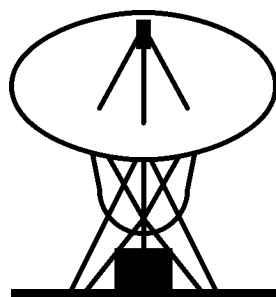


*Raport wewnętrzny Katedry Radioastronomii*

**ODBIÓR DANYCH  
TELEMETRYCZNYCH  
ze sztucznych satelitów Ziemi  
za pomocą 15-metrowego radioteleskopu  
UMK w Toruniu**



Opracował: K.M. Borkowski

Centrum Astronomii, Katedra Radioastronomii  
Piwnice, czerwiec 1998

# Spis treści

1. Wstęp .....	2
2. Przygotowanie terminarza obserwacji .....	3
3. Przygotowanie plików sterujących .....	4
4. Gdy trzeba zmodyfikować programy .....	6
5. Prowadzenie obserwacji .....	7
6. Obsługa systemu sterowania radioteleskopu RT15 .....	8
7. Dodatek 1: Aparatura odbiorczo-pomiarowa .....	11
8. Dodatek 2: Charakterystyki odbieranego sygnału .....	13
9. Uzupełnienie: Wydruk programu vect .....	16

## 1 Wstęp

Niniejsze opracowanie dotyczy projektu badawczego zarejestrowanego przez KBN pod numerem 2 PO3C 007 12. Wnioskodawcami (wykonawcami) projektu są: K. Borkowski, J. Hanasz, A. Kus i E. Pazderski.

Projekt ten obejmuje codzienne zbieranie i wstępną obróbkę sygnałów telemetrycznych z okołozemskiego satelity INTERBALL 2 (tzw. sonda zorzowa), w tym z polskiego eksperymentu POLRAD, za pomocą radioteleskopu o średnicy 15 m oraz aparatury odbiorczej i rejestrującej. Uruchomienie tej stacji telemetrycznej opiera się w głównej mierze na istniejącym sprzęcie Uniwersytetu M. Kopernika w Piwnicach k. Torunia (w Katedrze Radioastronomii) oraz podzespołach dostarczonych z Rosji (Moskwa).

Stacja ta jest pierwszą tego rodzaju placówką w Polsce. Zadanie uruchomienia stacji było nietypowe i wymagało znajomości technik radioastronomicznych. Ponieważ sygnał z satelity okazał się słabszy niż przewidywano, włożono wiele wysiłku na zoptymalizowanie wszystkich parametrów całej stacji (oświetlacze, przedwzmacniacz w.cz., dekodery sygnału cyfrowego, dokładność śledzenia anteną za satelitą, dokładność predykcji położenia satelity). Pierwsze testowe obserwacje wykonano w dniach 16–22 grudnia 1996 r. podczas pobytu delegacji IKI RAN w osobach V.N. Skorodunova i A.E. Tretyakova, którzy dostarczyli i zainstalowali dwa odbiorniki (R137 o Nr 7 i 8), wysokoczęstotliwościowy symulator (TC-2-CK, Nr 7403), symulator strumienia danych cyfrowych (CTO-II-02), demodulator częstotliwości podnośnych (DFD-4) i komputerowy program zbierający cyfrowe dane telemetryczne.

Regularne zbieranie danych telemetrycznych rozpoczęto dnia 1 stycznia 1997 r.

Satelita Interball 2 obiega Ziemię po bardzo wydłużonej orbicie z okresem nieco krótszym niż 6 godz. Transmisje telemetryczne typu STO (Sistema Telemetricheskogo Obepecheniya) nadawane są w czasie, gdy satelita przechodzi przez określone obszary zorzowe. Zwykle czas trwania transmisji (seansu) wynosi ok. 3 godziny na jednym okrążeniu. Takich seansów może być do czterech w ciągu doby.

Programy do przygotowania terminarza obserwacji i plików sterujących znajdują się na komputerze euka (pok. 31) w katalogu d:\intrb\telem. Program sterujący teleskopem oraz pliki do sterowania są złożone na komputerze trao2 oraz (rt3) w kabinie pod teleskopem. Archiwum na CD obok danych zawiera również katalog intrb, w którym mieszczą się wszystkie programy oraz pliki sterujące z całego okresu realizacji projektu (do końca czerwca 1998 r.).

## 2 Przygotowanie terminarza obserwacji

Szczegółowy terminarz nadawania sygnałów STO ustalany jest na podstawie propozycji kierowników projektów realizowanych na satelicie INTERBALL 2 oraz analizy przewidywanego ruchu satelity. Znany jest zwykle z wyprzedzeniem tygodnia lub więcej — niebył regularnie. Pliki zawierające informacje o terminach i czasach przekazu STO składane są w Moskwie na komputerze **romance** (ftp 193.232.8.13 w katalogu `/interball/intrb/av/control/strsns`). Stamtąd przez pewien czas były automatycznie „ściągane” na komputer toruńskiego Centrum Badań Kosmicznych PAN **fox**. Jeśli ten automat zawodzi, należy „wlogować się” (np. poprzez **telnet**) na **foxa** jako użytkownik **polrad** z niejawnym hasłem i dokonać transmisji **ftp** z Moskwy. Pliki noszą nazwy **strNSSvM.txt**, gdzie **NSS** jest kolejnym 3-cyfrowym numerem, a **M** wersją (0 lub wyżej). Pliki nowe, których jeszcze nie ma na **foxie**, należy przesłać do katalogu `d:\intrb\str`. W tym celu będąc „wlogowanym” do komputera **euka** w trybie DOS należy wykonać następujące operacje:

```
d:
cd \intrb\str
ftp fox (z innego komputera: ftp fox.ncac.torun.pl)
login: polrad
password: ***** (tu należy podać niejawne hasło)
cd dos/intrb/av/control/strsns
dir
```

Jeśli lista wyświetlona po ostatnim poleceniu ujawni nowe pliki, to przesyłamy je do Piwnic poleceniem typu

```
mget str???.txt lub get pełna-nazwa-pliku.
bye
```

Ponieważ pliki `str...txt` zawierają także dużo innych informacji, przygotowano specjalny program `telem` przeznaczony do ekstrakcji danych STO i zamiany czasu moskiewskiego na uniwersalny, UT. Aby go wykorzystać należy (na komputerze **euka**) wykonać następujące kroki:

```
d:
cd \intrb\telem
telem [NSS]
```

Na ekranie pojawi się kilka komunikatów, przykładowo:

```
Program usage: TELEM [nss[,dir[,day]]]
where nss - the first 'STRnssV?.TXT' file number
      dir - parent directory of these files (eg.d:\intrb\str\)
      day - day of month replacing the today's date
Reading file: d:\intrb\str\STR130V2
Reading file: d:\intrb\str\STR133V1
Reading file: d:\intrb\str\STR135V0
Reading file: d:\intrb\str\STR137V0
Reading file: d:\intrb\str\STR140V0
See file TELEM.TXT
```

Opcjonalne **NSS** to nr pliku `d:\intrb\str\strNSSv?.txt`, od którego program ten ma zacząć analizę. Np. plik z **NSS** = 140 obejmuje dni od 19 do 25 maja 1997 r. W powyższym przykładzie został wyprodukowany zbiór `telem.txt`, którego początek wygląda tak:

## INTERBALL TELEMETRY

		1997 UT [hh.mm]				Moscow Time		Orbit
Dw	M D	Start	End	M	D	h.mm	h.mm	No
	Czw	05-01-08.50	- 12.08	5	1	11.50	- 15.08	1018
	Pią	05-02-08.00	- 11.14	5	2	11.00	- 14.14	1022
	Sob	05-03-07.01	- 10.20	5	3	10.01	- 13.20	1026
	Nie	05-04-12.30	- 15.36	5	4	15.30	- 18.36	1031

W tabelce tej prawa kolumna zawiera numer orbity, który pozwala na wyeliminowanie grubych błędów daty. Mianowicie, jeśli numery orbit różnią się o 1, to odpowiadające seanse powinny zaczynać się w odstępnie ok. 6 godz.; jeśli różnią się o 4, to odstęp winien wynosić  $23 \pm 1$  godzin.

Nowe wiersze lewej części tej tabelki można przenieść (edytorem) do pliku `plan.tex` (znajdującego się także w katalogu `d:\intrb\telem`) w celu uaktualnienia go. Kolumny `M D Start` w tych dwóch plikach (`telem.txt` i `plan.tex`) zawierają datę (miesiąc i dzień) oraz czas UT (godzinę i minuty) początku seansu STO, ale stanowią jednocześnie nazwę odpowiedniego pliku z komendami dla programu sterującego teleskopem RT15. Uaktualniony plik `plan.tex` można wydrukować na potrzeby obserwatorów, jednak należy złożyć go też na komputerze `trao2` w katalogu zawierającym pliki komend do sterowania:

```
ftp tra02
user: oper
password: ***** (tu należy podać niejawnne hasło)
cd /temp/oper/rt3cmd
```

`show plan.tex` (w celu stwierdzenia, czy obserwatorzy nie dokonali zmian rezerwacji swoich dyżurów w poprzedniej wersji tego pliku. Jeśli są zmiany, należy pobrać stary plik, np. `get plan.tex plan.` i najpierw skorygować nowy plik `plan.tex` wg. zawartości `plan.`, a dopiero potem powtórzyć czynności „ftp-owania” i przesłać go na `trao2`).

```
put plan.tex
bye
```

### 3 Przygotowanie plików sterujących

Do przygotowywania plików strujących, z których korzysta program `rt3` i `rt3load` (na `trao2`), służy program `vect` (znajduje się on na `euka` w katalogu `d:\intrb\telem`). Jego działanie polega na rozwiązaniu równania Keplera dla zadanej *anomalii średniej* i oskulacyjnych elementów orbity satelity Ziemi dostępnych w formie typowego dla INTERBALLa pliku `latest2.txt` lub w postaci wektora położenia i prędkości w pliku `a.f1`. Ponieważ orbita wykazuje dość regularną precesję, program uwzględnia to przed rozwiązaniem równania Keplera. Rozwiązanie w postaci *anomalii prawdziwej* jest transformowane na geocentryczne współrzędne równikowe, a następnie na topocentryczne współrzędne równikowe (rektascensję lub kąt godzinny i deklinację) dla miejsca anteny toruńskiej.

W ostatnim kroku współrzędne równikowe są wpisywane do pliku w formie poleceń sterujących radioteleskopem RT15, o ile satelita znajduje się w polu widzenia teleskopu, tj. jest nad horyzontem i jego kąt godzinny zawiera się w przedziale ok.  $\pm 90^\circ$ . Program może generować automatycznie pliki wynikowe zawierające polecenia sterujące dla kolejnych seansów według pliku `plan.tex` z zadaniem krokiem śledzenia satelity.

Oto przykład komunikatów wypisywanych na ekranie przez program `vect` w przypadku, gdy użytkownik odpowiada za pierwszym razem `e`, potem kolejno kilkakrotnym wciśnięciem klawisza `ENTER`, a na końcu `<Ctrl>c`:

```
Elements (e) or vector (v) as input [ENTER=v]: e

Output file (1=scan, 2=Ra-d, 3=Ha-d, 4=Test, 5=Data, 6=Vis, 7=*) [ENTER=1]:

Date (year, month, day) [ENTER=97, 5,13]:

Ephemeris calculated every [ENTER= 5.0] minutes:

13/05/1997 --> Start, stop time (UTC hh.mm) [Enter= 4.22, 7.00]:

Filename [ENTER=05-13-04.22]:

UT[hh.mm] = 4.02   Angle [deg], D [km]:  39.6   7359
UT[hh.mm] = 4.32   Angle [deg], D [km]:  94.7   10149
UT[hh.mm] = 5.02   Angle [deg], D [km]: 118.9   14449
UT[hh.mm] = 5.32   Angle [deg], D [km]: 132.1   17691
UT[hh.mm] = 6.02   Angle [deg], D [km]: 141.0   19740
UT[hh.mm] = 6.32   Angle [deg], D [km]: 147.0   20670
UT[hh.mm] = 7.02   Angle [deg], D [km]: 149.8   20558

13/05/1997 --> Start, stop time (UTC hh.mm) [Enter=10.30,13.40]: <Ctrl>c
```

W wyniku takich operacji został wczytany plik `latest2.txt` (skutek wyboru opcji `e`) i utworzony zbiór `05-13-04.22` z danymi do sterowania na sesję z dnia 13 maja 1977 r. w godz. 4:22 – 7:00 (te dane o sesji zostały pobrane automatycznie z pliku `plan.tex`). Zbiór `05-13-04.22` ma następującą strukturę:

```
comment 06/05/1997
epoch 1997.36
comment 39.6/07359
13/05/1997 4:02:00 ps 274.722 -7.835
  po -3.396 -7.080
  pos 5.00
comment 50.3/07346
13/05/1997 4:07:00 ps 281.191 5.667
  po -3.074 -6.422
  pos 5.00
.....
comment 149/20646
13/05/1997 6:57:00 ps 74.310 40.456
  po -0.817 0.879
  pos 5.00
comment 149/20558
13/05/1997 7:02:00 pb
end
```

(Pierwsze polecenie, opatrzone etykietą czasu, w tym pliku przypada na 20 min przed rozpoczęciem sesji STO po to, aby operator miał czas na ustawienie anteny.)

Wygenerowane pliki sterujące normalnie nie wymagają żadnych modyfikacji przed użyciem do sterowania radioteleskopem. Po wygenerowaniu trzeba je tylko przesłać na `trao2`:

```
ftp tra02
user: oper
password: ***** (tu należy podać niejawne hasło)
cd /temp/oper/rt3cmd
mput 05-*. * (przesłanie wszystkich plików z maja)
bye
```

Elementy orbity INTERBALLa dość szybko ulegają degradacji. Mniej więcej po tygodniu należałoby dysponować nowymi wyznaczeniami, co nie zawsze jest możliwe i grozi zwiększonym błędem obliczonej pozycji satelity. Trzeba więc często sprawdzać czy strona rosyjska nie złożyła nowych plików.

Pliki `latest2.txt` i/lub `a.f1` należy „ściągać” z sieci INTERNETU korzystając np. z Netscape’a. Oto adresy odpowiednich URLi:

```
http://flox.kiam1.rssi.ru/INTERBALL/latest2.html    oraz
http://www.iki.rssi.ru/vprokhor/indap/a.f1
```

Należy sprawdzić, gdzie są świeższe dane. Zarówno plik `latest2.txt` (uwaga: nie `latest2.html`) jak i `a.f1` nadają się wprost do użycia przez program `vect` z opcją `e` (korzystanie z elementów orbity) lub `v` (domyślna). W przypadku pliku `a.f1` dla celów edytorskich można dodatkowo wykonać:

- konwersję do DOS’a: `dos2unix a.f1 x.f1` (nie `unix2dos!`) i zmienić nazwę pliku `x.f1` na `a.f1`.
- plik `a.f1` zawiera wszystkie wyznaczone wektory od początku misji INTERBALL 2. W razie konieczności przeprowadzenia obliczeń orbitalnych z wektorem innym niż ostatni, z pliku `a.f1` możemy wyciąć dowolny wektor tworząc nowy plik o nazwie `a.f1` lub innej.
- `vect` i wybieramy domyślną opcję `v` oraz podajemy nazwę pliku z wektorem orbity (gdy jest inna niż `a.f1`). Jeśli plik zawiera więcej wektorów niż jeden, program wykorzysta wektor ostatni.

## 4 Gdy trzeba zmodyfikować programy

W przypadku, gdy format sprowadzanych plików danych o orbicie (`latest2.txt` lub `a.f1`) albo o czasach nadawania (`str...txt`) zostanie zmodyfikowany przez stronę rosyjską (przypadek wcale nie rzadki), wspomniane programy z nich korzystające (`vect` i `telem`) okażą się bezużyteczne (wykażą błędy czytania). Możliwe są dwa podejścia do zaradzenia temu mankamentowi: (1) przeredagowanie odpowiednich fragmentów oryginalnych plików do formatu poprzedniego, albo (2) zmodyfikowanie programów w miejscach definiujących format danych wejściowych.

W celu zmodyfikowania programów najwygodniej skorzystać z `batch`’a `f77kb` znajdującego się na komputerze `euka`. Po poleceniu:

```
>f77kb telem
```

powoduje on wywołanie edytora (`NE`) z plikiem `telem.for` (pobieranym z bieżącego katalogu), a po wyjściu z edytora następuje kolejno: kompilacja (używany jest kompilator Fortranu firmy Lahey), łączenie (`link`) i wykonanie nowo powstałego programu `telem.exe`.

Błędy podczas kompilacji powodują powrót do edycji `telem.for`. Analogicznie postępujemy w przypadku potrzeby zmodyfikowania innego programu fortranowskiego, np. `f77kbvect`.

## 5 Prowadzenie obserwacji

- 15 minut przed seansem przygotować teleskop do sterowania w kabinie pod anteną: sprawdzić datę i czas systemowy na komputerze (musi być to czas uniwersalny, a nie strefowy) w przypadku rozbieżności ustawić datę i godzinę komendami DOS'a `date` oraz `time`). Włączyć zasilanie i odblokować napędy obu osi.

Przed **każdą** sesją obserwacyjną należy prócz czasu i daty w komputerze sprawdzić poprawność „bazowania” radioteleskopu, gdyż błędy (szczególnie nierzadkie błędy grube, np. wielokrotności jednego stopnia) oznaczają systematyczną utratę sygnału podczas całej obserwacji. Jeśli po włączeniu zasilania obie lampki kontrolne położenia dokładnego (dolne, które w czasie ruchu zapalają się na krótko co każdy 1° obrotu teleskopu) są zapalone, a od położenia zgrubnego (górne, pokazujące obszar w okolicy bazy) zgaszone, to oznacza że teleskop rzeczywiście stoi w położeniu bazowym.

Gdy jest inaczej, należy zablokować teleskop, przestawić sterowanie na „ręczne” (pozycja R na przełączniku przy ścianie pomieszczenia pod teleskopem) i po odblokowaniu tak przestawić teleskop (operując przełącznikiem kierunku W-Z lub N-S i pokrętłem prędkości ruchu) w danej osi, aby przy **wygaszonej lampce górnej zapaliła się lampka dolna**.

W pozycji bazowej na indykatorach powinniśmy mieć wskazania 358.635 i 51.150 (kąt godzinny i deklinacja, odpowiednio). Jeśli wskazania te są znacząco (więcej niż np. o 0.01) różne, to poprawne wartości należy wprowadzić za pomocą specjalnego przyrządu (niewielkie plastikowe pudełko z 11 klawiszami i kablem zakończonym 25 wtykowym złączem). W tym celu podłączamy ten przyrząd do gniazdka na płycie czołowej aparatury (znajdującego się tuż poniżej wskaźnika deklinacji) i wdużamy jeden z dwóch ostatnich klawiszy na pudełku (lewy od kąta godzinnego, a prawy od deklinacji).

Po zakończeniu bazowania należy przestawić sterowanie z „ręcznego” z powrotem na „automatyczne” (z pozycji R na A).

W sterowni: włączyć program sterowania anteny RT15 (`rt3` i `rt3loadplik-sterujacy`) i sprawdzić datę i czas uniwersalny.

- Jeśli są po temu warunki, włączyć także analizator widma HP i uruchomić program do panoramicznego monitorowania zakłóceń.
- Włączyć komputer zbierający dane, oscyloskop i odbiornik R137. Także na tym komputerze sprawdzić datę i czas systemowy. Jeśli występuje niezgodność z czasem uniwersalnym, ustawić czas i datę komputera.
- Sprawdzić, czy jest dostatecznie dużo miejsca na twardym dysku. Jedna sesja (seans) zajmuje ok. 30 MB przy typowej szybkości transmisji 20 kb/s (kilobitów na sekundę).
- Sprawdzić/ustawić zestrojenie odbiornika. Częstość dla telemetry cyfrowej wynosi 137.85 MHz (dla telemetry analogowej 136.6 MHz). W celu przestrojenia należy przyciskiem AFC wybrać stan *OFF* (zapali się czerwona dioda), pokrętłem TUNING ustawić potrzebną częstość, ustawić AFC w stan *ON* (zielona dioda). W tym momencie nastąpi automatyczne przeszukiwanie pasma, widoczne na wyświetlaczu i na wskaźniku

dostrojenia. Teraz należy ustawić typ modulacji przyciskiem DEM w stan *PM*, oraz pasmo przenoszenia przyciskiem IF-BW na wartość *250 kHz*.

- Zwrócić uwagę na fakt, że w jednym z odbiorników przy paśmie 250 kHz strojenie nie odbywa się: należy wtedy zmienić pasmo (IF-BW) na 60 kHz. Gdy pojawi się sygnał, przejść na pasmo 250 kHz.
- Gdy pojawi się sygnał z satelity, dioda PLL zaświeci się. W tym stanie wskaźnik dostrojenia powinien się ustabilizować na wartości 0. Jeśli to nie nastąpi, trzeba ręcznie pokrętelem TUNING ustawić go na wskazanie 0.
- Indykacja CNTRL powinna być w stanie *LOCAL* (zielona dioda).
- Uruchomić program `tmiod.exe` w katalogu `c:\intrb\exe`. Wybrać opcję *Start*.
- Po zakończonym seansie przerwać działanie programu naciskając klawisz Esc i wyłączyć odbiornik.
- Przekopiować świeżo odebrane dane (plik `aot7...dat`, gdzie ... są dwucyfrowym numerem miesiąca roku 1997 oraz dwucyfrowym numerem dnia miesiąca) oraz zbiór informacyjny (o podobnej nazwie ale z rozszerzeniem `.lst`) z katalogu `c:\intrb\av` na serwer `trao2` do katalogu `/temp/av` (uwaga: plik `aot...dat` należy przesyłać jako *binarny*).
- Program sesji w zasadzie automatycznie powoduje ustawienie anteny w bazie, ale z pewnym opóźnieniem (na wypadek, gdyby seans się przedłużył), dlatego po zakończeniu zbierania danych należy przerwać sterowanie z pliku i ustawić antenę w bazie, a następnie zablokować ją i wyłączyć zasilanie napędów.
- Okresowo, komendą `tar rv av` wydaną z katalogu `/temp` na serwerze `trao2` zarchiwizować dane na taśmie (czynność zarezerwowana dla administratora!).

## 6 Obsługa systemu sterowania radioteleskopu RT15

### Uruchomienie programu

Przed uruchomieniem programu sterującego należy sprawdzić, czy czas systemowy danego komputera jest równy wskazaniom zegara czasu uniwersalnego i czy jest właściwa data. Jeśli istnieje rozbieżność, należy ją usunąć ustawiając czas w komputerze komendą `time`, a datę — `date`.

### Na komputerze w kabinie pod anteną

Program uruchamiany jest spod systemu DOS przez komendę `rt3`

```
rt3 [command-file]
```

Podawanie `command-file` jest opcjonalne. Program automatycznie wczytuje plik `rt3.cmd`, zawierający podstawowe komendy potrzebne do uruchomienia systemu. Po uruchomieniu program oczekuje na impuls minutowy, po czym w pełni instaluje się przez około 15 sekund. Następnie należy włączyć zasilanie i odblokować napędy obu osi radioteleskopu. Po zakończeniu obserwacji i ustawieniu teleskopu w pozycji bazowej trzeba zablokować napędy i wyłączyć ich zasilanie.



## Na komputerze w sterowni (trao2)

Należy się wlogować na jeden z komputerów UNIXowych jako użytkownik `oper` (trzeba znać niejawne hasło!). Program sterujący uruchamiamy w oknie `hpterm` poleceniem:

```
rt3
```

(bez argumentów). Pozwala to na sterowanie anteną z klawiatury. W celu wprowadzenia komend sterujących ze zbioru o nazwie *plik* piszemy `cd /temp/oper/rt3cmd` (w tym katalogu powinien znajdować się ten zbiór komend), a następnie:

```
rt3load plik
```

Poprawne byłoby też polecenie `rt3load /temp/oper/rt3cmd/plik` wydane z innego katalogu.

Jeśli zachodzi potrzeba przerwania wykonywania komend z pliku należy w oknie `hpterm` napisać `<Ctrl>c` i dalej sterować ręcznie lub ponownie uruchomić czytanie innego pliku. Zakończenie wykonywania programu `rt3` następuje po komendzie `1` wydanej w oknie RT3 Command Module.

## Komendy sterowania

### *Sterowanie ruchem anteny*

`ps`  $\alpha$   $\delta$  – śledzenie źródła o rektascensji  $\alpha$  i deklinacji  $\delta$  [deg lub h/d m s]  
`po`  $\Delta\alpha$   $\Delta\delta$  – *offset* pozycji  $\alpha$   $\delta$  w stopniach [ $-9.9 \div 9.9$ ]  
`pop`  $r$   $\varphi$  – *offset* w postaci biegunowej: odjazd na odległość  $r$  pod kątem  $\varphi$   
`pos`  $n$  – przejazd z pozycji *offset* do pozycji  $-offset$  w czasie  $n$  minut  
`pod` – wyłączenie *offset*-u pozycji  
`pp`  $HA$   $\delta$  – najazd na pozycję ( $HA$ ,  $\delta$ ) wyrażoną w stopniach  
`pb` – pozycja bazy; to samo co `pp 358.635 51.150`  
`pser` – pozycja serwisowa; to samo co `pp 88.240 -22.160`  
`psd` – anulowanie powyższych komend  
`epoch rok` – epoka współrzędnych polecenia `ps`

### *Zbieranie i prezentacja danych*

Dane z odbiornika mocy całkowitej gromadzone są w katalogach o nazwach złożonych z numeru dnia i numeru miesiąca (np. 10VIII – 10 sierpnia) i w zbiorach o nazwach złożonych z numeru godziny i rozszerzenia `.dat` (np. 03.dat)

`addch`  $n$  – włączenie  $n$ -tego kanału A/D  
`rmch`  $n$  – usunięcie  $n$ -tego kanału A/D  
`samp`  $m$  – ustalenie liczby  $m$  próbek na sekundę  
`inf`  $m$  – zapis do zbioru informacji o czasie i położeniu anteny co  $m$  minut  
`sc`  $n$  – zmiana skali wykresu mocy całkowitej  $n$  razy  
`of`  $V$  – wprowadzenie przesunięcia sygnału na wykresie o  $V$  woltów (*offset*)

## Operacje na zbiorach

<code>lf zbior</code>	– otwarcie zbioru roboczego
<code>sa</code>	– zapis danych ze skanu po częstotliwościach do zbioru roboczego
<code>sap</code>	– zapis profilu pulsara lub skanu po położeniu do zbioru roboczego
<code>cofi zbior</code>	– otwarcie zbioru z komendami sterowania
<code>end</code>	– zamknięcie zbioru z komendami sterowania

*Inne komendy* (komendy `v . . .` są przeznaczone tylko dla osób upoważnionych!)

<code>comment napis</code>	– wprowadzenie napisu na ekran
<code>he</code>	– pomoc
<code>vh V</code>	– ustawienie napięcia $V$ na przetworniku D/A kąta godzinnego
<code>vd V</code>	– ustawienie napięcia $V$ na przetworniku D/A deklinacji
<code>vr V</code>	– ustawienie napięcia $V$ na przetworniku D/A kąta godzinnego do śledzenia

## Zbiory z komendami

Program może być sterowany automatycznie za pomocą komend podanych z pliku. Zbiór oprócz komend może zawierać daty i godziny ich wykonania. Jeżeli ich nie ma program natychmiast wykonuje wszystkie komendy. Wykona je także, jeśli podany czas będzie wcześniejszy od aktualnego. Po wczytaniu zbioru komend program w dalszym ciągu przyjmuje komendy z klawiatury. Każdy zbiór musi kończyć się komendą `end`.

Przykład:

```
lo 1820e6
lf test.dat
psrp 0.2530
24/08/1995 10:35:00 ps 125.32 45.69
24/08/1995 10:40:00 psr 120
24/08/1995 12:43:00 sap
pb
end
```

**UWAGA !** Data i godzina muszą być podane dokładnie w takim formacie, jak w przykładzie.

### Porady praktyczne na temat zbiorów z komendami

- Po komendzie `ps` należy dać teleskopowi czas na dojazd na ustaloną pozycję. Na początek może to być 5 do 7-miu minut (teleskop w tym czasie powinien wszędzie zdążyć). Czas ten można skrócić po zmierzeniu czasu koniecznego na dojazd.
- Początek ustawiania pozycji może być ustawiony na czas wcześniejszy niż moment wywołania zbioru. Nie należy jednak przesadzać.
- Komenda `end` może być wydana z klawiatury. Także wtedy kończy wykonywanie komend ze zbioru.
- Przy nazwie zbioru można wywołać inny dysk, zbiór musi być jednak na katalogu głównym tego dysku.

## 7 Dodatek 1: Aparatura odbiorczo-pomiarowa

### RT15 - współrzędne i parametry

Współrzędne podane w załączonej tabelce odnoszą się formalnie do międzynarodowego układu ziemskiego ITRF (IERS Terrestrial Reference Frame). Wskazują one na punkt na osi biegunowej teleskopu, przez który przechodzi płaszczyzna prostopadła do tej osi i zawierająca oś deklinacji. Przyjmując elipsoidę IAU (wielka półoś 6378140 m i spłaszczenie 1/298,257) dostaje się dla tego referencyjnego punktu anteny:

$x$ , składowa równikowa na $\lambda = 0^\circ$	3638609,62 $\pm$ 0,19 m
$y$ , składowa równikowa na $\lambda = 90^\circ$ E	1221773,23 $\pm$ 0,54 m
$z$ , składowa biegunowa ( $\phi = 90^\circ$ )	5077024,50 $\pm$ 1,66 m
Promień wodzący	6364619,98 $\pm$ 1,42 m
Długość geograficzna (wschodnia)	18°33'39,72" $\pm$ 0,03"
Szerokość geocentryczna	52°54'37,93" $\pm$ 0,03"
Szerokość geodezyjna	53°05'43,79" $\pm$ 0,03"
Wysokość nad elipsoidą	112,35 $\pm$ 1,43 m

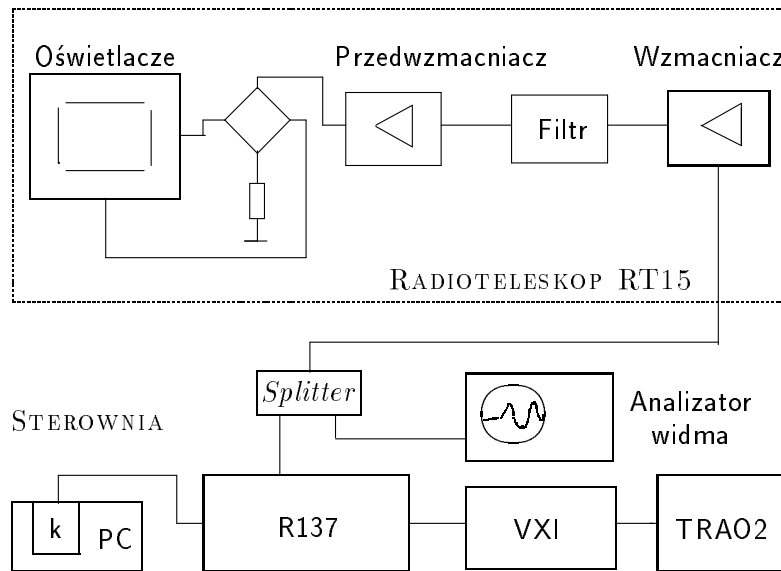
#### Parametry 15-metrowego radioteleskopu

Typ .....	sterowalna antena paraboliczna
Montaż .....	równikowy
Optyka .....	ognisko pierwotne
Średnica reflektora .....	$d$ 15,0 m
Odległość ogniskowa .....	$f$ 4,9650 m
Stosunek ogniskowej do średnicy .....	$f/d$ 0,331
Głębokość czaszy .....	$H = d^2/(16f)$ 2,8323 m
Kąt rozwarcia .....	$2\Theta_o = 4 \arctg[d/(4f)]$ 148,2531 °
Powierzchnia całkowita .....	$8\pi f^2[\cos^{-3}(\Theta_o/2) - 1]/3$ 199,92 m <sup>2</sup>
Apertura (powierzchnia zbierająca) .....	$\pi d^2/4$ 176,71 m <sup>2</sup>
Równanie paraboli <sup>†</sup>	$r = 2\sqrt{f(f-z)} = 2f \operatorname{tg}(\Theta/2)$
Odległość: ognisko – oś deklinacji .....	5,9097 m
— oś deklinacji – oś biegunowa .....	3,250 m
Nachylenie osi biegunowej .....	53°5'43"
Największa wysokość (bez pojemnika ogniskowego) .....	17,478 m
Dokładność powierzchni reflektora (rms) .....	2 mm
— ustawienia kierunku .....	0,01 °
Szybkość ruchu (obie osie) .....	0,25 ÷ 25°/min
Zakres ruchu w osi deklinacji .....	-32 ÷ +97 °
— — — biegunowej .....	-6 <sup>h</sup> 12 <sup>m</sup> ÷ +6 <sup>h</sup> 12 <sup>m</sup>
Przyspieszenie (średnio) .....	~1 °/s <sup>2</sup>
Całkowity ciężar (bez balastu w przeciwwadze) .....	45 t
Ilość paneli (24 + 4 + 1) .....	29

<sup>†</sup>  $r$  jest odległością od osi symetrii radioteleskopu, wzdłuż której mierzona jest współrzędna  $z$ , poczynając od ogniska paraboloidy i dodatnio w kierunku czaszy.  $\Theta$  jest kątem pomiędzy osią  $z$  i promieniem wodzącym.

## System odbiorczy

Na poniższym schemacie zaznaczono główne elementy toru przetwarzania odbieranego sygnału: od oświetlacza (składającego się z dwóch par dipoli o przeciwnych polaryzacjach) znajdującego się w ognisku RT15, do komputera (PC) ze specjalną kartą *interface'u* (k), oraz komputera *trao2*, gdzie rejestruje się poziom sygnału analogowego (z kanału AM odbiornika R137), znajdujących się w pomieszczeniu sterowni Katedry.



Schemat blokowy aparatury toruńskiej stacji telemetrycznej

Dostępny radioteleskop jest w zasadzie przeznaczony do obserwacji na wyższych częstotliwościach (ze względu na wzajemną bliskość czterech nóg wsporczych oświetlacza), dlatego szczególnej uwagi wymagał wybór typu oświetlacza. Po przeprowadzeniu wstępnych oszacowań i pomiarów z zastosowaniem różnych typów oświetlacza ostatecznie został zbudowany oświetlacz składający się z dwóch par wzajemnie prostopadłych par dipoli półfalowych umieszczonych nad płaskim reflektorem. Do wzmacniacza doprowadzany jest sygnał o jednej z dwóch polaryzacji kołowych (wybór oparto na bezpośrednich pomiarach sygnału z satelity) otrzymywanych za pomocą układu hybrydowego.

O własnościach szumowych systemu odbiorczego (a zatem także o jego czułości) w największym stopniu decyduje pierwszy stopień wzmocnienia (przedwzmacniacz). Wykonany w Katedrze przedwzmacniacz ma następujące własności:

Częstość środkowa	136.6 MHz	Wzmocnienie	88 dB
Pasma przenoszenia	10 MHz	Temperatura szumowa	100 K

Kalibracje na radioźródłach o znanych gęstościach strumienia promieniowania (Casiopeia A i Słońce) wykazały, że system ten ma charakterystyki (kierunkowość charakterystyki promieniowania anteny oraz czułość) zgodne z oczekiwaniami, a zatem jest optymalny.

## 8 Dodatek 2: Charakterystyki odbieranego sygnału

Sonda zorzowa, satelita Interball-2, obiega Ziemię około 4 razy na dobę. Jej orientacja w przestrzeni jest taka, że zawsze jest zwrócona osią obrotu na Słońce w celu skutecznego oświetlenia baterii słonecznych. W tym kierunku też zwrócona jest antena nadawcza telemetrii STO. Stabilność tego kierunku zapewnia obrót sondy wokół własnej osi, który odbywa się z okresem ok. 2 minut. Z powodu niesymetrycznej budowy satelity charakterystyka promieniowania jego anteny nie jest osiowo symetryczna, co sprawia, że emitowany sygnał dociera do stacji odbiorczych na Ziemi zmodulowany z częstością rotacji satelity. Jeśli oś obrotu satelity (a zatem także charakterystyka kierunkowa anteny nadawczej) jest w przybliżeniu skierowana na stację odbiorczą, to obserwuje się najmniejszą modulację spowodowaną rotacją satelity oraz nasilniejszy sygnał (maksimum charakterystyki kierunkowej). W miarę, jak powiększa się kąt  $\beta$  między kierunkami satelita–stacja i satelita–Słońce, odbierany sygnał telemetryczny słabnie i ulega coraz głębszej modulacji z okresem  $\sim 2$  minut. W czasie 3-godzinnego seansu telemetrii kąt ten zmienia się w dużych granicach, np. o  $100^\circ$ . Przy kątach  $\beta$  większych od ok.  $90^\circ$  modulacja ta sięga nawet 100 % sygnału. W praktyce oznacza to proporcjonalną stratę danych.

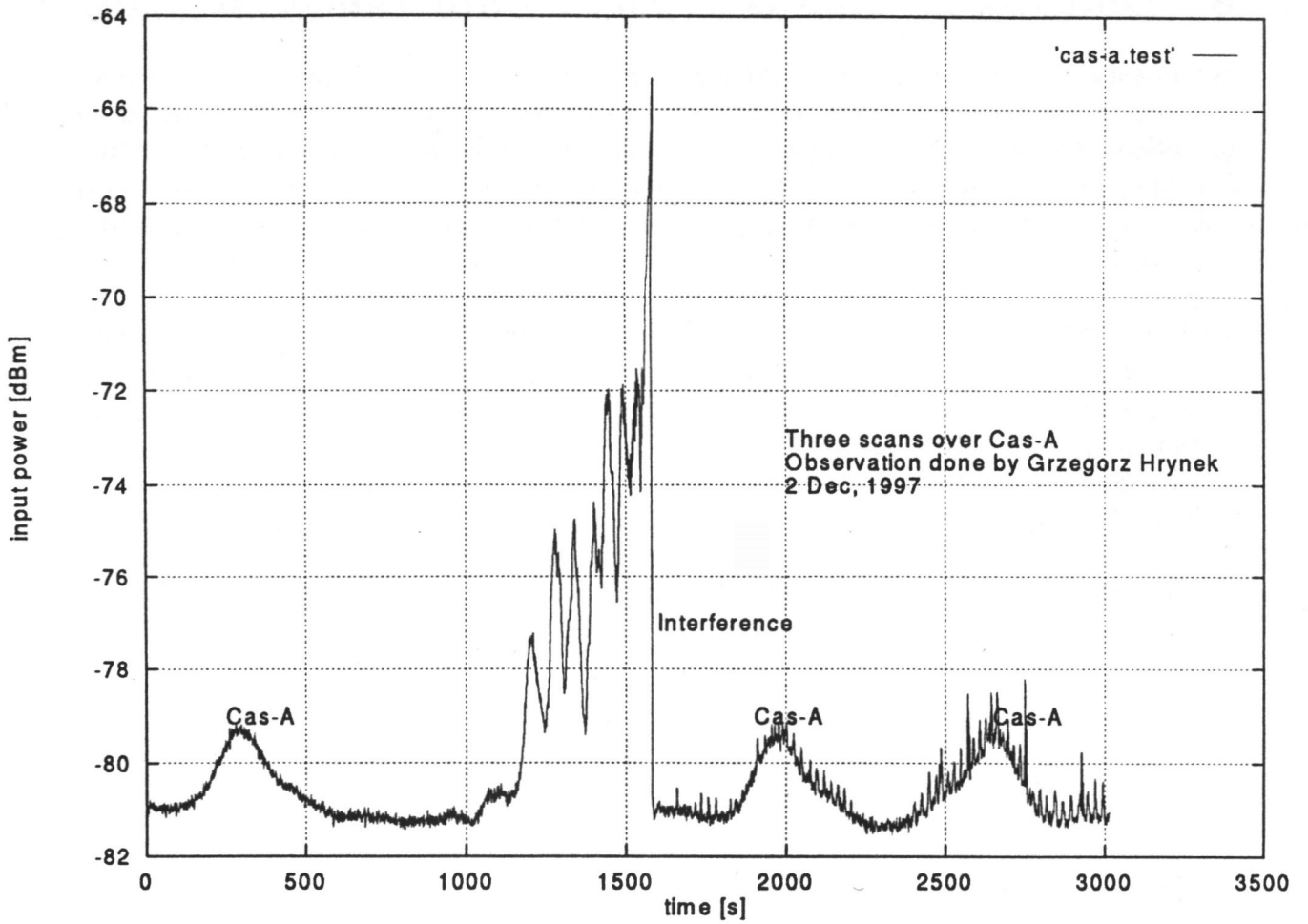
### Objaśnienia do rysunków

Rys. 1. Zapis kontrolnej kalibracji systemu na radioźródle Cassiopea A. Dobrze widać silne zakłócenia, które uniemożliwiły zaobserwowanie jednego z przejazdów (*skanów*) przez źródło, oraz słabsze ale narastające z czasem ku końcowi obserwacji.

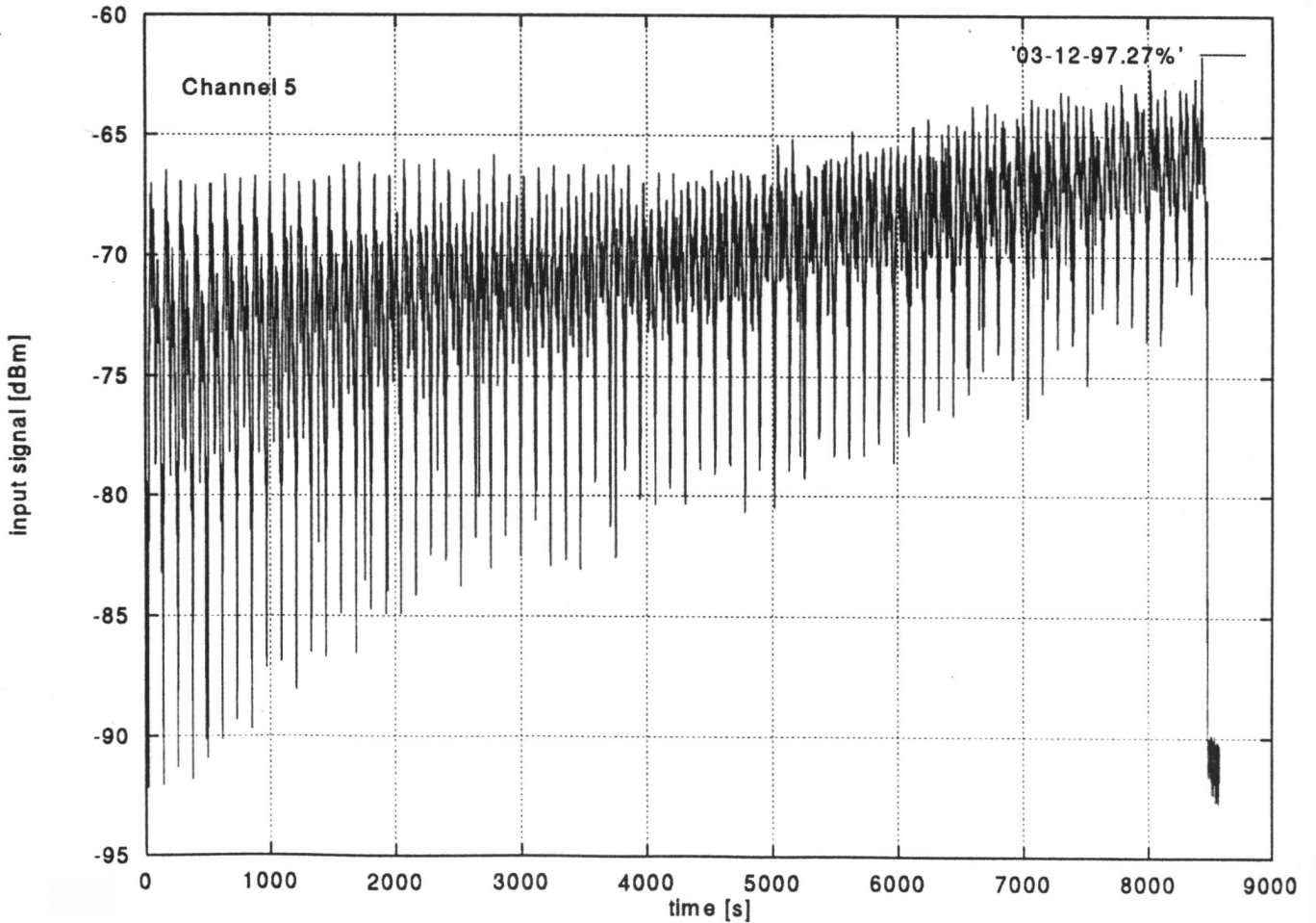
Rys. 2. Przebieg sygnału w kanale AM odbiornika R137 podczas seansu z dnia 3 grudnia 1997 r. Widać tutaj przez cały czas silną modulację pochodzącą od rotacji satelity. W seansie tym straty wyniosły 27 % danych telemetrycznych.

Rys. 3. Wyniki analizy telemetrycznych danych cyfrowych z równoległego zbierania za pomocą dwóch różnych kart *interface*'u w dniu 20 marca 1998 r. Na dolnym wykresie (b) zaznaczono dodatkowo przebiegi kąta  $\beta$  (od ok.  $35^\circ$ , do ok.  $135^\circ$ ) oraz odległości ze stacji do satelity (od ok. 8500 km do ok. 21000 km). Skala pionowa pokazuje bieżący procent bezbłędnie odebranych danych (w postaci 512-bajtowych ramek). Również na tych rysunkach dobrze widać efekty związane z brakiem symetrii osiowej charakterystyki anteny nadawczej na satelicie. Zwraca uwagę duża różnica w jakości danych zapisanych różnymi kartami *interface*'u (36.8 i 24.4 %, odpowiednio, strat danych w całym seansie).

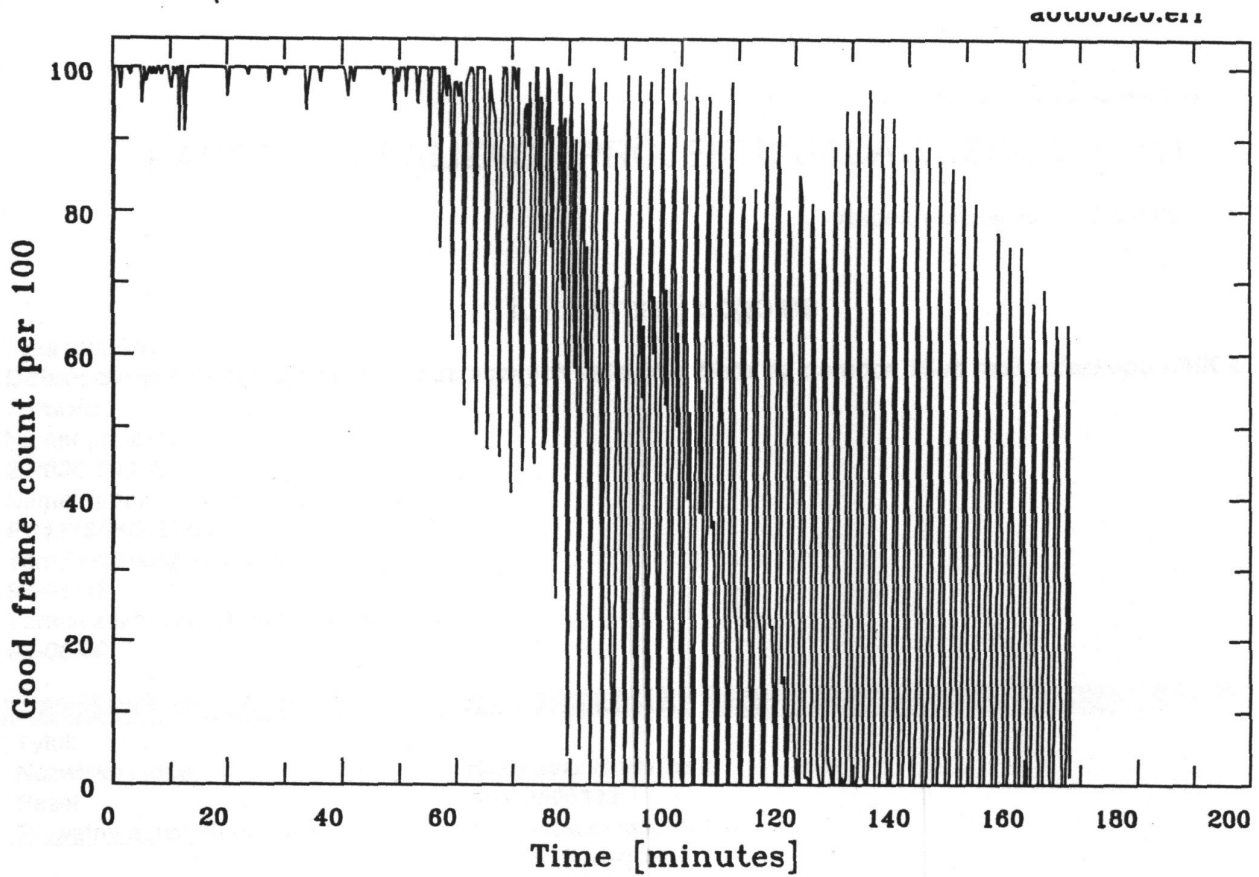
Rys. 1.



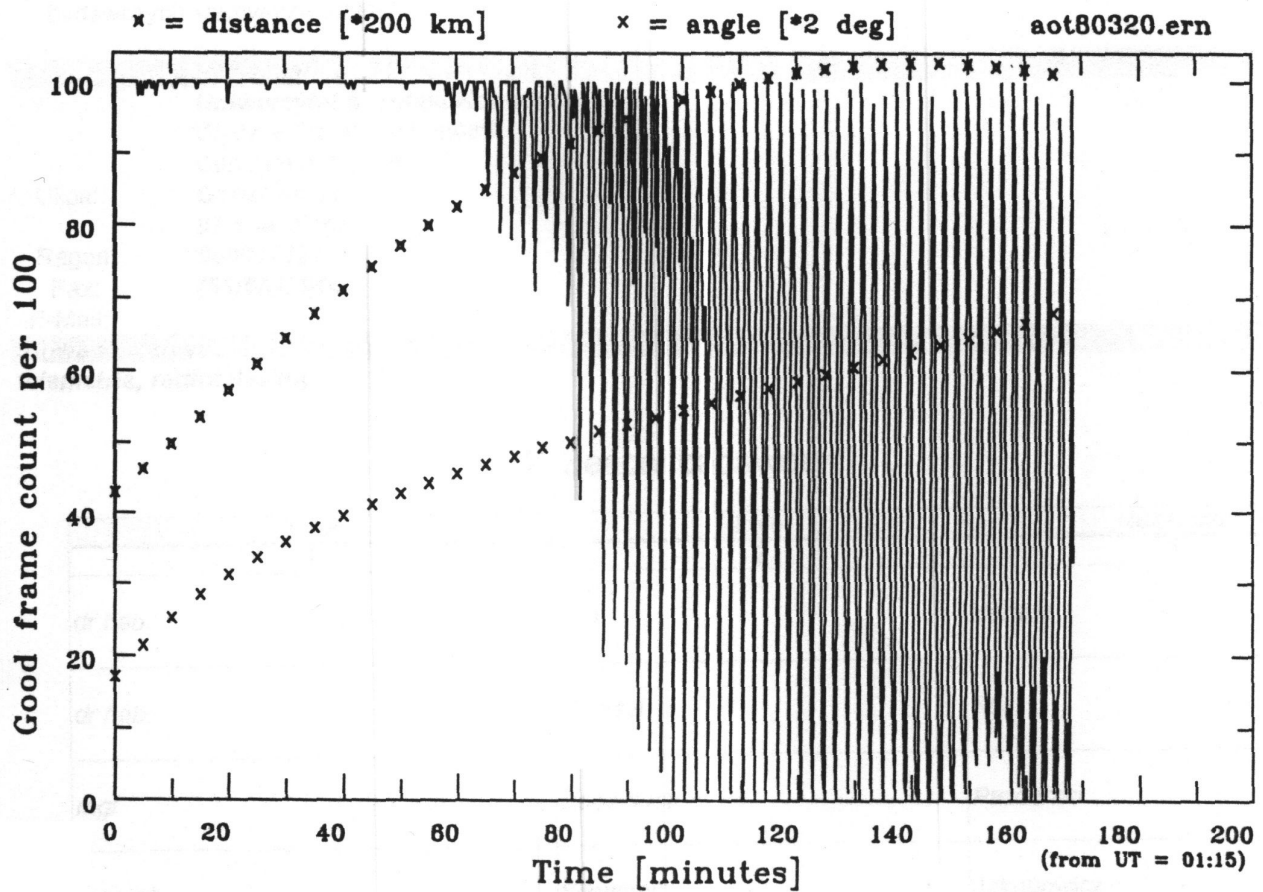
Rys. 2.



Rys. 3a



Rys. 3b.



## 9 Uzupełnienie: Wydruk programu vect

(Niniejsze **Uzupełnienie** nie znajdowało się w oryginalnej wersji raportu!  
Korzystanie z tego oprogramowania opisano od str. 3. raportu.)

```
program VECT
c Oblicza współrzędne topocentryczne satelity INTERBALL (Aurorar)
c na zadana date i godziny, oraz tworzy plik sterujacy dla RT15

implicit real*8 (a-h,o-z)
logical log
character*8 name*60,day,rec*80,rec2*80,tyg(7)*3
data tyg/'Poniedziałek','Wtorek','Środa','Czwartek','Piątek',
*'Sobota','Niedziela'/
c data proc/'Scan (RA-D)','RA-D','HA-D','Test (RA-D)'/
dmss(r)=1d4*(r-dmod(r,1.d0)*.4d0)-dmod(r*60.d0,1.d0)*40.d0
dms(r)=dsign(dmss(dabs(r)+.1d-5/3600d0),r)
pi=3.141592653589793d0
deg=180d0/pi
HALim1=273.1/deg ! w progr. ster. <273 - no action
HALim2=86.5/deg ! " >87 "
HAwait=40/deg
ALTlim=-15./deg ! nie -5, gdyz bywa silny sygnal
step= 5.d0 ! step (in min's)
RAdot= -.3612147d0/deg
argdot=+.0178285d0/deg

print*,' Elements (e) or vector (v) as input [ENTER=v]: '
read(*,'(a)') day
if(day(1:1).ne.'e') then
call vector(a,e,ai,RAaN,argp,p,EpochP,EpochD,No,ido,imo,iyo)
else
call latest2(a,e,ai,RAaN,argp,p,EpochP,EpochD,No,ido,imo,iyo)
endif

print*
ifo=1
15 print'(a,i1,a)',
*' Output file (1=scan, 2=Ra-d, 3=Ha-d, 4=Test, 5=Data, 6=Vis, 7=*)
*[ENTER=',ifo,']: '
read(*,'(a)') rec
if(rec.eq.' ') go to 19
read(rec,*) ifo
go to 15

19 call date(day)
read(day,'(i2,2(1x,i2))') imies,iday,irok1
21 print'(a,i2.2,1h,,i2,1h,,i2,a)',
*' Date (year, month, day) [ENTER=',irok1,imies,iday,']: '
```



```

read(*,'(a)') rec
if(rec.eq.' ') go to 29
read(rec,*) irok1,imies,iday
go to 21

29  irok=1900+irok1
    jd0=jd(irok,imies,iday,0)
    if(ifo.eq.6) go to 6

3   print'(a,f4.1,a)',
*' Ephemeris calculated every [ENTER=',step,'] minutes: '
read(*,'(a)') day
if(day.eq.' ') go to 5
read(day,*) step
go to 3

5   rec='plan.tex'
    rec2=' '
    inquire(file=rec,exist=log)
    if(.not.log) go to 59
    open(3,file=rec,status='old')
    idt=mod(jd0,7)+1
55  read(3,'(a)',end=58) rec
    if(rec(1:3).ne.tyg(idt).or.rec(7:7).ne.'-') go to 55
    read(rec(5:9),'(i2,1x,i2)') ima,ida
    if(ida.ne.iday.or.imies.ne.ima) go to 55

c Petla kolejnych sesji wg pliku 'plan.tex'
56  read(rec(5:23),'(2(i2,1x),2(f5.2,3x))') imies,ida,stahm,stohm
    iday=ida
54  jd0=jd(irok,imies,iday,0)
    idt=mod(jd0,7)+1
cc57 read(rec(11:15),*) stahm
cc   read(rec(18:23),*) stohm
    idoubl=1
52  read(3,'(a)',end=44) rec2
    if(rec2(1:3).eq.tyg(idt)) idoubl=2

c Czy rekord zawiera takie dane
do 51 i=1,7
    if(rec2(1:3).eq.tyg(i)) go to 4
51  continue
    go to 52
44  rec2=' '
    go to 4
58  print*, ' No suitable data in the STO (plan.tex) file!'
59  stohm=24.

```

```

c Czas uniwersalny (Universal Time of start and stop)
4   print'(i3,1h/,i2.2,1h/,i4,a,f5.2,1h,,f5.2,a)',iday,imies,irok,
   *' --> Start, stop time (UTC hh.mm) [Enter=',stahm,stohm,']: '
   read(*,'(a)') rec
   if(rec.eq.' ') go to 6
   read(rec,*) stahm,stohm
   go to 4

6   ida=iday
   if(idoubl.eq.2) ida=-ida
cc   print*,'|||:',ida,'|||',rec2,'|||'
c   call fname0(name,ida,imies,ifo)
   call fname(name,imies,ida,stahm,ifo)
   if(ifo.eq.6) write(name,'(i2.2,2(1h-,i2.2),a)') mod(irok,100)
   *,imies,iday,'.pre'
7   do 8 i=60,1,-1
   if(name(i:i).ne.' ') go to 9
8   continue
9   print'(a,a,a)', ' Filename [ENTER=',name(1:i),']: '
   read(*,'(a)') rec
   if(rec.ne.' ') name=rec
   if(rec.ne.' ') go to 7
   if(name.eq.' ') print*,' The FILENAME must be nonempty! Try again'
   if(name.eq.' ') go to 9

   open(1,file=name)
   rok=irok+(jd0-jd(irok,1,0,0))/365.25
   if(ifo.eq.5) write(1,'(a,i5,a,2(i2.2,1h/),i4,1h)//a)')
   *' INTERBALL track based on data of orbit No',No,' (' ,ido,imo,iyo,
   *' Date      UTC  ALT/km Fi=Dec Long RA(T) HA(T) Dec(T) Wys  Azym
   *' D/km  Angl'
   if(ifo.lt.5) write(1,'(a,2(i2.2,1h/),i4/a,f8.2)')
   *'comment ',ido,imo,iyo,'epoch',rok
   if(ifo.eq.6) write(1,'(a/a,i5,a,2(i2.2,1h/),i4,1h)/a/a//a/a)')
   *' INTERBALL AV visibility at Torun (Poland)',
   *'based on data of orbit No',No,' (' ,
   *ido,imo,iyo,
   *'(Aspect is the angle: station-satell.-Sun,',
   *'Altit.=antenna elevation, and HA=hour angle)',
   *'Orbit Date      UTC Aspect Altit.  HA',
   *' No  Year M D    h m s    deg    deg    deg'

   sta=int(stahm)+amod(stahm,1.d0)/.60
   if(ifo.lt.5) sta=sta - 20./60 ! 20 min. early start
   sto=int(stohm)+amod(stohm,1.d0)/.60
   if(sto.lt.sta) sto=sto+24.
   dje=jd0-.5d0+sto/24.

```

```

        if(ifo.eq.6) dje=dje+29 ! predykcja na 30 dni
        dj=jd0-.5d0+(sta-step/60.)/24

1      dj=dj+step/(60.*24)
        uth=dmod(dj+.5d0,1.d0)*24.
        call DATA(int(dj+.5d0),irok,imies,iday,0)
        irok1=mod(irok,100)

        aM=2*pi*dmod((dj-EpochP)/p,1d0) !anomalia srednia
        call kepl(aM,a,e,V,rho,ierror) !V - anom. prawdz., rho - odleglosc
        if(ierror.gt.2) go to 1
c Korekta orbity
        delT=dj-EpochD
        Nn=No+delT/p+.5
        u=argp+V +argdot*delT
        RAaNp=RAaN +RAdot*delT

c      print*,delT,(argp+argdot*delT)*deg,RAaNp*deg

        call coord
        *(dj,ai,u,rho,RAaNp,x,y,z,ra,dec,al,rat,dect,rhot,st,altn,azt,par)
        HA=dmod(st-rat+pi*2,pi*2)
c-----
c Kat miedzy kierunkiem anten satelity i nasza antena
45      call ZIEMIA(DJ,zx,zy,zz,zr,dlu)
        xt=rhot*cos(dect)*cos(rat)
        yt=rhot*cos(dect)*sin(rat)
        zt=rhot*sin(dect)
        rs=sqrt((zx+x)**2+(zy+y)**2+(zz+z)**2)
        cosA=((zx+x)*xt+(zy+y)*yt+(zz+z)*zt)/(rhot*rs)
        if(abs(cosA).gt.1.) then
            print*, ' |cos(A)|>1 ==>',cosA
            cosA=1.
        endif
        Ang=acos(cosA)*deg
        if(ifo.eq.6) go to 96
        uthm=int(uth+.005)+amod(uth+.005,1d0)*.6
        if(dmod(dj*48+.5d0/60,1d0)*30.gt.step) go to 50 !co .5 godz.
        print '(a,f6.2,a,f7.1,i8)'
        *, ' UT[hh.mm] =',uthm, ' Angle [deg], D [km]:',Ang,nint(rhot)
c-----
50      if(ifo.ge.7)
        *write(*,'(i3.1,1h:,i2.2,f9.1,f10.1,f9.1,f10.1,f11.1,i7,3i6)')
        *int(uth),nint(amod(uth,1.)*60.),dms(rat*deg/15),dms(dect*deg),
        *rhot,dms(altn*deg),dms(azt*deg),dms(par*deg),x,y,z

        if(name.eq.' ') go to 101
        if(altn.le.ALtlim.and.ifo.lt.5) go to 101

```

```

if(HA.gt.HALim2.and.HA.lt.HALim1) then
    if(HA.gt.HALim2.and.HA.lt.pi) then
        dHA=HA-HALim2
        HA=HALim2
    else ! if(HA.lt.HALim1) then
        dHA=HALim1-HA
        HA=HALim1
    endif
print'(a,f4.1,a)', ' RT15 Hour angle limit +',dHA*deg,' deg'
if(dHA.gt.HAwait.and.ifo.lt.5) go to 101
uthm=-100.
rat=dmod(st-HA+pi*2,pi*2)
endif

c Procedury do drukowania pliku wyjsciowego
c Plik do sledzenia komenda skanowania
    if(ifo.eq.1)call outsc(irok,imies,iday,uth,step,rat,dect,rhot,ang)
c Plik z efemeryda wspolrz. rownikowych (RA-Dec)
    if(ifo.eq.2)call outRA(irok,imies,iday,uth,step,rat,dect,rhot,ang)
c Plik z efemeryda HA-Dec
    if(ifo.eq.3)call outHA(irok,imies,iday,uth,step,HA,dect)
c Do testowych zjazdow z satel.
    if(ifo.eq.4) call outt (irok,imies,iday,uth,step,rat,dect,HA,st)
c Wydruk danych do analizy
    if(ifo.eq.5) call outdat
    *(irok,imies,iday,uth,dec,al,rho,rat,dect,HA,altt,azt,rhot,ang)
c Wydruk danych o widzialnosci
96    if(ifo.eq.6)call visib(dj,uth,step,HA*deg,altt*deg,ang,Nn)

    if(uthm.eq.-100..and.ifo.lt.5)
* write(1,'(a,i2.2)') 'comment |HA-lim|',nint(dHA*deg)

101    if(dj.lt.dje) go to 1
        if(ifo.eq.6) stop

        if(ifo.ge.5) go to 102
        uthm=amod(uth+.5d0/3600,1.)*60.
        write(1,'(2(i2.2,1h/),i4,i3.1,2(1h:,i2.2),3h pb/3hend)') iday,
*imies,irok,int(uth+.5d0/3600),int(uthm),nint(amod(uthm,1.)*60)
102    close(1)
        rec=rec2
        print*,' '
        if(rec.ne.' ') go to 56
103    end

```

```

        subroutine outdat(ir,im,id,uth,d,al,r,rat,dt,H,altr,azt,rt,a)
c Generuje tabele danych o orbicie
        implicit real*8 (a-h,o-z)
        deg=57.2957795131d0
        utm=amod(uth+.3d0/3600,1.)*60.
        write(1,'(2(i2,1h/),i2.2,i3.1,2(1h:,i2.2),i6,7f6.1,i6,f6.1)')
* id,im,mod(ir,100),int(uth+.3d0/3600),int(utm),nint(amod(utm,1.)
**60),nint(r-6371.),d*deg,al*deg,rat*deg,H*deg,dt*deg
*,altr*deg,amod(azt*deg+360,360d0),nint(rt),a

        if(deg.gt.0.) return
c Test wsp. topoc. (zgodno/s/c 10 km i lepiej niz 0.1 deg)
        clat=(53+05/60d0+43.79D0/3600)/deg ! conventional latitude
        clong=(18+33/60d0+39.72D0/3600)/deg ! conventional longitude
        rz=6371
        dd2=rz*rz+rt*rt+2*rz*rt*sin(altr)
        dd=sqrt(dd2)
        cosx=(dd2+rz*rz-rt*rt)/(2*rz*dd)
        sinx=sqrt(1-cosx**2)
        sinfi=sin(clat)*cosx-cos(clat)*sinx*cos(azt)
        fi=asin(sinfi)
        along=clong+datan2(-sin(azt)*sinx*cos(clat),cosx-sin(clat)*sinfi)
        write(1,'(17x,i6,2f6.1)') nint(dd-rz),fi*deg,along*deg

        end

        subroutine outsc(irok,imies,iday,uth,step,rat,dect,rhot,ang)
c Procedura tworzy plik do sledzenia komendami skanowania (ps,po,pos)
        implicit real*8 (a-h,o-z)
        data utp/-12d0/
        deg=57.2957795131d0

        crho=crho+step
        dut=amod(uth-utp+24,24.)
        if(dut.gt.5.*step/60.or.utp.eq.-12d0) go to 99
        uts=utp
c+dut/2+.4d0/3600
        ida=iday
        if(uts.gt.24) uts=uts-24.
        if(uts.gt.uth) ida=iday-1

        dra=(-rap+rat)*deg
        rap=rap*deg
        if(abs(dra).gt.180d0) dra=dra-dsign(360d0,dra)
        dra=dra/2
        ras=dmod(rap+dra+360,360d0)
        ddec=(-decp+dect)/2

```

```

    utm=amod(uts+.5d0/3600,1.)*60.
    write(1,'(2(i2.2,1h/),i4,i3.1,2(1h:,i2.2),3h ps,2f8.3
*,a,2f7.2,i4)')
* ida,imies,irok,int(uts+.5d0/3600),int(utm),nint(amod(utm,1.)*60),
*ras,(decp+ddec)*deg
c,' prev: ',rap,(decp)*deg

    write(1,'(a,f6.3,1h ,f6.3/a,f6.2)')
*' po ',-dra,-ddec*deg,' pos',dut*60
    if(abs(dra).gt.9.9.or.abs(ddec*deg).gt.9.9)
*print*,' Too large STEP in RA or DEC for scan !!!!!'
cc if(crho.lt.15.) go to 99
    crho=0
99  if(ang.lt.100)write(1,'(a,f5.1,1h/,i5.5)')'comment',ang,nint(rhot)
    if(ang.ge.100) write(1,'(a,i5,1h/,i5.5)') 'comment',ang,nint(rhot)
    rap=rat
    decp=dect
    utp=uth

    end

    subroutine outHA(ir,im,id,uth,step,HA,dect)
c,altr,rhot)
    implicit real*8 (a-h,o-z)
c pi=3.141592653589793d0
    deg=57.2957795131d0

    ida=id
    uts=uth-step/(2*60)+.4/3600
        if(uts.lt.0) then
    ida=ida-1
    uts=uts+24.
        endif

    utm=amod(uts,1.)*60.
    write(1,'(2(i2,1h/),i4,i3.1,2(1h:,i2.2),3h pp,2f7.2)
c**/a,i7)
c,a,f5.1,a)
*' ) ida,im,ir,int(uts),int(utm),nint(amod(utm,1.)*60),
*HA*deg,dect*deg
c**,'comment ',nint(rhot)
c' km Alt =',altr*deg,
c if(sstep.eq.0.) write(1,'(a,i7,a)') 'comment',nint(rhot),' km'
c sstep=sstep+step
c if(sstep.ge.9.) sstep=0
    end

```

```

subroutine outRA(ir,im,id,uth,step,rat,dect,rhot,ang)
c,HA,altr)

implicit real*8 (a-h,o-z)
pi=3.141592653589793d0
deg=180/pi

if(ang.lt.100)write(1,'(a,f5.1,1h/,i5.5)')'comment',ang,nint(rhot)
if(ang.ge.100) write(1,'(a,i5,1h/,i5.5)') 'comment',ang,nint(rhot)
uts=uth-step/(3*60)+.5/3600 !1/3 kroku wczesniejszy najazd
ida=id
      if(uts.lt.0) then
ida=ida-1
uts=uts+24.
      endif

utm=amod(uts,1.)*60.
write(1,'(2(i2.2,1h/),i4,i3.1,2(1h:,i2.2),3h ps,2f7.2,a,2i4)')
*ida,im,ir,int(uts),int(utm),nint(amod(utm,1.)*60),
*rat*deg,dect*deg
c,' HA, Altr:',nint(HA*deg),nint(altr*deg)
c  *,rhot,dms(altr*deg),dms(azt*deg),dms(par*deg),nint(x),nint(y),z
99  end

```

```

subroutine outt(ir,im,id,uth,step,rat,dect,HA,st)

implicit real*8 (a-h,o-z)
pi=3.141592653589793d0
deg=57.2957795131d0

uts=uth-step/(2*60)+.5/3600
ida=id
      if(uts.lt.0) then
ida=ida-1
uts=uts+24.
      endif

utm=amod(uts,1.)*60.
write(1,'(2(i2,1h/),i4,i3.1,2(1h:,i2.2),3h ps,2f7.2,i4)')
*ida,im,ir,int(uts),int(utm),nint(amod(utm,1.)*60),
*rat*deg,dect*deg
c,nint(HA*deg)
c  *,rhot,dms(altr*deg),dms(azt*deg),dms(par*deg),nint(x),nint(y),z
HA=HA+0.

loff=loff+1
if(loff.ne.3) go to 99

```

```

loff=0
uti=uth+.5/3600 !dla zjazdu moment efemer.
      if(uti.gt.24.) then
ida=ida+1
uti=uti-24.
      endif
utm=amod(uti,1.)*60.
rap=rat-rap
if(abs(rap).gt.pi) rap=rap-sign(2*pi,rap)
phi=atan2(dect-decp,rap)

c Offset do kalibracji - prostop. do tracku
offset=10/deg
rap=rat-offset*sin(phi)
decp=dect+offset*cos(phi)

rap=dmod(rap+2*pi,2*pi)
HAp=dmod(st-rap+pi*2,pi*2)
if(decp.gt.pi/2) decp=2*dect-decp

write(1,'(2(i2,1h/),i4,i3.1,2(1h: ,i2.2),3h ps,2f7.2,i4)')
*id,im,ir,int(uti),int(utm),nint(amod(utm,1.)*60),
*rap*deg,decp*deg,nint(HAp*deg)

99  rap=rat
    decp=dect
    end

SUBROUTINE TOPOC(ST,RA,D,R,X0,Z0)
IMPLICIT REAL*8(A-H,O-Z)
T=ST-RA
X=R*DCOS(D)
Y=X*DSIN(T)
X=X*DCOS(T) - X0
Z=R*DSIN(D) - Z0
R=DSQRT(X*X+Y*Y+Z*Z)
D=DASIN(Z/R)
RA=ST-DATAN2(Y,X)
END

SUBROUTINE XOZO(FI,H,X0,Z0)
IMPLICIT REAL*8(A-H,O-Z)
DATA A,FREC/6378.140D0,298.257D0/
PSI=DATAN((FREC-1D0)*DTAN(FI)/FREC)
X0 =a*DCOS(PSI) + H*DCOS(FI)
Z0 =a*(FREC-1D0)/FREC*DSIN(PSI) + H*DSIN(FI)

c Wynik w km
END

```



```

subroutine coord
*(dj,ai,u,rho,RAaN,x,y,z,ra,dec,al,ra,dect,rhot,st,al,altt,azt,par)
implicit real*8 (a-h,o-z)
c wspolrz. obserwatorium (Piwnice, KRA)
pi=3.141592653589793d0
clat=(53+05/60d0+43.79D0/3600)*pi/180d0 ! conventional latitude
clong=(18+33/60d0+39.72D0/3600)*pi/180d0 ! conventional longitude
Height=112.35d-3 ! height a.s.l. [km]
call X0Z0(clat,Height,X0,Z0)
c wspolrzedne rownikowe geocentryczne
dec=asin(sin(u)*sin(ai))
ra= RAaN+atan2(cos(ai)*sin(u),cos(u))
ra=dmod(ra+2*pi,2*pi)

x=rho*cos(dec)*cos(ra)
y=rho*cos(dec)*sin(ra)
z=rho*sin(dec)

c wspolrzedne rownikowe topocentryczne
st=sid(dj,clong)
al=dmod(ra-st+clong+2*pi,2*pi)
rat=ra
dect=dec
rhot=rho
call TOPOC(ST,RAt,Dect,Rhot,X0,Z0)
rat=dmod(rat+2*pi,2*pi)
c wspolrzedne horyzontalne
alt=asin(sin(clat)*sin(dec)+cos(clat)*cos(dec)*cos(st-ra))
altt=asin(sin(clat)*sin(dect)+cos(clat)*cos(dect)*cos(st-rat))
cc azt=atan2(sin(st-ra),-tan(dec)*cos(clat)+sin(clat)*cos(st-ra))
azt=atan2(sin(st-rat),-tan(dect)*cos(clat)+sin(clat)*cos(st-rat))
par=alt-altt
end

subroutine latest2(a,e,ai,RAaN,argp,p,EpocP,Ed,No,ido,im,iyo)
implicit real*8(a-h,o-z)
logical log
dimension elm(10)
character latest*14,rec*80
c,mon(12)*3,imo*3
data latest/'latest2.txt'/
c,mon/'Jan','Feb','Mar','Apr',
c *'May','Jun','Jul','Aug','Sep','Oct','Nov','Dec'/
pi=3.141592653589793d0

c orbita nr:
c No=85

```

```

c Epoka przejścia przez węzeł wstępujący
  iyo=1996
  im=9
  ido=18
  iho=7
  mo=14
  so=24.581d0

c elementy
  a=16368.155d0      !semimajor axis, km
  e=.56240209d0     !eccentricity
c   perig=7162.671d0 !perigee radius, km
c   apog=25573.640d0 !apogee radius, km
  ai=62.78857d0*pi/180 !inclination, deg=>rad
  RAaN=.83469d0*pi/180 !RA of ascending node, deg=>rad
  argp=288.73558d0*pi/180 !argument of perigee, deg=>rad
c**  p=20840.617d0/(24*3600) !period, s=>d
     p=20843.83269d0/(24*3600) !średni period od 18 IX do 14 I 97

c Czytanie elementów orbity z pliku LATEST2.TXT (jesli jest)
14  inquire(file=latest,exist=log)
     if(.not.log) then
       print '(5a)', ' Brak pliku ', latest, '. Można liczyć dalej według da
*nych z 18 Sep ''96 lub wczytać inny plik (14zn.) [Enter=(go!)] : '
       read(*,'(a)') latest
       if(latest.ne.' ') go to 14
     else
15  open(3,file=latest)
     read(3,'(a)') rec
     if(rec(1:18).ne.'Semimajor axis, km') go to 15
16  iel=iel+1
     read(rec(26:40),*) elm(iel)
17  read(3,'(a)') rec
     if(iel.lt.10) go to 16
     if(rec(1:6).ne.'Orbit ') go to 17

c Semimajor axis, km      16374.766
  a=elm(1)
c Eccentricity           .56237119
  e=elm(2)
c Perigee radius, km     7166.070
c Apogee radius, km      25583.463
c Inclination, deg       62.79036
  ai=elm(5)*pi/180
c RAAN, deg              1.62708
  RAaN=elm(6)*pi/180
c Argument of perigee, deg 288.68258
  argp=elm(7)*pi/180

```

```

c Period, sec                20853.245
  p=elm(8)/(3600*24)
c True anomaly, deg         257.49004
c ta=elm(9)*pi/180
c Mean motion, rad/sec      .301304921D-03

c Crossing of ascending nodes (calculation is based on vector N 014):
c Orbit 076 Sep 16, 1996    03:08:01.219 UTC
c Czytanie danych orbity nastepnej z dwoch podanych
c Orbit 840 1997.03.19 10:08:19.057 UTC !od 11 Mar 97

c   read(3,'(5x,i4,2x,a3,1x,i2,2x,i4,4x,2(i2,1x),f6.3)')
c   * No,imo,ido,iyo,iho,mo,so
c   read(3,'(5x,i5,i6,1x,i2,1x,i2,2x,2(i2,1x),f6.3)')
c   * No,iyo,im,ido,iho,mo,so

c   do 18 im=1,12
c   if(imo.eq.mon(im)) go to 19
c18  continue
c   stop 9
                                     endif

19  Ed=jd(iyo,im,ido,0)+((so/60+mo)/60d0+iho)/24d0-.5d0
    Ei=2*atan(sqrt((1-e)/(1+e))*tan(argp/2))
    EpocP=Ed+p*(Ei-e*sin(Ei))/(pi+pi) !Epoka przejścia przez peric.
c*** print*, ' UTnode - UTperig [h]: ',p*(Ei-e*sin(Ei))/(pi+pi)*24,Epoch
    end

    subroutine kepl(aM,a,e,V,rho,ierror)
c  rozwiązanie równania Keplera na anomalie mimosrodowa Ei
c  przy danej sredniej (aM), polosi (a) i ekscentrycznosci (e)
    implicit real*8 (a-h,o-z)
    ierror=0
    Ei=0
    aM0=aM
1   do 10 i=1,100
    Ei1=Ei-(aM0-Ei+e*sin(Ei))/(e*cos(Ei)-1)
    if(abs(Ei1-Ei).lt.1e-7) go to 11
10  Ei=Ei1
    ierror=ierror+1
    if(ierror.gt.1) print '(a,2f9.6,i2)',
* ' Brak zbieżn. r. Keplera; M, E = ',aM,Ei,ierror
    if(ierror.lt.3) then
        aM0=aM0+2*3.141592653589793d0
        go to 1
    endif
11  Ei=Ei1
c print*, ' Ilosc iteracji, Ei: ',i,Ei

```

```

c anomalia prawdziwa i odleglosc od Ziemi
  V=2*atan(sqrt((1+e)/(1-e))*tan(Ei/2.))
  rho=a*(1-e*cos(Ei))
  end

  function JD(L,M,N,J1G0)

c Input:  L - calendar year (years BC numbered 0, -1, -2, ...)
c         M - calendar month (for January M=1, February M=2, ..., M=12)
c         N - calendar day of the month M (1 to 28/29/30/31)
c         J1G0 - to be set to 1 for Julian and to 0 for Gregorian calendar
c Output: JD - Julian Day number
c Calculates the Julian Day number (JD) from Gregorian or Julian
c calendar dates. This integer number corresponds to the noon of
c the date (i.e. 12 hours of Universal Time).
c The procedure was tested to be good since 1 March, -100100 (of both
c the calendars) up to a few millions (10**6) years into the future.
c The algorithm is based on D.A. Hatcher, Q.Jl.R.Astron.Soc. 25(1984), 53-55
c slightly modified by me (K.M. Borkowski, Post.Astron. 25(1987), 275-279).

  JD=(L+(M-8)/6+100100)*1461/4+(153*mod(M+9,12)+2)/5+N-34840408
  if(J1G0.LE.0)      JD = JD-(L+100100+(M-8)/6)/100*3/4+752
c  MJD=JD-2400000.5  ! this formula gives Modified Julian Day number
  END

  subroutine DATA(JD,L,M,N,J1G0)

c Input:  JD   - Julian Day number
c         J1G0 - to be set to 1 for Julian and to 0 for Gregorian calendar
c Output: L - calendar year (years BC numbered 0, -1, -2, ...)
c         M - calendar month (for January M=1, February M=2, ..., M=12)
c         N - calendar day of the month M (1 to 28/29/30/31)
c Calculates Gregorian and Julian calendar dates from the Julian Day number
c (JD) for the period since JD=-34839655 (i.e. the year -100100 of both
c the calendars) to some millions (10**6) years ahead of the present.
c The algorithm is based on D.A. Hatcher, Q.Jl.R.Astron.Soc. 25(1984), 53-55
c slightly modified by me (K.M. Borkowski, Post.Astron. 25(1987), 275-279).

  J = 4*JD+139361631
  IF(J1G0.LE.0)      J = J+(4*JD+183187720)/146097*3/4*4-3908
  I = MOD(J,1461)/4*5+308
  N = MOD(I,153)/5+1
  M = MOD(I/153,12)+1
  L = J/1461-100100+(8-M)/6
  END

```

```

      double precision function sid(dj,dlg)
c prawdziwy miejscowy czas gwiazdowy (w radianach!)
c dj - dzien julianski
c dlg - dlugosc geogr. w rad
      implicit real*8 (a-h,o-z)
      data pi/3.141592653589793d0/
      t=dj-2451545.D0
      sid=pi*dmod(1.55811454652d0+dmod(t+t,2d0)+t*
* (.547581870159d-2+t*(1.61549d-15-t*1.473d-24))+dlg/pi,2d0)
      call nutation(dj,dpsi,deps,eps)
      sid=sid+dpsi*dcos(eps+deps)
      if(sid.lt.0d0) sid=sid+pi+pi
      end

      subroutine nutation(dj,dpsi,deps,eps)
c oblicza nutacje w dlugosci (dpsi) i nachyleniu ekliptyki do rownika
c (deps) oraz nachylenie (eps, w radianach) na date julianska dj.
      implicit real*8 (d,e,t)
      integer*2 k,k1,k2,ide*4
      dimension da(5),k(5,106),k1(5,61),k2(5,45),ide(2,106),bd(15)
* ,be(15)
      equivalence (k(1,1),k1(1,1)),(k(1,62),k2(1,1))
      data bd/-174.2,.2,-1.6,-3.4,1.2,-.5,.1,-.1,.1,-.2,.1,-.4,0.,.1,-.1
*/ ,be/8.9,.5,-3.1,-.1,-.6,.3,0.,0.,0.,-.5,0.,0.,-.1,2*0./ ,ide/
*-171996,92025,2062,-895,-13187,5736,1426,54,-517,224,217,-95,
*129,-70,17,0,-16,7,-2274,977,712,-7,-386,200,-301,129,63,-33,
*-58,32,      46,-24,11,0,-3,1,-3,0,-2,1,1,0,48,1,-22,0,-15,9,
*-12,6,-6,3,-5,3,4,-2,4,-2,-4,0,1,0,1,0,-1,0,1,0,1,0,-1,0,
*-158,-1,123,-53,63,-2,-59,26,-51,27,-38,16,29,-1,29,-12,-31,13,
*26,-1,21,-10,16,-8,-13,7,-10,5,-7,0,7,-3,-7,3,-8,3,6,0,6,-3,-6,3,
*-7,3,6,-3,-5,3,5,0,-5,3,-4,0,4,0,-4,0,-3,0,3,0,-3,1,-3,1,-2,1,
*-3,1,-3,1,2,-1,-2,1,2,-1,-2,1,2,0,2,-1,1,-1,-1,0,1,-1,-2,1,-1,0,
*1,-1,-1,1,-1,1,1,0,1,0,1,-1,-1,0,-1,0,1,0,1,0,-1,0,1,0,1,0,
*-1,0,-1,0,-1,0,-1,0,-1,0,-1,0,-1,0,-1,0,1,0,-1,0,1,0/
      data k1/4*0,1, 4*0,2, 0,0,2,-2,2, 0,1,4*0,1,2,-2,2 ,0,-1,2,-2,2,
c0,0,2,-2,1, 0,2,4*0,2,2,-2,2, 0,0,2,0,2, 1,6*0,2,0,1, 1,0,2,0,2,
c1,3*0,1,-1,3*0,1,      -2,0,2,0,1, 2,0,-2,0,0, -2,0,2,0,2, 1,-1,0,-1
c,0,0,-2,2,-2,1, 2,0,-2,0,1, 2,0,0,-2,3*0,2,-2,0, 0,1,0,0,1,0,-1,0,
c0,1,-2,0,0,2,1,0,-1,2,-2,1,2,0,0,-2,1,0,1,2,-2,1,1,0,0,-1,0,2,1,0,
c-2,0,0,0,-2,2,1,0,1,-2,2,0, 0,1,0,0,2, -1,0,0,1,1, 0,1,2,-2,0,
c1,0,0,-2,0,-1,0,2,0,2,3*0,2,0, -1,0,3*2,1,0,2,0,1,0,0,4*2,4*0,
c1,0,2,-2,2,2,0,2,0,2,0,0,2,0,0, -1,0,2,0,1, -1,0,0,2,1, 1,0,0,-2,1
c, -1,0,2,2,1, 1,1,0,-2,0, 0,1,2,0,2, 0,-1,2,0,2, 1,0,3*2, 1,0,0,2,
c0, 2,0,2,-2,2, 3*0,2,1, 0,0,2,2,1, 1,0,2,-2,1, 3*0,-2,1, 1,-1,3*0/
c62
* , k2/2,0,2,0,1, 0,1,0,-2,0, 1,0,-2,5*0,1,0, 1,1,3*0, 1,0,2,0,0,
c1,-1,2,0,2, 2*-1,3*2, -2,3*0,1, 3,0,2,0,2, 0,-1,3*2, 1,1,2,0,2,

```

```

c-1,0,2,-2,1, 2,3*0,1, 1,3*0,2, 3,6*0,2,1,2, -1,3*0,2, 1,0,0,-4,0,
c-2,0,3*2,-1,0,2,4,2, 2,0,0,-4,0, 1,1,2,-2,2, 1,0,2,2,1, -2,0,2,4,2
c, -1,0,4,0,2, 1,-1,0,-2,0, 2,0,2,-2,1,2,0,3*2, 1,0,0,2,1, 0,0,4,-2,
c2, 3,0,2,-2,2, 1,0,2,-2,0, 0,1,2,0,1, 2*-1,0,2,1, 0,0,-2,0,1,
c0,0,2,-1,2, 0,1,0,2,0, 1,0,2*-2,0, 0,-1,2,0,3*1,0,-2,1, 1,0,-2,2,0
c, 2,0,0,2,3*0,2,4,2, 0,1,0,1,0/
df(d1,i2,d2,d3,d4)=(d1+dmod(i2*t,1d0)*360*3600+
*t*(d2+t*(d3+t*d4)))*dpi/(3600*180)
dpi=3.141592653589793d0
t=(dj-2451545d0)/36525d0
da(1)=df( 485866.733d0,1325, 715922.633d0, 31.310d0,.064d0)
da(2)=df(1287099.804d0, 99,1292581.224d0, -.577d0,-.012d0)
da(3)=df( 335778.877d0,1342, 295263.137d0,-13.257d0,.011d0)
da(4)=df(1072261.307d0,1236,1105601.328d0, -6.891d0,.019d0)
da(5)=df( 450160.28d0, -5,-482890.539d0, 7.455d0,.008d0)
dpsi=0
deps=0
cd=0
ce=0
do 3 i=106,1,-1
dfaz=0
do 1 j=1,5
1 dfaz=dfaz+k(j,i)*da(j)
if(i.gt.15) go to 2
cd=t*bd(i)
ce=t*be(i)
2 dpsi=dpsi+(ide(1,i)+cd)*dsin(dfaz)
3 deps=deps+(ide(2,i)+ce)*dcos(dfaz)
dpsi=dpsi/(10000*3600*180)*dpi
deps=deps/(10000*3600*180)*dpi
eps=df(84381.448d0,0,-46.815d0,-.00059d0,.001813d0)
end

function dtuts(y)
dimension idt(32),dt(227) ! dt(227=2006-1780+1)
c wg: FR Stephenson, LV Morrison, 1984, Phil Trans R Soc Lond A313, 47
c iy 1630 - 1775 (1625 dodalem wg rownania; 1780 z nast.)
Data idt/78, 85,72,62,54,48,43,37,32,26,21,16,12,10,9,9,9,
* 10,10,11,11,11,12,12,13,13,14,15,16,16,17, 17/
c iy 1780 - 1859
Data dt/2*16.9,17.,6*17.1,17.,16.9,16.7,16.5,16.2,15.9,15.6,15.2
*,14.8,14.4,14.1,13.7,13.4,13.1,12.9,12.7,12.6,11*12.5,12.4,12.3,
*12.2,12.,11.7,11.4,11.1,10.6,10.2,9.6,9.1,8.6,8.,7.5,7.,6.6,6.3,6.
*,5.8,5.7,3*5.6,5.7,5.8,5.9,6.1,6.2,6.3,6.5,6.6,6.8,6.9,7.1,7.2,
*7.3,7.4,7.5,7.6,2*7.7,2*7.8,
c iy 1860 - 1980
*7.88,7.82,7.54,6.97,6.4,6.02,5.41,4.1,2.92,1.82,1.61,

```

```

*.1,-1.02,-1.28,-2.69,-3.24,-3.64,-4.54,-4.71,-5.11,-5.4,-5.42,
*-5.2,2*-5.46,-5.79,-5.63,-5.64,-5.8,-5.66,-5.87,-6.01,-6.19,
*-6.64,-6.44,-6.47,-6.09,-5.76,-4.66,-3.74,-2.72,-1.54,-.02,1.24,
*2.64,3.86,5.37,6.14,7.75,9.13,10.46,11.53,13.36,14.65,16.01,17.2,
*18.24,19.06,20.25,20.95,21.16,22.25,22.41,23.03,23.49,23.62,
*23.86,24.49,24.34,24.08,24.02,24. ,23.87,23.95,23.86,23.93,23.73,
*23.92,23.96,24.02,24.33,24.83,25.3,25.7,26.24,26.77,27.28,27.78,
*28.25,28.71,29.15,29.57,29.97,30.36,30.72,31.07,31.35,31.68,
*32.18,32.68,33.15,33.59,34. ,34.47,35.03,35.73,36.54,37.43,38.29,
*39.2,40.18,41.17,42.22,43.37,44.48,45.47,46.46,47.52,48.53,49.59,
*50.54,
c iy  1981   82   83   84   85   86   87   88   89   90   91
cc   *51.38,52.17,52.95,53.79,54.35,54.87,55.50,56.20,57.00,58.00,59.0/
*51.38,52.17,52.96,53.79,54.34,54.87,55.32,55.82,56.30,56.86,57.57
cc  1992   93   94   95   96   97   98   99  2000  1   2   3
*,58.31,59.12,59.98,60.79,61.63,62.6 ,63.6 ,64.6 ,66. ,67. ,68. ,69. ,
*70. ,71. ,72./
    itab=2006 !wymiar tab. dt(ii): ii=itab+1-1780
    iy=y
    f=y-iy
    if(iy.gt.itab-1.or.iy.lt.1637) go to 2

    if(iy.lt.1780) go to 1
    dtuts=dt(iy-1779)*(1-f)+dt(iy-1778)*f
    return

1    i=(iy-1620)/5
cc dtuts=idt(i)+(iy-i*5-1619.5)*(idt(i+1)-idt(i))/5.
    dtuts=idt(i)+(y-i*5-1620)*(idt(i+1)-idt(i))/5.
    return
2    t=(y-1800)/100.
c w zasadzie wstecz tylko do ok. 700 BC
    if(iy.lt.948) dtuts=(44.3*t+320)*t+1360.
    if(iy.ge.948) dtuts=25.5*t*t ! 36 s dla ciaglosci; nizej
cTest if(iy.ge.948) dtuts=27.*t*t ! - " -

c musze odjac 36 s dla ciaglosci z danymi obserw. w XX w.
    if(iy.gt.itab-1) dtuts= dtuts-36.
    end

    SUBROUTINE ZIEMIA(DJE,dxh,dyah,dzah,dr,dlu)

C GIVEN DJE = JULIAN EPHEMERIS DATE.
C OUTPUT REFERRED TO MEAN EQUATOR AND EQUINOX OF DATE (DJE).

C WG: P. STUMPF, 1980, ASTRON. ASTROPHYS. SUPPL. SER. 41, 1-8.
    IMPLICIT REAL*8 (D)
    DIMENSION F(3),SN(4)

```

DIMENSION DCFEL(3,8),DCEPS(3),CCSEL(3),DCARGS(2,15),  
\* CCAMPS(5,15),CCSEC(3,4),DCARGM(2,3),CCAMPM(4,3)

DATA DC2PI/6.2831853071796D0/,CC2PI/6.283185/,  
\* DC1/1.0D0/,DCT0/2415020.D0/,DCJUL/36525D0/, CCIM /8.978749E-2/

C CONSTANTS DCFEL(I,K) OF FAST CHANGING ELEMENTS

C DCFEL(I,1)=L srednia dl. Slonca

DATA DCFEL/1.7400353D+00, 6.2833195099091D+02, 5.2796D-6,  
\* 6.2565836D0, 6.2830194572674D2, -2.6180D-6,  
\* 4.7199666D0, 8.3997091449254D3, -1.9780D-5,  
\* 1.9636505D-1, 8.4334662911720D3, -5.6044D-5,  
\* 4.1547339D0, 5.2993466764997D1, 5.8845D-6,  
\* 4.6524223D0, 2.1354275911213D1, 5.6797D-6,  
\* 4.2620486D0, 7.5025342197656D0, 5.5317D-6,  
\* 1.4740694D0, 3.8377331909193D0, 5.6093D-6/

C sredn. anom. Ks. 5.1680003d0, 1717915922.633"t 31.31"t\*\*2

DATA DCEPS/4.093198D-1,-2.271110D-4,-2.860401D-8/  
DATA CCSEL/1.675104E-2,-4.179579E-5,-1.260516E-7/

DATA DCARGS/5.0974222D0,-7.8604195454652D2,  
\*3.9584962D0,-5.7533848094674D2,  
\*1.6338070D0,-1.1506769618935D3,  
\*2.5487111D0,-3.9302097727326D2,  
\*4.9255514D0,-5.8849265665348D2,  
\*1.3363463D0,-5.5076098609303D2,  
\*1.6072053D0,-5.2237501616674D2,  
\*1.3629480D0,-1.1790629318198D3,  
\*5.5657014D0,-1.0977134971135D3,  
\*5.0708205D0,-1.5774000881978D2,  
\*3.9318944D0, 5.296346478D1,  
\*4.8989497D0, 3.9809289073258D1,  
\*1.3097446D0, 7.7540959633708D1,  
\*3.5147141D0, 7.9618578146517D1,  
\*3.5413158D0,-5.4868336758022D2/

DATA CCAMPS/  
\*-2.279594E-5, 1.407414E-5, 8.273188E-6, 1.340565E-5,-2.490817E-7,  
\*-3.494537E-5, 2.860401E-7, 1.289448E-7, 1.627237E-5,-1.823138E-7,  
\* 6.593466E-7, 1.322572E-5, 9.258695E-6,-4.674248E-7,-3.646275E-7,  
\* 1.140767E-5,-2.049792E-5,-4.747930E-6,-2.638763E-6,-1.245408E-7,  
\* 9.516893E-6,-2.748894E-6,-1.319381E-6,-4.549908E-6,-1.864821E-7,  
\* 7.310990E-6,-1.924710E-6,-8.772849E-7,-3.334143E-6,-1.745256E-7,  
\*-2.603449E-6, 7.359472E-6, 3.168357E-6, 1.119056E-6,-1.655307E-7,  
\*-3.228859E-6, 1.308997E-7, 1.013137E-7, 2.403899E-6,-3.736225E-7,  
\* 3.442177E-7, 2.671323E-6, 1.832858E-6,-2.394688E-7,-3.478444E-7,  
\* 8.702406E-6,-8.421214E-6,-1.372341E-6,-1.455234E-6,-4.998479E-8,  
\*-1.488378E-6,-1.251789E-5, 5.226868E-7,-2.049301E-7,0.E0,



```

*-8.043059E-6,-2.991300E-6, 1.473654E-7,-3.154542E-7,0.E0,
* 3.699128E-6,-3.316126E-6, 2.901257E-7, 3.407826E-7,0.e0,
* 2.550120E-6,-1.241123E-6, 9.901116E-8, 2.210482E-7,0.e0,
*-6.351059E-7, 2.341650E-6, 1.061492E-6, 2.878231E-7,0.e0/

```

```

DATA CCSEC/ 1.289600E-6,5.550147E-1,2.076942E0,
*3.102810E-5,4.035027E0, 3.525565E-1,
*9.124190E-6,9.990265E-1,2.622706E0,
*9.793240E-7,5.508259E0, 1.559103E1/,CCSEC3/-7.757020E-8/

```

```

DATA CCKM/3.122140E-5/
DATA DCARGM/5.1679830D0,8.3286911095275D3,
*5.4913150D0,-7.2140632838100D3,
*5.9598530D0, 1.5542754389685D4/
DATA CCAMP/
* 1.097594E-1,2.896773E-7,5.450474E-2, 1.438491E-7,
*-2.223581E-2,5.083103E-8,1.002548E-2,-2.291823E-8,
* 1.148966E-2,5.658888E-8,8.249439E-3, 4.063015E-8/

```

#### C EXECUTION

```

AU=150000000d0
DT=(DJE-DCT0)/DCJUL
T=DT
DTSQ=DT*DT
DO 100 K=1,4
DLOCAL=DMOD(DCFEL(1,K)+DT*DCFEL(2,K)+DTSQ*DCFEL(3,K),DC2PI)
IF(K.EQ.1) DML=DLOCAL
100 IF(K.NE.1) F(K-1)=DLOCAL
E=AMOD(CCSEL(1)+T*CCSEL(2)+sngl(dTSQ)*CCSEL(3),CC2PI)
DO 300 K=1,4
300 SN(K)=SIN(AMOD(CCSEC(2,K)+T*CCSEC(3,K),CC2PI))
PERTL= CCSEC(1,1)*SN(1)+CCSEC(1,2)*SN(2)+
+(CCSEC(1,3)+T*CCSEC3)*SN(3)+CCSEC(1,4)*SN(4)
PERTR =0.
DO 400 K=1,15
A=DMOD(DCARGS(1,K)+DT*DCARGS(2,K),DC2PI)
COSA=COS(A)
SINA=SIN(A)
PERTL=PERTL+CCAMPS(1,K)*COSA+CCAMPS(2,K)*SINA
400 PERTR=PERTR+CCAMPS(3,K)*COSA+CCAMPS(4,K)*SINA
G=F(1)
ESQ=E*E
TWOE=E+E
TWOG=G+G
PHI=TWOE*((1.-ESQ*.125)*SIN(G)+E*.625*SIN(TWOG))+
+ ESQ*.5416667*SIN(G+TWOG))
DPSI=(DC1-ESQ)/(DC1+E*COS(G+PHI))

```

```

D1PDRO=DC1+PERTR
DML=(-(17.1822+.01742*T)*sin(f(2)-f(3))-1.3187*dsin(DML+DML)+
*.2062*sin(2*(f(2)-f(3)))-.2274*sin(f(2)*2)+.1426*dsin(f(1))
*-20.490)/206265.+DML
c +.0712*dsin(1)-.0517*sin(f(1)+2*DML)-.0386*sin(f(2)+f(3))- .0301*sin(1+2*f(2))
DTL=DMOD(DML+PHI+PERTL,DC2PI)
dlu=dmod(dtl+dc2pi*1.5,dc2pi)
DSINLS=DSIN(DTL)
DCOSLS=DCOS(DTL)

PERTL=0.
PERTP=0.
DO 500 K=1,3
A=DMOD(DCARGM(1,K)+DT*DCARGM(2,K),DC2PI)
SINA=SIN(A)
COSA=COS(A)
PERTL=PERTL+CCAMPM(1,K)*SINA
500 PERTP=PERTP+CCAMPM(3,K)*COSA

TL=F(2)+PERTL
SINLM=SIN(TL)
COSLM=COS(TL)
SIGMA=CCKM/(1.+PERTP)
DR=DPSI*D1PDRO
FLATM=CCIM*SIN(F(3))
A=SIGMA*COS(FLATM)
DXH=DR*DCOSLS-A*COSLM
DYH=DR*DSINLS-A*SINLM
DZH=-SIGMA*SIN(FLATM)
DEPS=DMOD(DCEPS(1)+DT*DCEPS(2)+DTSQ*DCEPS(3),DC2PI)
deps=deps+(9.2025*cos(f(2)-f(3))+.5736*dcos(DML+DML)
*+.0977*dcos(f(2)+f(2))- .0895*dcos(2*(f(2)-f(3))))/206264.8d0
c +.0224*cos(f(1)+2*DML)+.0200*cos(f(2)+f(3))+.0129*cos(1+2*f(2))
DCOSEP=DCOS(DEPS)
DSINEP=DSIN(DEPS)
dr=dsqrt(dxh*dxh+dyh*dyh+dzh*dzh)
DYAH=(DCOSEP*DYH-DSINEP*DZH)*au
DZAH=(DSINEP*DYH+DCOSEP*DZH)*au
dxh=dxh*au
END

subroutine fname(name,im,id,uthm,ifo)
character*60 name,ext(10)*1
real*8 uthm
data ext/' ','R','H','t','d',5*' '/
write(name,'(2(i2.2,1h-),f5.2,a1)') im,iabs(id),uthm,ext(ifo)
if(name(7:7).eq.' ') name(7:7)='0'
end

```

```

subroutine fname0(name,ida,im,ifo)
character*60 name,mi(12)*4,ext*5,ex(5)*3
data mi/'i','ii','iii','iv','v','vi','vii','viii','ix','x','xi',
* 'xii'//,ext/' '//,ex/'cmd','RaD','HaD','tst','dat'/
name=' '
id=abs(ida)
if(ext(1:1).eq.'A') then
  ext(1:1)='B'
else
  ext='.'//ex(ifo)
endif
if(ida.lt.0) ext='A.'//ex(ifo)
if(id.lt.10) write(name,'(i1,a)') id,mi(im)
if(id.ge.10) write(name,'(i2,a)') id,mi(im)
do 1 i=8,1,-1
  if(name(i:i).ne.' ') go to 2
1 continue
2 name(i+1:i+5)=ext
end

```

```

subroutine vector(a,e,ai,RAaN,argp,p,EpochP,ED,No,ido,im,iyo)
implicit real*8(a-h,o-z)
logical log
character latest*14,rec0*25,rec(5)*36
data latest/'A.F1'/
pi=3.141592653589793d0
GMz=3.986004418D5 !km**3/s**2 IERS Conv.
l=1
lr=36

c orbita nr:
c   No=85
c   Epoka przejścia przez węzeł wstępujący
c     iyo=1996
c     im=9
c     ido=18
c     iho=7
c     mo=14
c     sec=24.581d0
c elementy
c   a=16368.155d0 !semimajor axis, km
c   e=.56240209d0 !eccentricity
c   perig=7162.671d0 !perigee radius, km
c   apog=25573.640d0 !apogee radius, km

```

```

ai=62.78857d0*pi/180 !inclination, deg=>rad
RAaN=.83469d0*pi/180 !RA of ascending node, deg=>rad
argp=288.73558d0*pi/180 !argument of perigee, deg=>rad
c**  p=20840.617d0/(24*3600) !period, s=>d
      p=20843.83269d0/(24*3600) !sredni period od 18 IX do 14 I 97

c Czytanie elementow orbity z pliku a.F1 (jesli jest)
14  inquire(file=latest,exist=log)
      if(.not.log) then
        print'(5a)', ' Brak pliku ',latest,'. Mo/zna liczyc dalej wedlug da
*nych z 18 Sep ''96 lub wczytac inny plik (14zn.) [Enter=(go!)]: '
        read(*,'(a)') latest
        if/latest.ne.' ') go to 14
          else
15  open(3,file=latest)
c    read(3,'(a)',end=100) rec
c    if(rec(1:8).ne.'INT/IAZ/') go to 15
c    read(rec(9:24),'(3x,i4,1x,i2,1x,i5)') MesNo,MesVer,No
c    MesNo= MesNo+0 ! by kompilator nie krzyczal
c    MesVer= MesVer+0 ! ditto
c    read(3,'(a)',end=100) rec
c    read(rec,'(3i2,1x,3i2,i3)') ido,im,iyo,iho,mo,iso,ms
c    iyo=iyo+1900
c    read(3,'(a)',end=100) rec
c    read(rec,'(3(f11.3,1x))') x,y,z
c    read(3,'(a)',end=100) rec
c    read(rec,'(3(f11.3,1x))') vx,vy,vz
151  read(3,'(a)',end=10) rec0
      if(rec0(1:8).ne.'INT/IAZ/') go to 151
      rec(1)=rec0
      do 16 j=2,5
16  read(3,'(a)',end=10) rec(j)(1:1r)
      go to 151

10  do 11 j=1,5
      do 11 i=1,1r
        if(ichar(rec(j)(i:i)).lt.20) rec(j)(i:i)=' '
11  if(rec(j)(i:i).eq.'/') rec(j)(i:i)=' '
      print*, ' Ref. vector/orbit: '//rec(1)(1:25)//' of '//rec(2)(1:17)
c//'| '//rec(3)(1:1r)//
c  *'| '//rec(4)(1:1r)//'| '//rec(5)(1:1r)//'| '
      read(rec(1)(9:25),*) NoFor,MesNo,MesVer,No
      MesNo= MesNo+0 ! by kompilator nie krzyczal
      MesVer= MesVer+0 ! ditto
      NoFor= NoFor+0 ! ditto
      read(rec(2),*) idmyo,ihmsmo
      ido=idmyo/10000
      im=mod(idmyo,10000)/100

```

```

iyo=mod(idmyo,100)+1900
iho=ihmsmo/10000000
mo=mod(ihmsmo,10000000)/100000
sec=mod(ihmsmo,100000)/1000.d0
read(rec(3),*) x,y,z
read(rec(4),*) vx,vy,vz

c Format danych F1
cINT/IAZ/01/0054/01/00634/
c280197/210000000/
c-002550.903/+016488.275/+018294.531/
c-1.95658519/+0.68048275/-1.88450451/
c5874/ZZ

c=dsqrt((y*vz-z*vy)**2+(z*vx-x*vz)**2+(x*vy-y*vx)**2)
r=dsqrt(x*x+y*y+z*z)

c Semimajor axis, km          16374.766
a=1d0/(2/r-(vx*vx+vy*vy+vz*vz)/GMz)
c Period, sec                 20853.245; ni/zej w dobach
p=(pi+pi)*dsqrt(a*a*a/GMz)/(3600*24)
c Eccentricity                .56237119
pm=c*c/GMz
e=dsqrt(1d0-pm/a)
c Inclination, deg           62.79036
ai=dacos((x*vy-y*vx)/c)
c RAAN, deg                  1.62708 - Omega=dlug.wezla wstep.
RAAN=datan2(y*vz-z*vy,-z*vx+x*vz)
c True anomaly, deg         257.49004
Ae=datan2(x*vx+y*vy+z*vz,dsqrt(a*GMz)*(1-r/a))
ta=2*datan(dsqrt((1+e)/(1-e))*dtan(Ae/2))
c Anomalia srednia
aM=Ae-e*dsin(Ae)
c Argument of perigee, deg  288.68258
argp=datan2(z,dsin(ai)*(x*dcos(RAaN)+y*dsin(RAaN)))-ta
endif

c19 ED=jd(iyo,im,ido,0)+(((iso+ms*1d-3)/60+mo)/60d0+iho)/24d0-.5d0
19 ED=jd(iyo,im,ido,0)+((sec/60+mo)/60d0+iho)/24d0-.5d0
EpochP=ED-aM/(pi+pi)*p
c print*, 'a,e,i,p,argp,ta', a,e,ai*180/pi,p,argp*180/pi,ta*180/pi
return
c100 print*, ' End of the vector file '//latest
end

```

```

subroutine visib(dj,uth,step,HA,alt,a,No)

implicit real*8 (a-h,o-z)
data H Amin,H Amax/270.,90./,altmin/7./,index/1/

      if(index.gt.0) then
c      print*, 'Poczatek, HA alt...',ha,alt
      if(alt.lt.altmin) go to 100
      if(HA.gt.HAmax.and.HA.lt.HAmin) go to 100
          if(index.eq.2) go to 2
          index=2
          step=.5
          dj=djp
          return
2      step=10
          call DATA(int(dj+.5d0),ir,im,id,0)
          print'(a,i5,2i3,i4)', 'Date (Y M D), Orbit No:',ir,im,id,No
          utm=amod(uth+.3d0/3600,1.)*60.
          write(1,'(i4,i5,2i2.2,i5,2i2.2,3f7.1)')
          * No,ir,im,id,int(uth+.5d0/3600),int(utm),nint(amod(utm,1.)*60),a
          *,alt,HA
          index=-1
              else
          if(alt.ge.altmin.and.(HA.lt.HAmax.or.HA.gt.HAmin)) go to 100
              if(index.eq.-2) go to 12
              index=-2
              step=.5
              dj=djp
              return
12      step=10
          call DATA(int(djp+.5d0),ir,im,id,0)
          utm=amod(utp+.3d0/3600,1.)*60.
          write(1,'(6h end,3x,2i2.2,i5,2i2.2,5f7.1)')
          * im,id,int(utp+.5d0/3600),int(utm),nint(amod(utm,1.)*60)
          *,ap,altp,HAp
          index=1
              endif

100     HAp=HA
          altp=alt
          ap=a
          djp=dj
          utp=uth

      end

```