Program Listing for File cpu_resource_monitor.hpp

```cpp
/*
 * SPDX-FileCopyrightText: Copyright (c) 2023 NVIDIA CORPORATION & AFFILIATES.
All rights reserved.
 * SPDX-License-Identifier: Apache-2.0
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
#ifndef HOLOSCAN_CORE_SYSTEM_CPU_RESOURCE_MONITOR_HPP
#define HOLOSCAN_CORE_SYSTEM_CPU_RESOURCE_MONITOR_HPP

#include <sched.h>  // for sched_getaffinity
#include <memory>

#include "cpu_info.hpp"

namespace holoscan {

constexpr uint64_t kDefaultCpuMetrics = CPUMetricFlag::CORE_COUNT |
CPUMetricFlag::CPU_COUNT | CPUMetricFlag::AVAILABLE_PROCESSOR_COUNT;

class CPUResourceMonitor {
 public:
  explicit CPUResourceMonitor(void* context,
uint64_t metric_flags = kDefaultCpuMetrics);
  virtual ~CPUResourceMonitor() = default;
  uint64_t metric_flags() const;
  void metric_flags(uint64_t metric_flags);
  CPUInfo update(uint64_t metric_flags = CPUMetricFlag::DEFAULT);
  CPUInfo& update(CPUInfo& cpu_info, uint64_t metric_flags = CPUMetricFlag::DEFAULT);
  CPUInfo cpu_info(uint64_t metric_flags = CPUMetricFlag::DEFAULT);
  cpu_set_t cpu_set() const;

 protected:
  void* context_ = nullptr;
  uint64_t metric_flags_ = kDefaultCpuMetrics;
  bool is_cached_ = false;
  CPUInfo cpu_info_ = {};
  cpu_set_t cpu_set_ = {};
  bool is_last_total_stats_valid_ = false;
  uint64_t last_total_stats_[4] = {0};
};

}  // namespace holoscan
#endif/* HOLOSCAN_CORE_SYSTEM_CPU_RESOURCE_MONITOR_HPP */
```