

```

26. MPI_Status status;
27. int left_neighbor = (pid > 0) ? (pid - 1) : MPI_PROC_NULL;
28. int right_neighbor = (pid < np - 2) ? (pid + 1) : MPI_PROC_NULL;

/* Upload stencil coefficients */
upload_coefficients(coeff, 5);

29. int left_halo_offset = 0;
30. int right_halo_offset = dimx * dimy * (4 + dimz);
31. int left_stagel_offset = 0;
32. int right_stagel_offset = dimx * dimy * (dimz - 4);
32. int stage2_offset = num_halo_points;

34. MPI_Barrier( MPI_COMM_WORLD );
35. for(int i=0; i < nreps; i++) {
    /* Compute boundary values needed by other nodes first */
36. launch_kernel(d_output + left_stagel_offset,
                d_input + left_stagel_offset, dimx, dimy, 12, stream0);
37. launch_kernel(d_output + right_stagel_offset,
                d_input + right_stagel_offset, dimx, dimy, 12, stream0);

    /* Compute the remaining points */
38. launch_kernel(d_output + stage2_offset, d_input + stage2_offset,
                dimx, dimy, dimz, stream1);

```