Embedded systems: Tutorial 4, software architectures

24 October

TP3, exercise 1, erratum

A circuit is composed of two LEDs and a microcontroller equipped with a timer with tunable frequency. The frequency of this timer can be set to

- 24 kHz,
- 8 kHz,
- 6 kHz, or
- 1 kHz.

LEDs ~~should respectively blink~~ should have their respective states toggled at 8 kHz and 3 kHz. How can we program a solution for this, without any busywaiting, if we assume that the instruction clock frequency of the MCU is 1 MHz and that the timer sends an interrupt request at each tick?

Tips for choosing the architecture

Decision guidelines:

1. Make an inventory of all tasks.
2. Identify those that can be (or need to be) performed by interrupt routines.
3. Is preemption needed? (Yes $\rightarrow$ RTOS).
4. Are interrupts needed? (No $\rightarrow$ RR).
5. Is static CPU assignment acceptable? (Yes $\rightarrow$ RR/I, No $\rightarrow$ WQ).

Important points:

1. Only urgent and short operations in interrupt routines!
2. When peripherals are busy, the processor can do something else.

Boat safety sytem

We want to program a system responsible for managing sensors on a boat. This system is equipped with six sensors that asynchronously send data to the microcontroller. To tell that a new measurment is available, the corresponding sensor sends an interrupt request and the microcontroller is responsible for fetching the corresponding data. This is assumed to take negligible time. Sensors may send data at any time, with variable frequency. As soon as information coming from a sensor has been fetched by the MCU, it has to be processed (this takes around 100 μs). Then a summary has to be sent every 10 ms. This amounts to copying a few bytes to registers and setting flags, it can be assumed instantaneous. If the measurment obtained by some sensor could be the sign of a life-threatening situation, an alarm should immediately be sounded. This is done by setting a dedicated pin to a high voltage. Moreover, the six sensors monitor different physical quantities and some are better estimators of a hazard than others. We will assume that sensor 1 is the most critical one, just before sensor 2, down to sensor 6.

What software architecture is best suited for this problem?

Give a pseudo-code of this architecture that is precise enough to illustrate the different tasks, their communication objects as well as interrupt routines (if some are needed).

Robotic vision and control

We want to program the system responsible for processing image sensed by a robot, deciding what action to perform accordingly and controlling the two motors of the robot.
There is a camera sending frames one after the other through a series communication protocol. A new frame is sent every 100 ms. The microcontroller is equipped with a peripheral able to acquire each frame. When a new frame has been saved in the MCU's memory, an interrupt request is sent by this peripheral. Processing an image is a complex task that typically takes 10 ms, the output of this computation is the target angle of the robot. On the other hand, each motor has to be continuously controlled. The speed of each motor is controlled by a PWM signal. To have some feedback on the actual speed of each wheel, 8 magnets are attached to it and a coil is connected to a microcontroller pin. Each time a magnet passes by the coil, a pulse is obtained on the corresponding MCU pin. The rotating speed of the wheel is estimated by measuring the time elapsed between two pulses. We assume that the signal is clean enough to trigger an interrupt on a rising edge of each pulse. The typical rotational speed of a wheel is 60 rpm. The duty cycle of each motor has to be recomputed every 5 ms and takes at most 500 μs.
What is the best architecture for this problem?
Give a pseudo-code of this architecture that is precise enough to illustrate the different tasks, their communication objects as well as interrupt routines (if some are needed).