

# INTER – SPECTRO

## Manuel de référence

Pour CORALIE

24 octobre 2012



# Table des matières

<b>1</b>	<b>COMMANDES</b>	<b>2</b>
1.1	ATTCALIB . . . . .	3
1.2	ATTGUIDE . . . . .	4
1.3	DCORRECTOR . . . . .	5
1.4	ENTRYSHUT . . . . .	7
1.5	FOCUS . . . . .	8
1.6	LAMP . . . . .	9
1.7	LED . . . . .	10
1.8	LFAPE . . . . .	11
1.9	MASK . . . . .	12
1.10	MCALIB . . . . .	13
1.11	MFAPE . . . . .	14
1.12	SHUTTER . . . . .	15
1.13	SPECTRO . . . . .	18
1.14	TEMPERATURE . . . . .	21
1.15	VISUSHUT . . . . .	22
<b>2</b>	<b>DICTIONNAIRE DES VARIABLES</b>	<b>23</b>
<b>3</b>	<b>REFERENCES CROISEES</b>	<b>25</b>
<b>4</b>	<b>QUICKREF</b>	<b>28</b>

# **Chapitre 1**

## **COMMANDES**

## 1.1 ATTCALIB

Commande de l'atténuateur d'étalonnage

---

### SYNTAXES:

ATTCALIB /POSITION=<position>

ATTCALIB /ECHO /STAT

---

### VARIABLES RESULTATS PRINCIPALES:

**ATTCALIB**            position atténuateur étalonnage (-2000 - +2000)

---

### QUALIFICATEURS A DISPOSITION:

**/POSITION=<position>**

Donne la position de l'atténuateur d'étalonnage en position codeur (comprise entre 0 et 4095)

**/ECHO**

Affiche le status à l'écran

**/STAT**

Retourne le status dans la variable "ATTCALIB"

---

---

### VARIABLES DU BLOC UTILISEES PAR CETTE COMMANDE

Variables	Mode	Description
ATTCALIB	I/O	position atténuateur étalonnage (-2000 - +2000)

## 1.2 ATTGUIDE

Commande de l'atténuateur de la caméra de guidage

---

### SYNTAXES:

ATTGUIDE /MIN  
 ATTGUIDE /HALF  
 ATTGUIDE /MAX  
 ATTGUIDE /ECHO /STAT

---

### VARIABLES RESULTATS PRINCIPALES:

**ATTGUIDE**            position atténuateur guidage (1=maxi,2=mini)

---

### DESCRIPTION:

On ne connaît la position de l'atténuateur de la caméra de guidage que lorsqu'il est en butée, il ne possède pas de mesure de position. C'est pour cette raison que seul les deux positions extrême et la position intermédiaire sont commandées.

---

### QUALIFICATEURS A DISPOSITION:

**/MINIMUM**  
           Va en position de densité minimum

**/HALF**  
           Va en position de densité milieu

**/MAXIMUM**  
           Va en position de densité maximum

**/ECHO**  
           Affiche le status à l'écran

**/STAT**  
           Retourne le status dans la variable "ATTGUIDE"

---

### VARIABLES DU BLOC UTILISEES PAR CETTE COMMANDE

Variables	Mode	Description
ATTGUIDE	I/O	position atténuateur guidage (1=maxi,2=mini)

## 1.3 DCORRECTOR

Commande du plateau principal du correcteur de réfraction

---

### SYNTAXES:

DCORRECTOR /NUMBER=<number>  
DCORRECTOR /POSITION=<position>  
DCORRECTOR /ANGLE=<angle>  
DCORRECTOR /CLOCKWISE  
DCORRECTOR /ANTICLOCKWISE  
DCORRECTOR /ECHO /STAT

---

### VARIABLES RESULTATS PRINCIPALES:

<b>DCONUM</b>	No du correcteur (1-5)
<b>DCOPOS</b>	Position du correcteur (0-255)
<b>DCOANG</b>	Angle du correcteur (0-255)

---

### QUALIFICATEURS A DISPOSITION:

**/NUMBER=<number>**  
Donne le numéro du correcteur de réfraction (compris entre 1 et 5)

**/POSITION=<position>**  
Donne la position du correcteur de réfraction en position codeur (compris entre 0 et 255)

**/ANGLE=<angle>**  
Donne l'angle du correcteur de réfraction en position codeur (compris entre 0 et 255)

**/CLOCKWISE**  
Rotation du correcteur de réfraction dans le sens des aiguilles d'une montre

**/ANTICLOCKWISE**  
Rotation du correcteur de réfraction dans le sens inverse des aiguilles d'une montre

**/ECHO**  
Affiche le status à l'écran

**/STAT**  
Retourne le status dans les variables "DCONUM", "DCOPOS" et "DCOANG"

---

---

---

---

**VARIABLES DU BLOC UTILISEES PAR CETTE COMMANDE**

Variables	Mode	Description
DCOANG	I/O	Angle du correcteur (0-255)
DCONUM	I/O	No du correcteur (1-5)
DCOPOS	I/O	Position du correcteur (0-255)

## 1.4 ENTRYSHUT

Commande du volet d'entrée dans la tête de fibre

---

### SYNTAXES:

ENTRYSHUT /OPEN  
ENTRYSHUT /CLOSE  
ENTRYSHUT /ECHO /STAT

---

### VARIABLES RESULTATS PRINCIPALES:

**ENTRYSHUT**      état volet d'entrée (1=close,2=open)

---

### QUALIFICATEURS A DISPOSITION:

**/OPEN**  
Ouverture du volet d'entrée

**/CLOSE**  
Fermeture du volet d'entrée

**/ECHO**  
Affiche le status à l'écran

**/STAT**  
Retourne le status dans la variable "ENTRYSHUT "

---

---

### VARIABLES DU BLOC UTILISEES PAR CETTE COMMANDE

Variables	Mode	Description
ENTRYSHUT	I/O	état volet d'entrée (1=close,2=open)

## 1.5 FOCUS

Mis au point du spectro

---

**SYNTAXES:**

FOCUS /POSITION=<position>

FOCUS /ECHO /STAT

---

**VARIABLES RESULTATS PRINCIPALES:**

**FOCUS**                    position focus (0 - 4095)

---

**QUALIFICATEURS A DISPOSITION:**

**/POSITION=<position>**

Donne la position de focus en position codeur(comprise entre 0 et 4095) 1[ADU] == 6[ $\mu$ ]

**/ECHO**

Affiche le status à l'écran

**/STAT**

Retourne le status dans la variable " FOCUS "

---

---

**VARIABLES DU BLOC UTILISEES PAR CETTE COMMANDE**

Variables	Mode	Description
FOCUS	I/O	position focus (0 - 4095)

## 1.6 LAMP

Contrôle des lampes de calibration

---

### SYNTAXES:

LAMP /OFF  
 LAMP /TUNGSTENE  
 LAMP /THORIUM  
 LAMP /THORIUM /NOWAIT  
 LAMP /ECHO /STAT

---

### VARIABLES RESULTATS PRINCIPALES:

**LAMP**                      No de lampe allumé (0=off,1=thorium,2=tungstene)

---

### QUALIFICATEURS A DISPOSITION:

**/OFF**  
 Extinction des lampes

**/TUNGSTENE**  
 Allumage lampe tungstene

**/THORIUM**  
 Allumage lampe thorium

**/NOWAIT**  
 Pas d'attente sur l'allumage. Valide uniquement avec l'allumage lampe thorium

**/ECHO**  
 Affiche le status à l'écran

**/STAT**  
 Retourne le status dans la variable "LAMP"

---

### VARIABLES DU BLOC UTILISEES PAR CETTE COMMANDE

Variables	Mode	Description
LAMP	I/O	No de lampe allumé (0=off,1=thorium,2=tungstene)

## 1.7 LED

Contrôle de la LED de pré-flash

### SYNTAXES:

LED /ON  
 LED /OFF  
 LED /MSEXPOSE=<time[ms]>  
 LED /ECHO /STAT

### VARIABLES RESULTATS PRINCIPALES:

**LED**                      état Diode pré flash (1=on 0=off)

### QUALIFICATEURS A DISPOSITION:

**/ON**

Allumage de la LED

**/OFF**

Extinction de la LED

**/MSEXPOSE=<time[ms]>**

Allumage de la LED pour une durée donnée en millisecondes. Le contrôle est rendu seulement après l'extinction de la led. Le temps effectif de l'allumage est retourné dans TIMEFF (en millisecondes).

**/ECHO**

Affiche le status à l'écran

**/STAT**

Retourne le status dans la variable "LED"

### VARIABLES DU BLOC UTILISEES PAR CETTE COMMANDE

Variables    Mode    Description

Variables	Mode	Description
LED	I/O	état Diode pré flash (1=on 0=off)
TIMEFF	I/O	temps d'exposition effectif [ms]

## 1.8 LFAPE

Contrôle de la lampe du Fabry-Perot

---

### SYNTAXES:

LFAPE /ON  
LFAPE /OFF  
LFAPE /ECHO /STAT

---

### VARIABLES RESULTATS PRINCIPALES:

**LFAPE**                    état Lampe du Fapry-Perot (1=on 0=off)

---

### QUALIFICATEURS A DISPOSITION:

**/ON**  
Allumage de la lampe  
**/OFF**  
Extinction de la lampe  
**/ECHO**  
Affiche le status à l'écran  
**/STAT**  
Retourne le status dans la variable " LFAPE "

---

---

### VARIABLES DU BLOC UTILISEES PAR CETTE COMMANDE

Variables	Mode	Description
LFAPE	I/O	état Lampe du Fapry-Perot (1=on 0=off)

## 1.9 MASK

Commande du masque des trous d'entrée de la tête de fibre

---

### SYNTAXES:

MASK /STAR  
 MASK /SKY  
 MASK /OUT  
 MASK /POSITION=<position>  
 MASK /ECHO /STAT

---

### VARIABLES RESULTATS PRINCIPALES:

**MASKPO**            position codeur masque des trous d'entrée (31-129)  
**MASKNU**            position calibrée masque des trous d'entrée (1=sky,2=star,3=out)

---

### QUALIFICATEURS A DISPOSITION:

**/STAR**  
 Place le masque sur l'étoile

**/SKY**  
 Place le masque sur le ciel

**/OUT**  
 Enlève le masque

**/POSITION=<position>**  
 Donne la position du masque en position codeur (compris entre 31 et 129)

**/ECHO**  
 Affiche le status à l'écran

**/STAT**  
 Retourne le status dans la variable "MASK"

---

### VARIABLES DU BLOC UTILISEES PAR CETTE COMMANDE

Variables	Mode	Description
MASKNU	I/O	position calibrée masque des trous d'entrée (1=sky,2=star,3=out)
MASKPO	I/O	position codeur masque des trous d'entrée (31-129)

## 1.10 MCALIB

Positionnement du miroir étalonnage

---

### SYNTAXES:

MCALIB /IN  
 MCALIB /OUT  
 MCALIB /SIMULTANEOUS  
 MCALIB /ECHO /STAT

---

### VARIABLES RESULTATS PRINCIPALES:

**MCALIB**                    position du miroir d'étalonnage (1=out,2=simultane,3=in)

---

### QUALIFICATEURS A DISPOSITION:

#### **/IN**

Place le miroir d'étalonnage complètement dans le faisceau et renvoie ainsi la lampe de calibration dans les deux fibres.

#### **/OUT**

Place le miroir d'étalonnage complètement hors du faisceau des lampes de calibration.

#### **/SIMULTANEOUS**

Place le miroir d'étalonnage avec la lampe de calibration sur la fibre ciel et laisse la fibre objet en vision directe.

#### **/ECHO**

Affiche le status à l'écran

#### **/STAT**

Retourne le status dans la variable "MCALIB"

---

### VARIABLES DU BLOC UTILISEES PAR CETTE COMMANDE

Variables	Mode	Description
MCALIB	I/O	position du miroir d'étalonnage (1=out,2=simultane,3=in)

## 1.11 MFAPE

Commande du renvoi du miroir du fabry-perrot dans la tête de fibre.

---

### SYNTAXES:

MFAPE /IN  
 MFAPE /MIDDLE  
 MFAPE /OUT  
 MFAPE /ECHO /STAT

---

### VARIABLES RESULTATS PRINCIPALES:

**MFAPE**                    position miroir Fabry-Perrot (1=out,2=middle,3=in)

---

### QUALIFICATEURS A DISPOSITION:

**/IN**  
                          Place le miroir dans le faisceau  
**/MIDDLE**  
                          Place le miroir en position intermédiaire  
**/OUT**  
                          Place le miroir hors du faisceau  
**/ECHO**  
                          Affiche le status à l'écran  
**/STAT**  
                          Retourne le status dans la variable "MFAPE "

---



---

### VARIABLES DU BLOC UTILISEES PAR CETTE COMMANDE

Variables	Mode	Description
MFAPE	I/O	position miroir Fabry-Perrot (1=out,2=middle,3=in)

## 1.12 SHUTTER

Commande de l'obturateur principal

---

### SYNTAXES:

```
SHUTTER /ARM
SHUTTER /OPEN
SHUTTER /ARM /OPEN
SHUTTER /CLOSE
SHUTTER /CLOSE /ARM
SHUTTER /MSEXPOSE=<time[ms]>
SHUTTER /EXPOSE=<time[s]>
SHUTTER /ADJUST=<time[s]>
SHUTTER /REMAIN
SHUTTER /WAIT
SHUTTER /ECHO /STAT
```

---

### VARIABLES RESULTATS PRINCIPALES:

<b>SHUTTER</b>	état shutter principal (1=arm,2=open,3=close)
<b>TIMEFF</b>	temps d'exposition effectif [ms]
<b>REMAIN</b>	temps restant [s]

---

### DESCRIPTION:

L'obturateur doit être armé pour pouvoir être ouvert.

---

---

**QUALIFICATEURS A DISPOSITION:****/ARM**

Armement

**/OPEN**

Ouverture

**/MSEXPOSE=<time[ms]>**

Ouvre l'obturateur pour une durée donnée en millisecondes. Le contrôle est rendu seulement après la fermeture du shutter (mode wait). Le temps effectif de l'exposition est retourné dans TIMEFF (en millisecondes).

**/EXPOSE=<time[s]>**

Ouvre l'obturateur pour une durée donnée en secondes. Le contrôle est immédiatement rendu à l'utilisateur (mode nowait). Une fois le temps écoulé, l'obturateur se ferme et le temps effectif de l'exposition est retourné dans TIMEFF (en millisecondes) avec l'utilisation de /WAIT.

**/ADJUST=<time[s]>**

Donne une nouvelle consigne de durée d'exposition. Le temps donné correspond à la durée totale de l'exposition. Si la nouvelle consigne est inférieure à la durée d'exposition déjà écoulée, l'obturateur se ferme et le temps effectif de l'exposition est retourné dans TIMEFF (en millisecondes).

**/REMAIN**

Retourne le temps restant avant la fin de l'exposition (en secondes) dans REMAIN.

**/WAIT**

Suspend la commande jusqu'à la fermeture du shutter.

**/CLOSE**

Fermeture de l'obturateur. Le temps effectif de l'exposition est retourné dans TIMEFF (en millisecondes).

**/ECHO**

Affiche le status à l'écran

**/NOECHO**

N'affiche pas le status à l'écran

**/STAT**

Retourne le status dans la variable " SHUTTER "

---

---

---

---

**VARIABLES DU BLOC UTILISEES PAR CETTE COMMANDE**

Variables	Mode	Description
REMAIN	I/O	temps restant [s]
SHTUNIX	I/O	tunix d ouverture
SHUTTER	I/O	état shutter principal (1=arm,2=open,3=close)
TIMEFF	I/O	temps d'exposition effectif [ms]

## 1.13 SPECTRO

Commande globale de gestion du spectrographe

---

### SYNTAXES:

SPECTRO /CLIENT  
SPECTRO /VGOP=<level>  
SPECTRO /DWRITE=<port>,<value>  
SPECTRO /DREAD=<port> [/ECHO]  
SPECTRO /AREAD=<port> [/ECHO]  
SPECTRO /ON  
SPECTRO /OFF  
SPECTRO /CLEAR  
SPECTRO /SDB  
SPECTRO /LOGBOOK  
SPECTRO /QUIT  
SPECTRO /EXIT

---

### VARIABLES INTERACTIVES PRINCIPALES:

<b>SVHOST</b>	host name server
<b>SVNAME</b>	server name
<b>SVPORT</b>	server port number

---

### VARIABLE RESULTAT PRINCIPALE:

<b>VPORT</b>	contenu d'un port (/PREAD, /AREAD)
--------------	------------------------------------

---

---

**QUALIFICATEURS A DISPOSITION:****/CLIENT**

Effectue une connection sur le serveur selon les variables "SVHOST" (défaut="localhost"), "SVNAME" (défaut="run\_spectro\_srv.csh") et "SVPORT" (défaut=7810). "SVHOST" est la machine ou se trouve le logbook, le simulateur et donc ou tournera le serveur de coralie (spectro\_srv lancé par run\_serveur\_srv.csh)

**/VGOP=<level>**

Donne le niveau de verbosité du protocole de communication (compris en 0 et 9).

**/ON**

Allume le rack d'électronique.

**/OFF**

Eteint le rack d'électronique.

**/DWRITE=<port>,<value>**

Ecrit <value> sur le port digital <port>.

**/DREAD=<port>**

Lit sur le port digital <port> et retourne le résultat dans VPORT

**/AREAD=<port>**

Lit sur le port analogique <port> et retourne le résultat dans VPORT

**/SDB**

Reconnecte le serveur sur le SDB.

**/LOGBOOK**

Reconnecte le serveur sur le Logbook.

**/ECHO**

Affiche la valeur du port à l'écran.

**/CLEAR**

Met tout les ports du rack d'electronique à zéro (attention ferme le shutter de guidage s'il est ouvert (voir :VISUSH) mais laisse le rack allumé.

**/QUIT**

Stoppe la communication avec le serveur, mais laisse le serveur en attente.

**/EXIT**

Stoppe la communication avec le serveur, et arrête le serveur.

---

---

---

---

**VARIABLES DU BLOC UTILISEES PAR CETTE COMMANDE**

Variables	Mode	Description
DOCONNECT	I/O	1 si on a lancé le serveur (fait le rsh)
SVHOST	I	host name server
SVNAME	I	server name
SVPORT	I	server port number
VPORT	I/O	contenu d'un port (/PREAD, /AREAD)

## 1.14 TEMPERATURE

Lecture des températures du système

---

### SYNTAXES:

TEMPERATURE <numéro>  
 TEMPERATURE /ALL  
 TEMPERATURE /ECHO /STAT

---

### PARAMETRES:

<numéro>            numéro de la voie (compris entre 1 et 8)

---

### VARIABLES RESULTATS PRINCIPALES:

**TEMPER**            tête de fibre

---

### DESCRIPTION:

Attention les numéros de voie sont compris entre 1 et 8 et correspondent aux voies 0 à 7 du spectro.

---

### QUALIFICATEURS A DISPOSITION:

**/ALL**  
 Lecture de toutes les voies

**/ECHO**  
 Affichage de la ou des voies désirées

**/STAT**  
 Retourne le status dans la ou les variables "TEMPER "

---



---

### VARIABLES DU BLOC UTILISEES PAR CETTE COMMANDE

Variables	Mode	Description
TEMPER	I/O	tête de fibre

## 1.15 VISUSHUT

commande de l'obturateur de la caméra de visée

---

### SYNTAXES:

VISUSHUT /OPEN

VISUSHUT /CLOSE

VISUSHUT /ECHO /STAT

---

### VARIABLES RESULTATS PRINCIPALES:

**VISUSHUT**            état obturateur de visée (1=open,2=close)

---

### QUALIFICATEURS A DISPOSITION:

**/OPEN**

Ouvre l'obturateur de la caméra de visée

**/CLOSE**

Ferme l'obturateur de la caméra de visée

**/ECHO**

Affiche le status à l'écran

**/STAT**

Retourne le status dans la variable "VISUSHUT "

---

### REMARQUES:

Attention en cas de reset du rack d'électronique (SPECTRO /CLEAR), l'obturateur se ferme.

---



---

### VARIABLES DU BLOC UTILISEES PAR CETTE COMMANDE

Variables	Mode	Description
VISUSHUT	I/O	état obturateur de visée (1=open,2=close)

## Chapitre 2

# DICTIONNAIRE DES VARIABLES

<b>AAAAAAAAAA</b>	code
<b>AREFRA</b>	coefficient de refraction (A)
<b>ATTCALIB</b>	position atténuateur étalonnage (-2000 - +2000)
<b>ATTGUIDE</b>	position atténuateur guidage (1=maxi,2=mini)
<b>BDFCUN</b>	DF - cunit
<b>BDFCUT</b>	DF - cuts
<b>BDFIDE</b>	DF - ident
<b>BDFNPI</b>	DF - npix
<b>BDFSTA</b>	DF - start
<b>BDFSTE</b>	DF - step
<b>BLKCRT</b>	bloc de donnees courant
<b>BLKTIM</b>	date du sauvetage de ce block
<b>BREFRA</b>	coefficient de refraction (B)
<b>CHRONO</b>	de chronometre interne en mu sec
<b>CMDAFF</b>	options pour l'afficheur (aff)
<b>DAY</b>	jour (TCL)
<b>DCOANG</b>	Angle du correcteur (0-255)
<b>DCONUM</b>	No du correcteur (1-5)
<b>DCOPOS</b>	Position du correcteur (0-255)
<b>DEBUG</b>	selon DEBUG /on /off
<b>DIRHLP</b>	2eme directory pour help
<b>DIRPRC</b>	directory pour procedure
<b>DOCONNECT</b>	1 si on a lancé le serveur (fait le rsh)
<b>DTUNIX</b>	delta tunix depuis le lancement de inter
<b>ENTRYSHUT</b>	état volet d'entrée (1=close,2=open)
<b>EOF</b>	end of file
<b>ERREUR</b>	erreur
<b>ERRNAM</b>	nom pour le fichier d'erreur
<b>ERRSTAT</b>	status (==1) si erreur /on
<b>FDATE</b>	date formattée (TCL)
<b>FOCUS</b>	position focus (0 - 4095)
<b>FPESTAT</b>	status (==1) si erreur /fpeon
<b>FPE_ERREUR</b>	floating point exception
<b>FUSEAU</b>	heures de decalage par rapport au TU (+ouest)
<b>GANG</b>	angle en deg de la ligne de base (normal=0) pour le texte
<b>GBCOL</b>	index couleur background pour le texte (<0 == transparent)
<b>GCOL</b>	index couleur des caracteres pour le texte
<b>GDRIVE</b>	marge pour la feuille 0=pas <0=mm >0=marge
<b>GFONT</b>	No fonte courante (1-4) (norm,roman,italic,script)
<b>GJUS</b>	justification (g->d)=(0 - 0.5 - 1)
<b>GMARGE</b>	marge pour la feuille 0=pas <0=mm >0=marge
<b>GSIZ</b>	hauteur de la font courante
<b>HOUR</b>	heure (TCL)
<b>I</b>	variable

<b>J</b>	variable
<b>K</b>	variable
<b>LAMP</b>	No de lampe allumé (0=off,1=thorium,2=tungstene)
<b>LASTF</b>	nom du dernier fichier cree
<b>LATITU</b>	latitude du lieu d'observation
<b>LED</b>	état Diode pré flash (1=on 0=off)
<b>LEAPE</b>	état Lampe du Fapry-Perot (1=on 0=off)
<b>LONGIT</b>	longitude du lieu d'observation (+est)
<b>MASKNU</b>	position calibrée masque des trous d'entrée (1=sky,2=star,3=out)
<b>MASKPO</b>	position codeur masque des trous d'entrée (31-129)
<b>MATCOU</b>	Nb de couches pour la matrice(i)
<b>MATNCL</b>	nombre de colonne sur l'ecran
<b>MATNDG</b>	nombre de digits
<b>MATNLI</b>	nombre de lignes
<b>MATSIZ</b>	Nb de pixel pour la matrice(i)
<b>MCALIB</b>	position du miroir d'étalonnage (1=out,2=simultane,3=in)
<b>MFAPE</b>	position miroir Fabry-Perrot (1=out,2=middle,3=in)
<b>MINUTE</b>	minute (TCL)
<b>MONTH</b>	mois (TCL)
<b>NX</b>	nb de pixel en X de la matrice 1
<b>NY</b>	nb de pixel en Y de la matrice 1
<b>PROMPT</b>	prompt
<b>REMAIN</b>	temps restant [s]
<b>SECOND</b>	seconde (TCL)
<b>SHAMEM</b>	indicateur de shared memory. Si diff de zero on alloue en shared memory
<b>SHTUNIX</b>	tunix d ouverture
<b>SHUTTER</b>	état shutter principal (1=arm,2=open,3=close)
<b>SRVANS</b>	reponse du serveur
<b>SRVCI</b>	identificateur de canal
<b>SVHOST</b>	host name server
<b>SVNAME</b>	server name
<b>SVPORT</b>	server port number
<b>TCL</b>	temps civil local (heures decimales)
<b>TEMPER</b>	tête de fibre
<b>TIMEFF</b>	temps d'exposition effectif [ms]
<b>TUNIX</b>	temps Unix nb de seconde depuis 1.1.1970 0h UTC
<b>TUNIXUS</b>	microsecondes de TUNIX
<b>UTC</b>	Coordinated Universal Time (heures decimales)
<b>UTC_DAY</b>	jour (UTC)
<b>UTC_FDATE</b>	date formattée (UTC)
<b>UTC_HOUR</b>	heure (UTC)
<b>UTC_MINUTE</b>	minute (UTC)
<b>UTC_MONTH</b>	mois (UTC)
<b>UTC_SECOND</b>	seconde (UTC)
<b>UTC_YEAR</b>	annee (UTC)
<b>VERBOS</b>	selon VERBOSE /on /off
<b>VISUSHUT</b>	état obturateur de visée (1=open,2=close)
<b>VPORT</b>	contenu d'un port (/PREAD, /AREAD)
<b>XCORIG</b>	coord w. de dep en X de l'image d'origine 1
<b>XSIZE</b>	taille de l'espace graphique en mm
<b>XSTART</b>	coord. world de depart en X de la matrice 1
<b>XSTEP</b>	pas en X de la matrice 1
<b>YCORIG</b>	coord w. de dep en Y de l'image d'origine 1
<b>YEAR</b>	annee (TCL)
<b>YSIZE</b>	taille de l'espace graphique en mm
<b>YSTART</b>	coord. world de depart en Y de la matrice 1
<b>YSTEP</b>	pas en Y de la matrice 1

## **Chapitre 3**

# **REFERENCES CROISEES**

<b>Variables</b>	<b>lues par</b>	<b>mises à jour par</b>
AAAAAAAAAA	main	—
BDFCUN	patch	—
BDFIDE	patch	—
BLKCRT	main	—
BLKTIM	main	—
CMDAFF	aff	—
DIRHLP	main	—
DIRPRC	main	—
ERRNAM	main	—
FDATE	—	date
GDRIVE	print	—
LASTF	main	—
PROMPT	main	—
SHTUNIX	shutter	shutter
SRVANS	server	server
SVHOST	spectro	—
SVNAME	spectro	—
TUNIX	—	date
UTC_FDATE	—	date
AREFRA	main	—
ATTCALIB	attcalib	attcalib
ATTGUIDE	attguide	attguide
BDFCUT	patch	—
BDFNPI	patch	—
BDFSTA	patch	—
BDFSTE	patch	—
BREFRA	main	—
CHRONO	main	—
DAY	—	date
DCOANG	dcorrector	dcorrector
DCONUM	dcorrector	dcorrector
DCOPOS	dcorrector	dcorrector
DEBUG	debug	debug
DOCONNECT	spectro	spectro
DTUNIX	—	date
ENTRYSHUT	entryshut	entryshut
EOF	main	—
ERREUR	erreur	erreur
ERRSTAT	erreur	erreur
FOCUS	focus	focus
FPESTAT	erreur	erreur
FPE_ERREUR	erreur	erreur
FUSEAU	main	—
GANG	print	—
GBCOL	print	—
GCOL	print	—
GFONT	print	—
GJUS	print	—
GMARGE	print	—
GSIZ	print	—
HOUR	—	date
I	main	—
J	main	—
K	main	—
LAMP	lamp	lamp
LATITU	main	—
LED	led	led
LFAPF	lfape	lfape
LONGIT	main	—
MASKNU	mask	mask

<b>Variables</b>	<b>lues par</b>	<b>mises à jour par</b>
MASKPO	mask	mask
MATCOU	matrix	—
MATNCL	matrix	—
MATNDG	matrix	—
MATNLI	matrix	—
MATSIZ	matrix	—
MCALIB	mcalib	mcalib
MFAPE	mfape	mfape
MINUTE	—	date
MONTH	—	date
NX	fimage matrix patch server	fimage matrix
NY	fimage matrix patch server	fimage matrix
REMAIN	shutter	shutter
SECOND	—	date
SHAMEM	main	—
SHUTTER	shutter	shutter
SRVCI	server	server
SVPORT	spectro	—
TCL	—	date
TEMPER	temperature	temperature
TIMEFF	led shutter	led shutter
TUNIXUS	—	date
UTC_DAY	—	date
UTC_HOUR	—	date
UTC_MINUTE	—	date
UTC_MONTH	—	date
UTC_SECOND	—	date
UTC_YEAR	—	date
UTC	—	date
VERBOS	verbose	verbose
VISUSHUT	visushut	visushut
VPORT	spectro	spectro
XCORIG	fimage	fimage
XSIZE	main	—
XSTART	fimage matrix patch	fimage
XSTEP	fimage matrix patch	fimage
YCORIG	fimage	fimage
YEAR	—	date
YSIZE	main	—
YSTART	fimage matrix patch	fimage
YSTEP	fimage matrix patch	fimage

**Chapitre 4**

**QUICKREF**

## Fonctions

\* .....  
 \*\* .....  
 + .....  
 - .....  
 / .....  
 .AND .....  
 .EQ .....  
 .GE .....  
 .GT .....  
 .LE .....  
 .LT .....  
 .NE .....  
 .NOT .....  
 .OR .....  
 ABS(x) .....  
 ACOS(x) .....  
 ACOSD(x) .....  
 ADD(variable) .....  
 AHDE2AZ(angle\_horaire, delta) .....  
 AHDE2EL(angle\_horaire, delta) .....  
 AND(a,b) .....  
 ANGLE("angle") .....  
 ANUT(alpha,delta) .....  
 APPNAME() .....  
 APREC(alpha,delta) .....  
 ASIN(x) .....  
 ASIND(x) .....  
 ASSIGN(str[,prefix]) .....  
 ATAN(x) .....  
 ATAN2(y,x) .....  
 ATAN2D(y,x) .....  
 ATAND(x) .....  
 ATOR(str) .....  
 AZEL2AH(azimut, élévation) .....  
 AZEL2DE(azimut, élévation) .....  
 CAL(x,y) .....  
 CDE(x,y) .....  
 CHAR(code) .....  
 CLEAN([mat],seuil,size,valrep) .....  
 CLEARSV() .....  
 COMPAC(str) .....  
 CONNECT() .....  
 COS(alpha) .....  
 COSD(alpha) .....  
 DCDIV(x,y) .....  
 DCMINUS(x,y) .....  
 DCMUL(x,y) .....  
 DCPLUS(x,y) .....  
 DDDIV(x,y) .....  
 DDTOD(angle) .....  
 DDTOD2(angle) .....  
 DEG2RAD(x) .....  
 DIM(variable) .....  
 DMINUS(x,y) .....  
 DMUL(x,y) .....  
 DNUOT(alpha,delta) .....  
 DPLUS(x,y) .....

## Descriptions

multiplication.  
 exponentiation.  
 addition.  
 soustraction.  
 division.  
 ET logique.  
 égalité numérique.  
 plus grand ou égal.  
 strictement plus grand.  
 plus petit ou égal.  
 strictement plus petit.  
 non égalité numérique.  
 négation de la condition.  
 OU logique.  
 retourne la valeur absolue de  $x$ .  
 retourne l'arc cosinus de  $x$  en radian.  
 retourne l'arc cosinus de  $x$  en degré.  
 retourne l'adresse de la *variable* numérique dans les tableaux d'Inter.  
 calcul l'azimut.  
 calcul l'élévation.  
 retourne le résultat du 'and' binaire  
 retourne la valeur numérique d'un angle donné sous forme de chaîne de caractères.  
 nute la coordonnée donnée en alpha.  
 retourne le nom de l'application.  
 précessionne la coordonnée donnée en alpha.  
 retourne l'arc sinus de  $x$  en radian.  
 retourne l'arc sinus de  $x$  en degré.  
 assignation de variables selon une chaîne formatée contenant une suite variable, contenu.  
 retourne l'arc tangente de  $x$  en radian.  
 retourne l'arc tangente selon  $x,y$  en radian.  
 retourne l'arc tangente selon  $x,y$  en degré.  
 retourne l'arc tangente de  $x$  en degré.  
 retourne le nombre écrit dans *str*.  
 calcul angle horaire.  
 calcul delta.  
 conversion de coordonnées cartésiennes en coordonnées équatoriales.  
 conversion de coordonnées cartésiennes en coordonnées équatoriales.  
 retourne le caractère ASCII donné par *code*.  
 patch une matrice centrée de cote= $2*size+1$  contenant valrep pour chaque point plus élevé que seuil  
 Efface les flags d'erreur et de status ainsi que les code et message d'erreur ainsi que le texte de la commande courante dans le bloc de communication.  
 retourne *str* sans espace.  
 Se (re)connecte sur les sémaphores du serveur courant s'ils ont été tués.  
 retourne le cosinus de *alpha* donné en radian.  
 retourne le cosinus de *alpha* donné en degré.  
 Division. Même remarque que pour DCPLUS.  
 Soustraction. Même remarque que pour DCPLUS.  
 Multiplication. Même remarque que pour DCPLUS.  
 Addition. Les arguments sont caractères, le résultat est caractère.  
 Division. Même remarque que pour DPLUS.  
 formate *angle* en : "SDDd MMm SSs ".  
 formate *angle* en : "SDDD:MM:SS ".  
 conversion degrés radians.  
 retourne la dimension d'une *variable*.  
 Soustraction. Même remarque que pour DPLUS.  
 Multiplication. Même remarque que pour DPLUS.  
 nute la coordonnée donnée en delta  
 Addition. Les arguments sont numériques ou caractères, le résultat est numérique en simple précision.

## Fonctions

DPREC(alpha,delta) .....  
 ENVDEF(evar) .....  
 EXIST(fichier) .....  
 EXP(x) .....  
 EXPAND(val|vec, mul [,size]) .....  
 FETCH() .....  
 FIND(str1,str2) .....  
 FITGAU(vec) .....  
 FORMAT(val|vec|str, fmt) .....  
 FZ(dist\_zénitale) .....  
 GENF(xvec, polyvec) .....  
 GENGAU(vec, nb\_points) .....  
  
 GETENV(evar) .....  
 GETLU() .....  
 GROUP() .....  
 HDTOH(heure) .....  
 HDTOH2(heure) .....  
 HTOHD("heure") .....  
 ICHAR(str) .....  
 INDEX(str1,str2) .....  
 INILOG(host) .....  
 INMAXV(vec) .....  
 INMINV(vec) .....  
 INT(x) .....  
 ISNUM(xxx) .....  
 ISVNUM(xxx) .....  
 ITOA(val) .....  
 JD([j],[m],[a]) .....  
 JD0IN([j],[m],[a]) .....  
 LCAT(str1, ..., strN) .....  
 LCEQ(str1,str2) .....  
 LEN(str) .....  
 LEQ(str1,str2) .....  
 LGE(str1,str2) .....  
 LGT(str1,str2) .....  
 LLE(str1,str2) .....  
 LLT(str1,str2) .....  
 LNE(str1,str2) .....  
 LNO(str) .....  
 LOG(x) .....  
 LOG10(x) .....  
 LOWER(str) .....  
 LSCAT(str1,str2,...,strN) .....  
  
 LSFIT(No\_de\_matrice) .....  
 LTRIM(str) .....  
 LYES(str) .....  
 MAX(x1,x2,...,xn) .....  
 MAXSIZ(No\_mat) .....  
  
 MAXV(vec) .....  
 MEDDEV(vec) .....  
 MEDIAN(vec) .....  
 MIN(x1,x2,...,xn) .....  
 MINV(vec) .....  
 MOD(x,y) .....  
 NBCOU(No\_mat) .....  
 NBMAT() .....  
 NBPIX() .....

## Descriptions

précessionne la coordonnée donnée en delta.  
 retourne 1 si la variable d'environnement est définie, 0 sinon.  
 retourne 1 si le *fichier* existe.  
 retourne l'exponentiel de *x*.  
 expand chaque position d'un vecteur ou d'une matrice.  
 assigne les variable Inter selon les couples donnés dans le bloc de communication.  
 retourne 1 si *str2* existe dans *str1*.  
 Fit une gaussienne sur les données de *vec*.  
 retourne *val|vec|str* formaté le format *fmt*.  
 calcul de la masse d'air (voir slalib).  
 fabrique *f(xvec)* avec *polyvec* comme coefficient du polynôme.  
 génère une gaussienne selon les 4 premières valeurs du vecteur *vec*. (1=max, 2=centre, 3=forme, 4=background)  
 retourne le contenu de la variable d'environnement *evar*.  
 retourne une unité logique inutilisée.  
 retourne le numéro de groupe.  
 formate *heure* en : "HHh MMm SS.Ss".  
 formate *heure2* en : "HH:MM:SS.S".  
 retourne la valeur numérique d'une heure donnée sous forme de chaîne de caractères.  
 retourne le code ASCII du premier caractère de la chaîne *str*.  
 retourne la position de *str2* dans *str1*.  
 Initialise une connection sur le logbook de la machine *host*.  
 retourne l'index du premier plus grand élément de *vec*.  
 retourne l'index du premier plus petit élément de *vec*.  
 retourne la valeur entière de *x*.  
 retourne 1 si "xxx" est un nombre (lexicalement parlant).  
 retourne 1 si "xxx" est une variable numérique.  
 retourne le nombre *val* formaté en entier sans blanc.  
 retourne le jour julien formaté (1x,f11.3,1x).  
 initialise le jour julien de référence (jd0) pour les calculs de précession et de nutation.  
 retourne la concaténation des tous les arguments (type caractère) .  
 retourne 1 si *str1 = str2*. Cette fonction est insensible aux minuscules et aux majuscules  
 retourne le nombre de caractères dans *str*.  
 retourne 1 si *str1 = str2*.  
 retourne 1 si *str1 ≥ str2*.  
 retourne 1 si *str1 > str2*.  
 retourne 1 si *str1 ≤ str2*.  
 retourne 1 si *str1 < str2*.  
 retourne 1 si *str1 ≠ str2*.  
 retourne 1 si *str* vaut "n", "no" ou "non".  
 retourne le logarithme naturel de *x*.  
 retourne le logarithme base 10 de *x*.  
 retourne *str* mis en minuscule.  
 retourne la concaténation des tous les arguments (type caractère) en placant un espace entre chaque argument.  
 résoud *m* équations à *n* inconnues.  
 retourne *str* sans les espaces à gauche du texte utile.  
 retourne 1 si *str* vaut "y", "yes", "o" ou "oui" .  
 retourne le maximum parmi les argument *x1...xn*.  
 retourne la taille actuelle ou maximum (dans le cas d'une matrice en mémoire partagée) de la matrice *No\_mat*.  
 retourne le plus grand élément de *vec*.  
 retourne le sigma median (nag :g07daf)  
 retourne le median (nag :g07daf)  
 retourne le minimum parmi les argument *x1...xn*.  
 retourne le plus petit élément de *vec*.  
 retourne le modulo de *x* par *y*.  
 retourne le nombre de couche pour la matrice *No\_mat*.  
 retourne le nombre de matrices accessibles.  
 retourne le nombre de pixels alloués par l'ensemble des matrices en mémoire partagée.

## Fonctions

NINT(x) .....  
 NOCOU(str|num) .....  
 NOMAT(str|num) .....  
 NOT(a) .....  
 NUTIN([j][.][m][.][a]) .....  
 OR(a,b) .....  
 PAD(str) .....  
  
 PARSE(str,No) .....  
 POLYF(xvec, yvec, degre [,pvec]) .....  
 PRECIN([j][.][m][.][a]) .....  
 RAD2DEG(x) .....  
 RAND(flag) .....  
 REFCO(tdk, pmb, rh, [wl], [eps], [ttr], [hm], [phi]) ..  
 REFRAC(dist\_zénitale) .....  
 RFITS(file, key) .....  
 RM ?(string) .....  
  
 ROTHPOS(ah,delta) .....  
 ROTHVEL(ah,delta) .....  
 ROTPOS(alpha,delta) .....  
 ROTVEL(alpha,delta) .....  
 RTRIM(str) .....  
 SELECT(server) .....  
 SEMAGET(No) .....  
 SEMASET(No, val) .....  
 SETTENV("evar=val") .....  
 SETV(start,stop[,step]) .....  
 SHMACK() .....  
  
 SHMADD(key, content) .....  
  
 SHMCLI([name]) .....  
  
 SHMCMD(cmd [,to]) .....  
 SHMCMDW(cmd [,to\_wait [,to\_cmd]]) .....  
  
 SHMCONT() .....  
  
 SHMFREE() .....  
 SHMGACK() .....  
 SHMGCOD() .....  
 SHMGERR() .....  
 SHMGET(key) .....  
 SHMGID() .....  
 SHGMES() .....  
 SHMGSRV() .....  
 SHMGSTA() .....  
 SHMINIT() .....  
 SHMNCNT() .....  
 SHMPCOD() .....  
 SHMPUT(key, content) .....  
  
 SHMSHOW() .....  
 SHMSRV() .....  
 SHMWACK([timeout]) .....  
 SHMWAIT([timeout]) .....  
 SHMZCNT() .....  
 SHMZERO() .....  
 SHOWSEL() .....

## Descriptions

retourne la plus proche valeur entière de *x*.  
 retourne le No de couche d'un No de matrice.  
 retourne le No de matrice d'un No de matrice.  
 retourne le résultat du 'not' binaire  
 initialise les variables internes pour le calcul de la nutation.  
 retourne le résultat du 'or' binaire  
 retourne *str* sans les espaces en début et fin de chaîne et avec une seule barre de soulignement ("\_") pour chaque suites d'espaces entres les mots.  
 retourne le champ *No* de *str*, avec pour séparateur le premier caractère de *str*.  
 fit polynômial à une inconnue de degré *degré*.  
 initialise les variables internes pour le calcul de la précession.  
 conversion radians degrés.  
 retourne un nombre aléatoire.  
 calcul des parametres AREFRA et AREFRA du modèle de réfraction atmospherique.  
 calcul de la réfraction atmosphérique pour La Silla.  
 Lit un keyword *key* du fichier *file*.  
 Cette fonction supprime les points d'interrogation en fin de ligne. (utile pour les commandes @@ et @@@ ...).  
 angle du derotateur. Remarque : utilise longit et latitu.  
 vitesse du derotateur [deg/s]. Remarque : utilise longit et latitu.  
 angle du derotateur moment courant. Remarque : utilise longit et latitu.  
 vitesse du derotateur moment courant [deg/s]. Remarque : utilise longit et latitu.  
 retourne *str* sans les espaces à droite du texte utile.  
 Sélectionne le serveur sur lequel vont travailler toutes les fonctions de communication.  
 retourne la valeur du sémaphore *No*.  
 Met le sémaphore *No* à *val*.  
 assigne une variable d'environnement *evar*.  
 remplit un vecteur selon les paramètres de boucle.  
 indique au serveur d'exécuter la commande placée dans "COMMAND" et lui signale de ne pas rendre la main.  
 ajoute un paramètre référencié par *key* et son contenu *content*, dans le bloc de communication .  
 retourne 1 si Inter a été lancé en mode client. Si name est précisé, alors retourne 1 si Inter est client de *name*  
 Suspend le client, envoie une commande et libère le client.  
 Suspend le client, envoie une commande et attend la fin de la commande pour libérer le client.  
 indique au serveur d'exécuter la commande placée dans "COMMAND" et lui signale de rendre la main.  
 Rend la main (après un SHMWACK()).  
 retourne la valeur du flag ackno.  
 retourne le code d'erreur (ascii).  
 retourne le code d'erreur (numérique).  
 lit un paramètre référencié par *key* dans le bloc de communication.  
 retourne l'identificateur du bloc de mémoire partagée.  
 retourne le texte du message d'erreur.  
 retourne le nom du serveur courant.  
 retourne le status d'erreur.  
 initialise le bloc de communication en le vidant.  
 retourne le nombre de client en attente sur le serveur courant.  
 met le code d'erreur, l'erreur et le message d'erreur en shared memory.  
 écrit un paramètre référencié par *key* et son contenu *content* dans le bloc de communication et initialise le bloc de communication.  
 visualise le bloc de communication à l'écran.  
 retourne 1 si Inter a été lancé en mode serveur.  
 attend que le serveur ait finis (après un SHMACK()).  
 suspend le client jusqu'à que le serveur soit prêt.  
 retourne le nombre de process en attente de zero sur sem No.  
 met le semaphore zéro à 1 (reset).  
 affiche le nom de tous les serveurs possible ainsi que le serveur actuellement sélectionné.

**Fonctions**

SIGNAL(signal) .....  
 SIN(alpha) .....  
 SIND(alpha) .....  
 SLEEP(sec) .....  
 SPARSE(str,No) .....  
 SQRT(x) .....  
 SRVEXIS() .....  
 SRVWAIT() .....  
 SRVWORK() .....  
 STDDEV(vec) .....  
 STRIP(str) .....  
  
 SUM(vec) .....  
 SYSTEM(cmd) .....  
 TAN(alpha) .....  
 TAND(alpha) .....  
 TEMP(flag) .....  
 TIME() .....  
 TPARSE(str,No) .....  
 TRIM(str) .....  
 TS([tcl][.][j][.][m][.][a][.]) .....  
 UPPER(str) .....  
 USER() .....  
 VEC(arg1,arg2,...,argn) .....  
 VERS() .....  
 XBARY([mat]) .....  
 XOR(a,b) .....  
 YBARY() .....  
 \$(string) .....

**Descriptions**

Envoie un signal au serveur.  
 retourne le sinus de *alpha* donné en radian.  
 retourne le sinus de *alpha* donné en degré.  
 Suspend le process durant un temps donné en seconde  
 retourne le champ *No* de *str*, avec pour séparateur l'espace.  
 retourne la racine carrée de *x*.  
 retourne 1 si le serveur existe.  
 indique si le client a la main sur le serveur courant.  
 indique si le serveur est en train de travailler.  
 retourne le sigma (nag :g07daf)  
 supprime les blancs, les 0 non significatifs et le point non significatif sur une chaîne numérique.  
 retourne la somme des valeur contenus dans *vec*.  
 retourne le résultat fournis par la commande system *cmd*.  
 retourne la tangente de *alpha* donné en radian.  
 retourne la tangente de *alpha* donné en degré.  
 retourne certaines températures du CCD.  
 donne l'heure de la machine (heures décimales).  
 retourne le champ *No* de *str*, avec pour séparateur le tabulateur.  
 retourne *str* sans les espaces avant et apres le texte utile.  
 donne l'heure sidérale.  
 retourne *str* mis en majuscule.  
 retourne le No d'utilisateur.  
 fabrique un vecteur en joignant tous les arguments.  
 retourne la date de la dernière compilation d'Inter.  
 retourne le x barycentre  
 retourne le résultat du 'xor' binaire  
 retourne le y barycentre (il faut utilise *xbary(mat)* avant)  
 retourne le contenu de la variable nommée dans *string*.