



UNIVERSIDAD DE SEVILLA  
ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA  
DEPARTAMENTO DE INGENIERÍA DE SISTEMAS Y AUTOMÁTICA

PROYECTO FINAL DE CARRERA  
INGENIERO DE TELECOMUNICACIÓN

---

**Diseño e Implementación de Interfaz  
Persona Máquina y Sistema de Análisis  
Post-Misión para robot autónomo móvil  
Romeo-4R**

---

*Autor:*

D. DANIEL PÉREZ RODRÍGUEZ

*Tutor:*

Dr. D. JESÚS IVÁN MAZA ALCAÑIZ

*Ponente:*

Dr. D. LUIS MERINO CABAÑAS

Enero, 2013



Este documento ha sido generado con L<sup>A</sup>T<sub>E</sub>X.



A mis padres,  
por ser responsables de ser como soy ...



Proyecto realizado en el Departamento de Ingeniería de Sistemas y Automática

<http://www.esi2.us.es/ISA/GAR/>

En colaboración con el Grupo de Robótica, Visión y Control, dirigido por el  
Catedrático Dr. D. Anibal Ollero Baturone

<http://grvc.us.es/>





## AGRADECIMIENTOS

En primer lugar, me gustaría agradecer sinceramente a todas las personas que componen el Grupo de Robótica, Visión y Control de la Escuela Superior de Ingenieros de Sevilla, liderados por el catedrático del Departamento de Ingeniería de Sistemas y Automática, Dr. D. Aníbal Ollero Baturone, por darme la oportunidad de aprender y trabajar a su lado y hacer de todo este tiempo una maravillosa experiencia de aprendizaje, tanto profesional como personal.

Agradecer, ya de forma más específica a mi tutor en el proyecto URUS y director de este proyecto fin de carrera, el Dr. D. Luis Merino Cabañas, por sus indicaciones durante el trabajo realizado y agradecer también a mis compañeros Francisco Real Pérez, Pablo Soriano Tapia y Miguel Ángel Rodríguez, por tener paciencia conmigo y aportarme tanto durante este tiempo en URUS y en otros proyectos.

Mencionar sin ninguna duda el apoyo y conocimientos recibidos de los profesores del Departamento de Ingeniería de Sistemas y Automática de la Escuela Superior de Ingenieros de Sevilla, principalmente al Dr. D. Iván Maza Alcañiz, tutor de este proyecto fin de carrera y al Dr. D. Fernando Caballero Benítez por toda la experiencia aprendida en múltiples proyectos así como al Dr. D. Guillermo Heredia Benot, al Dr. D. Ángel Rodríguez Castaño y al resto de profesores del Grupo de Robótica, Visión y Control.

Gracias también al resto de compañeros y miembros del grupo con lo que mi trabajo y día a día ha sido más cercano, Víctor, Jesús, Roberto, José Antonio, Sabine, Adrián, ... y alguno que otro que seguro me dejo en el tintero.

Tengo que buscar un hueco los habitantes de la Palmilla, Vito, Jordi, Fran, Eugenio, ... Ellos saben por qué están aquí.

Sin duda no se me ocurre mejor forma de casi acabar esta página de agradecimientos que dándole las gracias a mi familia, a mis padres y a mi hermano, porque sin duda todo lo que soy y todo lo que este trabajo significa no habría tenido sentido sin su apoyo y cariño a lo largo de todo este tiempo, que no ha sido poco.

El último gesto de agradecimiento no podría ser para otra persona que para ti Mari Carmen. No tengo palabras para agradecerte el apoyo y cariño que durante tanto tiempo han sido el motor que ha impulsado mi vida y la brújula que siempre me ha ayudado a encontrar el camino, deseando que sigan siéndolo en las nuevas aventuras que encontraremos en nuestro camino juntos.



## PREFACIO

La introducción de la robótica en la vida cotidiana de los seres humanos es un proceso que continua creciendo. Existen múltiples iniciativas y proyectos que apuestan y tratan de avanzar en este campo, como es el caso de proyecto europeo URUS [8] *Ubiquitous Networking Robotics in Urban Settings*, que trata de desarrollar nuevas formas de cooperación entre redes de robots y sensores en zonas peatonales del área urbana.

Este proyecto fin de carrera se centra en recoger el trabajo desarrollado por el autor en el contexto del proyecto URUS, desde septiembre de 2008 hasta junio de 2009, a través de una beca en el Grupo de Robótica, Visión y Control, participante activo del proyecto URUS. Dicho trabajo consistió principalmente en el diseño y realización de una interfaz hombre máquina para el control del robot Romeo-4R, así como alguna labor de soporte y apoyo en otros aspectos del mantenimiento y desarrollo de los módulos del robot, destacando en estos aspectos el desarrollo del sistema de análisis Post-Misión, o aplicación Logplayer, para la reproducción de los experimentos realizados en la realidad en un entorno de simulación.

Por ello, el primer capítulo de la memoria se dedica a introducir la robótica móvil, los problemas principales que aparecen en la navegación autónoma y específicamente en entornos urbanos; también dedicamos una introducción al diseño de interfaces hombre máquina/computador, así como otra breve descripción de los aspectos y resultados más importantes del proyecto URUS para acabar recapitulando los objetivos principales del proyecto fin de carrera, consistentes en las dos aplicaciones que ya hemos anticipado en el párrafo anterior.

El segundo capítulo se dedica a una completa descripción tanto a nivel hardware como software del robot Romeo-4R. Dicho conocimiento del sistema es básico para realizar la integración de la interfaz hombre máquina en el robot.

El tercer capítulo recoge una descripción de las principales librerías software, basadas completamente en software libre, utilizadas en los módulos del robot y por lo tanto en la propia interfaz, que como se verá, se entenderá como un módulo más del sistema completo.

El cuarto capítulo realiza una descripción completa del diseño y funcionamiento de la interfaz gráfica Romeo HMI mientras que el quinto capítulo realiza una descripción análoga del Sistema de Análisis Post-Misión, o módulo Logplayer.

El sexto capítulo recoge las principales conclusiones y resultados obtenidos en el desarrollo de la interfaz y del módulo Logplayer y de su uso posterior en los experimentos finales del proyecto URUS.



# Índice general

<b>1. Introducción</b>	<b>21</b>
1.1. Introducción a la robótica . . . . .	21
1.2. Robótica móvil . . . . .	28
1.3. Navegación en entornos urbanos . . . . .	31
1.4. Interacción humana con el robot . . . . .	32
1.5. Proyecto URUS . . . . .	36
1.6. Objetivos del proyecto fin de carrera . . . . .	37
<b>2. Vehículo Autónomo Romeo-4R</b>	<b>39</b>
2.1. Introducción . . . . .	39
2.2. Equipamiento Hardware . . . . .	41
2.3. Arquitectura Software . . . . .	45
<b>3. Librerías Software</b>	<b>55</b>
3.1. Introducción . . . . .	55
3.1.1. Comunicaciones . . . . .	55
3.1.2. Interfaces gráficas . . . . .	56
3.1.3. Visión por Computador . . . . .	57
3.2. YARP . . . . .	58
3.2.1. Definiciones . . . . .	60
3.2.2. Propiedades de una red YARP . . . . .	61
3.2.3. Gestión de los puertos . . . . .	63
3.2.4. Empaquetamiento de la información . . . . .	63
3.3. Qt . . . . .	63
3.4. OPENCV . . . . .	71
<b>4. Interfaz Romeo HMI</b>	<b>75</b>
4.1. Introducción . . . . .	75
4.2. Definición de clases principales . . . . .	76
4.2.1. RomeoMainWindow . . . . .	76
4.2.2. MapScene . . . . .	78
4.3. Variables . . . . .	83
4.4. Descripción de la interfaz gráfica . . . . .	86

<b>5. Sistema de Análisis Post-Misión (Logplayer)</b>	<b>93</b>
5.1. Introducción . . . . .	93
5.2. Definición de Clases . . . . .	95
5.2.1. LogPlayer . . . . .	95
5.2.2. PlayerFactory . . . . .	96
5.2.3. TimeReference . . . . .	96
5.2.4. TypedPlayer<T> . . . . .	99
5.3. Variables . . . . .	100
5.4. Descripción de la aplicación . . . . .	101
5.4.1. Opciones línea de comandos . . . . .	101
5.4.2. Fichero de Configuración . . . . .	101
<b>6. Conclusiones y Resultados</b>	<b>103</b>
6.1. Conclusiones . . . . .	103
6.2. Futuras líneas de desarrollo . . . . .	104
6.2.1. Mejoras en la interfaz gráfica actual . . . . .	104
6.2.2. Interfaz gráfica para usuarios finales . . . . .	104
6.3. Resultados obtenidos en Proyecto URUS . . . . .	105
6.4. Uso de las lecciones aprendidas en posteriores proyectos . . . . .	110
<b>A. Romeo HMI - Include dependency graph for main.cpp</b>	<b>111</b>
<b>B. LogPlayer - Include dependency graph for main.cpp</b>	<b>113</b>
<b>C. LogPlayer - Mecanismo de Sincronización de Logs</b>	<b>115</b>

# Índice de figuras

1.1. Robot y su interacción con el entorno . . . . .	22
1.2. Robot Konabot . . . . .	23
1.3. Robot KUKA KR150 . . . . .	24
1.4. Robots industriales Staubli RX90. Laboratorio de Sistemas y Automática . . . . .	26
1.5. Robot móvil Romeo-4R. Experimento de evitación de obstáculo .	26
1.6. Izquierda) Robot Tibi (Universidad Politécnica de Cataluña). Derecha) Robot Asimo (Honda) . . . . .	27
1.7. Robot Big Dog. Boston Dynamics . . . . .	27
1.8. Proyecto URUS. Trayectorias de Robots . . . . .	29
1.9. Proyecto URUS. Romeo-4R World Laser Generation . . . . .	31
1.10. Interfaz de uso del Buscador Google . . . . .	34
1.11. Interfaz de uso Herramienta Dolphin Plus . . . . .	34
1.12. Disciplinas relacionadas con la IPO . . . . .	35
1.13. Proyecto URUS. Interfaz de Usuario Robot Tibi . . . . .	36
1.14. Proyecto URUS. Identificación de persona realizando una petición al sistema . . . . .	36
1.15. Proyecto URUS. Tracking de personas mediante red de cámaras IP	37
1.16. Equipo del Proyecto URUS . . . . .	38
2.1. Robot Romeo-4R . . . . .	40
2.2. Sistemas de referencia global y local . . . . .	40
2.3. Particularización de los sistemas de referencia para el caso plano .	41
2.4. Robot Romeo-4R. Sensores (I) . . . . .	43
2.5. Robot Romeo-4R. Sensores (II) . . . . .	43
2.6. Láser Hokuyo URG-04LX . . . . .	44
2.7. Láser Hokuyo UTM-30LX . . . . .	45
2.8. Videocámara DFK 21BF04 . . . . .	45
2.9. Red YARP - Modo peer to peer . . . . .	46
2.10. Conexiones módulos software Romeo-4R y puertos YARP . . . . .	48
2.11. Mapa generado sensorialmente . . . . .	51
2.12. Proyecto URUS. Mapas de Elevación y Transversabilidad . . . . .	52
2.13. Splashscreen Interfaz Romeo HMI . . . . .	53

3.1. Logo YARP . . . . .	58
3.2. Red YARP en varias máquinas y sistemas operativos . . . . .	62
3.3. Logo Qt . . . . .	64
3.4. Diagrama de conexión de señales y slots diversos objetos Qt . . . . .	69
3.5. Qt Designer . . . . .	70
3.6. OpenCV Logo . . . . .	71
3.7. Estructura librería OpenCV . . . . .	72
3.8. Proyecto URUS. Seguimiento visual de personas para experimento de guiado . . . . .	73
4.1. Interfaz gráfica Romeo HMI . . . . .	76
4.2. Diagrama de Herencia de clase RomeoMainWindow . . . . .	77
4.3. Llamadas desde el constructor RomeoMainWindow . . . . .	78
4.4. Diagrama de colaboración de clase RomeoMainWindow . . . . .	81
4.5. Gráfico de dependencias fichero mapscene.cpp . . . . .	81
4.6. Diagrama estados funcionamiento aplicación Romeo HMI . . . . .	86
4.7. Frames aplicación Romeo HMI . . . . .	87
4.8. Monitor del sistema . . . . .	87
4.9. QToolBox Frame izquierdo . . . . .	88
4.10. Controles visualización de los sensores . . . . .	89
4.11. Controles del Launcher . . . . .	89
4.12. Controles de inserción de trayectoria . . . . .	90
4.13. Accesos directos frame izquierdo . . . . .	90
4.14. Zona de visualización mapa (I) . . . . .	91
4.15. Zona de visualización mapa (II) . . . . .	91
4.16. Sensores Romeo HMI . . . . .	92
4.17. Mapa de elevación integrado en la visualización del mapa . . . . .	92
5.1. Módulo LOGPLAYER reemplazando sensores . . . . .	94
5.2. LogPlayer - Herencia TypedPlayer<T> . . . . .	99
5.3. Salida error consola LogPlayer . . . . .	101
5.4. Salida correcta consola LogPlayer . . . . .	102
6.1. Proyecto URUS. Recreación entorno 3D . . . . .	105
6.2. Proyecto URUS. Reconocimiento facial . . . . .	106
6.3. Proyecto URUS. Red cámaras IP - Identificación y seguimiento . . . . .	106
6.4. Proyecto URUS. Experimentos Navegación Romeo-4R . . . . .	107
6.5. Proyecto URUS. Redes de Sensores (I) . . . . .	108
6.6. Proyecto URUS. Redes de Sensores (II) . . . . .	108
6.7. Proyecto URUS. Experimento de guiado (I) . . . . .	109
6.8. Proyecto URUS. Experimento de guiado (II) . . . . .	109
6.9. Qt Multi UAV Ground Control Station . . . . .	110
A.1. Romeo HMI - Include dependency graph for main.cpp . . . . .	112

B.1. LogPlayer - Include dependency graph for main.cpp . . . . .	114
C.1. LogPlayer - Mecanismo de Sincronización de Logs . . . . .	115



# Índice de tablas

2.1. Proyecto URUS – Comms.h . . . . .	47
3.1. Plataformas con soporte Qt . . . . .	64
3.2. Módulos Qt . . . . .	65
3.3. Módulos Extra Qt en Sistemas Windows . . . . .	66
3.4. Clases Módulo QtCore . . . . .	66
3.5. Clases Módulo QtGui . . . . .	67
4.1. Proyecto URUS – main.cpp . . . . .	77
4.2. Proyecto URUS – romeomainwindow.h . . . . .	79
4.3. Proyecto URUS – ui-MainWindow-HMI2.h . . . . .	80
4.4. Proyecto URUS – ui-MainWindow-HMI2.h . . . . .	80
4.5. Proyecto URUS – romeomainwindow.cpp . . . . .	81
4.6. Proyecto URUS – mapscene.h . . . . .	82
4.7. Romeo HMI – Variables locales de localización del robot . . . . .	83
4.8. Romeo HMI – Variables de localización del robot - módulo EKFLOC . . . . .	83
4.9. Romeo HMI – Variables odométricas del robot - módulos IMU y DCX . . . . .	84
4.10. Romeo HMI – Variables vista gráfica del mapa . . . . .	84
4.11. Romeo HMI – Variables Monitor Estado del Sistema . . . . .	85
4.12. Romeo HMI – Variables estado aplicación . . . . .	85
4.13. Romeo HMI – Tabla de estados aplicación . . . . .	85
5.1. LogPlayer - Fichero de log dcx.log . . . . .	94
5.2. LogPlayer - Fichero de log imu.log . . . . .	95
5.3. LogPlayer – Declaración clase LogPlayer (LogPlayer.h) . . . . .	96
5.4. LogPlayer – Declaración clase PlayerFactory . . . . .	97
5.5. LogPlayer – Añadir nueva clase de comunicaciones en LogPlayer.cpp (PlayerFactory.h) . . . . .	98
5.6. LogPlayer – Declaración clase TimeReference (timeutil.h) . . . . .	98
5.7. LogPlayer – Declaración clase TypedPlayer (LogPlayer.h) . . . . .	99
5.8. LogPlayer – Lista de variables (main.cpp) . . . . .	100
5.9. Fichero de Configuración logPlayer.conf . . . . .	102

