OpenCL Tutorial

David Castells-Rufas
Microelectronics & Electronics Systems Department
Universitat Autònoma de Barcelona
david.castells@uab.cat

Centre de Prototips i Solucions Hardware - Software
Center for Hardware – Software Prototypes & Solutions

Basic Kernels

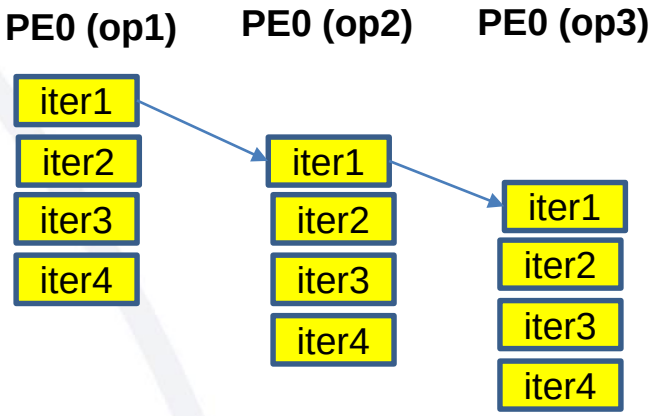
Goals

- Implement Single Work Item and NDRange Kernels

Review: Pipelined Kernels

- In the host we ask for 1 work item
- In the kernel We do NOT use get_global_id()
- The important factor is the Interval Innitiation Factor (II)

- Pipelining



gammaImage as a Single Work Item Kernel

```
unsigned char doGamma(unsigned char inv, double nGamma)
{
    double dvin = inv;
    double dvout = 255.0*pow(dvin/255.0, 1.0/nGamma);

    if (dvout < 0)
        return 0;
    else if (dvout > 255)
        return 255;
    else
        return dvout;
}
...

__kernel void gammaImage(__global unsigned char* inputImage, int w, int h, double nGamma, __global
unsigned char* outputImage)
{
    for (int y = 0; y < h; y++)
        for (int x = 0; x < w; x++)
        {
            unsigned char r,g,b;
            image_getRGB(inputImage, w, h, x, y, &r, &g, &b);

            r = doGamma(r, nGamma);
            g = doGamma(g, nGamma);
            b = doGamma(b, nGamma);

            bitmap_setRGB(outputImage, w, h, x, y, r, g, b);
        }
}
```

Testing it

YOU DO...

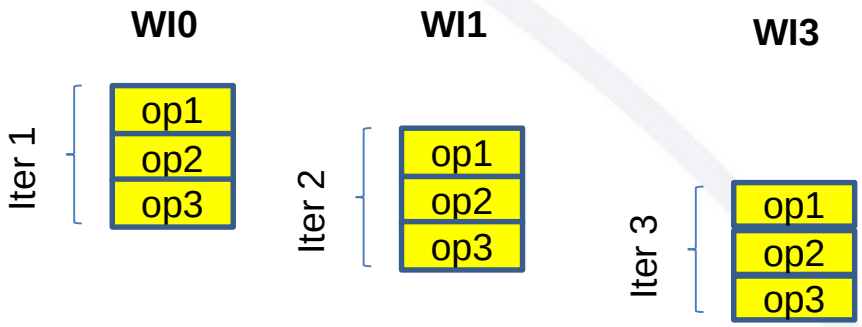
- go to LAB2.1
- Compile / emulate / early / deploy / execute
- Test and measure performance

gamma 0.5

gamma 2

Alternative Implementation

- Using NDRange kernel (GPU like)
- Going to assign a Work Item (thread) to every line of the image



gammaImage as NDRange Kernel

```
unsigned char doGamma(unsigned char inv, double nGamma)
{
    double dvin = inv;
    double dvout = 255.0*pow(dvin/255.0, 1.0/nGamma);

    if (dvout < 0)
        return 0;
    else if (dvout > 255)
        return 255;
    else
        return dvout;
}
...

__kernel void gammaImage(__global unsigned char* inputImage, int w, int h, double nGamma, __global
unsigned char* outputImage)
{
    int y = get_global_id(1);
    int x = get_global_id(0);
    //for (int y = 0; y < h; y++)
        //for (int x = 0; x < w; x++)
        {
            unsigned char r,g,b;
            image_getRGB(inputImage, w, h, x, y, &r, &g, &b);

            r = doGamma(r, nGamma);
            g = doGamma(g, nGamma);
            b = doGamma(b, nGamma);

            bitmap_setRGB(outputImage, w, h, x, y, r, g, b);
        }
}
```

Testing it

YOU DO...

- go to LAB2.2
- Compile / emulate / early / deploy / execute
- Test and measure performance