Timing with CUPTI Callback functions consists of just a few steps:

- Initialize trace:
  - Enable collection of kernel activity records
  - Register callback functions

```
cuptiActivityEnable(CUPTI_ACTIVITY_KIND_CONCURRENT_KERNEL);
cuptiActivityRegisterCallbacks(bfrRequest, bfrCompleted);
```

- Trigger callback functions; schematically, they look like this:

```
void CUPTI API bfrRequest (uint8_t **bfr, ...)
{
      // Signal to CUPTI client that an empty buffer is needed by CUPTI
}
void CUPTI API bfrCompleted (uint8_t *bfr, ...)
{
      // Return a buffer of completed activity records to CUPTI client
}
```

Results: implemented CUPTI-Callback timer in AMReX (Adaptive Mesh Refinement library), and tested with simple kernels

- We tested the CUPTI-based timer using a simple device sleep function
- With NVIDIA Volta V100 (peak clock frequency: 1.53 GHz), we launched sleep kernels on separate streams for multiples of 1, 2, 3, and 4×10^8 cycles (≈ 65 ms)