

# Poros plazma kísérletek támogatása multiprocesszoros környezetben

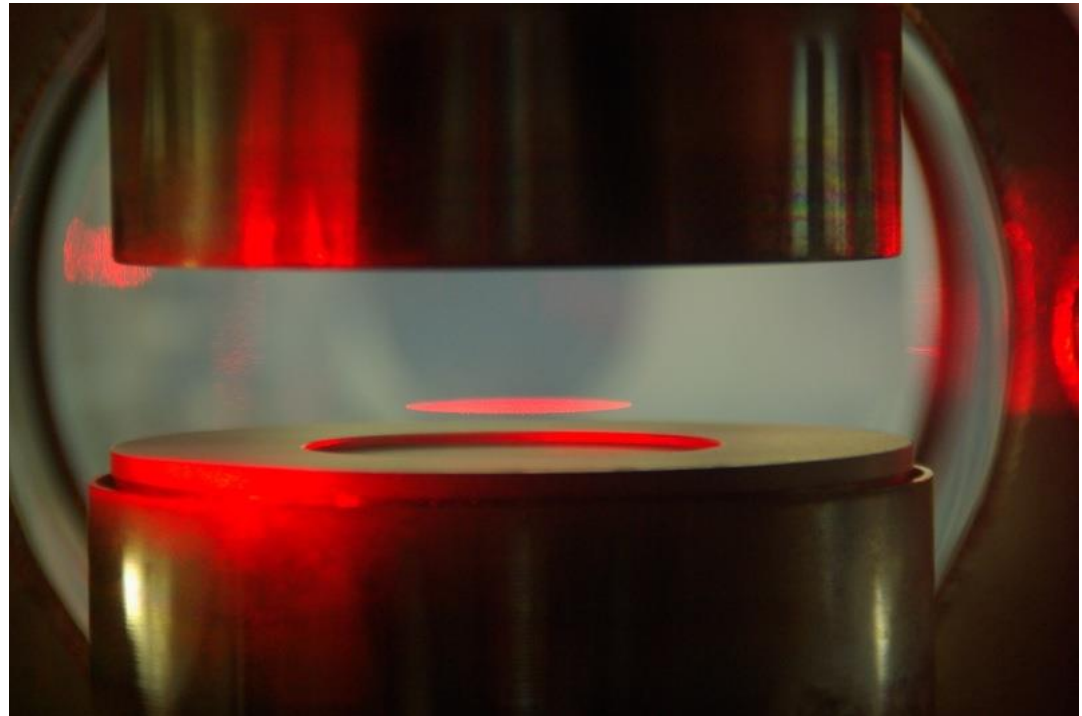
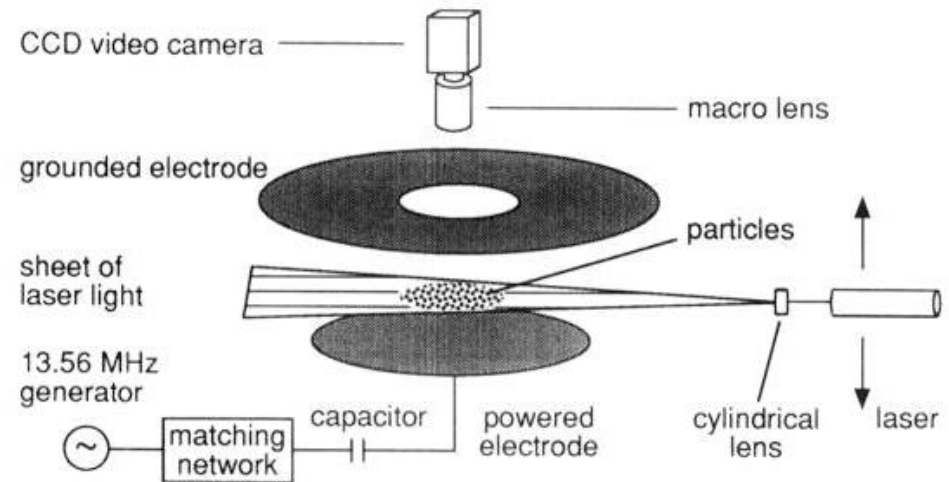
---

KÉSZÍTETTE: BAKRÓ NAGY ISTVÁN

KONZULENS: HARTMANN PÉTER (MTA WIGNER FK,SZFI)  
REICHARDT ANDRÁS (BME SZHVT)

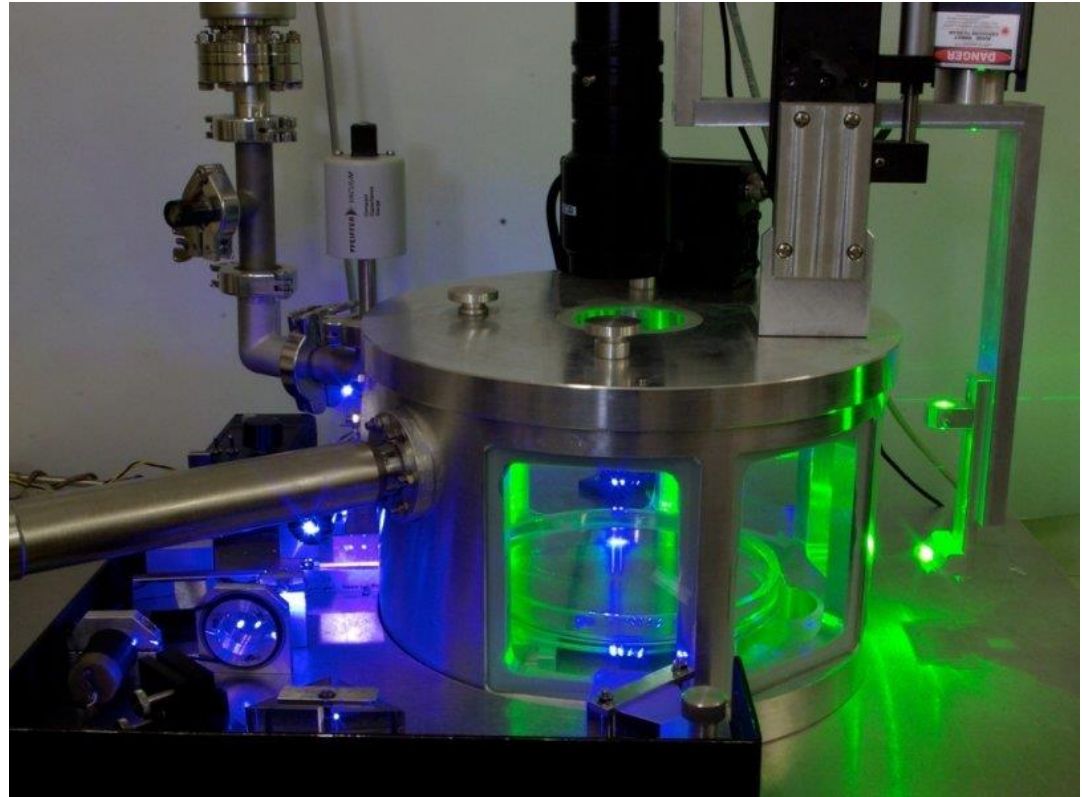
# A poros plazma kísérlet

- Alacsony nyomású ionizált nemesgáz
- RF gerjesztés
- Plazmába szórt porrészecskék
- Gravitációs, villamos, szóródásos, hőmérséklet gradiensi, ion sodrási erő
- Részecskék erős vagy gyenge kölcsönhatása: Coulomb csatolási param. ( $\Gamma$ )



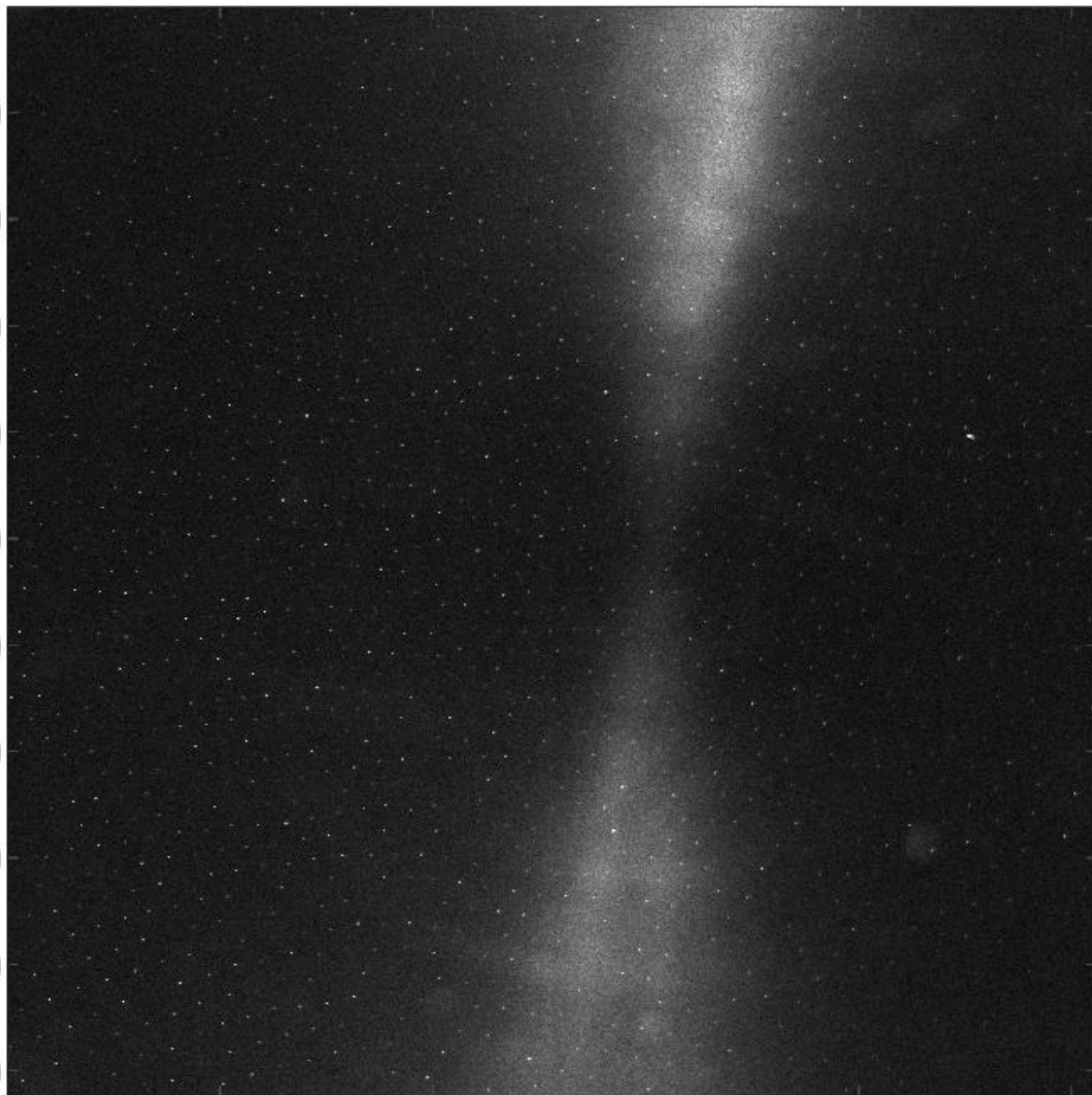
# A kísérlet

- Elővákuum, középvákuum szivattyú
- Nemesgáz áramlás
- RF gerjesztés
- Porrészecskék
- Megvilágító lézer és kamera



Á  
l  
l  
ó  
e  
s  
e  
t

100  
200  
300  
400  
500  
600  
700  
800  
900  
1000



200

400

600

800

1000

60

50

40

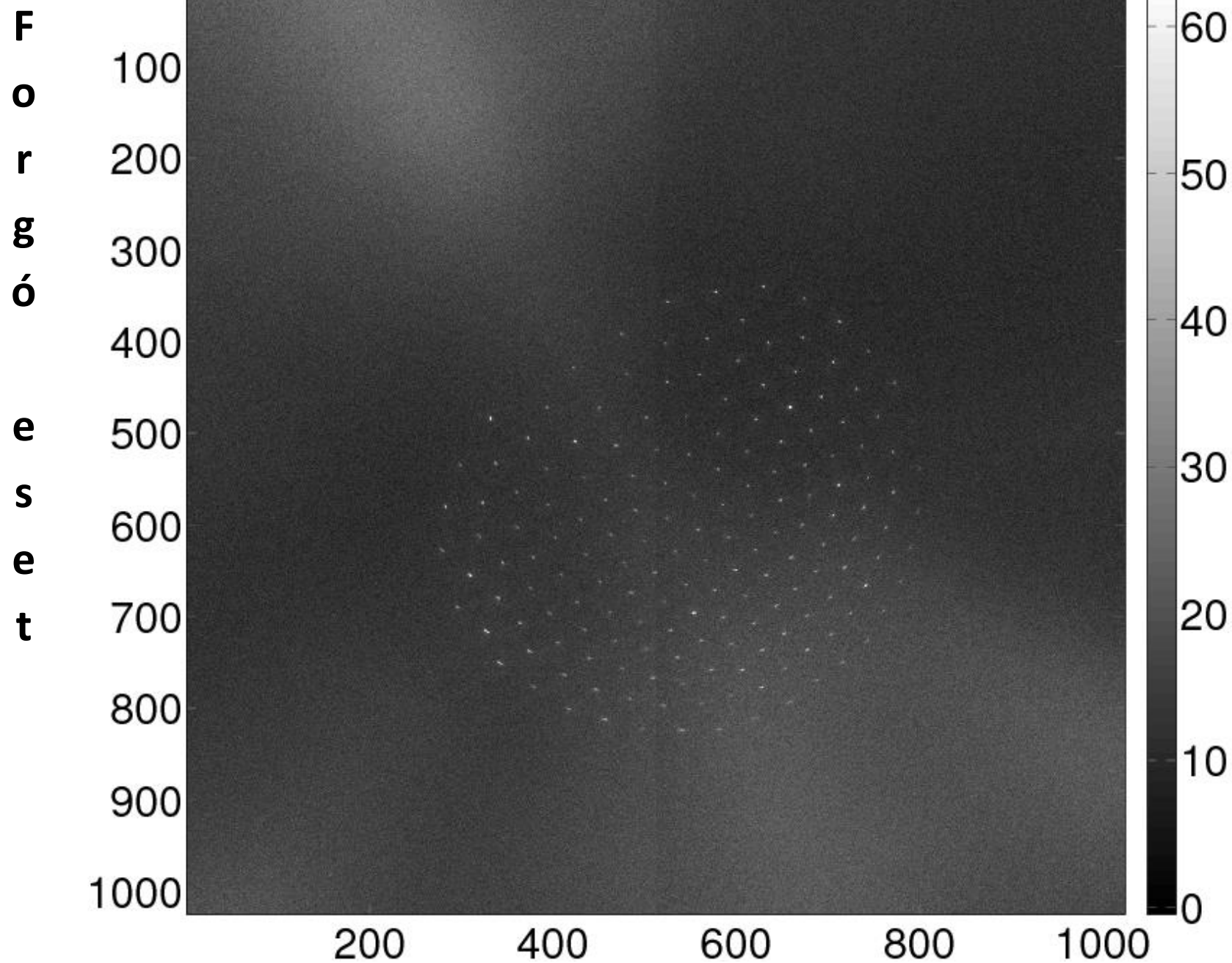
30

20

10

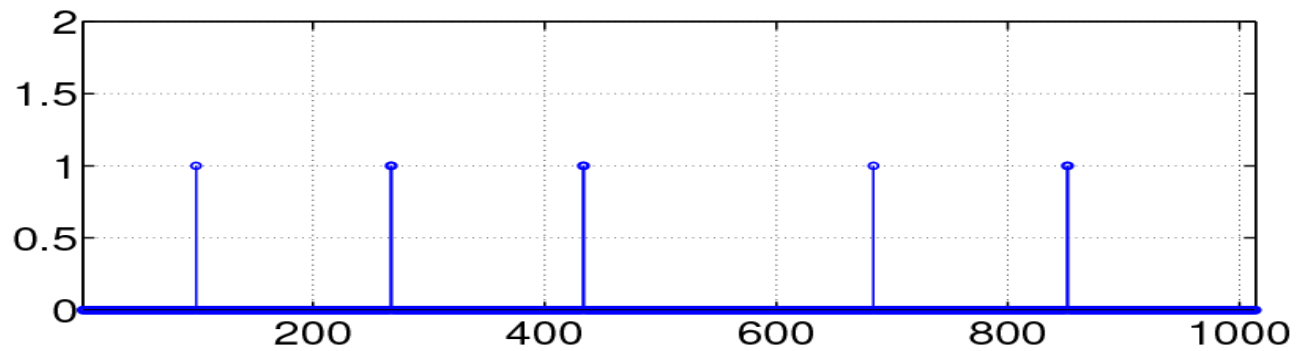
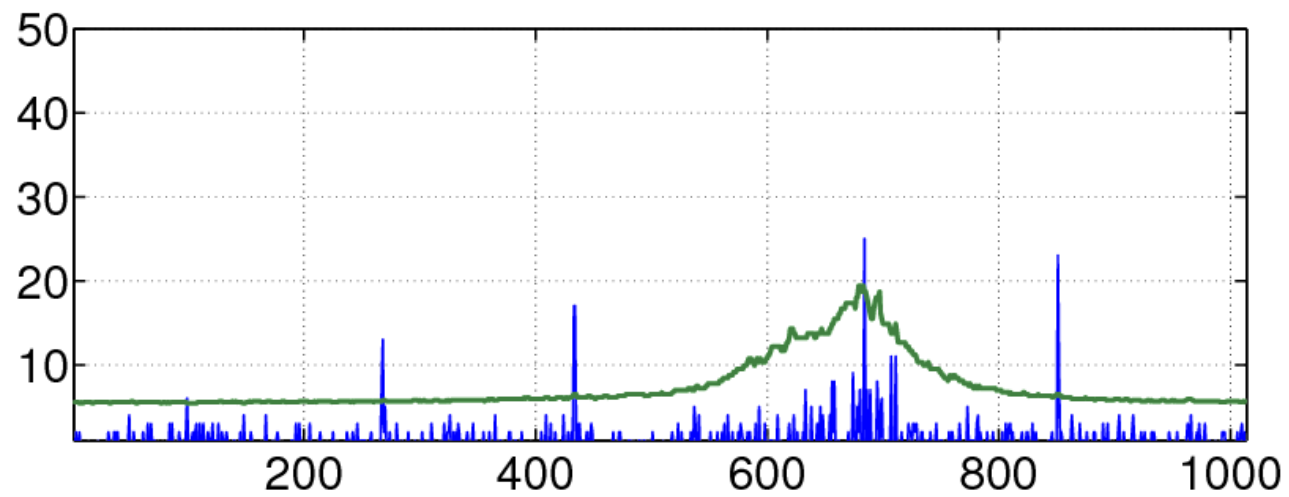
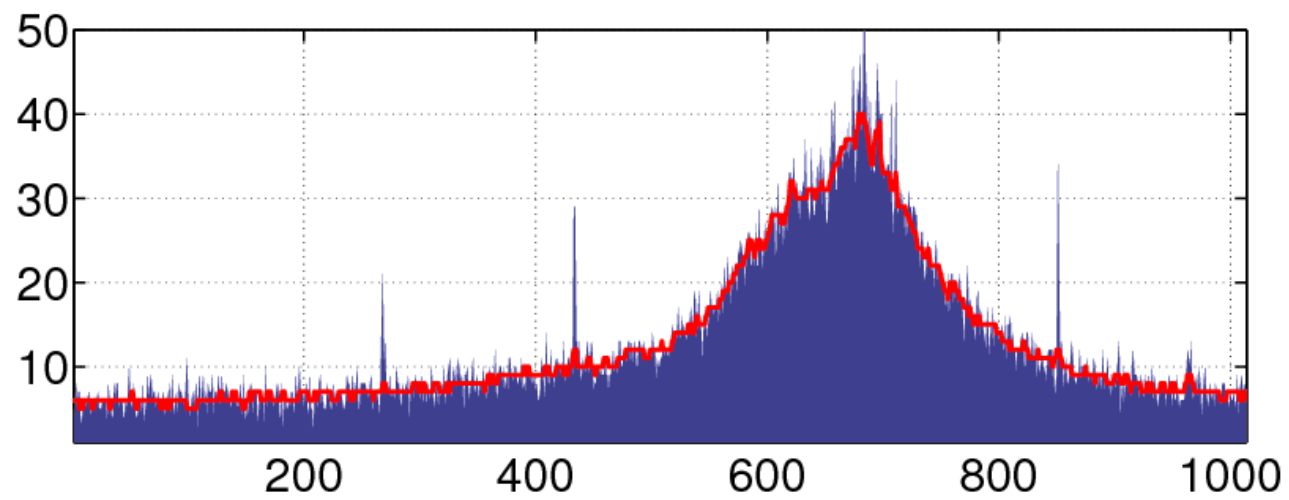
0





# Részecskék detektálása

- Küszöb módszer
- Küszöb módszer szűréssel
  - Gauss szűrővel
  - Medián szűrő
- Adaptív küszöb módszer szűréssel
  - Fényképezés okozta részletesség-különbség problémája
  - Fényes területekben kevésbé bízhatunk meg

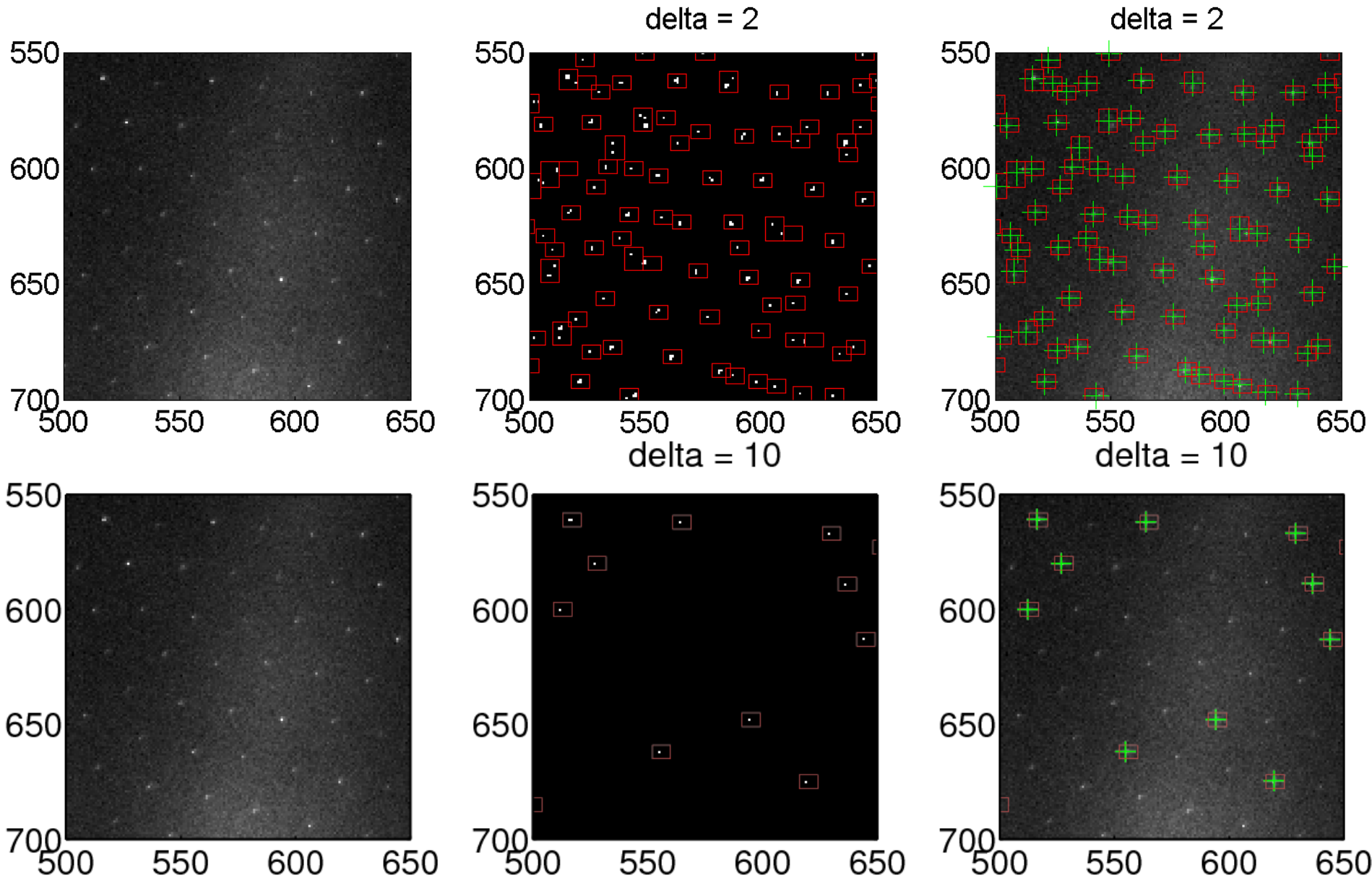


# Részecske pozíciójának számítása momentum módszerrel

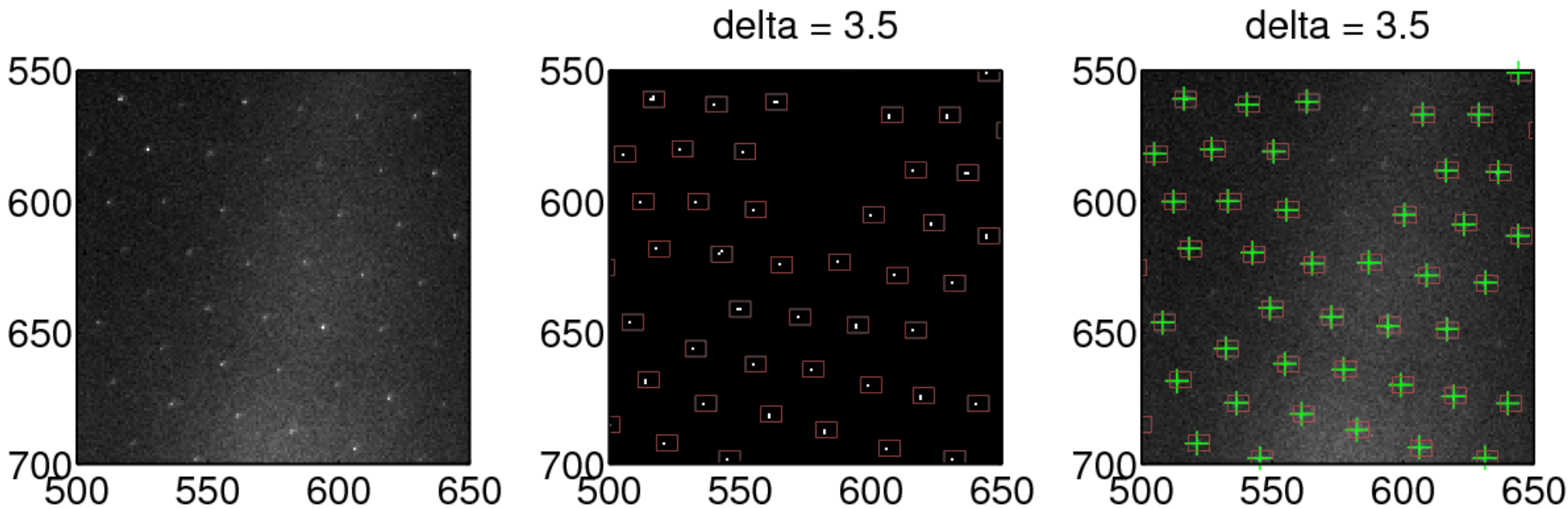
- Ditherelés a fókusz elállításával
- Megjelölt pixel kiterjesztése
- Flood-fill algoritmussal a ROI megkeresése
- Maximálisan világos pont
- Súlypont számítása



# Momentum módszer

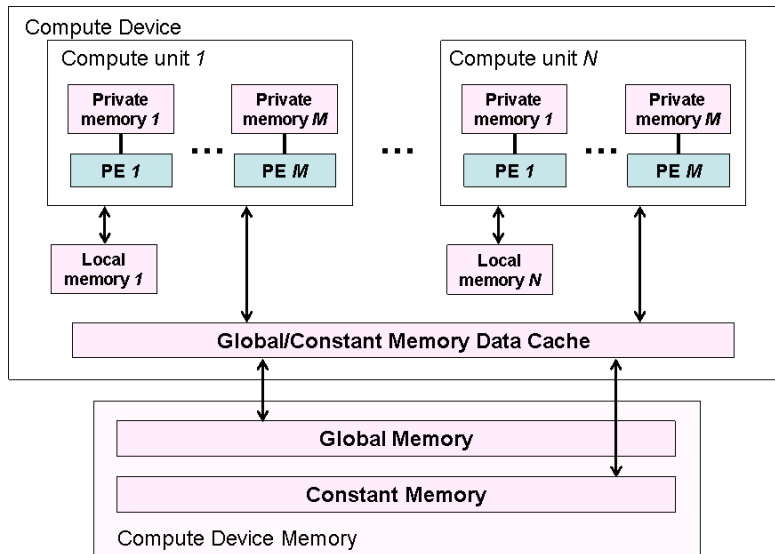


# Momentum módszer



# OpenCL architektúrája

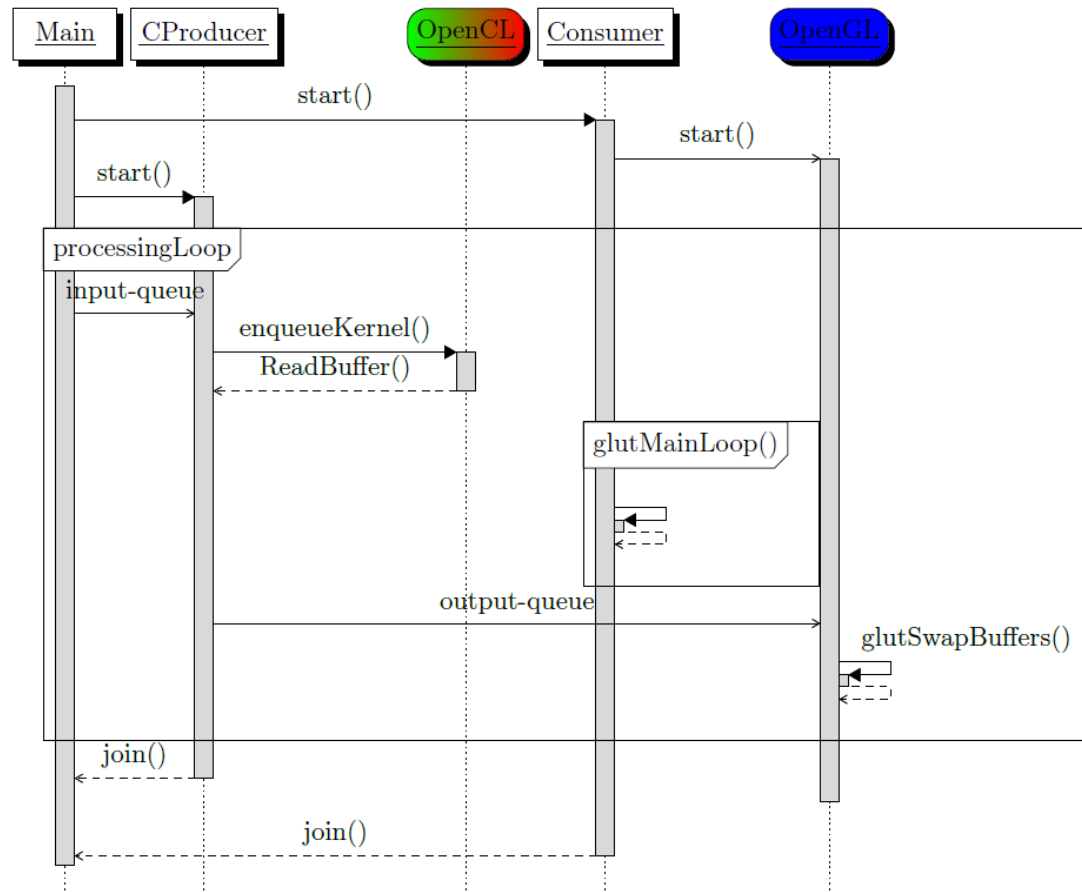
## ESZKÖZ ARCHITEKTÚRÁJA



## MEMÓRIA SZINTEK

	Allokálás		Sebesség
	Hoszt	Kernel	
Globális	Din	Statik.	Lassú
Konstans	Din	Statik.	Gyors
Lokális	Din	Statik.	Gyors
Privát	Din	Statik.	Regiszter

# A host-program működése

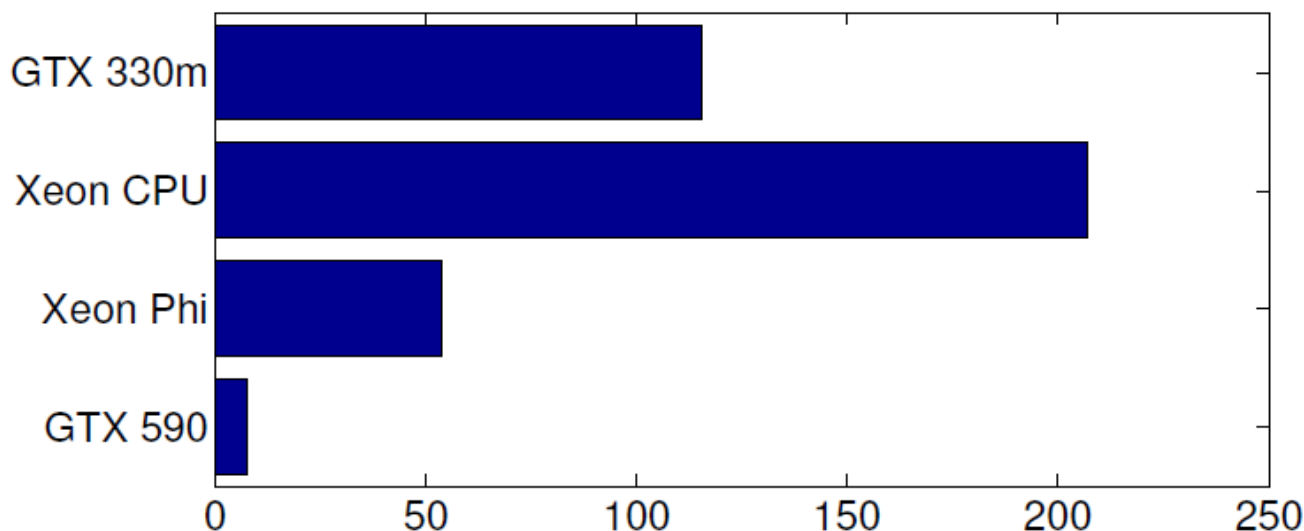


# A meghívott kernelek

- Medián szűrő
- Átlag számító
- Szórás számító
- Detektáló
- ROI-t megkereső és tömegpont számító

	GTX 330m	Xeon E5-1620	Xeon PHI	GTX 590
MAX_COMPUTE_UNITS	6	8	224	16
MAX_CLOCK_FREQUENCY	1265 MHz	3000 MHz	1100 MHz	1225 MHz
MAX_MEM_ALLOC_SIZE	~ 0.25 Gbyte	~ 8 Gbyte	~ 1.5 Gbyte	~ 0.4 Gbyte
LOCAL_MEM_SIZE	16 Kbyte	32 Kbyte	32 Kbyte	48 Kbyte
LOCAL_MEM_TYPE	Local	Global	Global	Local
Futási idő $E\{T\}$	114.12 s	202.01 s	52.74 s	7.74 s
Feldolgozási seb. $\frac{1000}{E\{T\}}$	8.65 FPS	4.82 FPS	18.46 FPS	128.51 FPS

**6.1. táblázat.** Az eszközök erőforrásainak és a program futási idejének összehasonlítása.

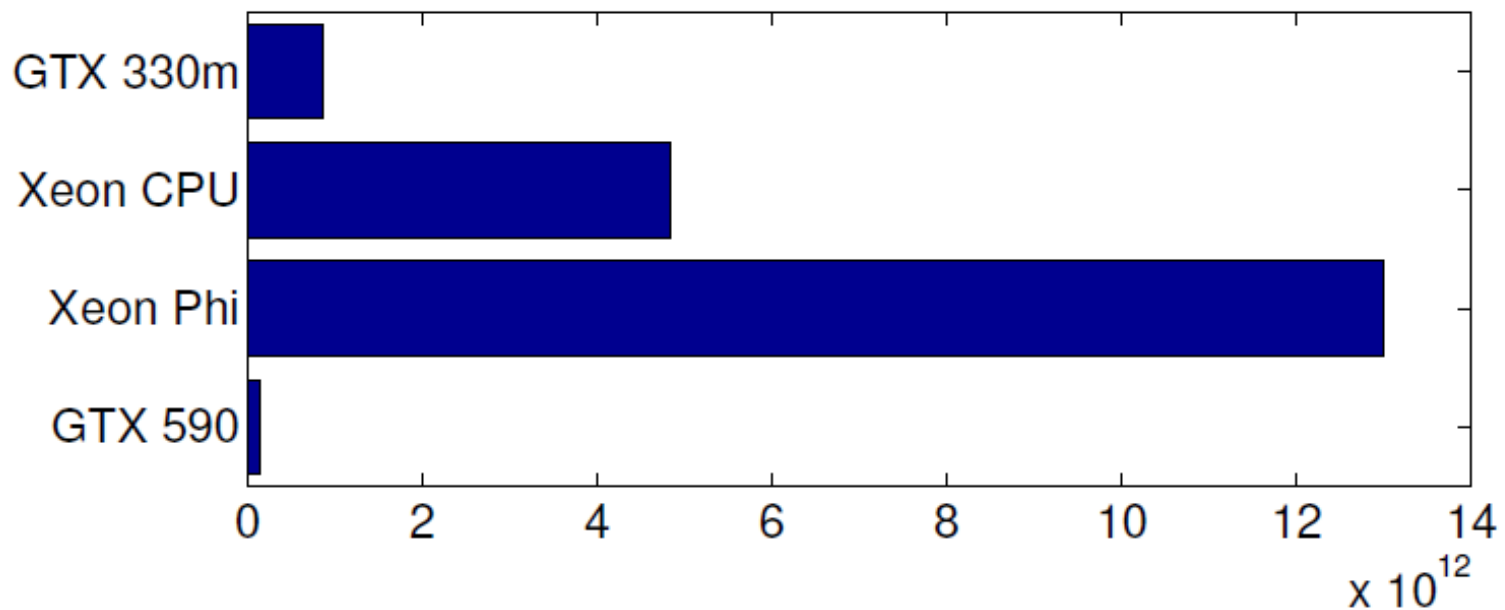


**6.1. ábra.** 1000 kép feldolgozásának futási ideje [s]. A kisebb érték a kedvezőbb.



	GTX 330m	Xeon E5-1620	Xeon PHI	GTX 590
MAX_COMPUTE_UNITS	6	8	224	16
MAX_CLOCK_FREQUENCY	1265 MHz	3000 MHz	1100 MHz	1225 MHz
Futási idő $E\{T\}$	114.1 s	202.0 s	52.7 s	7.74 s
Fajlagos futási idő $T_{\text{fajl}}$	$0.86 \times 10^{12}$	$4.85 \times 10^{12}$	$13.00 \times 10^{12}$	$0.15 \times 10^{12}$

**6.2. táblázat.** *Eszközök fajlagos futási idejének összehasonlítása*



**6.2. ábra.** *1000 kép feldolgozásának fajlagos futási ideje. A kisebb érték a kedvezőbb.*

# Összegzés

- Bemutattam a poros plazma kísérletek apparátusát
- Részecske detektálása szűréssel és adaptív döntési küszöbvel
- Szűrés Gauss helyett medián szűrővel
- Host-program producer-consumer szálba rendezése
- Eredmény grafikus megjelenítése (OpenGL)

## **További lehetőségek:**

- Paraméterek GUI-val történő online állítása
- Eredeti stream megjelenítése

# Bírálóí kérdések

- 1) Hogyan történt a program tesztelése, validálása, különös tekintettel a kernelre?
- 2) Mikor kell a „ditherelés” a gyakorlatban? (Lásd 2.2 fejezet!)
- 3) A 2. oldalon található  $F_n$  és  $F_i$  függetlenek-e egymástól? Válaszát indokolja!
- 4) A 2.1 egyenletben található paramétereket hogyan választjuk meg? (Lásd 10. oldal!)

$$K = \mathbf{E} \left\{ P - \hat{M}P \right\} + \\ + \delta \cdot \text{STD} \left\{ P - \hat{M}P \right\} \cdot \left[ 1 + a \left( \frac{\hat{M}P}{\max \left\{ \hat{M}P \right\} - \min \left\{ \hat{M}P \right\}} \right)^b \right] \quad (2.1)$$

# Bírálóí kérdések

5) A 19. oldal alján olvasható: „Az összehasonlíthatóság végett a legkisebb memóriájú eszközre fogom a problémát skálázni. Tehát maximálisan 16Kbyte lokális memóriát fogok használni. A többi eszköz memóriája nagyobb, így a kód mindegyiken tud futni.”

E fenti feltételezés viszont azt jelenti, hogy csak az egyik eszközre lesz optimális a kernel. Ez módosítja a 6.1 ábrán vázolt futási sebességeket. Mi a helyzet, ha a 16Kbyte helyett az egyes hardverek optimum memória blokkját használjuk a párhuzamosításhoz? Történt-e ez irányú vizsgálat?

6) A diploma dolgozatban ismertetett program alkalmazása hogyan történik a poros plazma készüléken végrehajtott mérések során? A jelölt lehetőleg konkrét példán keresztül mutassa be az általa kifejlesztett program alkalmazását