

MES Homework week 4

René Rosendahl

Introduction

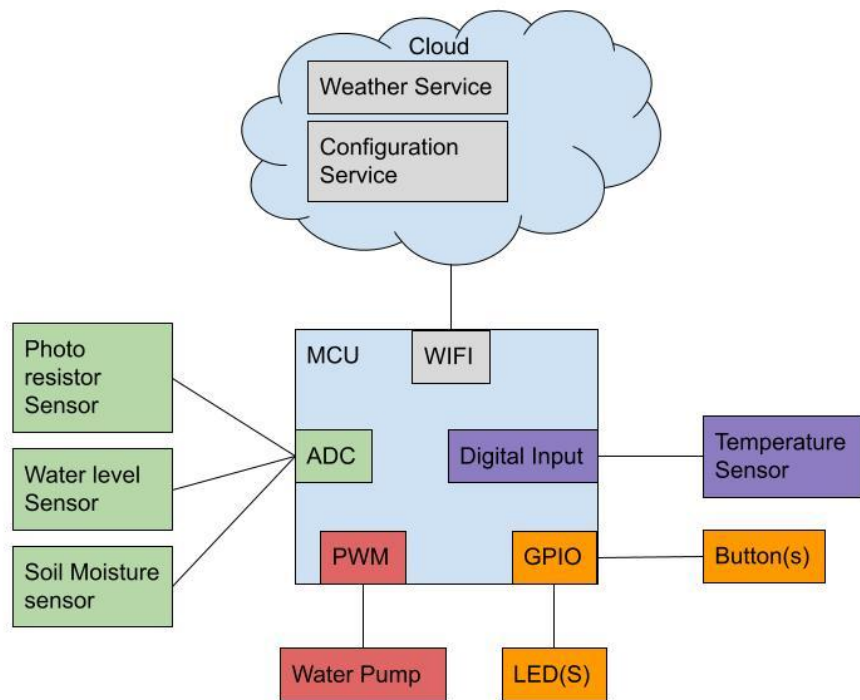
For my final project I want to make an automatic watering system. This is a quite classic project and might not push any boundaries for innovation, but for me the most important part of the course and project is to learn Embedded programming. With this project I will use different peripherals, so I imagine I will learn a lot.

Hardware Block Diagram

For now I am planning to use the Disco board that was handed out as part of the course(STM32F411VET6)

I want to try to use these peripherals:

- Soil moisture sensor - To measure how dry or wet the soil is around the plant, so we water when it starts to dry up and stop when it is wet.
- Temperature and Photo resistor sensors - To “detect” if it is a warm day and if the sun is out. In general it is better to water the plant during night time.
- Water level sensor - I imagine to have a bucket of water to get water from, and this sensor can measure how much water is left.
- Button - I will use the button on the Disco board to switch the system between states of e.g. idle and active.
- LEDs - I will use the LEDs on the Disco board to indicate different state os the system e.g. idle, active/collecting data, watering, failure etc.
- Water pump - To actually pump water out of the bucket into the plant
- Wifi - This is an idea for the future where the watering system is only collecting information and a cloud service calculates if it needs to start watering or not (If I want to do this I should consider using a less powerful MCU and one where Wifi is build into the chip)



State Machine Table

I have created my state table so the green rows indicates states you can go from, and blue is the states you can go to. The data within this is the transitions between states. X denotes that it is not possible, ? denotes that I haven't figured out yet.

This is a high level state table and within e.g. "Collecting Data" I believe I can make a mini state machine to decide if watering shall happen or not.

From States -> To States	Off	Booting	Idle	Activate	Collecting Data	Watering	Turning Off	Fail state
Off	X	System receives power	X	X	X	X	Pull Power	
Booting	Pull Power	X	Board is powered up Pheriphirals checked for connection	X	X	X	Pull Power	Board is not powered correctly 1 or more pheriphirals not working
Idle	Pull Power	X	X	Short push User button	X	X	Pull Power Hold user button	?
Activate	Pull Power	X	Short push User button	X	Start data flow Store data	X	Hold user button	data flow cannot start on 1 or more sensors
Collectin g data	Pull Power	X	Short push User button	X	X	Soil is too dry Water left in the tank Not much daylight Not too hot	Hold user button	
Watering	Pull Power	X	Short push User button	X	X	X	Hold user button	?
Turning Off	Power down sequence complete	X	X	X	X	X	X	?
Fail state	?	?	?	?	?	?	?	X

Software architecture

The software architecture is build up around a state machine that uses a layered structure to hide away some of the complexity of the hardware.

I do not expect to use an RTOS, as this prototype system is rather small and simple plus I believe that is a whole exercise in itself.

