(36),DUM(8)
C
      RES=DAT(1)-DAT(N1)
      DENOM=1.
      DO 12 I=1,NP
      R=0.
      DO 10 J=1,NP
      L=I*(I-1)/2+J
      IF (J.GT.I) L=J*(J-1)/2+I
      M=K1+J
      IF (J.GT.NA) M=M+K1
      IF (J.GT.NAB) M=M+K1
   10 R=R+P(L)*DAT(M)
      DUM(I)=R
      M=K1+I
      IF (I.GT.NA) M=M+K1
      IF (I.GT.NAB) M=M+K1
      DENOM=DENOM+R*DAT(M)
   12 RES=RES-DAT(M)*TH(I)
C
      DO 20 I=1,NP
      R=DUM(I)/DENOM
      TH(I)=TH(I)+R*RES
      DO 20 J=1,I
      L=I*(I-1)/2+J
   20 P(L)=(P(L)-R*DUM(J))/RL
C
      R=0.
      DO 30 I=1,NP
      L=I
      IF (I.GT.NA) L=L+K1
      IF (I.GT.NAB) L=L+K1
   30 R=R-TH(I)*DAT(L)
C
      DO 32 I=2,NDAT
```

```
      L=NDAT1-I
32    DAT(L+1)=DAT(L)
      DAT(NU1)=R
C
      RETURN
      END
```

```
      SUBROUTINE STURB(DAT,TH,P,DUM,RL,NA,NAB,NP,K1,NDAT,NDAT1,NU1)
C
C     SELFTUNING REGULATOR BASED ON LEAST SQUARES IDENTIFICATION
C     AND MINIMUM VARIANCE CONTROL, ADMITS FEEDFORWARD AND
C     EXPLOITS SYMMETRY OF P,
C
C     AUTHOR, C.KALLSTROM 1974-07-04.
C     REVISED, C.KALLSTROM 1975-02-04.
C
C     THE ALGORITHM IS BASED ON THE MODEL
C
C     Y(T)+A(1)*Y(T-K-1)+...+A(NA)*Y(T-K-NA)=
C          B(1)*U(T-K-1)*B(2)*U(T-K-2)+...+B(NB)*U(T-K-NB)+
C          C(1)*V(T-K-1)+C(2)*V(T-K-2)+...+C(NC)*V(T-K-NC)+EPS(T)
C
C     AT EACH STEP THE LEAST SQUARES ESTIMATES OF THE PARAMETERS
C     OF THE MODEL ARE COMPUTED. THE CONTROL VARIABLE U(T) TO
C     BE APPLIED AT TIME T IS THEN COMPUTED FROM
C
C     U(T)=( AE(1)*Y(T)+...+AE(NA)*Y(T-NA+1)
C             -BE(2)*U(T-1)-...-BE(NB)*U(T-NB+1)
C             -CE(1)*V(T)-...-CE(NC)*V(T-NC+1)  )/BE(1)
C
C     WHERE AE,BE AND CE ARE THE PARAMETER ESTIMATES.
C
C     WHEN USING THE ALGORITHM THE PROCESS OUTPUT Y(T) AND THE
C     FEEDFORWARD SIGNAL V(T) ARE READ AT TIME T AND THE CONTROL
C     SIGNAL U(T) TO BE APPLIED AT TIME T IS THEN COMPUTED
C
C     DAT- VECTOR OF DIMENSION NA+NB+NC+3*K+2 CONTAINING
C          PROCESS OUTPUTS Y, SCALED CONTROL VARIABLES U
C          AND FEED FORWARD SIGNALS V ORGANIZED AS FOLLOWS
C     DAT(1)=Y(T)                              RETURNED AS Y(T)
C     DAT(2)=Y(T-1)                            RETURNED AS Y(T)
C     DAT(3)=Y(T-2)                            RETURNED AS Y(T-1)
C     .
C     DAT(NA+K+1)=Y(T-K-NA)                    RETURNED AS Y(T-K-NA+1)
C     DAT(NA+K+2)=U(T-1)                      RETURNED AS U(T)
C     DAT(NA+K+3)=U(T-2)                      RETURNED AS U(T-1)
C     .
C     DAT(NA+NB+2*K+1)=U(T-K-NB)            RETURNED AS U(T-K-NB+1)
C     DAT(NA+NB+2*K+2)=V(T)                    RETURNED AS US(T-K-NB)
C     DAT(NA+NB+2*K+3)=V(T-1)                  RETURNED AS V(T)
C     .
C     DAT(NA+NB+NC+3*K+2)=V(T-K-NC)            RETURNED AS V(T-K-NC+1)
C
C
C     TH- VECTOR OF DIMENSION NP=NA+NB+NC CONTAINING THE PARAMETER
C          ESTIMATES ORGANIZED AS FOLLOWS
C     TH(1)=-AE(1)
C     TH(2)=-AE(2)
C     .
C     TH(NA)=-AE(NA)
C     TH(NA+1)=BE(1)
C     TH(NA+2)=BE(2)
C     .
C     TH(NA+NB)=BE(NB)
C     TH(NA+NB+1)=CE(1)
C     TH(NA+NB+2)=CE(2)
C     .
C     TH(NA+NB+NC)=CE(NC)
C
C
```

```
C     P- COVARIANCE MATRIX STORED AS FOLLOWS
C     P(1)=P(1,1)
C     P(2)=P(2,1)
C     P(3)=P(2,2)
C     .
C     P(I*(I-1)/2+J)=P(I,J)
C     .
C     P(NP*(NP+1)/2)=P(NP,NP)
C
C
C     DUM- DUMMY VECTOR OF DIMENSION NP
C     RL- BASE OF EXPONENTIAL WEIGHTING FACTOR
C     NA- NUMBER OF A-PARAMETERS
C        NB- NUMBER OF B-PARAMETERS
C        NC- NUMBER OF C-PARAMETERS
C        K -NUMBER OF TIME DELAYS IN THE MODEL
C     NAB- NA+NB
C     NP- NA+NB+NC
C     K1- K+1
C     NDAT- NP+3*K+2
C     NDAT1- NDAT+1
C     NU1- NA+K+2
C
C     SUBROUTINE REQUIRED
C          NONE
C
      DIMENSION DAT(1),TH(1),P(1),DUM(1)
C
      RES=DAT(1)
      DENOM=1.
      DO 12 I=1,NP
      R=0.
      DO 10 J=1,NP
      L=I*(I-1)/2+J
      IF (J.GT.I) L=J*(J-1)/2+I
      M=K1+J
      IF (J.GT.NA) M=M+K
      IF (J.GT.NAB) M=M+K1
10    R=R+P(L)*DAT(M)
      DUM(I)=R
      M=K1+I
      IF (I.GT.NA) M=M+K
      IF (I.GT.NAB) M=M+K1
      DENOM=DENOM+R*DAT(M)
12    RES=RES-DAT(M)*TH(I)
C
      DO 20 I=1,NP
      R=DUM(I)/DENOM
      TH(I)=TH(I)+R*RES
      DO 20 J=1,I
      L=I*(I-1)/2+J
20    P(L)=(P(L)-R*DUM(J))/RL
C
      R=0.
      NP1=NP-1
      DO 30 I=1,NP1
      M=I
      IF(I .GT. NA) M=I+1
      L=I
      IF (I.GT.NA) L=L+K1
      IF (I.GE.NAB) L=L+K1
30    R=R-TH(M)*DAT(L)
C
      DO 32 I=2,NDAT
```

```
      L=NDAT1-I
   32 DAT(L+1)=DAT(L)
      NA1=NA+1
      DAT(NU1)=R/TH(NA1)
C
      RETURN
      END
```

```
      SUBROUTINE STURE3(YSC,U,TETA,P,S,RL,RO,Q2,NA,NB,K,ITER,
      1   IND,IP,IS)
C
C     SELFTUNING REGULATOR BASED ON LEAST SQUARES IDENTIFICATION
C     AND MINIMUM VARIANCE CONTROL
C     THE ALGORITHM IS BASED ON THE MODEL
C
C     Y(T)+A(1)*Y(T-1)+...+A(NA)*Y(T-NA)=
C     B(1)*U(T-K-1)+...+B(NB)*U(T-K-NB)      (*)
C
C     AT EACH STEP THE LEAST SQUARES ESTIMATES OF THE MODEL
C     PARAMETERS ARE COMPUTED. THE PROCESS INPUT U(T) TO BE
C     APPLIED TO THE PROCESS AT TIME T IS THEN COMPUTED
C     FROM THE SOLUTION OF THE RICCATI EQUATION WHICH MINIMIZES
C     SUM Y(T)**2 UNDER THE CONSTRAINT THAT ALL POLES OF
C     THE CLOSED LOOP SYSTEM ARE WITHIN THE CIRCLE WITH RADIUS RO
C
C     WHEN APPLYING THE ALGORITHM THE PROCESS OUTPUT IS THUS READ
C     AT TIME T. THE PROCESS INPUT U(T) TO BE APPLIED AT TIME T
C     IS THEN IMMEDIATELY COMPUTED. I.E. THERE IS NO TIME
C     DELAY IN THE REGULATOR.
C
C
C     AUTHOR B WITTENMARK 72-08-02  REV 05-74 BB/BE
C     REVISED, C.KALLSTROM 1975-02-04.
C
C
C     YSC- VECTOR OF SCALED PROCESS OUTPUTS OF DIMENSION NA+1
C             AND ORGANIZED AS FOLLOWS
C
C     YSC(1)=YSC(T)           RETURNED AS YSC(T)
C     YSC(2)=YSC(T-1)         RETURNED AS YSC(T)
C     ...                     ...
C     YSC(NA+1)=YSC(T-NA)     RETURNED AS YSC(T-NA+1)
C
C
C     U- VECTOR OF CONTROL VARIABLES OF DIMENSION NB+K+1
C             AND ORGANIZED AS FOLLOWS
C
C     U(1)=U(T-1)             RETURNED AS U(T)
C     U(2)=U(T-2)             RETURNED AS U(T-1)
C     ...                     ...
C     U(NB+K+1)=U(T-NB-K-1)   RETURNED AS U(T-NB-K)
C
C
C     TETA- VECTOR OF ESTIMATED PARAMETERS OF DIMENSION NA+NB
C             AND ORGANIZED AS FOLLOWS
C
C     TETA(1)=AE(1)
C     TETA(2)=AE(2)
C     ...
C     TETA(NA)=AE(NA)
C     TETA(NA+1)=BE(1)
C     ...
C     TETA(NA+NB)=BE(NB)
C
C
C     P- COVARIANCE MATRIX OF PARAMETER ESTIMATES OF
C             ORDER (NA+NB)*(NA+NB).
C     S- SOLUTION TO THE RICCATI EQUATION WHICH GIVES THE CONTROL LAW.
C             THE INITIAL VALUE OF THE MATRIX SET IN THE CALLING
C             PROGRAM MUST BE POSITIVE DEFINITE. THE MATRIX S
C     IS OF ORDER N*N (N=MAX(NA,NB+K)),
```

```
C       RL- THE BASE OF THE EXPONENTIAL WEIGHTING FUNCTION.
C       R0- CIRCLE RADIUS (SEE ABOVE).
C       Q2- CRITERIA SCALAR FOR U.
C       NA- NUMBER OF A-PARAMETERS  SEE (*).   MAX(NA)=8
C       NB- NUMBER OF B-PARAMETERS  SEE (*).   MAX(NB+K)=8
C       K- NUMBER OF TIME DELAYS    SEE (*).   MAX(NB+K)=8
C       ITER- MAXIMUM NUMBER OF STEPS OF THE RICCATI EQUATION.
C       IND- RETURNED 1 IF THE RICCATI EQUATION HAS NOT CONVERGED
C                AFTER ITER STEPS.
C                RETURNED -N IF CONVERGENCE AFTER N STEPS.
C       IP- DIMENSION PARAMETER OF MATRIX P.   MAX(IP)=32
C       IS- DIMENSION PARAMETER OF MATRIC S.   MAX(IS)=8
C
C      SUBROUTINES REQUIRED:
C          RTLS1
C          SCAPRO
C          MOVE
C          CORI
C          NORM
C
      DIMENSION YSC(1),U(1),TETA(1),P(1,1),S(1,1)
C
      COMMON/STUR/DUM(16),AS(8),BS(8),US(8),YS(8),X(8),AL(8),
     1             R(8,8),FI(64),DUM8(64)
C
      NP=NB+K
      IF (NA-NP) 2,2,4
2     N=NP
      GO TO 6
4     N=NA
6     NM1=N-1
      NAM1=NA-1
      NAP1=NA+1
      NBM1=NB-1
      NP1=N+1
      NP=NA+NB
C
C     SET FIXED PARAMETERS
C
      EPS=1.E-5
      NLOP=ITER
      IR=8
C
C     ORGANIZE DATA FOR IDENTIFICATION ROUTINE
C
      YIN=YSC(1)
      DO 10 I=1,NA
10    FI(I)=-YSC(I+1)
      IF(NB) 12,12,11
11    NS=2*NB
      CALL MOVE(U(K+1),FI(NA+1),NS)
C
12    CALL RTLS1(TETA,P,FI,YIN,NP,IP,RL,RES,DENOM)
C
C     SET PARAMETERS OF STATE MODEL
C
      CALL MOVE (TETA(1),AS(1),NA+NA)
      IF (N-NA-1) 20,21,22
22    AS(NA+1)=0.0
      NS=N-NA-1
      CALL MOVE (AS(NA+1),AS(NA+2),NS+NS)
      GO TO 20
21    AS(NA+1)=0.0
20    IF (K-1) 24,25,26
```

```
 26   BS(1)=0.0
      NS=K-1
      CALL MOVE (BS(1),BS(2),NS+NS)
      GO TO 24
 25   BS(1)=0.0
 24   CALL MOVE (TETA(NA+1),BS(K+1),NB+NB)
      N1=NB+K+1
      IF (N-N1) 28,29,30
 30   BS(N1)=0.0
      NS=N-N1
      CALL MOVE(BS(N1),BS(N1+1),NS+NS)
      GO TO 28
 29   BS(N1)=0.0
C
C     SCALE SYSTEM PARAMETERS
C
 28   SF=1.
      DO 32 I=1,N
      SF=SF*R0
      AS(I)=AS(I)/SF
 32   BS(I)=BS(I)/SF
C
C     SCALE U AND Y
C
      SF=1.
      DO 34 I=1,NA
      SF=SF*R0
 34   YS(I)=YSC(I)*SF
      SF=1.
      NS=NB+K
      DO 36 I=1,NS
      SF=SF*R0
 36   US(I)=U(I)*SF
C
C     COMPUTE STATE VARIABLES
C
      X(1)=YS(1)
      IF (NA-NB-K) 40,41,42
 40   IF (NA-2) 70,71,71
 71   DO 43 I=2,NA
      NS=NA-I+1
      NSB=N-I+1
 43   X(I)=-SCAPRO(AS(I),1,YS(2),1,NS)+SCAPRO(BS(I),1,US(1),1,NSB)
 70   DO 44 I=NAP1,N
      NS=N-I+1
 44   X(I)=SCAPRO(BS(I),1,US(1),1,NS)
      GO TO 49
 41   DO 45 I=2,N
      NS=N-I+1
 45   X(I)=-SCAPRO(AS(I),1,YS(2),1,NS)+SCAPRO(BS(I),1,US(1),1,NS)
      GO TO 49
 42   N1=NB+K
      DO 46 I=2,N1
      NSA=N-I+1
      NS=N1-I+1
 46   X(I)=-SCAPRO(AS(I),1,YS(2),1,NSA)+SCAPRO(BS(I),1,US(1),1,NS)
      N1=N1+1
      DO 47 I=N1,N
      NS=N-N1+1
 47   X(I)=-SCAPRO(AS(I),1,YS(2),1,NS)
 49   CONTINUE
C
C     COMPUTE CONTROL LAW
C
```

```
      NLOOP=0
 60   NLOOP=NLOOP+1
      DO 62 I=1,N
      DO 62 J=1,N
 62   R(I,J)=S(I,J)
C
      CALL CORI(AS,BS,Q2,S,AL,N,R3,IS)
C
      DO 64 I=1,N
      DO 64 J=1,N
 64   R(I,J)=R(I,J)-S(I,J)
C
C     TEST IF ITERATION HAS CONVERGED
C
      CALL NORM(R,N,IR,RNORM)
      CALL NORM(S,N,IS,SNORM)
      IF (RNORM-EPS*SNORM) 66,66,65
 66   IND=-NLOOP
      GO TO 68
 65   IF (NLOOP-NLOP) 60,67,67
 67   IND=1
 68   CONTINUE
C
C
C     REORGANIZE DATA FOR NEXT STEP
C
      NS=2*NA
      CALL MOVE(YSC(1),YSC(2),-NS)
      NS=2*(NB+K)
      CALL MOVE(U(1),U(2),-NS)
C
C     COMPUTE CONTROL SIGNAL
C
      U(1)=-SCAPRO(AL(1),1,X(1),1,N)
C
C     THE CELL YSC(1) IS NOW READY TO RECEIVE THE NEXT SCALED
C     PROCESS OUTPUT Y(I+1)-YREF AND THE CELL U(1) NOW CONTAINS THE
C     CONTROL SIGNAL U(I) TO BE USED AT TIME T
C
      RETURN
      END
```

```
      SUBROUTINE RTLS1(T,P,FI,Y,N,NMAX,RL,RES,DENOM)
C
C
C     REAL-TIME IDENTIFICATION USING MODIFIED LEAST SQUARES METHOD.
C     SUBROUTINE UPDATES ARGUMENTS T, P, RES AND DENOM.
C     REFERENCE, REPORT 6810 DEC.-68.
C     AUTHOR, JOHAN WIESLANDER  26/06  -69.
C     PDP-VERSION, JOHAN WIESLANDER 1970-12-01
C       REVISED,C.KALLSTROM 1975-02-04.
C
C     T IS VECTOR CONTAINING ESTIMATED SYSTEM PARAMETERS.
C     P CORRESPONDS TO THE INVERSE OF THE INFORMATION MATRIX.
C     FI CONTAINS OLD INPUT AND OUTPUT VALUES. FI SHOULD BE UPDATED
C     OUTSIDE THE ROUTINE.
C     Y IS LAST OUTPUT FROM THE SYSTEM.
C     N IS NUMBER OF PARAMETERS. MAX=32.
C     NMAX IS DIMENSION PARAMETER OF P. MAX=32.
C     RL IS THE BASE OF THE EXPONENTIAL WEIGHTING FUNCTION.
C     RES IS THE RESIDUAL.
C     DENOM IS THE FACTOR 1+FI*P*FIT.
C
C
C     SUBROUTINES REQUIRED
C             NONE
C
      DIMENSION T(1),P(1,1),FI(1),RK(32),S(32),DUMY(192)
      COMMON /STUR/DUMY,RK,S
C
      DO 5 I=1,N
5     S(I)=SCAPRO(P(I,1),NMAX,FI(1),1,N)
      DENOM=1.+SCAPRO(FI(1),1,S(1),1,N)
      RES=Y-SCAPRO(FI(1),1,T(1),1,N)
      DO 20 I=1,N
      RK(I)=S(I)/DENOM
      T(I)=T(I)+RK(I)*RES
      DO 20 J=1,I
      P(I,J)=(P(I,J)-RK(I)*S(J))/RL
20    P(J,I)=P(I,J)
      RETURN
      END
```

```
      SUBROUTINE CORI(A,B,Q2,S,AL,N,R3,IS)
C     THIS SUBROUTINE ITERATES THE RICCATI EQUATION
C             S=AT*S*(A-B*L)+Q1      (*)
C             L=BT*S*A/(Q2+BT*S*B)   (**)
C     IN THE SPECIAL CASE WHEN
C             A IS A COMPANION MATRIX
C             Q1=DIAG(1,0,...,0)
C             B IS A VECTOR
C
C     AUTHOR K,J. ASTROM 72-01-03
C     REVISED, C.KALLSTROM 1975-02-04.
C
C             A   - VECTOR CONTAINING THE FIRST COLUMN
C                   OF THE MATRIX A IN (*) I,E. A(I,1)=-A(I)
C             B   - VECTOR B IN (*) AND (**)
C             Q2   - SCALAR Q2 IN (**)
C             S    - SOLUTION OF RICCATI EQUATION
C             AL   - VECTOR AL IN (**)
C             N    - ACTUAL ORDER OF SYSTEM
C             IS   - DIMENSION PARAMETER OF MATRIX S
C             R3   - DENOMINATOR (Q2+BT*S*B)
C
      DIMENSION A(1),B(1),AL(1),S(1,1)
      COMMON /STUR/DUM(128),S1(8,8),S2(8,8)
C
      DO 10 I=1,N
10    S1(I,1)=SCAPRO(B(1),1,S(1,I),1,N)
      R3=SCAPRO(B(1),1,S1(1,1),1,N)+Q2
C
C     R3 NOW CONTAINS BT*S*B+Q2
C
      AL(1)=-SCAPRO(S1(1,1),1,A(1),1,N)/R3
      DO 14 I=2,N
      I1=I-1
14    AL(I)=S1(I1,1)/R3
C
C     COMPUTATION OF L=BT*S*A/(Q2+BT*S*B) COMPLETE
C     RESULT STORED INVECTOR AL
C
      R=AL(1)
      DO 20 I=1,N
20    S1(I,1)=-A(I)-B(I)*R
      DO 22 J=2,N
      R=AL(J)
      DO 22 I=1,N
      R1=0.
      IF (I+1-J) 22,23,22
23    R1=1.
22    S1(I,J)=R1-B(I)*R
C
C     S1 NOW CONTAINS A-B*L
C
      DO 24 I=1,N
24    S2(I,1)=-SCAPRO(A(1),1,S(I,1),IS,N)
      NM1=N-1
      DO 26 J=1,NM1
26    CALL MOVE(S(1,J),S2(1,J+1),N+N)
C
C     S2 NOW CONTAINS S*A
C
      DO 30 I=1,N
      DO 30 J=1,N
30    S(I,J)=SCAPRO(S2(1,I),1,S1(1,J),1,N)
```

```
S(1,1)=S(1,1)+1.
RETURN
END
```

```
      SUBROUTINE NOIS1(NNOI,NODD,IERR)
C
C        DISCRETE SYSTEM, TO BE INCLUDED IN SIMNON,
C        THAT GENERATES NOISE WITH GIVEN COVARIANCE.
C
C        NNOI   - NUMBER OF OUTPUTS
C        NODD   - INITIAL STATE FOR GENERATOR
C                 (ODD, POSITIVE INTEGER)
C        IERR   - ERROR INDICATOR
C                 1: BAD NUMBER OF NOISE COMPONENTS
C                 2: BAD COVARIANCE MATRIX
C
C        OUTPUTS:
C        E1,E2,...
C
C        PARAMETERS:
C        R11,R12,...,R1N
C            R22      R2N
C                     .
C                     RNN
C
C               - COVARIANCE MATRIX
C        DT1    - TIME FOR FIRST SAMPLING RELATIVE TO THE START TIME
C        DT     - DISTANCE BETWEEN SAMPLINGS
C        SAME   - SWITCH TO RESET STATE OF GENERATOR
C                 >0.5: RESET STATE
C
C        NOTE, NNOI MUST BE LESS THAN 6
C
C        AUTHOR  HILDING ELMQVIST  1974-05-03
C        REVISED, C.KALLSTROM 1974-09-26.
C
C        SUBROUTINE REQUIRED
C                 IDENT,...
C                 MNODI3
C
C
      DIMENSION R(5,5),XN(10)
      COMMON /TIME/ T
      COMMON /DESTIN/ IDUM,IPART
      COMMON /USER/ ISTOP
      DATA MAX /5/
C
C
      GO TO(1,2,3,4,5,6,7,8),IPART
C
 1    CALL IDENT('DISCR','NOIS1')
      RETURN
C
 2    IF(NNOI.LE.0 .OR. NNOI.GT.MAX) GO TO 810
      CALL OUTPUV(XN,NNOI,'E')
C
      DO 20 I=1,NNOI
      CALL CRENAM('R',I,VNAM1)
      DO 20 J=I,NNOI
      CALL CRENAM(VNAM1,J,VNAM2)
 20   CALL PAR(R(I,J),VNAM2)
C
      CALL TSAMP(TS,'TS')
      CALL PAR(DT1,'DT1')
      CALL PAR(DT,'DT')
      CALL PAR(SAME,'SAME')
      RETURN
```

```
C
 3    N=NNOI
      NODD1=NODD
      NODOLD=NODD1
      DO 30 I=1,N
      I1=I+1
      DO 30 J=I1,N
 30   R(I,J)=0.
      DO 32 I=1,N
 32   R(I,I)=1.
C
      DT1=0.
      DT=1.
      SAME=0.
      RETURN
C
 4    ILOG=0
C
C       FILL THE COVARIANCE MATRIX
      N1=N-1
      DO 40 I=1,N1
      I1=I+1
      DO 40 J=I1,N
 40   R(J,I)=R(I,J)
C
      TS=T+DT1
      IF(SAME.GE.0.5) NODD1=NODOLD
      NODOLD=NODD1
      RETURN
C
C       COMPUTE OUTPUTS
 5    CALL MNODI3(XN,R,N,5,NODD1,ILOG,IND)
      IF(IND.EQ.1) GO TO 820
      TS=T+DT
      RETURN
C
 6    RETURN
C
 7    RETURN
C
 8    RETURN
C
C
C       BAD NUMBER OF NOISE COMPONENTS
 810  IERR=1
      RETURN
C
C       BAD COVARIANCE MATRIX
 820  IERR=2
      RETURN
C
      END
```

```
      SUBROUTINE MNODI3(E,R,N,IA,NODD,ILOG,IND)
C
C     GENERATES GAUSSIAN RANDOM VECTORS E WITH COVARIANCE MATRIX R.
C     SUITED FOR REPEATED USE.
C     AUTHOR, C.KALLSTROM 1971-07-20.
C     REVISED  HILDING ELMQVIST  1974-05-15
C
C     E- GAUSSIAN RANDOM VECTOR OF DIMENSION N.
C     R- COVARIANCE MATRIX (SYMMETRIC) OF ORDER N*N, NOT DESTROYED.
C     N- DIMENSION OF THE VECTOR E (MAX 10, MIN 1).
C     IA- DIMENSION PARAMETER OF R.
C     NODD- BY FIRST CALL MNODI, NODD MUST EQUAL AN ODD, POSITIVE
C        INTEGER (SAY 19). NODD IS RETURNED CONTAINING A NEW ODD,
C        POSITIVE INTEGER WHICH IS USED BY REPEATED CALLS.
C     ILOG- INPUT PARAMETER:
C        ILOG .EQ. 0: THE COVARIANCE MATRIX R IN THE CALL IS USED.
C        ILOG IS RETURNED CONTAINING 1.
C        ILOG .NE. 0: THE MATRIX R IN THE PREVIOUS CALL OF MNODI IS USED
C        ILOG IS NOT CHANGED.
C     IND- OUTPUT PARAMETER:
C        MNODI CALLED WITH ILOG .EQ. 0:
C        IND=1 WHEN THE DECOMPOSITION OF R IN SUBROUTINE DESYM
C        HAS FAILED.
C        IND=0 OTHERWISE.
C        MNODI CALLED WITH ILOG .NE. 0:
C        THE VALUE OF IND IS NOT CHANGED.
C
C     ILOG MUST BE EQUAL TO ZERO BY THE FIRST CALL OF MNODI WITH A NEW
C     COVARIANCE MATRIX R. IF SEVERAL VECTORS WITH THE SAME
C     COVARIANCE MATRIX SHOULD BE GENERATED, THE COMPUTATION SPEEDS UP
C     IF THE FOLLOWING CALLS OF MNODI IS MADE WITH ILOG DIFFERENT
C     FROM ZERO. (THE VALUE OF ILOG IS CHANGED IN MNODI AFTER
C     THE FIRST CALL).
C
C     SUBROUTINE REQUIRED
C            NORM
C            DESYM
C            MCNODI
C
      DIMENSION E(1),R(1,1),S(5,5),E1(10)
C
      IS=5
      IF(ILOG) 20,10,20
10    CALL NORM(R,N,IA,SS)
      EPS=1.E-07*SS
      CALL DESYM (R,S,N,EPS,IRANK,IS)
      IND=0
      IF(IRANK .EQ. (-1)) IND=1
      ILOG=1
C
20    DO 30 I=1,N
      CALL MCNODI(NODD,GAUSS)
30    E1(I)=GAUSS
C
      DO 50 I=1,N
50    E(I)=SCAPRO(S(I,1),IS,E1(1),1,I)
C
      RETURN
      END
```

```
C       SETNAM
      BLOCK DATA
C
C
      DOUBLE INTEGER UNAMES(11)
      COMMON /SETNAM/ NVAR,UNAMES
C
      DATA NVAR /11/
      DATA UNAMES /'NIN','FILEI'
    : ,'NOUT','FTYPE','FILEO'
    : ,'NNOI1','NODD1'
    : ,'NNOI','NODD'
    : ,'NDEL','METH'/
C
      END
```

```
C         SETVAR
        BLOCK DATA
C
        COMMON /SETVAR/ NIN,IDUM1,FILEI
     :  ,NOUT,IDUM2,FTYPE,FILEO
     :  ,NNOI1,IDUM3,NODD1,IDUM8
     :  ,NNOI,IDUM4,NODD,IDUM5
     :  ,NDEL,IDUM6,METH,IDUM7
C
        DATA NIN /1/, FILEI /4HDATA/
        DATA NOUT /1/, FTYPE /4HCONT/, FILEO /4HDATA/
        DATA NNOI1/1/,NODD1/19/
        DATA NNOI /1/, NODD /17/
        DATA NDEL /1/, METH /1/
C
        END
```

```
      SUBROUTINE DUMCK
C
C     DUMMY SUBROUTINE.
C
      DIMENSION A(64)
C
      RETURN
      END
```

```
CONNECTING SYSTEM TSHP1

TIME T

XI[LPFI1]=E1[NOIS1]
XI[LPFI2]=E2[NOIS1]
W1[SHIP1]=XO[LPFI1]
W2[SHIP1]=XO[LPFI2]
R[AUT1]=RM[SHIP1]+E3[NOISE]
PSI[AUT1]=PSIM[SHIP1]+E4[NOISE]
PSIM[AUT1]=PSIM[SHIP1]
DELS[SHIP1]=DELCO[AUT1]
C1[OFILE]=T/60.
C2[OFILE]=DELCO[AUT1]
C3[OFILE]=DELM[SHIP1]
C4[OFILE]=V1[SHIP1]
C5[OFILE]=V1[SHIP1]+E1[NOISE]
C6[OFILE]=V2[SHIP1]
C7[OFILE]=V2[SHIP1]+E2[NOISE]
C8[OFILE]=VM[SHIP1]
C9[OFILE]=RM[SHIP1]
C10[OFILE]=R[AUT1]
C11[OFILE]=PSIM[SHIP1]
C12[OFILE]=PSI[AUT1]
C13[OFILE]=XO[LPFI1]
C14[OFILE]=XO[LPFI2]
C15[OFILE]=VLOS1[AUT1]
C16[OFILE]=VLOS2[AUT1]
C17[OFILE]=VLOS3[AUT1]
C18[OFILE]=TH1[AUT1]
C19[OFILE]=TH2[AUT1]
C20[OFILE]=TH3[AUT1]
C21[OFILE]=TH4[AUT1]
C22[OFILE]=TH5[AUT1]
C23[OFILE]=TH6[AUT1]
C24[OFILE]=TH7[AUT1]
C25[OFILE]=TH8[AUT1]

RC=SC1*RM[SHIP1]+PC1
PSIC=SC2*PSIM[SHIP1]+PC2
DELCC=SC3*DELCO[AUT1]+PC3

SC1:30.
PC1:4.5
SC2:1.5
PC2:0.
SC3:0.15
PC3:-4.5

END
```

```
CONTINUOUS SYSTEM SHIP1

STATE DEL V R PSI

" DEL = RUDDER ANGLE [RAD]
" V   = TRANSVERSAL VELOCITY [M]/[S]
" R   = YAW RATE [RAD]/([S]*100)
" PSI = COURSE [RAD]

DER DDEL DV DR DPSI

INPUT DELS W1 W2

" DELS   = RUDDER ANGLE [DEGREES]
" W1, W2 = NOISE

OUTPUT DELM V1 V2 VM RM PSIM

" DELM = RUDDER ANGLE [DEGR]
" V1   = TRANSVERSAL VELOCITY OF BOW [KNOTS]
" V2   = TRANSVERSAL VELOCITY OF STERN [KNOTS]
" VM   = TRANSVERSAL VELOCITY [KNOTS]
" RM   = YAW RATE [DEGREES]/[S]
" PSIM = COURSE [DEGREES]

INITIAL

DEL:0
V:0
R:0
PSI:0

F1=(20.-TT)/9.5
F2=(TI-10.5)/9.5
CDV=YVD10*F1+YVD20*F2
YUVP=YUV10*F1+YUV20*F2
YUVM=YUR10*F1+YUR20*F2
YVVP=YVV10*F1+YVV20*F2
IZN=NRD10*F1+NRD20*F2
NUVP=NUV10*F1+NUV20*F2
NURM=NUR10*F1+NUR20*F2
NVRP=NVR10*F1+NVR20*F2
TS1=1/TS
TS2=TS1/CRG
DL1=DLIM/CRG
G1S=G1/L
G3S=G3*L
A11=CDV
A12=MXY*L
A21=MXN/L
A22=IZN
DET1=1/(A11*A22-A12*A21)
CUV=YUVP/L
CUR=YUVM
CV2=YVVP/L
CYC=YCCDP/L
CN2=YNNP*L
C2UV=NUVP/(L*L)
C2UR=NURM/L
C2VR=NVRP/L
C2CD=NCDP/(L*L)
C2TP=KTNP/L
CSIN=LV/(L*L)
```

```
FV1=CMK*L1
FV2=-CMK*L2
SINAL=SIN(1/CRG*ALFA)*K
COSAL=COS(1/CRG*ALFA)*K
NSIGN=SIGN(N)
N2=N*N
U2=U*U
TPM=G1S*U2+G2*U*N+G3S*NSIGN*N2
CCA=C1*U2+C2*NSIGN*U2+C3*U*N+C4*N2

OUTPUT

DELM=CRG*DEL
V1=FV1*R/100.+CMK*V
V2=FV2*R/100.+CMK*V
VM=CMK*V
RM=CRG*R/100.
PSIM=CRG*PSI

DYNAMICS

SINP=SIN(PSI)
COSP=COS(PSI)
RR=R/100.
DDEL1=-TS1*DEL+TS2*DELS
DDEL=IF DDEL1<-DL1 THEN -DL1 ELSE IF DDEL1>DL1 THEN DL1 ELSE DDEL1
KSIN=SINAL*COSP-COSAL*SINP
B0=CUV*U*V+CUR*U*RR+CV2*V*ABS(V)+CYC*CCA*DEL+KTYP*TPM
B1=B0+CN2*N2-KSIN+W1
B21=C2UV*U*V+C2UR*U*RR+C2VR*ABS(V)*RR+C2CD*CCA*DEL
B2=B21+C2TP*TPM+NNNP*N2+CSIN*KSIN+W2
DV=DET1*(B1*A22-B2*A12)
DR=DET1*(B2*A11-B1*A21) *100.
DPSI=RR

G:9.80665
CMK:1.943844
CRG:57.2958
L:329.18
TS:5.0
DLIM:2.0
YVD10:
YVD20:
MXY:0.
YUV10:
YUV20:
YUR10:
YUR20:
TI:20.
YVV10:
YVV20:
YCCDP:
KIYP:
YNNP:
K:0.0
ALFA:
MXN:0
NRD10:
NRD20:
NUV10:
NUV20:
NUR10:
NUR20:
NVR10:
```

```
NVR20:
NCDP:
KINP:
NNNP:
LV:
G1:
G2:
G3:
C1:
C2:
C3:
C4:
L1:148.7
L2:131.1
N:1.283
U:8.21

END
```

```
CONTINUOUS SYSTEM LPFI1
INPUT XI
OUTPUT XO
STATE X1 X2
DER DX1 DX2
INITIAL
X1:0
X2:0
T=TP/6.283185
A1=-2*CS/T
A2=-1/(T*T)
C2=AK/(T*T)
OUTPUT
XO=C2*X2
DYNAMICS
DX1=A1*X1+A2*X2+XI
DX2=X1

AK:1             " FILTER GAIN
TP:8            " PERIOD TIME OF PEAK FREQ.
CS:0.25         " DAMPING FACTOR

"   PEAK GAIN FOR FREQ.=1/TP : AK/(2*CS)

END
```

```
CONTINUOUS SYSTEM LPFI2
INPUT XI
OUTPUT XO
STATE X1 X2
DER DX1 DX2
INITIAL
X1:0
X2:0
T=TP/6.283185
A1=-2*CS/T
A2=-1/(T*T)
C2=AK/(T*T)
OUTPUT
XO=C2*X2
DYNAMICS
DX1=A1*X1+A2*X2+XI
DX2=X1

AK:1          " FILTER GAIN
TP:8         " PERIOD TIME OF PEAK FREQ.
CS:0.25      " DAMPING FACTOR

"  PEAK GAIN FOR FREQ.=1/TP : AK/(2*CS)

END
```