Lab 0x01 - Using a Scheduler

This assignment is to be completed in your lab groups. This assignment is meant to be brief and simply get you on board with the new code structure we will use the remainder of the term.

In this assignment you will modify your code from the previous lab assignment so that your code runs using the cooperative scheduler discussed in lecture and utilizes queues and shares for all data transfer between tasks.

Familiarization

Before you begin the assignment, download the scheduler files and run an example program on your microcontroller. Visit `https://github.com/spluttflob/ME405-Support` and download `cotask.py`, `taskshare.py`, and `basic_tasks.py`. Examine the contents of the example program `basic_tasks.py` while running the code on your microcontroller from Thonny.

Assignment

Once you have become familiar with the behavior of the example program and understand a bit more about using tasks, shares, and queues, adapt your Lab 0x00 program to run within the scheduler using tasks and proper shared variables.

1. Use a task diagram and a set of state transition diagrams to design your program flow. A starter design is provided in Figures 1 through 4; the design is not complete but will give you a head start. Keep in mind that code running in interrupts should be considered as its own task even if not implemented using generator functions and the scheduler.
2. You can use objects from `task_share.Share` to communicate information between tasks, such as whether or not the user button has been pressed as triggered by an interrupt callback.
3. You can also use objects from `task_share.Queue` to communicate a sequence of data between tasks, such as your collected ADC readings. Queues can also be used to synchronize tasks by having one task wait to perform an action until a queue is either full or empty.

Requirements and Deliverables

The functionality of your program should be identical to the previous lab, but must be fully implemented using.

You will submit a brief memo describing your setup and program flow along with completed task and state transition diagrams. Do not include the same content in your memo from the previous assignment. Focus on the changes made to fit your code into the new implementation style. Be sure to include your Python code as an attachment to the memo.

The memo will be submitted through Canvas as a PDF document.

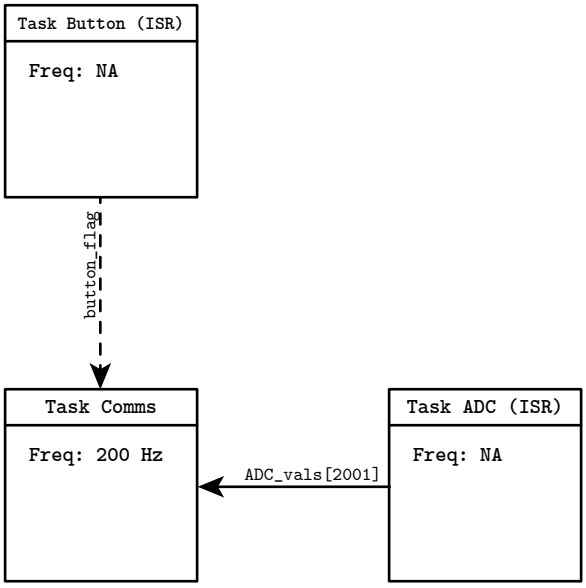

Figure 1: Preliminary task diagram for Lab 0x01.

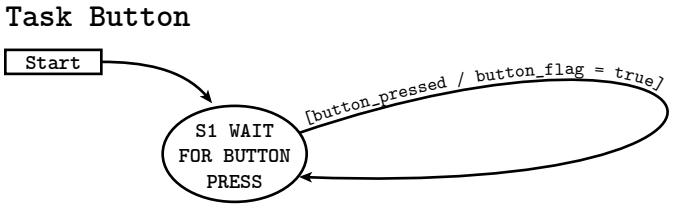

Figure 2: Preliminary button task design.

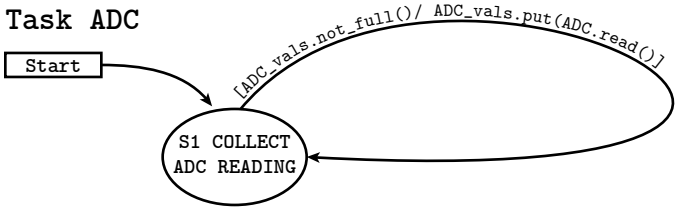

Figure 3: Preliminary ADC task design.

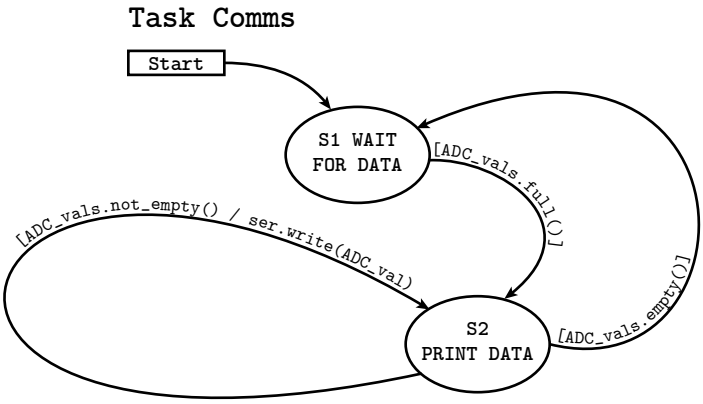

Figure 4: Preliminary communication task design.