Name: Wilmer Henao

Optimizations used.

1) Introduction of initial conditions
2) Introduction of measurements
3) Changing the stencil

In this opportunity the first thing I had to do was create the measurements and change the initial conditions in function f and changing the measurements.

The program registers the following times
MCells/sec: 1049.55
MFlops/sec: 8746.28
GBytes/sec: 31.4315

This is all done before the change in stencil B.

```
#define BLOCK_SIZE 16
#define A_BLOCK_STRIDE (BLOCK_SIZE * dim_x)
#define A_T_BLOCK_STRIDE (BLOCK_SIZE * dim_y)

__kernel void add_source_term(
    __global float *u,
    const unsigned points,
    const float dt,
    const float t,
    const unsigned dim_x,
    const unsigned dim_y,
    const float dx
)
{
  __local float u_local[BLOCK_SIZE][BLOCK_SIZE];
  int base_idx_a =
    get_group_id(0) * BLOCK_SIZE +
    get_group_id(1) * A_BLOCK_STRIDE;
  int glob_idx_a =
    base_idx_a + get_local_id(0)
    + dim_x * get_local_id(1);

    __local float x, y, z;

    const int i = get_global_id(1) + 1;
    const int j = get_global_id(0) + 1;
    const int k = base_idx_a;

    x = -1 + (float) i * dx;
    y = -1 + (float) j * dx;
    z = -1 + (float) k * dx;
    unsigned base = i + dim_x * (j + dim_y * k);

    u[base] = dt * dt * exp(-1600 * ((x - 0.5) * (x - 0.5) + (y - 0.2) * (y - 0.2) + (z - 0.3) * (z - 0.3) ));

}
```

After doing the change in stencil B The following times were registered

MCells/sec: 1370.1
MFlops/sec: 11417.5
GBytes/sec: 20.5156

A few changes in the layout of cl files were tried but they did not improve on the performance by much. I will be very happy to see a good version of this code at work.
Stencil B. Wave-Kernel.cl

```
__kernel void fd_update(
    const float dt2_over_dx2,
    __global float *new_and_hist_u,
    __global const float *u,
    const unsigned dim_x,
    const unsigned dim_y,
    const unsigned points
)
{
  for (int k = 2; k < points; ++k)
  {
    // + 2 to account for ghost cells
    const int i = get_global_id(1) + 2;
    const int j = get_global_id(0) + 2;

    unsigned base = i + dim_x*(j + dim_y * k);

    new_and_hist_u[base] =
      2 * u[base] - new_and_hist_u[base]
      + (1.0/12.0) *dt2_over_dx2 * (
          - 90*u[base]
          + 16.0 * u[base - 1]
          + 16.0 * u[base + 1]
          + 16.0 * u[base - dim_x]
          + 16.0 * u[base + dim_x]
          + 16.0 * u[base + dim_x*dim_y]
          + 16.0 * u[base - dim_x*dim_y]
          - 1.0 * u[base - 2]
          - 1.0 * u[base + 2]
          - 1.0 * u[base - 2*dim_x]
          - 1.0 * u[base + 2*dim_x]
          - 1.0 * u[base + 2*dim_x*dim_y]
          - 1.0 * u[base - 2*dim_x*dim_y]
          );
  }
}
```