Design of Embedded Systems (DES), Assignment 9

Richard Bisschops: s4448545 & Lisa Boonstra: s3018547

Exercise #9-2: Rotating LEDs

Everything was tested on pi #3.

Ex09a

a1) The average rotation speed is 83238615 ns, with an average cycle time of 720 rotations per minute. To measure the rotation time, we measure the time between two falling interrupts.

a2) The average time it takes for the light sensor to pass is 1660216 ns. We wait for an interrupt of either rising or falling at first, then we wait for a second interrupt and measure the time between them. We have the expectation that falling will be first, since this is the most likely case, as there is a very small window that rising will be first. However, we found that the rising turned out to be as frequent, or even more frequent, than falling. In that case we would end up measuring the time spend when the sensor is not passed. This would lead to a result close to the rotation length, with a difference between the two being the time that the sensor is indeed passed. If the sensor time is very close to the rotation time we subtract the sensor time from the rotation time in order to obtain the correct answer. In case we did not observe this, we left the sensor time value unchanged.

Ex09b

In order to create an X we split up the drawing of the X at different moment in time. Each led light except, led 17 and 27, are connected to two specific tasks: one task on the left of the center of the x, one task on the right of the center of the x. One task handles led light 17 and 27, which represent the center of the x.

The center itself is half a 'sensor time' (the result found at 9a.2) away from the rotation time to account for a small deficit from the hardware. Note that for 9b, the leds too far on the left have to wait almost a full rotation time before they can lit up. However, relatively, they are only several sensor times away (rotation_time-sensor_time).

Each task waits for a semaphore to be freed. Only an interrupt handler is able to do this: the interrupt_handler checks when we pass the light sensor (falling). When it does, it sends a broadcast over the semaphore, releasing all led tasks. All led tasks then wait for the above described period of time before showing the light. Exact locations for the x were found using trial and error.

We did not use an alarm or the letters. We only became aware of this potential solution too late into the assignment, and decided against redoing the assignment due a lack of time.

Ex09c

Exercise 09c is almost the same as 09b. The main difference in the code is that the rotation time is halved in order draw the x at the bottom. Minor adjustments had to be made to correct the drawing of x from ex09b for this exercise.

Ex09d

In order to be able to rotate the x along the clock, a new task was introduced. This task is periodic and activates every single second. When activated it adds to a variable named timeshift, which is 1/60th of the rotation time. All the led light tasks wait the time defined in the previous tasks + the timeshift value. When this has happend 60 times, the timeshift value is reset to zero, just as a normal clock would. An additional check is build into each led light task: if the time is over the rotation time then the rotation time is subtracted from the periodtime + timeshift otherwise the time the task sleeps is periodtime + timeshift.