

Simulation d'un drone intelligent en 2D

Présentation-Projet-G1d06-IGI-3006

Djiré Aboubakar Malick et Adjé Axel

23 Octobre 2024

- 1 Description du projet
- 2 Différents scénarios
- 3 Configurations des GPIO
- 4 Déroulement du programme
- 5 Illustration du scénario

Description

Pour notre projet nous allons modéliser un drone en 2D. L'utilisateur commence à programmer le robot pour le faire avancer d'une certaine distance , lorsque le drone rencontre un obstacle qui traduit une instabilité , il prends l'initiative de le contourner ce qui traduit le retablissement de la stabilité, il reproduit ce même schema jusqu'a ce qu'il s'arrête .

scénario 1

1. L'utilisateur n'appuie pas sur le switch 1 durant la phase de programmation Dans ce cas le robot n'avance pas.

scénario 2

L'utilisateur appuis n fois sur 2.le switch 1 durant la phase de programmation et il n'y a aucun obstacle sur la trajectoire du robot. Dans ce cas le robot avance n fois.

scénario 3

L'utilisateur appuis n fois sur le switch 1 durant la phase de programmation et il y a un obstacle sur le coté gauche du robot. Dans ce cas le robot avance jusqu'à l'obstacle, il le detecte grace au bumper gauche, allume uniquement la led droite et tourne à droite. Il continu à avancer du nombre de pas qu'il lui reste à faire.

scénario 4

L'utilisateur appuis n fois sur le switch 1 durant la phase de programmation et il y a un obstacle sur le coté droit du robot. Dans ce cas le robot avance jusqu'à l'obstacle, il le detecte grace au bumper droit, allume uniquement la led gauche et tourne à gauche. Il continu à avancer du nombre de pas qu'il lui reste à faire.

Configurations des GPIO

```
1 ; This register controls the clock gating logic in normal Run mode
2 SYSTCTL_PERIPH_GPIO EQU 0x400FE108 ; SYSTCTL_RCGC2_R (p291 datasheet de lm3s9B92.pdf)
3
4 ; Pulse Width Modulator (PWM) configuration
5 PWM_BASE EQU 0x040028000 ;BASE des Block PWM p.1138
6 PWMENABLE EQU PWM_BASE+0x008 ; p1145
7
8 ; The GPIODATA register is the data register
9 GPIO_PORTF_BASE EQU 0x40025000 ; GPIO Port F (APB) base: 0x4002.5000 (p416 datasheet de lm3s9B92.pdf)
10
11 ; The GPIODATA register is the data register
12 GPIO_PORTD_BASE EQU 0x40007000 ; GPIO Port D (APB) base: 0x4000.7000 (p416 datasheet de lm3s9B92.pdf)
13
14 GPIO_PORTE_BASE EQU 0x40024000
15
16 PORTH_BASE EQU 0x40027000
17 GPIODATA_H EQU PORTH_BASE
18
19 PORTD_BASE EQU 0x40007000
20 GPIODATA_D EQU PORTD_BASE
21
22 ; configure the corresponding pin to be an output
23 ; all GPIO pins are inputs by default
24 GPIO_O_DIR EQU 0x00000400 ; GPIO Direction (p417 datasheet de lm3s9B92.pdf)
25
26 ; The GPIODR2R register is the 2-mA drive control register
27 ; By default, all GPIO pins have 2-mA drive.
28 GPIO_O_DR2R EQU 0x00000500 ; GPIO 2-mA Drive Select (p428 datasheet de lm3s9B92.pdf)
29
30 ; Digital enable register
31 ; To use the pin as a digital input or output, the corresponding GPIODEN bit must be set.
32 GPIO_O_DEN EQU 0x0000051C ; GPIO Digital Enable (p437 datasheet de lm3s9B92.pdf)
33
34 ; Pul_up
35 GPIO_I_PUR EQU 0x00000510 ; GPIO Pull-Up (p432 datasheet de lm3s9B92.pdf)
36
```


Déroulement du programme

Selection des broches

```
30 BROCHE6_7      EQU      0xC0 ; bouton poussoir 1 et bouton poussoir 2
31
32 BROCHE0_1      EQU      0x03 ;bumper 1 et bumper 2
33
34 BROCHE4_5      EQU      0x30 ; led1 et 2 sur broche 4 5
35
36
```

Déroulement du programme

Durée et exportation des fonctions à définir

```
44 DUREE EQU 0x002FFFFFFF
45
46 DUREE_D EQU 0x9200c0
47
48 GPIO_1 EQU 0x2
49
50 AREA MyCode, CODE, READONLY
51 ENTRY
52
53 EXPORT LED_INIT
54 EXPORT SWITCH_INIT
55 EXPORT COUNT_CLICK
56 EXPORT CLOCK_INIT
57 EXPORT BUMPER_INIT
58 EXPORT ReadState BUMPER1
59 EXPORT ReadState BUMPER2
60
```

Déroulement du programme

Configurations horloge, leds, bumpers et switches

horloge

```
61
62 CLOCK_INIT
63     ldr r6, = SYSCTL_PERIPH_GPIO           ;; RCGC2
64     mov r0, #0x00000038                   ;; Enable clock sur GPIO D et F E o
65                                           ;; sont branchés les leds (0x38 == 0b111000)
66                                           ;; (GPIO::FEDCBA)
67     ; ;;
68     str r0, [r6]
69
70     ; ;; "There must be a delay of 3 system clocks before any GPIO reg. access (p413 datasheet de lm3s9B9)
71     nop                                   ;; tres tres important....
72     nop
73     BX LR
74
```

Déroulement du programme

Configurations horloge, leds, bumpers et switches

leds

```
75
76 LED_INIT
77     ;Set as output
78     ldr r6, = GPIO_PORTF_BASE+GPIO_O_DIR    ;; 1 Pin du portF en sortie (broche 4 : 00010000)
79     ldr r0, = BROCHE4_5
80     str r0, [r6]
81
82     ldr r6, = GPIO_PORTF_BASE+GPIO_O_DEN    ;; Enable Digital Function
83     ldr r0, = BROCHE4_5
84     str r0, [r6]
85
86     ldr r6, = GPIO_PORTF_BASE+GPIO_O_DR2R    ;; Choix de l'intensité de sortie (2mA)
87     ldr r0, = BROCHE4_5
88     str r0, [r6]
89
90     mov r2, #0x20
91     mov r9, #0x10
92
93     ; allumer la led broche 4 (PIN4)
94     mov r3, #BROCHE4_5    ;; Allume portF broche 4 : 00010000
95     ldr r6, = GPIO_PORTF_BASE + (BROCHE4_5<<2)    ;; @data Register = @base + (mask<<2) ==> LE
96
97     BX LR
98
```

Déroulement du programme

Configurations horloge, leds, bumpers et switches

switches

```
100 SWITCH_INIT
101     ldr r8, = GPIO_PORTD_BASE+GPIO_I_PUR    ;; Pul_up
102     ldr r0, = BROCHE6_7
103     str r0, [r8]
104
105     ldr r8, = GPIO_PORTD_BASE+GPIO_O_DEN    ;; Enable Digital Function
106     ldr r0, = BROCHE6_7
107     str r0, [r8]
108
109     ldr r8, = GPIO_PORTD_BASE + (BROCHE6_7<<2)    ;; @data Register = @base + (mask<<2) ==> Switch
110
111     BX LR
112
```

Déroulement du programme

Configurations horloge, leds, bumpers et switches

bumpers

```
114 BUMPER_INIT
115    ldr r7, = GPIO_PORTE_BASE+GPIO_I_PUR    ;; Pul_up
116    ldr r0, = BROCHE0_1
117    str r0, [r7]
118
119    ldr r7, = GPIO_PORTE_BASE+GPIO_O_DEN    ;; Enable Digital Function
120    ldr r0, = BROCHE0_1
121    str r0, [r7]
122
123    ldr r7, = GPIO_PORTE_BASE + (BROCHE0_1<<2) ;; @data Register = @base + (mask<<2) ==> Switcher
124
```

Déroulement du programme

Initialisation du compteur de click

code

```
125  
126 COUNT_CLICK  
127     ;Counts the number of time the button is pressed in 5 seconds  
128     mov r1, #0x00 ; initialisation du compteur de click  
129     mov r12, #0x00  
130     str r12, [r6]  
131
```

Déroulement du programme

Phases du programme

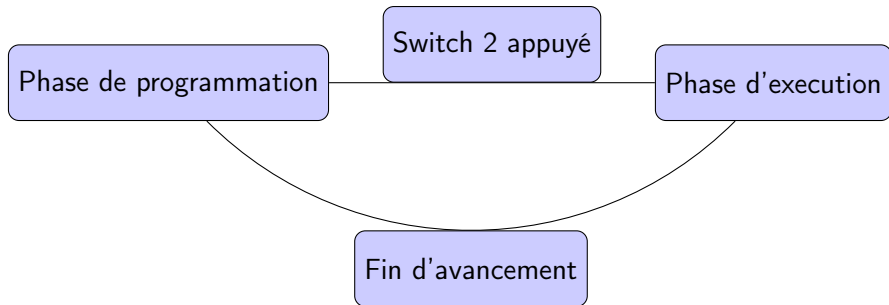


Illustration des différents scénarios

scénario 1 (cliquez sur l'image)



Illustration des différents scénarios

scénario 2 (cliquez sur l'image)



Illustration des différents scénarios

scénario 3 (cliquez sur l'image)



Illustration des différents scénarios

scénario 4 (cliquez sur l'image)



MERCI
😊