Software Detailed Design Requirements

# Résultat de recherche d'images pour "sambot esigelec"Introduction

A small bot on wheels needs to be designed. It should be able to move itself in an environment containing obstacles.

The obstacles can be detected and avoided thanks to an ultrasound sensor placed on a servomotor (sweeping).

Holes must be detected too with an infrared sensor (to prevent falls).

This document lists all the **system requirements**, for the **software only.**

Every requirement is composed of:

* One unique ID following this pattern: SYS\_XXXXX (Five digits),
* A name, which is always a small introduction of the requirement,
* A text, describing what is this requirement for.

# Software Detailed Design Requirements

DDR\_00100

Name: Moving forward

Text: if the user press the “move forward” key or if the robot is in autopilot mode and there is no obstacle and no hole, then the MSP430G2553 should use a timer to provide a PWM signal in order to drive both the wheels clockwise.

Function: R\_avancer.

DDR\_00110

Name: Turn right

Text: if the user press the “move right” key, then the MSP430G2553 should use a timer to provide a PWM signal in order to drive the right wheel counterclockwise and the left wheel clockwise.

Covers: SYS\_00100

Function: R\_tourner\_droite

DDR\_00120

Name: Turn left

Text: if the user press the “move left” key or if the robot is in autopilot mode and there is an obstacle or a hole, then the MSP430G2553 should use a timer to provide a PWM signal in order to drive the left wheel counterclockwise and the right wheel clockwise.

Covers: SYS\_00100

Function: R\_tourner\_gauche

DDR\_00130

Name: move backward

if the user press the “move backward” key, then the MSP430G2553 should use a timer to provide a PWM signal in order to drive both the wheels counterclockwise

Covers: SYS\_00100

Function: R\_reculer

DDR\_00200

Name: Get the distance with obstacle in front of the ultrasound sensor

Text: The MSP430G2231 should acquire data from the ultrasound sensor continuously.

Function: get\_distance\_ultrason

DDR\_00210

Name: Measure infrared sensor

Text: The MSP430G2231 should acquire data from the infrared sensor continuously.

Function: get\_distance\_infra

DDR\_00220

Name: Obstacle considered detected

Text: The MSP430G2553 should considered an obstacle detected if obstacle\_value is under a threshold

Function: main

DDR\_00230

Name: Obstacle considered detected

Text: The MSP430G2553 should considered a hole detected if hole\_value is under a threshold

Function: main

DDR\_00300

Name: Bot start autopilot mode

Text: If the user send a command ‘1’ through UART communication to the MSP430G2553, the variable on should be set to 1

Function: USCIABORX\_ISR (UART interruption)

DDR\_00310

Name: Bot stop autopilot mode

Text: If the user send a command ‘0’ to the MSP430G2553, the variable on should be set to 0

Function: USCIABORX\_ISR (UART interruption)

DDR\_00320

Name: Data stop display

Text: If the user send a command ‘2’ to the MSP430G2553, the variable log should be set to 0

Function: USCIABORX\_ISR (UART interruption)

DDR\_00330

Name: Data start display

Text: If the user send a command ‘3’ to the MSP430G2553, the variable log should be set to 1

Function: USCIABORX\_ISR (UART interruption)

DDR\_00340

Name: servomotor on

Text: If the user send a command ‘l’ to the MSP430G2553, the variable servo should be set to 1

Function: USCIABORX\_ISR (UART interruption)

DDR\_00350

Name: servomotor on

Text: If the user send a command ‘m’ to the MSP430G2553, the variable servo should be set to 0

Function: USCIABORX\_ISR (UART interruption)

DDR\_00360

Name: command list

Text: If the user send a command ‘h’ to the MSP430G2553, then the MSP430G2553 should answer back the list of commands available

Function: USCIABORX\_ISR (UART interruption)

DDR\_00400

Name: MSP430G2553 SPI listenning

Text: The MSP430G2553 should implement a function to listen to the MSP430G2231 when it answers back. Data coming from the MSP430G2231 is written on 8 bits. If the BIT #8 is 1 then it should compute the data written on the other BITS and put it in “hole\_value” variable. If the BIT #8 is 0 then the data should be put in “obstacle\_value” variable

Function: **USCIAB0RX\_ISR** (called on SPI interruption)

DDR\_00410

Name: MSP430G2231 SPI (ultrasound sensor)

Text: If the MSP430G2231 receive the command ‘x’ from the MSP430G2553 through the SPI communication, then it should answer back the value of the ultrasound sensor: BIT #8 should be set as 0 and the data should be written on the 7 others BITs (max value = (2^7)-1 = 127)

Function: universal\_serial\_interface(SPI interruption)

DDR\_00420

Name: MSP430G2231 SPI (infrared sensor)

Text: If the MSP430G2231 receive the command ‘y’ from the MSP430G2553 through the SPI communication, then it should answer back the value of the infrared sensor : BIT #8 should be set as 1 and the data should be written on the 7 others BITs (max value = (2^7)-1 = 127)

Function: universal\_serial\_interface(SPI interruption)

DDR\_00500

Name: 2553 Timer

Text: The MSP430G2553 should use a timer to do operations reccurently such as :

* send char ‘x’ to the MSP430G2231 (through SPI communication).
* send char ‘y’ to the MSP4302231 (through SPI communication).
* Execute baylayage() function to make a sweep with the servomotor
* Log sensors data to the users terminal (through UART communication)

Function: ma\_fnc\_timer (called on Timer interruption)

DDR\_00600

Name: Sweep obstacle

Text:

**If the servomotor rotates clockwise:**

If it hasn't reach 45° then it shall keep rotating clockwise else it should rotate counterclockwise.

**If the servomotor rotates counterclockwise :**

If it hasn't reach -45° then it shall keep rotating counterclockwise else it should rotate clockwise

Function: balayage