pyCreate2 Documentation

Release spring2016

Wolfgang Hoenig, Arash Tavakoli, Nora Ayanian

Apr 06, 2017

Contents

pyCreate2 is a package to control an iRobot Create2 robot. It supports seamless integration with the V-REP robotics simulator. Scripts can be run without any changes in simulation and on the physical robot. The iCreate2 robot is extended with an embedded computer, allowing fully autonomous operation.

CHAPTER 1

Getting Starting

Simulation

1. Download pyCreate2 from the github repository
2. Download V-REP
3. Open example1.ttt in V-REP
4. Execute:

```
python3 run.py example1 --sim
```

Physical Robot

1. Modify your robot by following this section *Hardware*
2. Install Ubuntu on the ODROID
3. Copy the python files to the ODROID
4. Execute:

```
python3 run.py example1
```

Contents:

Software

Contents:

pyCreate2

pyCreate2 package

Subpackages

pyCreate2.robot package

Submodules

pyCreate2.robot.create2_driver module

pyCreate2.robot.gpio module

Module to use ODROID's GPIOs.

class pyCreate2.robot.gpio.**Gpio**(*number*)
Bases: object

Class to use general purpose input/output (GPIO) pins.

This class specifically targets the standard linux support for GPIO as provided by ODROID. More details can be found here: http://odroid.com/dokuwiki/doku.php?id=en:c1_hardware_pwm. The sysfs interface is described here: https://www.kernel.org/doc/Documentation/gpio/sysfs.txt. It supports setting the value and waiting for a hardware interrupt.

set_direction(*direction*)
Set the direction (input/output) of the pin.

Parameters **direction** (*string*) – One of "in", "out", "low", "high".

set_edge(*edge*)
Set the edge trigger for HW interrupt support. Use *wait_for_interrupt* to wait for an interrupt afterwards.

Parameters **edge** (*string*) – One of "none", "rising", "falling", "both"

set_value(*value*)
Set the current value of the pin (only valid if configured as output.)

Parameters **value** (*integer*) – 0 or 1.

wait_for_interrupt(*timeout_in_ms=1000*)
Waits until timeout or interrupt occurs.

Parameters **timeout_in_ms** (*integer*) – maximum time to wait for an interrupt

Returns None if timeout occurred or the current value of the pin in case the interrupt was triggered.

pyCreate2.robot.pwm module

Module to use ODROID's HW PWM.

class pyCreate2.robot.pwm.**Pwm**(*number=0*)
Bases: object

Class to use general pulse-width-modulation.

This class specifically targets the standard linux support for PWM as provided by ODROID. More details can be found here: http://odroid.com/dokuwiki/doku.php?id=en:c1_hardware_pwm. The sysfs interface is described here: https://www.kernel.org/doc/Documentation/pwm.txt. It supports setting the value and waiting for a hardware interrupt.

disable()
Disables PWM.

enable()
Enables PWM.

set_duty_cycle(*duty_in_percent*)
Set the duty cycle of the pulse width

Parameters **duty_in_percent** (*float*) – duty signal in percent (i.e. 0.0 to 100.0)

set_frequency(*frequency_in_hertz*)
Set the frequency of the pulse width

Parameters **frequency_in_hertz** (*integer*) – frequency in Hertz.

pyCreate2.robot.servo module

pyCreate2.robot.sonar module

Module to interface a PING Sonar connected via GPIO

class pyCreate2.robot.sonar.**Sonar**(*pin*)
Bases: object

Class to use the PING Sonar

This class assumes that the PING sonar is connected using a GPIO pin. It implements the protocol specified in https://www.parallax.com/sites/default/files/downloads/28015-PING-Documentation-v1.6.pdf using hardware interrupts.

Parameters **pin** – GPIO pin number where the sonar is connected to.

get_distance()
Queries the current distance from the sonar.

Starts a new measurement cycle (which takes up to 19 ms) and returns the measured distance in m.

Returns Distance in m. If there was an error, it returns 3.3 m.

Module contents

pyCreate2.simulation package

Submodules

pyCreate2.simulation.create2_vrep module

pyCreate2.simulation.kuka_lbr4p_vrep module

Module to control the KUKA LBR4+ in V-REP.

class pyCreate2.simulation.kuka_lbr4p_vrep.**KukaLBR4PlusVrep**(*client_id*)
Bases: object

Class to control the KUKA LBR4+ Manipulator in V-REP.

disable_painting()
Disable spray painting end-effector.

enable_painting()
Enable spray painting end-effector.

go_to(*joint*, *angle*)
Go to specified target angle.

Parameters

- **joint** (*int*) – number of joint to change (0 to 7)
- **angle** (*float*) – radians

set_color(*r*, *g*, *b*)
Set spray painting color (RGB).

Parameters

- **r** (*float*) – red component (0 to 1)
- **g** (*float*) – green component (0 to 1)
- **b** (*float*) – blue component (0 to 1)

pyCreate2.simulation.servo module

Module to control a virtual Servo.

class pyCreate2.simulation.servo.**Servo**(*client_id*)
Bases: object

Class to control a virtual servo in V-REP. The servo is modeled as joint, using an integrated position controller in V-REP.

go_to(*angle*)
Go to specified target angle.

Parameters **angle** (*float*) – -90 - 90 degrees. 0 means facing forward. Negative numbers turn to the left.

pyCreate2.simulation.sonar module

Module to control a virtual Sonar.

class pyCreate2.simulation.sonar.**Sonar**(*client_id*)
Bases: object

Class to control a virtual sonar.

get_distance()
Queries the current distance from the sonar.

We use a proximity sensor in V-REP to model the sonar.

Returns Distance in m. If there was an error, it returns 3.3 m.

pyCreate2.simulation.time_helper module

Module to deal with simulation time.

class pyCreate2.simulation.time_helper.**TimeHelper**(*client_id*)
Bases: object

This class is similar to the default time module of python, however it uses V-REPs simulation time rather than real time.

sleep(*wait_in_sec*)
Wait for the specified number of seconds (simulation time).

The simulation will continue making progress.

Parameters **wait_in_sec** (*float*) – time (in seconds) to continue simulation.

time()
Query current time. Simulation starts at time 0.

Returns Elapsed simulated seconds.

Module contents

pyCreate2.visualization package

Submodules

pyCreate2.visualization.virtual_create module

Module to control a virtual create

class pyCreate2.visualization.virtual_create.**VirtualCreate**(*client_id*)
Bases: object

Class to control a virtual create in V-REP.

class Button
Bases: enum.Enum

An enumeration.

MoveForward = 3

Sense = 6

TurnLeft = 4

TurnRight = 5

VirtualCreate.**disable_buttons**()

VirtualCreate.**enable_buttons**()

VirtualCreate.**get_last_button**()

VirtualCreate.**set_point_cloud**(*data*)

VirtualCreate.**set_pose**(*position*, *yaw*)

Module contents

class pyCreate2.visualization.**VirtualCreate**(*client_id*)
Bases: object

Class to control a virtual create in V-REP.

class Button
Bases: enum.Enum

An enumeration.

MoveForward = 3

Sense = 6

TurnLeft = 4

TurnRight = 5

VirtualCreate.**disable_buttons**()

VirtualCreate.**enable_buttons**()

VirtualCreate.**get_last_button**()

VirtualCreate.**set_point_cloud**(*data*)

VirtualCreate.**set_pose**(*position*, *yaw*)

pyCreate2.vrep package

Submodules

pyCreate2.vrep.vrep module

pyCreate2.vrep.vrep.**simxAddStatusbarMessage**(*clientID*, *message*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxAppendStringSignal**(*clientID*, *signalName*, *signalValue*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxAuxiliaryConsoleClose**(*clientID*, *consoleHandle*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxAuxiliaryConsoleOpen**(*clientID*, *title*, *maxLines*, *mode*, *position*, *size*, *textColor*, *backgroundColor*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxAuxiliaryConsolePrint**(*clientID*, *consoleHandle*, *txt*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxAuxiliaryConsoleShow**(*clientID*, *consoleHandle*, *showState*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxBreakForceSensor**(*clientID*, *forceSensorHandle*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxClearFloatSignal**(*clientID*, *signalName*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxClearIntegerSignal**(*clientID*, *signalName*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxClearStringSignal**(*clientID*, *signalName*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxCloseScene**(*clientID*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxCopyPasteObjects**(*clientID*, *objectHandles*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxCreateBuffer**(*bufferSize*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxCreateDummy**(*clientID*, *size*, *color*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxDisplayDialog**(*clientID*, *titleText*, *mainText*, *dialogType*, *initialText*, *titleColors*, *dialogColors*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxEndDialog**(*clientID*, *dialogHandle*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxEraseFile**(*clientID*, *fileName_serverSide*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxFinish**(*clientID*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxGetAndClearStringSignal**(*clientID*, *signalName*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxGetArrayParameter**(*clientID*, *paramIdentifier*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxGetBooleanParameter**(*clientID*, *paramIdentifier*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxGetCollisionHandle**(*clientID*, *collisionObjectName*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxGetConnectionId**(*clientID*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxGetDialogInput**(*clientID*, *dialogHandle*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxGetDialogResult**(*clientID*, *dialogHandle*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxGetDistanceHandle**(*clientID*, *distanceObjectName*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxGetFloatSignal**(*clientID*, *signalName*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxGetFloatingParameter**(*clientID*, *paramIdentifier*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxGetInMessageInfo**(*clientID*, *infoType*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxGetIntegerParameter**(*clientID*, *paramIdentifier*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxGetIntegerSignal**(*clientID*, *signalName*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxGetJointForce**(*clientID*, *jointHandle*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxGetJointMatrix**(*clientID*, *jointHandle*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxGetJointPosition**(*clientID*, *jointHandle*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxGetLastCmdTime**(*clientID*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxGetLastErrors**(*clientID*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxGetModelProperty**(*clientID*, *objectHandle*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxGetObjectChild**(*clientID*, *parentObjectHandle*, *childIndex*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxGetObjectFloatParameter**(*clientID*, *objectHandle*, *parameterID*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxGetObjectGroupData**(*clientID*, *objectType*, *dataType*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxGetObjectHandle**(*clientID*, *objectName*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxGetObjectIntParameter**(*clientID*, *objectHandle*, *parameterID*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxGetObjectOrientation**(*clientID*, *objectHandle*, *relativeToObjectHandle*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxGetObjectParent**(*clientID*, *childObjectHandle*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxGetObjectPosition**(*clientID*, *objectHandle*, *relativeToObjectHandle*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxGetObjectSelection**(*clientID*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxGetObjectVelocity**(*clientID*, *objectHandle*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxGetObjects**(*clientID*, *objectType*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxGetOutMessageInfo**(*clientID*, *infoType*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxGetPingTime**(*clientID*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxGetStringParameter**(*clientID*, *paramIdentifier*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxGetStringSignal**(*clientID*, *signalName*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxGetUIButtonProperty**(*clientID*, *uiHandle*, *uiButtonID*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxGetUIEventButton**(*clientID*, *uiHandle*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxGetUIHandle**(*clientID*, *uiName*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxGetUISlider**(*clientID*, *uiHandle*, *uiButtonID*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxGetVisionSensorDepthBuffer**(*clientID*, *sensorHandle*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxGetVisionSensorImage**(*clientID*, *sensorHandle*, *options*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxJointGetForce**(*clientID*, *jointHandle*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxLoadModel**(*clientID*, *modelPathAndName*, *options*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxLoadScene**(*clientID*, *scenePathAndName*, *options*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxLoadUI**(*clientID*, *uiPathAndName*, *options*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxPackFloats**(*floatList*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxPackInts**(*intList*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxPauseCommunication**(*clientID*, *enable*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxPauseSimulation**(*clientID*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxQuery**(*clientID*, *signalName*, *signalValue*, *retSignalName*, *timeOutInMs*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxReadCollision**(*clientID*, *collisionObjectHandle*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxReadDistance**(*clientID*, *distanceObjectHandle*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxReadForceSensor**(*clientID*, *forceSensorHandle*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxReadProximitySensor**(*clientID*, *sensorHandle*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxReadStringStream**(*clientID*, *signalName*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxReadVisionSensor**(*clientID*, *sensorHandle*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxReleaseBuffer**(*buffer*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxRemoveModel**(*clientID*, *objectHandle*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxRemoveObject**(*clientID*, *objectHandle*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxRemoveUI**(*clientID*, *uiHandle*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxSetArrayParameter**(*clientID*, *paramIdentifier*, *paramValues*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxSetBooleanParameter**(*clientID*, *paramIdentifier*, *paramValue*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxSetFloatSignal**(*clientID*, *signalName*, *signalValue*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxSetFloatingParameter**(*clientID*, *paramIdentifier*, *paramValue*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxSetIntegerParameter**(*clientID*, *paramIdentifier*, *paramValue*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxSetIntegerSignal**(*clientID*, *signalName*, *signalValue*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxSetJointForce**(*clientID*, *jointHandle*, *force*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxSetJointPosition**(*clientID*, *jointHandle*, *position*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxSetJointTargetPosition**(*clientID*, *jointHandle*, *targetPosition*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxSetJointTargetVelocity**(*clientID*, *jointHandle*, *targetVelocity*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxSetModelProperty**(*clientID*, *objectHandle*, *prop*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxSetObjectFloatParameter**(*clientID*, *objectHandle*, *parameterID*, *parameterValue*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxSetObjectIntParameter**(*clientID*, *objectHandle*, *parameterID*, *parameterValue*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxSetObjectOrientation**(*clientID*, *objectHandle*, *relativeToObjectHandle*, *eulerAngles*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxSetObjectParent**(*clientID*, *objectHandle*, *parentObject*, *keepInPlace*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxSetObjectPosition**(*clientID*, *objectHandle*, *relativeToObjectHandle*, *position*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxSetObjectSelection**(*clientID*, *objectHandles*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxSetSphericalJointMatrix**(*clientID*, *jointHandle*, *matrix*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxSetStringSignal**(*clientID*, *signalName*, *signalValue*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxSetUIButtonLabel**(*clientID*, *uiHandle*, *uiButtonID*, *upStateLabel*, *downStateLabel*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxSetUIButtonProperty**(*clientID*, *uiHandle*, *uiButtonID*, *prop*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxSetUISlider**(*clientID*, *uiHandle*, *uiButtonID*, *position*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxSetVisionSensorImage**(*clientID*, *sensorHandle*, *image*, *options*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxStart**(*connectionAddress*, *connectionPort*, *waitUntilConnected*, *doNotReconnectOnceDisconnected*, *timeOutInMs*, *commThreadCycleInMs*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxStartSimulation**(*clientID*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxStopSimulation**(*clientID*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxSynchronous**(*clientID*, *enable*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxSynchronousTrigger**(*clientID*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxTransferFile**(*clientID*, *filePathAndName*, *fileName_serverSide*, *timeOut*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxUnpackFloats**(*floatsPackedInString*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxUnpackInts**(*intsPackedInString*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrep.**simxWriteStringStream**(*clientID*, *signalName*, *signalValue*, *operationMode*)
Please have a look at the function description/documentation in the V-REP user manual

pyCreate2.vrep.vrepConst module

Module contents

Submodules

pyCreate2.create2 module

Helper enumerations which are both used for simulation and driver.

class pyCreate2.create2.**ChargingState**
Bases: enum.Enum

An enumeration.

ChargingFaultCondition = 5

ChargingStateWaiting = 4

FullCharging = 2

NotCharging = 0

ReconditioningCharging = 1

TrickleCharging = 3

class pyCreate2.create2.**InfraredCharacter**
Bases: enum.Enum

An enumeration.

DockForceField = 242

DockGreenBuoy = 244

DockGreenBuoyAndForceField = 246

DockRedBuoy = 248

DockRedBuoyAndForceField = 250

DockRedBuoyAndGreenBuoyAndForceField = 254

DockRedBuoyGreenBuoy = 252

class pyCreate2.create2.**Mode**
Bases: enum.Enum

An enumeration.

Full = 3

Off = 0

Passive = 1

Safe = 2

class pyCreate2.create2.**Op**
Bases: object

Baud = 129

Buttons = 165

Clean = 135

Control = 130

DigitsLedsAscii = 164

DigitsLedsRaw = 163

Drive = 137

DriveDirect = 145

DrivePwm = 146

Full = 132

Leds = 139

Max = 136

Motors = 138

PauseResumeStream = 150

Play = 141

Power = 133

PwmMotors = 144

QueryList = 149

Reset = 7

Safe = 131

Schedule = 167

SchedulingLeds = 162

SeekDock = 143

Sensors = 142

SetDayTime = 168

Song = 140

Spot = 134

Start = 128

Stop = 173

Stream = 148

class pyCreate2.create2.**Sensor**

Bases: object

Angle = 20

BatteryCapacity = 26

BatteryCharge = 25

BumpsAndWheelDrops = 7

Buttons = 18

ChargingSourcesAvailable = 34

ChargingState = 21

CliffFrontLeft = 10

CliffFrontLeftSignal = 29

CliffFrontRight = 11

CliffFrontRightSignal = 30

CliffLeft = 9

CliffLeftSignal = 28

CliffRight = 12

CliffRightSignal = 31

Current = 23

DirtDetect = 15

Distance = 19

InfraredCharacterLeft = 52

InfraredCharacterOmni = 17

InfraredCharacterRight = 53

LeftEncoderCounts = 43

LeftMotorCurrent = 54

LightBumpCenterLeftSignal = 48

LightBumpCenterRightSignal = 49

LightBumpFrontLeftSignal = 47

LightBumpFrontRightSignal = 50

LightBumpLeftSignal = 46

LightBumpRightSignal = 51

MainBrushMotorCurrent = 56

NumberOfStreamPackets = 38

OIMode = 35

RequestedLeftVelocity = 42

RequestedRadius = 40

RequestedRightVelocity = 41

RequestedVelocity = 39

RightBumper = 45

RightEncoderCounts = 44

RightMotorCurrent = 55

SideBrushMotorCurrent = 57

SongNumber = 36

SongPlaying = 37

Stasis = 58

Temperature = 24

VirtualWall = 13

Voltage = 22

WheelOvercurrents = 14

class pyCreate2.create2.**Specs**
Bases: object

CountsPerRev = 508.8

WheelDiameterInMM = 72.0

WheelDistanceInMM = 235.0

class pyCreate2.create2.**State**
Bases: object

pyCreate2.factory module

Module with factory methods for different objects (either real or simulation)

class pyCreate2.factory.**FactoryCreate**
Bases: object

Class to create objects which are related to the physical iRobot Create2 robot.

close()
Clean-up

create_create()
Instantiates a new create robot (only a single one is supported!)

Returns (robot.Create2Driver) instance of robot.Create2Driver

create_pen_holder()
Instantiates a new pen holder (only a single one is supported!)

Returns (robot.PenHolder) instance of robot.PenHolder

create_servo()
Instantiates a new servo (only a single one is supported!)

Returns (robot.Servo) instance of robot.Servo

create_sonar()
Instantiates a new sonar (only a single one is supported!)

Returns (robot.Sonar) instance of robot.Sonar

create_time_helper()
Instantiates a new time object.

Returns (time) instance of time

create_tracker(*tag_id*, *sd_x=0*, *sd_y=0*, *sd_theta=0*)

create_virtual_create(*hostname*)
Instantiates a new virtual create for visualization (only a single one is supported!)

Returns (visualization.VirtualCreate) instance of visualization.VirtualCreate

class pyCreate2.factory.**FactorySimulation**
Bases: object

Class to create objects which are simulated.

close()
Clean-up

create_create()
Instantiates a new create robot (only a single one is supported!)

Returns (simulation.Create2Vrep) instance of simulation.Create2Vrep

create_kuka_lbr4p()
Instantiates a new robotic arm (only a single one is supported!)

Returns (simulation.KukaLBR4PlusVrep) instance of simulation.KukaLBR4PlusVrep

create_pen_holder()
Instantiates a new pen holder (only a single one is supported!)

Returns (simulation.PenHolder) instance of simulation.PenHolder

create_servo()
Instantiates a new servo (only a single one is supported!)

Returns (simulation.Servo) instance of simulation.Servo

create_sonar()
Instantiates a new sonar (only a single one is supported!)

Returns (simulation.Sonar) instance of simulation.Sonar

create_time_helper()
Instantiates a new time object.

Returns (simulation.TimeHelper) instance of simulation.TimeHelper

create_tracker(*tag_id*, *sd_x=0.01*, *sd_y=0.01*, *sd_theta=0.01*)

create_virtual_create()
Instantiates a new virtual create for visualization (only a single one is supported!)

Returns (visualization.VirtualCreate) instance of visualization.VirtualCreate

Module contents

class pyCreate2.**FactoryCreate**
Bases: object

Class to create objects which are related to the physical iRobot Create2 robot.

close()
Clean-up

create_create()
Instantiates a new create robot (only a single one is supported!)

Returns (robot.Create2Driver) instance of robot.Create2Driver

create_pen_holder()
Instantiates a new pen holder (only a single one is supported!)

Returns (robot.PenHolder) instance of robot.PenHolder

create_servo()
Instantiates a new servo (only a single one is supported!)

Returns (robot.Servo) instance of robot.Servo

create_sonar()
Instantiates a new sonar (only a single one is supported!)

Returns (robot.Sonar) instance of robot.Sonar

create_time_helper()
Instantiates a new time object.

Returns (time) instance of time

create_tracker(*tag_id*, *sd_x=0*, *sd_y=0*, *sd_theta=0*)

create_virtual_create(*hostname*)
Instantiates a new virtual create for visualization (only a single one is supported!)

Returns (visualization.VirtualCreate) instance of visualization.VirtualCreate

class pyCreate2.**FactorySimulation**
Bases: object

Class to create objects which are simulated.

close()
Clean-up

create_create()
Instantiates a new create robot (only a single one is supported!)

Returns (simulation.Create2Vrep) instance of simulation.Create2Vrep

create_kuka_lbr4p()
Instantiates a new robotic arm (only a single one is supported!)

Returns (simulation.KukaLBR4PlusVrep) instance of simulation.KukaLBR4PlusVrep

create_pen_holder()
Instantiates a new pen holder (only a single one is supported!)

Returns (simulation.PenHolder) instance of simulation.PenHolder

create_servo()
Instantiates a new servo (only a single one is supported!)

Returns (simulation.Servo) instance of simulation.Servo

create_sonar()
Instantiates a new sonar (only a single one is supported!)

Returns (simulation.Sonar) instance of simulation.Sonar

create_time_helper()
Instantiates a new time object.

Returns (simulation.TimeHelper) instance of simulation.TimeHelper

create_tracker(*tag_id*, *sd_x=0.01*, *sd_y=0.01*, *sd_theta=0.01*)

create_virtual_create()
Instantiates a new virtual create for visualization (only a single one is supported!)

Returns (visualization.VirtualCreate) instance of visualization.VirtualCreate

example1 module

Example to move robot forward for 10 seconds Use "python3 run.py [–sim] example1" to execute

class example1.**Run**(*factory*)
Bases: object

run()

run module

Actual helper script to execute code. It takes care of proper error handling (e.g. if you press CTRL+C) and the difference between running code on the robot vs. in simulation.

Usage: python3 run.py –sim lab1 [for simulation] python3 run.py lab1 [to run on a robot]

Hardware

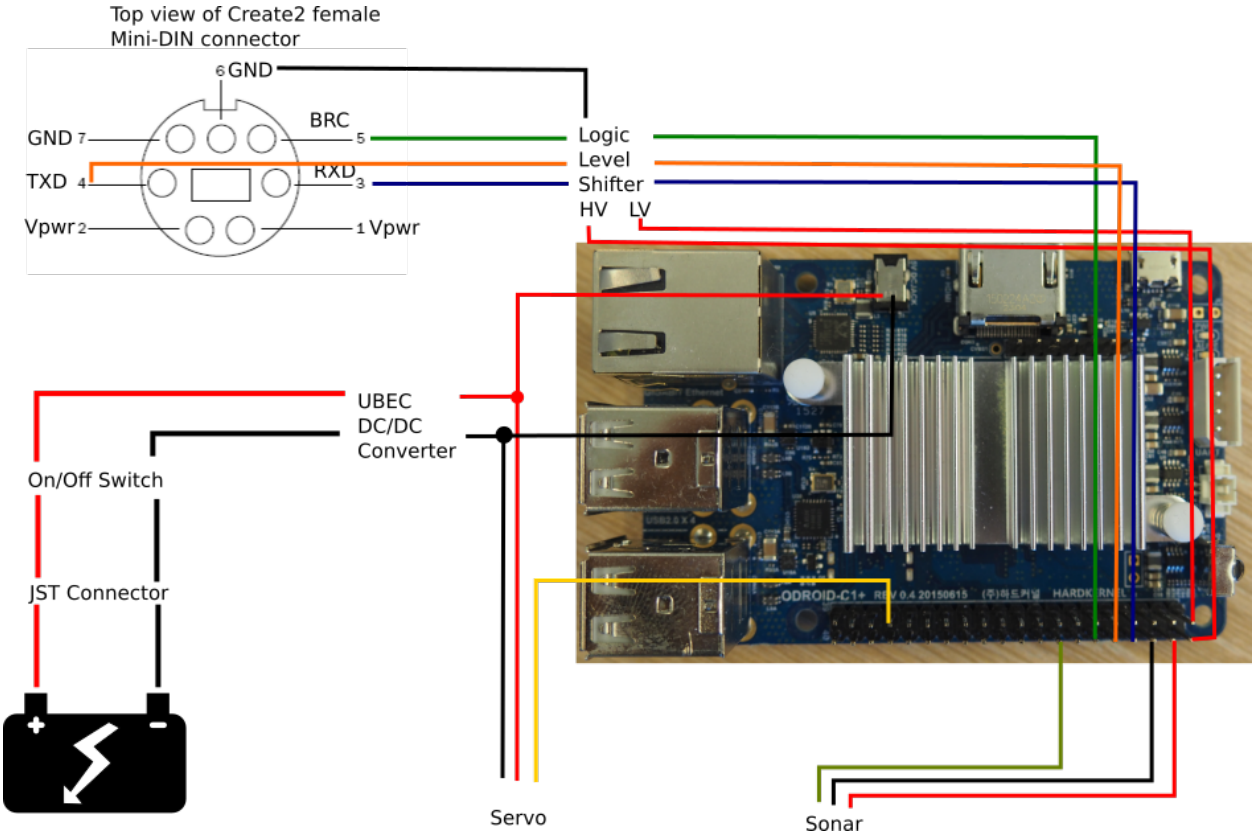
Components

(prices in USD and links as of spring 2016)

Table 1.1: Part List

Name	Link	Distributor	Price	Notes
iRobot Create2	https://www.adafruit.com/products/2388	Adafruit	199.99	
Mini-DIN Connector Cable for iRobot Create 2 - 7 Pins - 6 feet	https://www.adafruit.com/products/2438	Adafruit	6.95	
4-channel I2C-safe Bi-directional Logic Level Converter - BSS138	https://www.adafruit.com/products/757	Adafruit	3.95	
UBEC DC/DC Step-Down (Buck) Converter - 5V @ 3A output	https://www.adafruit.com/products/1385	Adafruit	9.95	
Rugged Metal On/Off Switch with Green LED Ring - 16mm Green On/Off	https://www.adafruit.com/products/482	Adafruit	4.95	
Silicone Cover Stranded-Core Wire - 25ft 26AWG - Red	https://www.adafruit.com/products/2513	Adafruit	4.95	For several Robots
Silicone Cover Stranded-Core Wire - 25ft 26AWG - Black	https://www.adafruit.com/products/2517	Adafruit	4.95	For several Robots
Premium Female/Female Jumper Wires - 40 x 6”	https://www.adafruit.com/products/266	Adafruit	3.95	For several Robots
Multi-Colored Heat Shrink Pack - 3/32” + 1/8” + 3/16” Diameters	https://www.adafruit.com/products/1649	Adafruit	4.95	For several Robots
Panel Mount USB Cable - A Male to A Female	https://www.adafruit.com/products/908	Adafruit	3.95	
Odroid C1+	http://ameridroid.com/products/odroid-c1	AmeriDroid	38.95	
WiFi Module 3	http://ameridroid.com/products/wifi-module-3	AmeriDroid	9.95	
DC Plug and Cable Assembly 2.5mm L-Type	http://ameridroid.com/products/dc-plug-and-cable-assembly-2-5mm-l-type	AmeriDroid	1.95	
Aluminum Standoff: 1/2” Length, 4-40 Thread, F-F (4-Pack)	https://www.pololu.com/product/2091	Pololu	1.39	For 2 robots each
Machine Screw: #4-40, 1/4” Length, Phillips (25-pack)	https://www.pololu.com/product/1960	Pololu	0.99	For up to 12 robots each
Machine Screw: #4-40, 5/16” Length, Phillips (25-pack)	https://www.pololu.com/product/1961	Pololu	0.99	For up to 12 robots each
JST RCY Connector Pack, Female	https://www.pololu.com/product/1934	Pololu	1.75	For up to 3 robots each
JST RCY Connector Pack, Male	https://www.pololu.com/product/1935	Pololu	1.75	For up to 3 robots each
Parallax Standard Servo	https://www.parallax.com/product/900-00005	Parallax	12.99	Optional
PING))) Ultrasonic Distance Sensor	https://www.parallax.com/product/28015	Parallax	29.99	Optional

Wiring



Software

Basic

- Download http://odroid.in/ubuntu_16.04lts/ubuntu-16.04-mate-odroid-c1-20160727.img.xz
- Extract:

```
unxz ubuntu-16.04-mate-odroid-c1-20160727.img.xz
```

- Verify MD5:

```
md5sum ubuntu-16.04-mate-odroid-c1-20160727.img
f5dfee4a8ea919dd8afc4384431574e5  ubuntu-16.04-mate-odroid-c1-20160727.img
```

- Copy to SD-Card:

```
sudo dd if=ubuntu-16.04-mate-odroid-c1-20160727.img of=</dev/path/of/card> bs=1M␣
↪conv=fsync
sync
```

Network

- Add */etc/wpa_supplicant/wpa_supplicant.conf* with following content:

```
network={
  ssid="<SSID>"
  psk="<password>"
  id_str="wifi"
}
```

- Update */etc/network/interfaces*:

```
# interfaces(5) file used by ifup(8) and ifdown(8)
# Include files from /etc/network/interfaces.d:
source-directory /etc/network/interfaces.d

auto lo
iface lo inet loopback

auto wlan0
# allow-hotplug wlan0
iface wlan0 inet manual
wpa-roam /etc/wpa_supplicant/wpa_supplicant.conf
iface wifi inet dhcp
iface default inet dhcp
```

- Disable persistent network (so that SD-card can be used with any WiFi dongle):

```
sudo ln -s /dev/null /etc/udev/rules.d/80-net-setup-link.rules
```

PWM

- Update */etc/modules*:

```
# /etc/modules: kernel modules to load at boot time.
#
# This file contains the names of kernel modules that should be loaded
# at boot time, one per line. Lines beginning with "#" are ignored.
# Parameters can be specified after the module name.

# ODROID HW PWM support (see http://odroid.com/dokuwiki/doku.php?id=en:c1_
↪hardware_pwm)
pwm-meson
pwm-ctrl
```

GPIO Support

- Add udev-rule: */etc/udev/rules.d/90-gpio.rules*:

```
SUBSYSTEM=="gpio", KERNEL=="gpiochip*", ACTION=="add", PROGRAM="/bin/sh -c 'chown␣
↪root:gpio /sys/class/gpio/export /sys/class/gpio/unexport ; chmod 222 /sys/
↪class/gpio/export /sys/class/gpio/unexport'"
SUBSYSTEM=="gpio", KERNEL=="gpio*", ACTION=="add", PROGRAM="/bin/sh -c 'chown␣
↪root:gpio /sys%p/active_low /sys%p/direction /sys%p/edge /sys%p/value ; chmod␣
↪660 /sys%p/active_low /sys%p/direction /sys%p/edge /sys%p/value'"
```

- Create GPIO group:

```
sudo groupadd gpio
```

- Add user to group:

```
sudo adduser odroid gpio
```

- Reboot

Additional Software

- Update the system:

```
sudo apt update
sudo apt upgrade
```

- Install additional packages:

```
sudo apt install python3 python3-serial python3-scipy python3-numpy python3-
↪matplotlib
```

Add User

- Add user and assign groups:

```
sudo adduser csci445
sudo adduser csci445 gpio
sudo adduser csci445 dialout
```

Debugging

You can use the USB UART Kit for debugging, see http://odroid.com/dokuwiki/doku.php?id=en:usb_uart_kit for more details. This will allow you to gain access to a shell using UART.

- On you host PC, add */etc/udev/rules.d/99-odroiduart.rules* with the following content:

```
SUBSYSTEM=="usb", ATTRS{idVendor}=="10c4", ATTRS{idProduct}=="ea60", MODE="0664",␣
↪GROUP="plugdev"
```

  Make sure that your user is member of the *plugdev* group.

- To connect, use:

```
picocom --baud 115200 /dev/ttyUSB0
```

  You can end the session by pressing Ctrl+A followed by Ctrl+X.

CHAPTER 2

Indices and tables

- genindex
- modindex
- search

Python Module Index

e

p

r

Index

D

E

F

G

I

K

L

M

N

O

P

Q

R

S

T

V

W