

# Tasks

## Scout

1. Wireless Message Reception ( $T_0$ ): this is a non-periodic task which becomes ready whenever the RF module receives data from the referee. The task execution consists of reading the received data from the RF module over SPI and placing this data in the global message queue. This task has no dependencies.
2. Sensor Sampling ( $T_1$ ): this is a periodic task that becomes ready every 0.1 seconds. The task execution consists of probing the ADC four times for the analog channel measurements corresponding to the four light sensors and storing these values in global memory. This task has no dependencies.
3. Collector Ping ( $T_2$ ): this is a periodic task that becomes ready every 0.5 seconds. The task execution consists of sending harvesting positions to the collector, or  $-1$  if none has been found. This task is dependent on the Sensor Sampling task.
4. Differential Drive ( $T_3$ ): this is a periodic task that becomes ready every 0.1 seconds. The task execution consists of calculating the required direction of movement (reference theta which is used as PID input) and speed based on the latest light sensor readings and referee messages. This task depends on the Sensor Sampling task.

All the scout's periodic tasks are synchronous (they are ready at time 0). The following table summarizes the scout's tasks periods, priorities, costs, and dependencies.

Task	Period (s)	Priority	Cost (s)	Dependencies
$T_0$	N/A	1	0.001	None
$T_1$	0.1	2	0.001	None
$T_2$	0.5	4	0.001	$T_1$
$T_3$	0.1	3	0.001	$T_1$

For this robot, we are going to use a non-preemptive scheduler that executes tasks according to their priorities. Each time the scheduler is invoked, it rearranges the available tasks according to decreasing priority and whenever a task is complete, the first ready task in the list is executed. We have given  $T_0$  the highest priority for the scheduler to execute it as soon as possible, thus updating location estimates and allowing for better performance by the remaining tasks. All cost estimates are rounded up to the nearest millisecond.

## Collector

1. Wireless Message Reception ( $T_0$ ): this is a non-periodic task which becomes ready whenever the RF module receives data from the referee or the scout. The task execution consists of reading the received data from the RF module over SPI and placing this data in the global message queue. This task has no dependencies.
2. Sensor Sampling ( $T_1$ ): this is a periodic task that becomes ready every 0.1 seconds. The task execution consists of sequentially polling the GPIOs to which the proximity sensors are attached and placing the readings in global memory. This task has no dependencies.
1. Differential Drive ( $T_2$ ): this is a periodic task that becomes ready every 0.1 seconds. The task execution consists of calculating the required direction of movement (reference theta which is used as PID input) and speed based on the latest proximity sensor readings in addition to scout and referee messages. This task depends on the Sensor Sampling as well as Wireless Message Reception tasks.

The collector's periodic tasks are synchronous. The following table summarizes the collector's tasks periods, priorities, costs, and dependencies.

Task	Period (s)	Priority	Cost (s)	Dependencies
$T_0$	N/A	1	0.001	None

<b>T<sub>1</sub></b>	0.1	2	0.001	None
<b>T<sub>2</sub></b>	0.1	3	0.001	T <sub>0</sub> , T <sub>1</sub>

For this robot, we are going to use a preemptive scheduler. Whenever a task becomes ready or finishes executing, the scheduler is invoked and immediately begins executing the highest priority task that is ready.