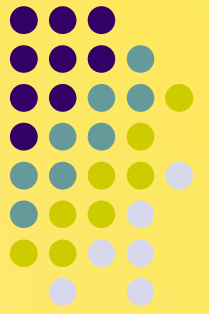


# Iránybecslési eljárások implementálása DSP-n

*Önálló laboratórium 2 beszámoló*

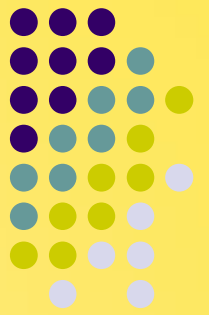
Készítette: *Kővári Balázs*  
Konzulens: *Balogh László*

2010. május 11.



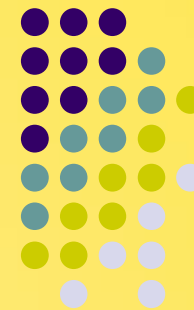
# Előzmények

- Iránybecslési eljárások megismerése
  - Rádió iránymérés alapjai
  - Az iránymérők felépítése
  - Pszeudo-Doppler eljárás
- Adatok feldolgozása MATLAB segítségével
  - Az algoritmus fejlesztése
  - Eredmények vizsgálata
- A kidolgozott algoritmus tesztelése DSP-n
  - Eredmények összehasonlítása



# Feladatok a félévben

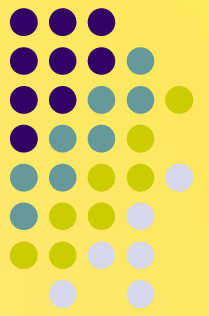
- Szoftverfejlesztés DSP-n
  - A hardver megismerése
  - Ismerkedés a Visual DSP++ fejlesztői környezettel és a Visual DSP++ Kernel (VDK) operációs rendszerrel
  - A szoftver felépítésének megtervezése
  - Implementálás
- Szoftvertesztelés
  - Részegységek tesztelése
  - Funkcionális tesztek
  - Eredmények összegzése



# ADSP-BF537 EZ-KIT Lite

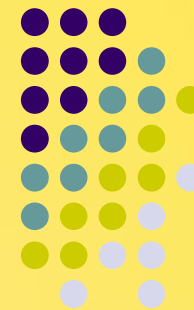
- Analog Devices fejlesztői kártyája
- Fixpontos, Blackfin processzor
- Fontos paraméterek
  - AD1871 48 kHz-es sztereó DA konverter 3.5 mm jack csatlakozóval
  - AD1854 48 kHz-es sztereó AD átalakító 3.5 mm jack csatlakozóval
  - RS-232 UART adó/vevő
  - USB-s debugger interfész



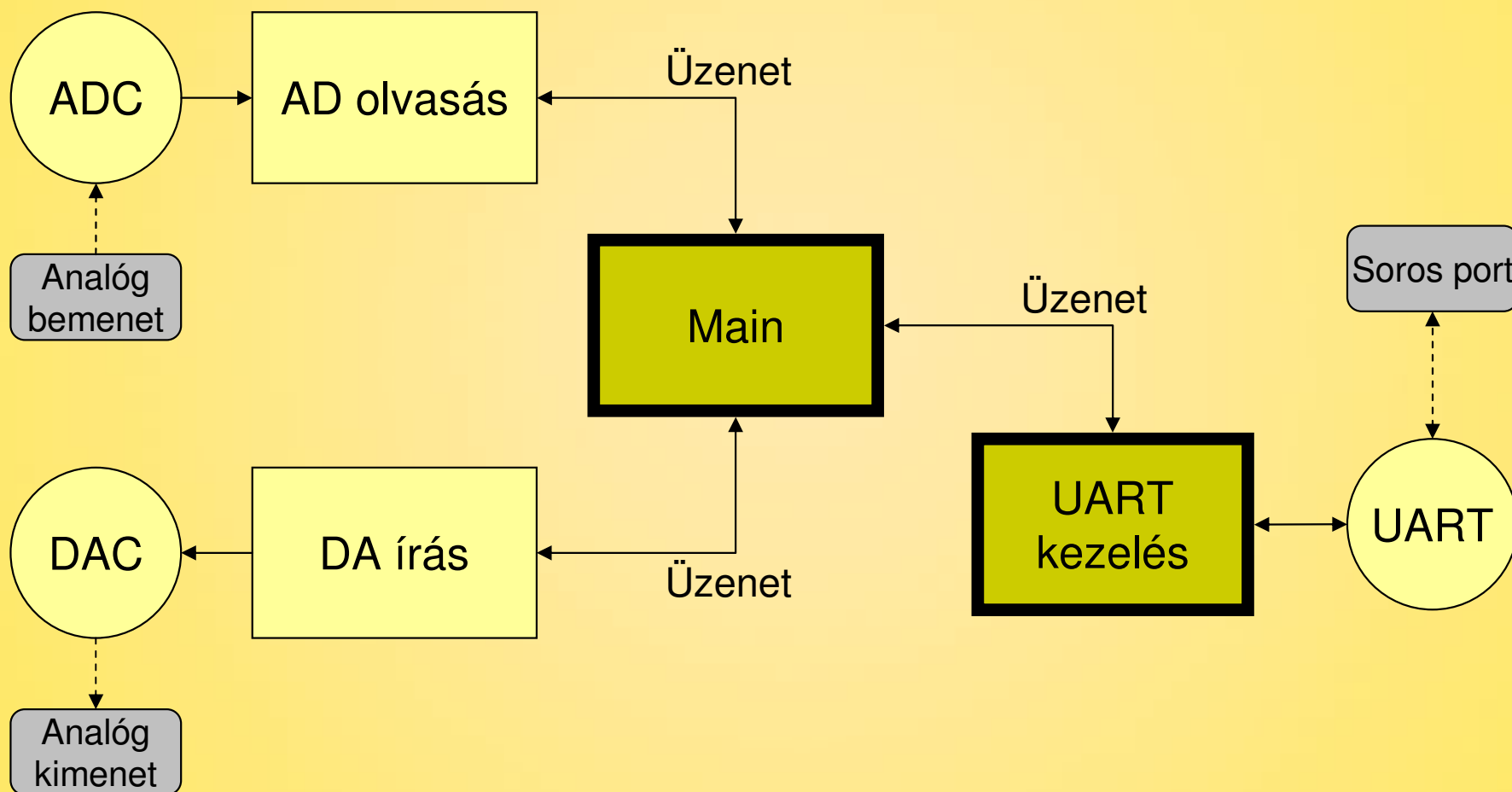


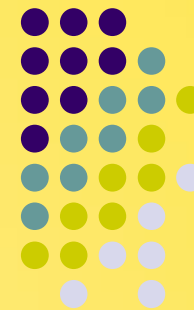
# Visual DSP++ Kernel

- Az Analog Devices által támogatott operációs rendszer, a Visual DSP++ fejlesztői környezet része
- Legfontosabb szolgáltatásai a feladat szempontjából
  - Multi-threading – több, különböző prioritású szál kezelése
  - Üzenetek a szálak közötti kommunikációra
  - Eszközillesztők (device driver) fejlesztésének lehetősége a hardver interfészek kezelésére
  - Debuggolási lehetőségek a működés ellenőrzésére

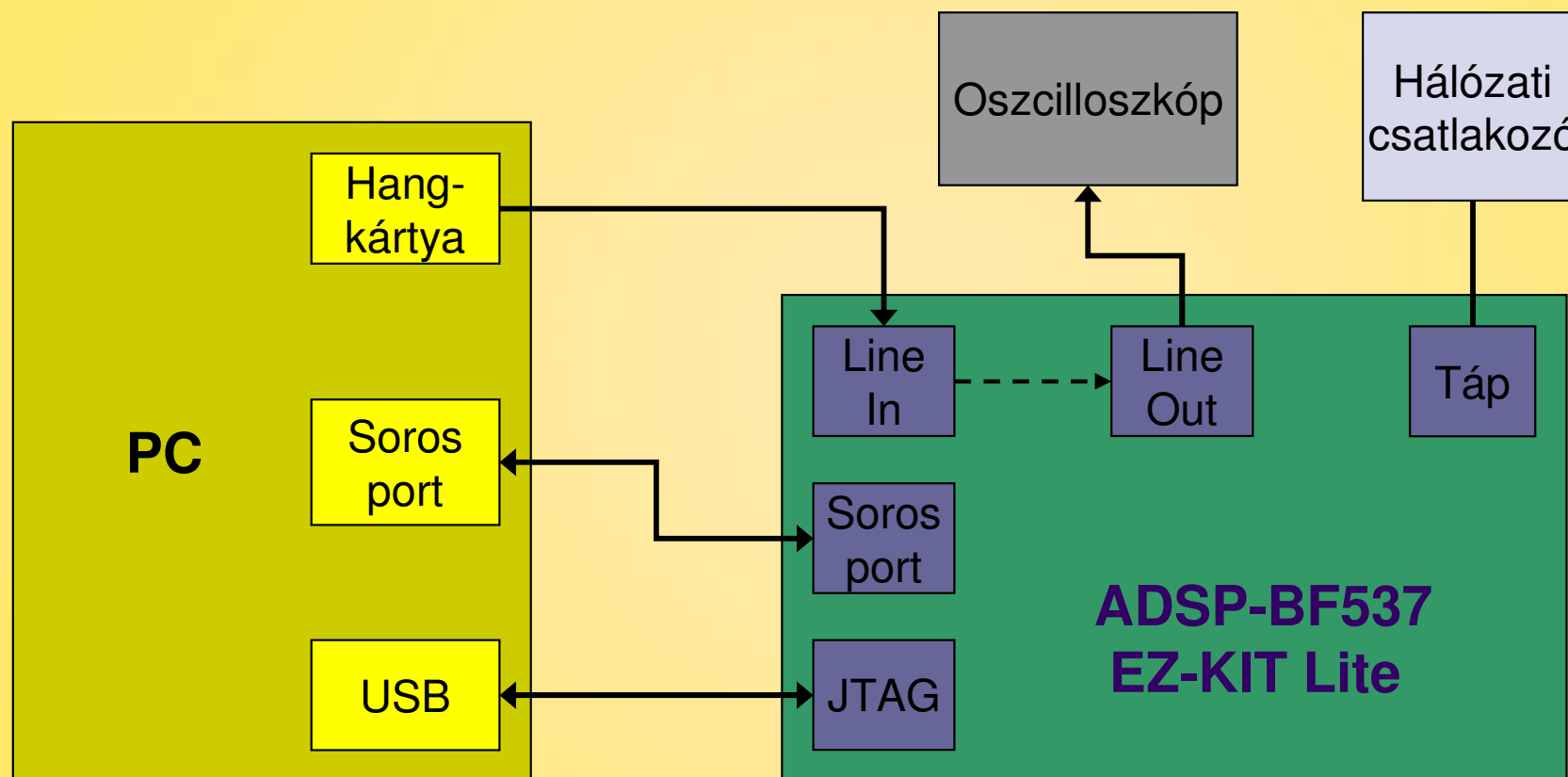


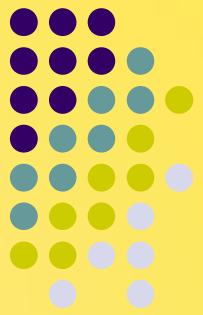
# DSP szoftver felépítése





# Tesztkörnyezet összeállítása





# UART blokk tesztelése

```
DSP - HyperTerminal
File Edit View Call Transfer Help
help
Command usage
Description

start
Starts the data processing

stop
Stops the data processing

help
Prints the list of usable commands

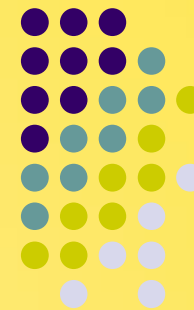
setN <256/512/1024>
Sets the processing variable N - must be a power of 2

setC <1..1000>
Sets the processing variable C, which indicates how many times the processing lo
op shall be executed

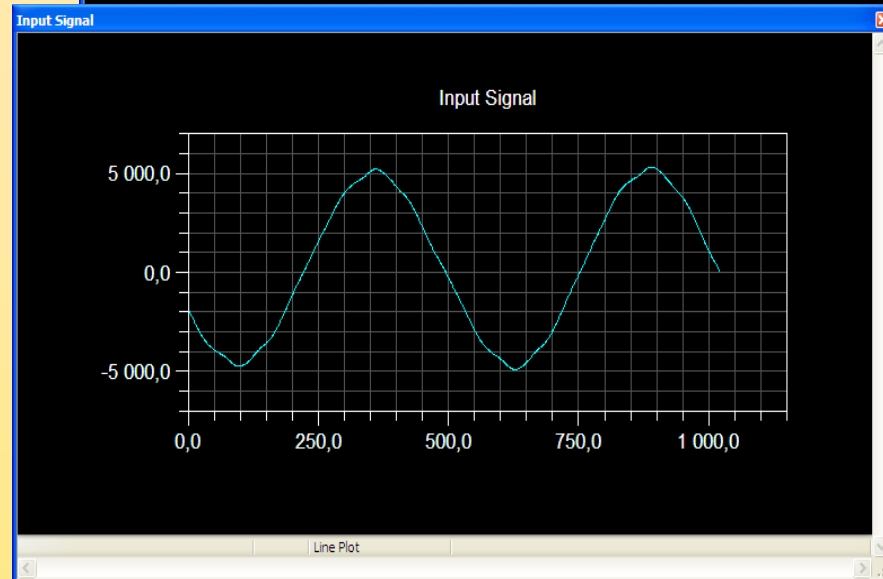
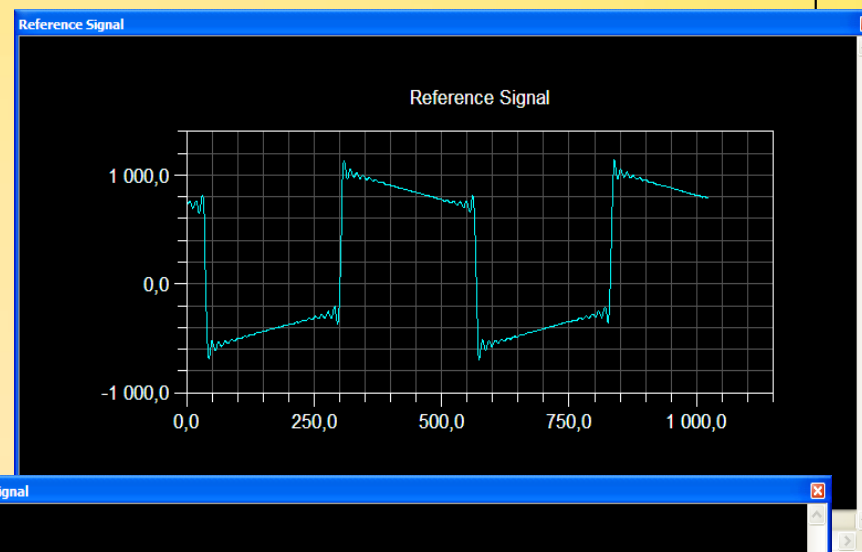
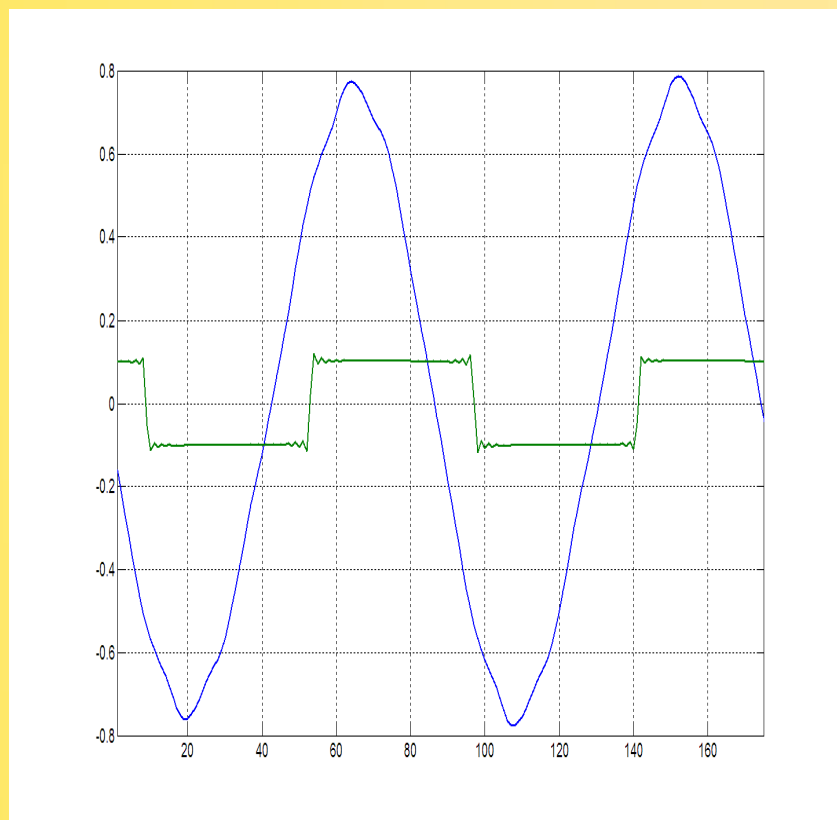
getstat
Prints information about current state

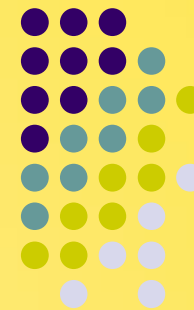
Connected 2:21:52 Auto detect 115200 8-N-1 SCROLL CAPS NUM Capture Print echo
```





# Main blokk tesztelése





# Funkcionális tesztelés

## Példa

- Beesési irány  $\rightarrow 315^\circ$

```
DSP - HyperTerminal
File Edit View Call Transfer Help
getstat
Data processing: stopped
Parameter N: 1024
Parameter C: 100
Latest results
Mean: 316.667725 degree
Deviation: 0.000553

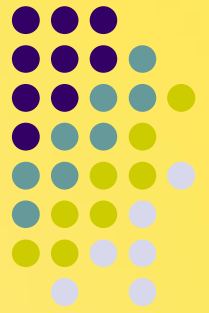
setC 10
Parameter C is set to 10

start
Data processing started successfully

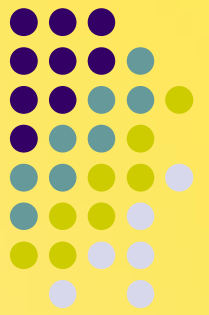
getstat
Data processing: stopped
Parameter N: 1024
Parameter C: 10
Latest results
Mean: 316.520900 degree
Deviation: 0.000458

Connected 0:01:25  Auto detect  115200 8-N-1  SCROLL  CAPS  NUM  Capture  Print echo
```

# További feladatok



- Továbbfejlesztési lehetőségek
  - A kód optimalizálása
  - További funkciók implementálása
  - Bővebb konfigurációs lehetőségek
  - Új parancsok



# Összegzés

- Teljesített feladatok
  - Hardver és fejlesztői környezet megismerése (20%)
  - DSP szoftver tervezése (20%)
  - Alkalmazás implementálása (30%)
  - Szoftver tesztelése és hibák javítása (30%)
- Eredmények
  - Megegyeznek a korábbiakban kiszámítottakkal (MATLAB és DSP emulátor)