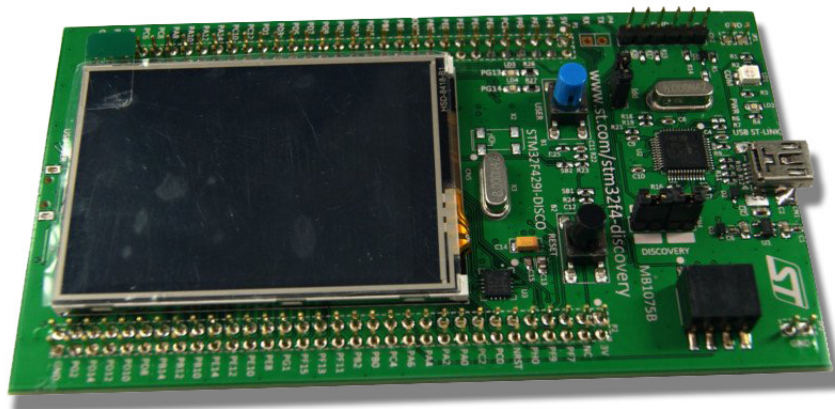


# laSalle

UNIVERSITAT RAMON LLULL

## Pràctica 1

Controlador de plataforma robòtica



Tecnologies en perifèrics

Curs 2018-2019

## Índex

<b>1</b>	<b>Enunciat</b>	<b>2</b>
<b>2</b>	<b>Memòria</b>	<b>5</b>
<b>3</b>	<b>Entrega</b>	<b>5</b>
<b>4</b>	<b>Consideracions</b>	<b>5</b>

## 1 Enunciat

Aquest any, el departament de robòtica de la universitat s'està preparant per competir en una *Hackathon* que té com a temàtica un robot de seguiment. Com no els convenç cap model comercial, han demanat ajuda als alumnes de l'assignatura de Tecnologies en perifèrics per a veure si els poden donar un cop de mà.

El que més els interessa actualment és crear una plataforma robòtica que permeti controlar el moviment del robot de forma precisa. Per facilitar-nos la feina, ens han donat les característiques dels motors que faran servir. La plataforma disposa doncs, de dues rodes, cadascuna d'elles accionada per un motor, i una tercera que pot girar lliurement. La trajectòria de la plataforma es controla variant la velocitat de rotació de cadascuna de les rodes motrius.

La plataforma es pot controlar mitjançant un emissor de radio-control de dos canals i el receptor corresponent, instal·lat en la mateixa plataforma. Les sortides del receptor són dues senyals PWM amb un període de 10ms i un duty cycle variable que codifica la velocitat i la direcció del motor i que es connectaran a dues línies qualssevol d'entrada del microcontrolador. Específicament en els PWM, un pols d'1,5ms significa motor aturat. Un pols de 2,3ms correspon a la velocitat màxima en la direcció d'avanç i un pols de 0,7ms correspon a la velocitat màxima marxa enrere. Amb aquest sistema es poden aconseguir velocitats de rotació diferents per cada roda i, per tant, fer girar la plataforma. Per il·lustrar aquest comportament se'ns facilita la següent figura.

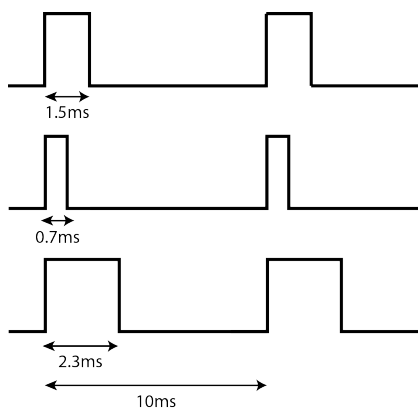


Figura 1: Esquema de funcionament del PWM del receptor de radio-control

La plataforma inclou també dos drivers que controlen els dos motors. Els senyals de control de cadascun dels drivers són un senyal de direcció de rotació del motor (0 = marxa endavant / 1 = marxa enrere) i un senyal de PWM amb una freqüència de 10kHz que controla la velocitat de rotació en la direcció seleccionada (0% = motor aturat / 100% = velocitat màxima). Aquests senyals de control de cada un dels drivers es connecten a quatre línies de sortida del microcontrolador. El programa de control del microcontrolador monitoritza els dos senyals de sortida del receptor i genera els quatre senyals d'entrada dels

dos drivers de motor (trieu algunes sortides de les disponibles en el connector d'expansió de la placa d'avaluació).

Per ajudar al desenvolupament del sistema cal fer un simulador del receptor del radio-control. El simulador genera dos senyals que simulen les sortides del receptor a partir de les dades introduïdes per l'usuari. Aquests dos senyals fan servir dues línies de sortida del microcontrolador. Per tal de fer les proves del sistema caldrà connectar amb un cable les dues línies de sortida del simulador amb les dues entrades seleccionades per a connectar el receptor en el sistema final. El disseny global es pot apreciar en el següent diagrama.

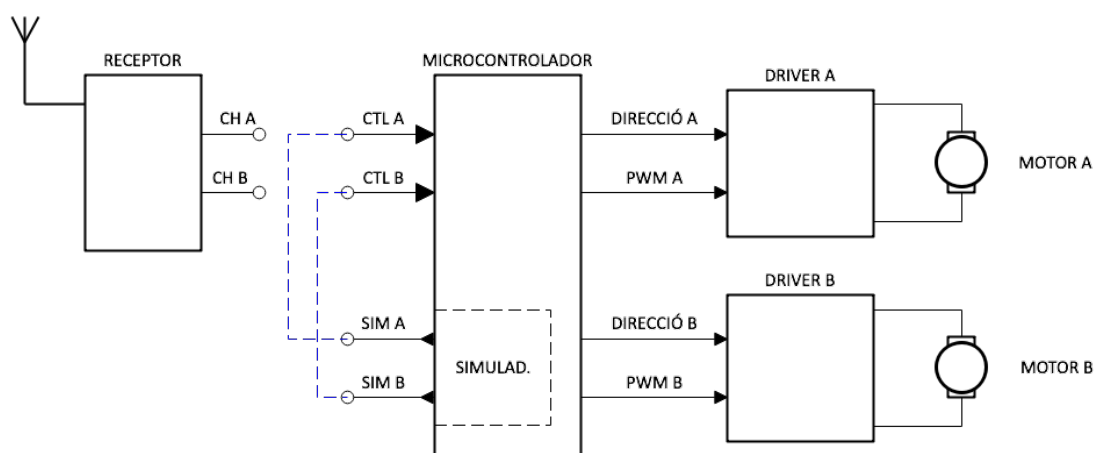


Figura 2: Diagrama global de la plataforma

Des del departament de professors, s'ha decidit utilitzar una placa de desenvolupament de *STMicroelectronics* STM32f429 per la implementació d'aquesta pràctica. Per a simular el control remot del robot i comprovar el correcte funcionament del sistema, ens ajudarem d'un dels LEDs i del pulsador per a introduir les dades de configuració. El flux del programa serà el següent:

1. En inicialitzar-se el microcontrolador, el controlador ha de generar els senyals pertinents a l'estat de repòs dels motors de forma interrompuda.
2. Seguidament, el LED verd començarà a fer pampallugues de forma ràpida, indicant a l'usuari que pot configurar el motor dret mitjançant el pulsador.
3. Cada cop que l'usuari faci una pulsació curta (<2.5seg), s'incrementarà un comptador que anirà lligat amb la velocitat i direcció del motor.
4. Si l'usuari ha mantingut premut el botó per més de 2.5seg, el LED verd començarà a fer pampallugues de forma lenta, indicant que es pot configurar el motor esquerre.
5. De la mateixa manera que abans, una pulsació curta incrementarà un altre comptador, i una llarga permetrà sortir del mode de configuració.
6. Quan ja s'hagin configurat ambdós motors, es canviaran els senyals PWM (SIM A i SIM B) per adequar-los a les noves dades introduïdes.

7. El controlador, que en tot moment estarà llegint les entrades provinents del simulador, veurà el canvi en el duty cycle dels PWM i modificarà, en conseqüència, els senyals que envia als drivers dels motors.
8. El programa esperarà una nova pulsació per part de l'usuari per a tornar a iniciar el procés de configuració (punt 2).

Pel que fa al nombre de pulsacions, es facilita la següent equivalència entre el comptador i l'estat del motor.

1	4	5	6	9
velocitat mínima endavant	velocitat màxima endavant	repòs	velocitat mínima endarrere	velocitat màxima endarrere

Els valors que falten són valors intermedis entre velocitats mínima i màxima. Aquesta taula és cíclica, és a dir, la 10a pulsació serà interpretada com un 1, la 11a com un 2, etc. Podeu assumir que si l'usuari no prem cap cop el pulsador, és equivalent a configurar l'estat de repòs.

Per a que el controlador sigui capaç de detectar els canvis de duty cycle dels senyals *SIM A* i *SIM B*, s'han de connectar aquestes sortides a dos línies d'entrada del microcontrolador capaces de generar interrupcions i implementar dos rutines de servei d'interrupcions (RSI) associades que siguin capaces de mesurar amb bona resolució temporal el duty cycle dels PWM generats anteriorment.

A més a més, es demanen les següents qüestions:

1. Comprovar que els temps mesurats per les RSI siguin correctes, i per tant, que els senyals d'entrada dels drivers dels motors s'adeqüin al que ha configurat l'usuari.
2. Mesurar la durada d'una RSI. Es pot implementar mitjançant una línia d'entrada/sortida que senyalitzi l'inici i el final de la RSI.
3. Mesurar la latència de la interrupció mitjançant el mateix procediment que a l'apartat anterior.
4. Realitzar una estimació de l'error màxim en la velocitat dels motors que es podria tenir si arribessin dues interrupcions simultàniament.
5. Justificació de les prioritats de totes les interrupcions configurades.
6. Calcular l'ample de banda de la CPU utilitzat per la gestió de la interrupció quan la velocitat és mínima i endarrere (0.7ms) i quan és màxima i endavant (2.3ms).

És important llegir les consideracions abans de començar a estructurar la pràctica.

## 2 Memòria

Caldrà realitzar una memòria que contingui els següents apartats:

1. Descripció de l'arquitectura del programa.
2. Diagrama de mòduls on constin tots aquells emprats en la pràctica i com es relacionen entre sí.
3. Descripció dels mòduls del controlador utilitzats i dels respectius processos de configuració/inicialització degudament descrits.
4. Resultats de les qüestions plantejades en l'enunciat.
5. Captures de pantalla i formes d'ona mesurades a l'oscil·loscopi que demostrin el correcte funcionament del sistema.
6. Conclusions.

## 3 Entrega

El format d'entrega ha de constar d'un únic fitxer ZIP que inclogui en primer lloc el codi font del programa en C correctament documentat i en segon lloc la memòria en format PDF. El nom del fitxer haurà de ser *login1\_login2\_P1.zip*, on *login1* i *login2* seran els *logins* dels integrants del grup.

La data límit d'entrega d'aquesta pràctica és el **diumenge 7 d'Abril de 2019**.

## 4 Consideracions

Aquesta pràctica es pot realitzar de forma individual o en grups de dues persones. Únicament es requereix d'una placa de desenvolupament STM32F429I per grup.

La implementació de la pràctica únicament requereix utilitzar alguns dels mòduls disponibles al microcontrolador. Es recomana utilitzar el manual *ST32Fxx Reference Manual - en.DM00031020*, disponible al eStudy, concretament els capítols: *General-purpose I/Os (GPIO)* (pàg. 269), *Interrupts and events* (pàg. 373) i *General-Purpose Timers* (pàg. 589).

Pel que fa a la implementació de la pràctica:

- Es valorarà la qualitat i estructura del codi.
- No es podran fer servir llibreries externes com ara *HAL*. Només es podran fer servir aquelles llibreries pròpies de la nostra arquitectura, com ara *stm32f4xx.h*.
- A l'hora de fer l'entrevista es farà servir una placa auxiliar que simularà el comportament del receptor.
- Per comptar temps, només es podrà fer servir un únic *timer free running*.