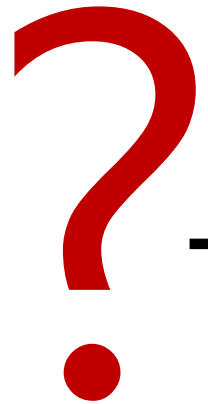
TIZEN™

Tizen.IoTivity.Sensoring

Junkyu Han

Tizen Talks

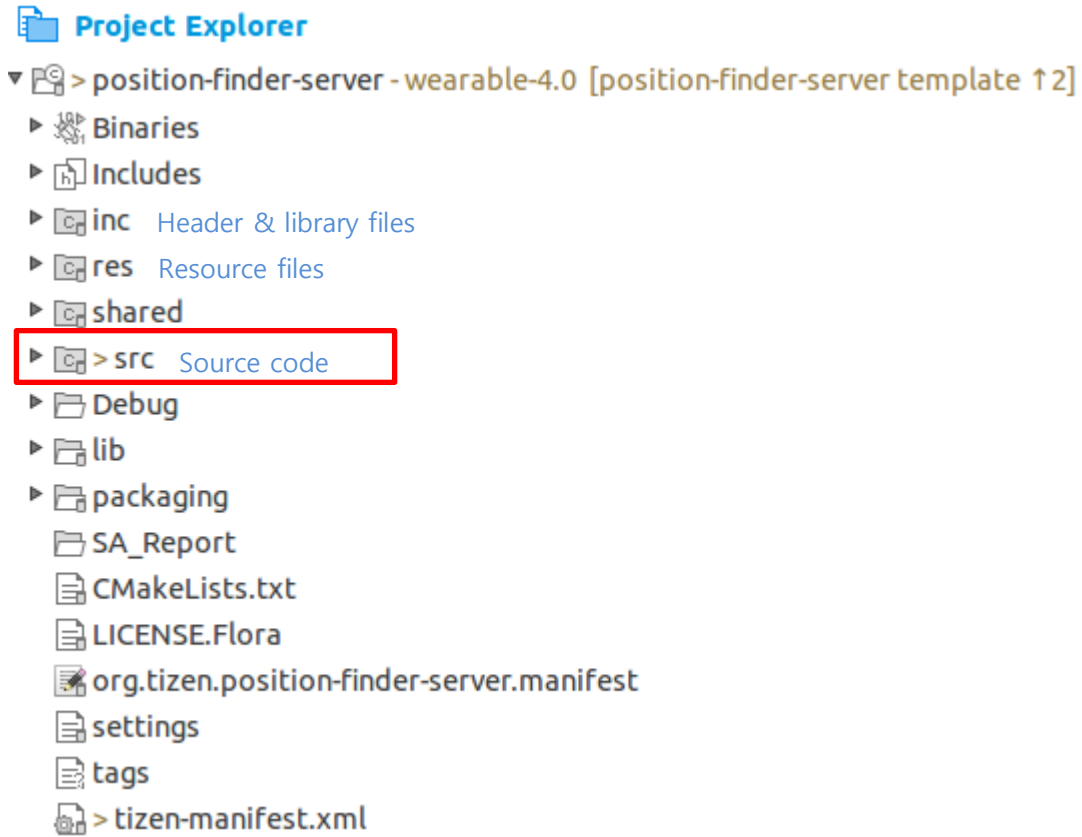
TIZEN



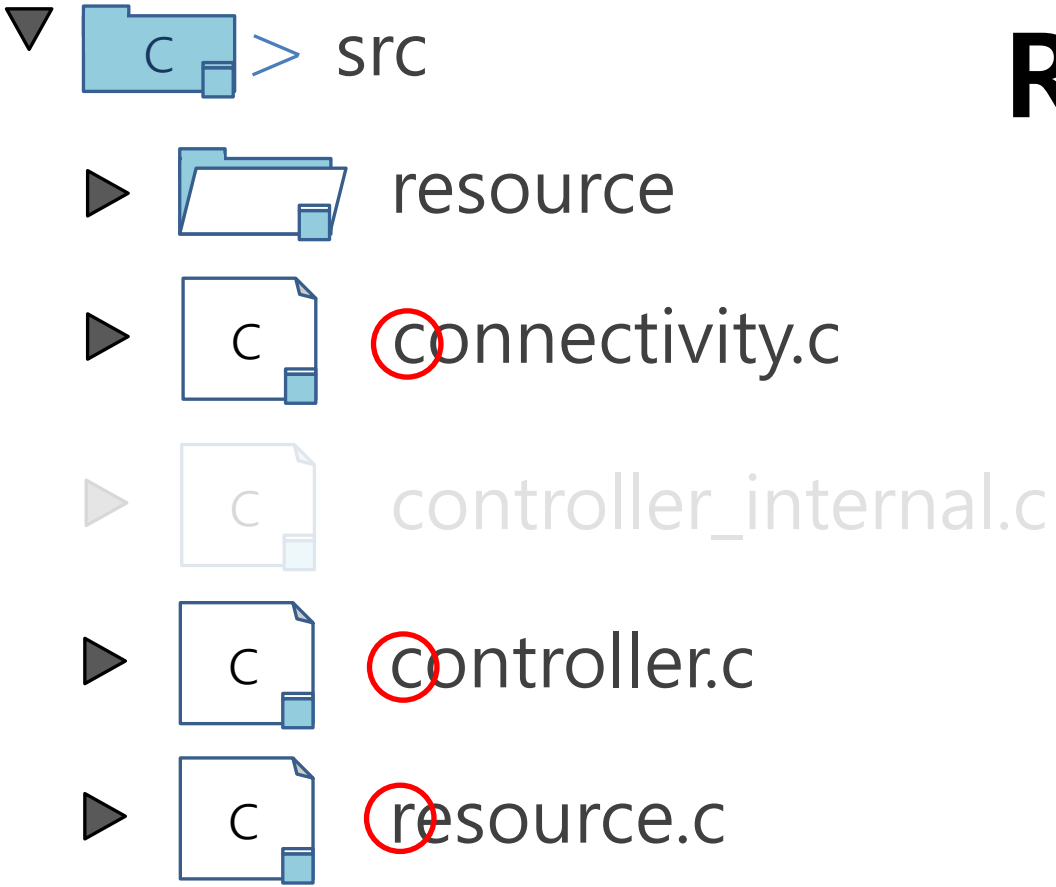
Template Structure

position-finder-server

position-finder-server



position-finder-server



R C C

Pattern

position-finder-server

Sensors, LED etc …
Collects and exchanges data
Represents real state content

Resource

Connectivity of devices
Local network or cloud network
Integrated into communication networks

Connectivity

Controller

Tizen IOTivity Application Start from Here !

Controls resources and connectivity

position-finder-server/src/controller.c

```
/* Register LifeCycle Callback Function called on each lifecycle step*/

int main(int argc, char* argv[ ])
{
    ...
    service_app_lifecycle_callback_s event_callback;
    ...

    event_callback.create = service_app_create;
    event_callback.terminate = service_app_terminate;
    event_callback.app_control = service_app_control;
    ...

    ret = service_app_main(argc, argv, &event_callback, ad);

    return ret;
}
```

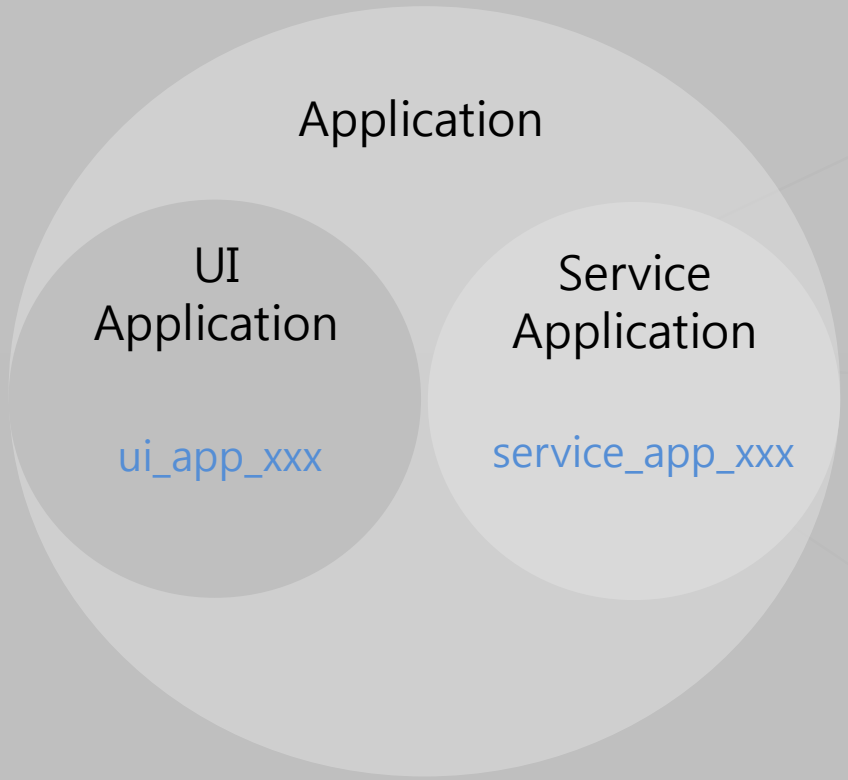
```
static bool service_app_create(void *data)
{
    ...

    return true;
}
```

```
static void service_app_terminate(void *data)
{
    ...
}
```

```
static void service_app_control(void *data)
{
    ...
}
```

position-finder-server/src/controller.c

Tizen

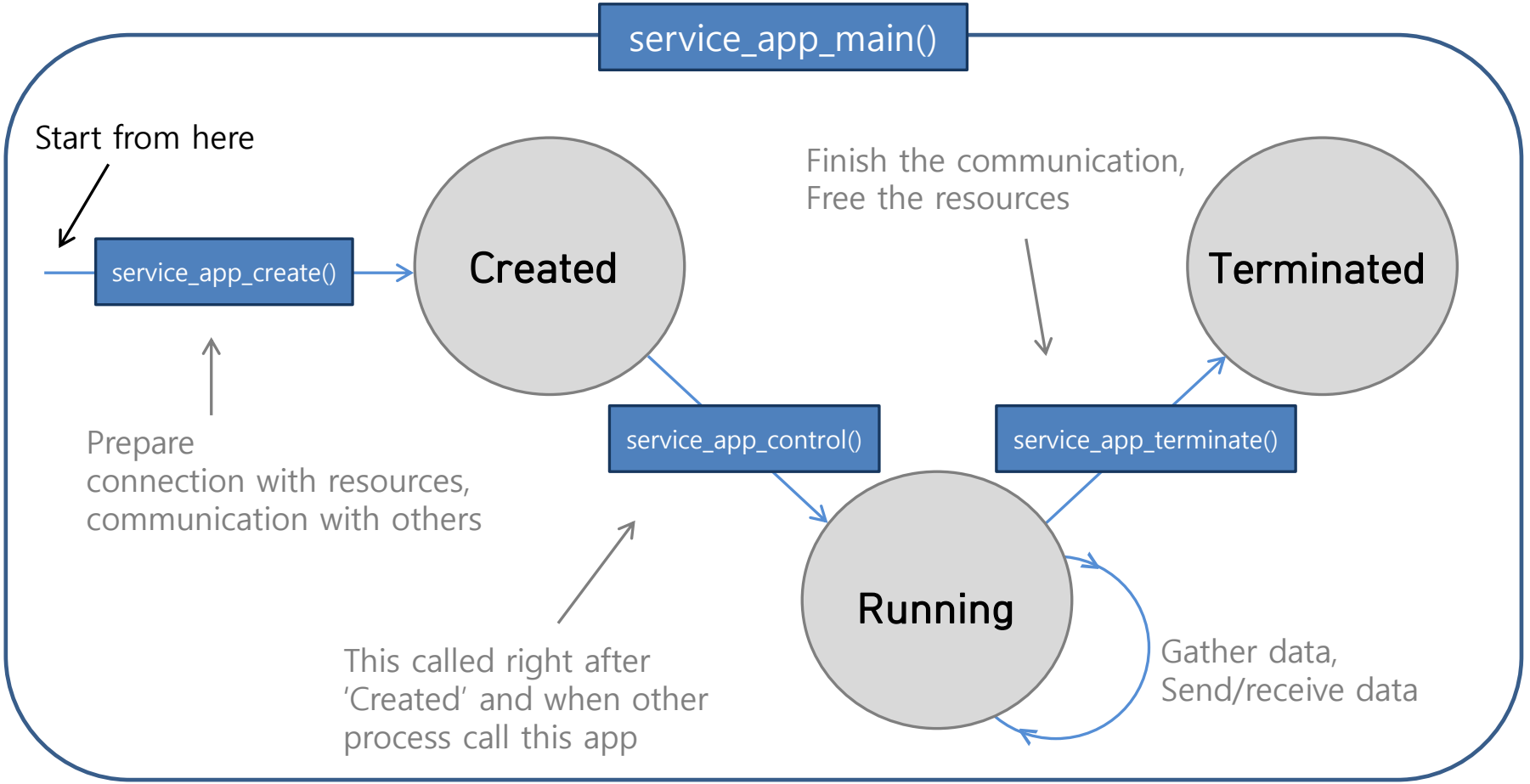



```
static bool service_app_create(void *data)
{
    ...

    return true;
}
```

```
static void service_app_terminate(void *data)
{
    ...
}
```

```
static void service_app_control(void *data)
{
    ...
}
```

Tizen Service Application LifeCycle



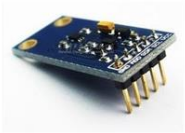
service_app_create()

```
static  bool service_app_create(void *data)
{
    …

    /* We should prepare for the connection with resources like sensors, led .. etc.. */

    …

    return true;
}
```

How to connect with resources ?

Is it easy ?

position-finder-server

- resource
  - C resource_illuminance_sensor.c 
  - C resource_infrared_motion_sensor.c 
  - C resource_infrared_obstacle_avoidance_sensor.c 
  - C resource_led.c 
  - C resource_touch_sensor.c 
  - C resource_ultrasonic_sensor.c 

Prepare for the communication with sensor

1. Check the function name with sensor name you want to use in **inc/resource** directory

2. Decide how often you want to check the sensor

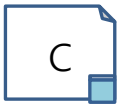
3. Read or write data from/to sensor and do useful with data

position-finder-server

1. Check the function name with sensor name you want to use in **inc/resource** directory

resource

resource_illuminance_sensor.c



resource_infrared_motion_sensor.c

resource_infrared_obstacle_avoidance_sensor.c

resource_led.c

resource_touch_sensor.c

resource_ultrasonic_sensor.c

service_app_create()

2. Decide how often you want to check the sensor

```
static  bool service_app_create(void *data)
{
   …

   /* We should prepare for the connection with resources like sensors, led .. etc.. */

   ad->getter_timer = ecore_timer_add(/* Duration */, control_sensors_cb, ad);
   …
   …

   return true;
}
```

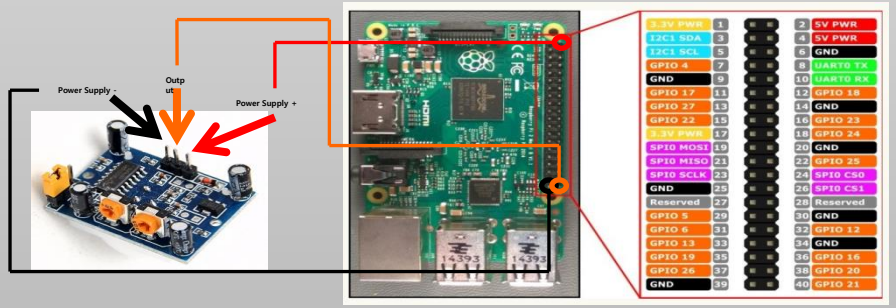
Type: float ex) 0.5f

This parameter decide how often callback function will be called.

control_sensors_cb()

3. Read or write data from/to sensor

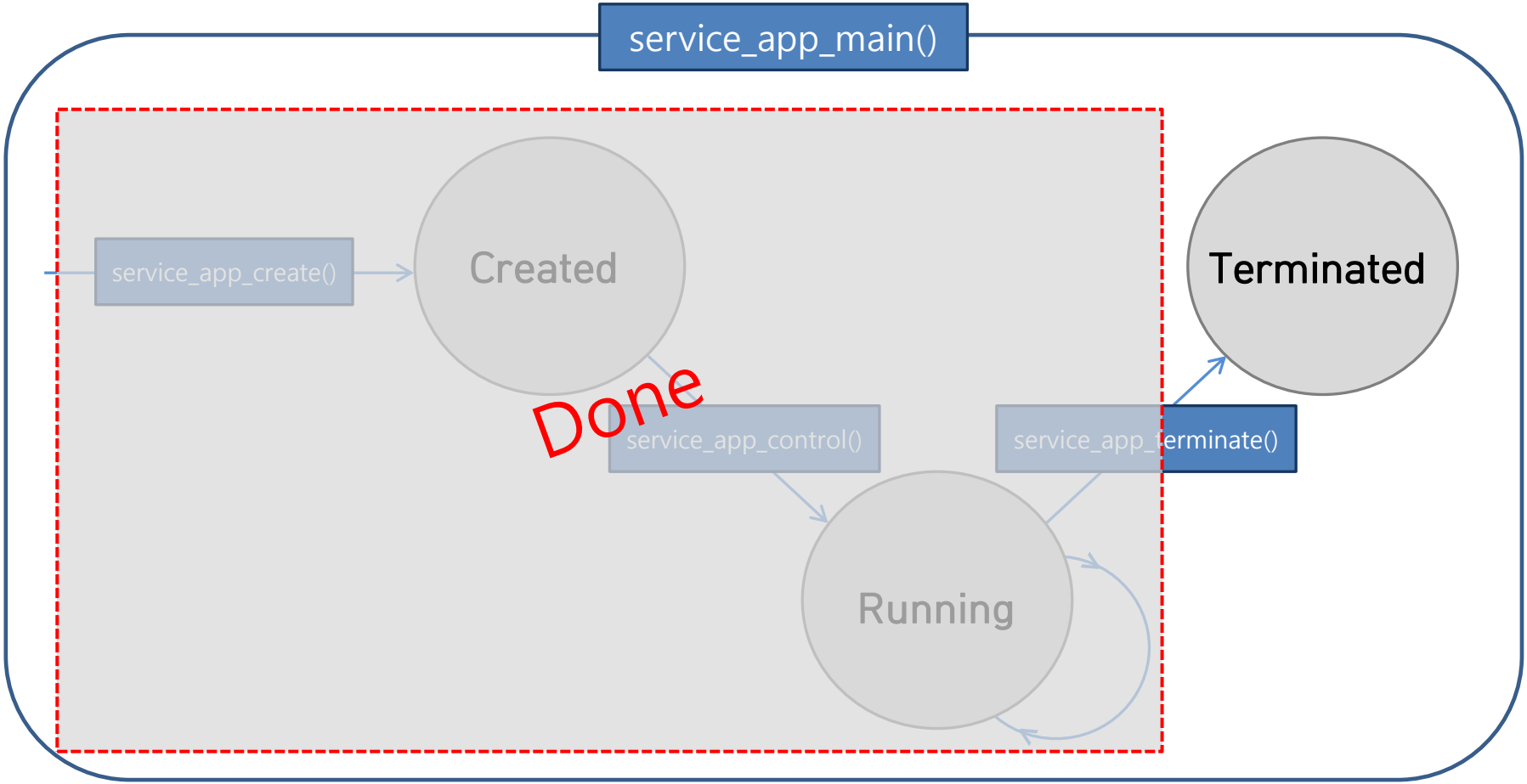
```
static  Eina_Bool control_sensors_cb(void *data)
{
   ...


   if (resource_read_infrared_motion_sensor( 21  &value) == -1)
      _E("Filed to get Infrared Motion value");

   ...
   ...

   return true;
}
```



resource/resource_infrared_motion_sensor.c

```
void resource_close_infrared_motion_sensor(int pin_num)
{
   ...

   ...
   ...
}

int resource_read_infrared_motion_sensor(int pin_num, int *out_value)
{
   int ret = PERIPHERAL_ERROR_NONE;

   if (!resource_get_info(pin_num)->opened) {
      ret = peripheral_gpio_open(pin_num,
                                 &resource_get_info(pin_num)->sensor_h);
      retv_if(!resource_get_info(pin_num)->sensor_h, -1);

      ret = peripheral_gpio_set_direction(resource_get_info
                  (pin_num)->sensor_h, PERIPHERAL_GPIO_DIRECTION_IN);
      retv_if(ret != PERIPHERAL_ERROR_NONE, -1);

      resource_get_info(pin_num)->opened = 1;
   }

   ret = peripheral_gpio_read(resource_get_info(pin_num)->sensor_h, out_value);
   retv_if(ret != PERIPHERAL_ERROR_NONE, -1);

   return 0;
}
```

Tizen Service Application LifeCycle

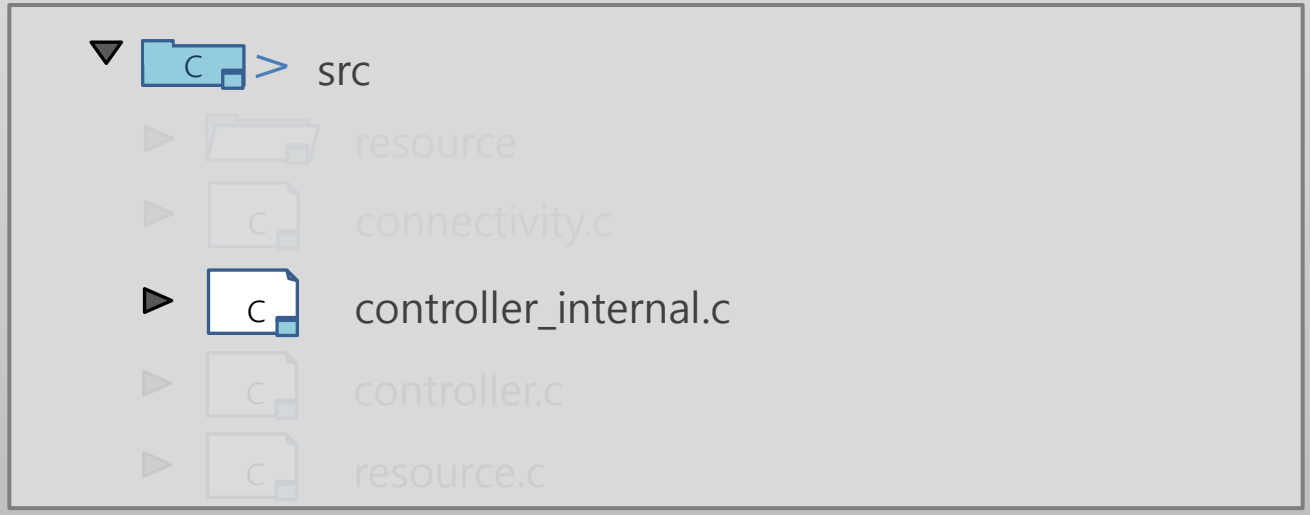


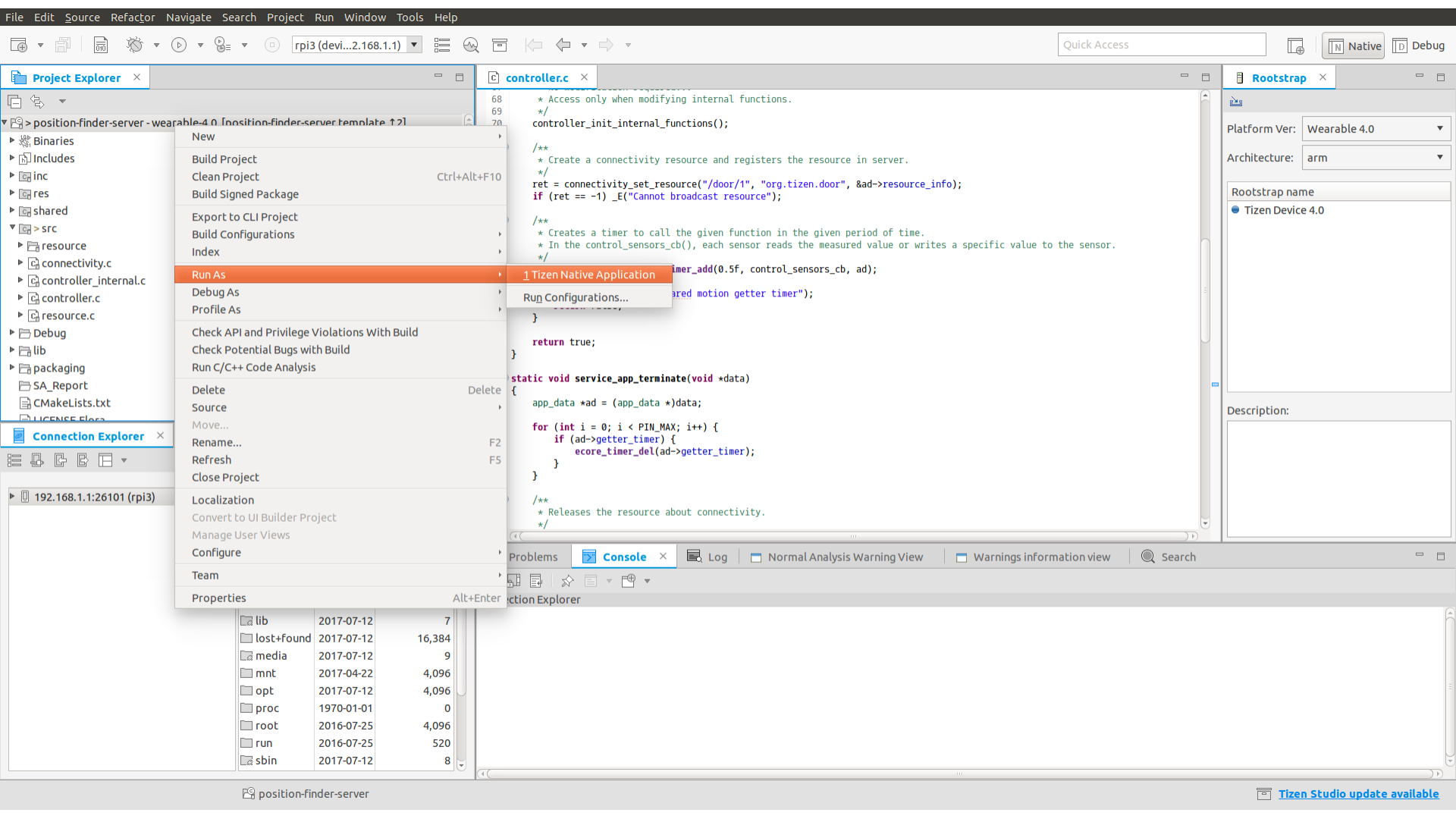
service_app_terminate()

```
static  void service_app_terminate(void *data)
{
    ...
    controller_fini_internal_function();
    ...
    ...
    return true;
}
```

NOTHING TO DO

All connections with resources are closed in this function




File Edit Source Refactor Navigate Search Project Run Window Tools Help
rpi3 (devi...2.168.1.1)
Quick Access
Native
Debug
Project Explorer
> position-finder-server - wearable-4.0 [position-finder-server template 1.2]
Binaries
Includes
inc
res
shared
> src
resource
connectivity.c
controller_internal.c
controller.c
resource.c
Debug
lib
packaging
SA_Report
CMakeLists.txt
New
Build Project
Clean Project Ctrl+Alt+F10
Build Signed Package
Export to CLI Project
Build Configurations
Index
Run As
Debug As
Profile As
Check API and Privilege Violations With Build
Check Potential Bugs with Build
Run C/C++ Code Analysis
Delete Delete
Source
Move...
Rename... F2
Refresh F5
Close Project
Localization
Convert to UI Builder Project
Manage User Views
Configure
Team
Properties Alt+Enter
1 Tizen Native Application
Run Configurations...
Connection Explorer
192.168.1.1:26101 (rpi3)
lib 2017-07-12 7
lost+found 2017-07-12 16,384
media 2017-07-12 9
mnt 2017-04-22 4,096
opt 2017-07-12 4,096
proc 1970-01-01 0
root 2016-07-25 4,096
run 2016-07-25 520
sbin 2017-07-12 8
controller.c
* Access only when modifying internal functions.
*/
controller_init_internal_functions();
/**
* Create a connectivity resource and registers the resource in server.
*/
ret = connectivity_set_resource("/door/1", "org.tizen.door", &ad->resource_info);
if (ret == -1) _E("Cannot broadcast resource");
/**
* Creates a timer to call the given function in the given period of time.
* In the control_sensors_cb(), each sensor reads the measured value or writes a specific value to the sensor.
*/
imer_add(0.5f, control_sensors_cb, ad);
ared motion getter timer");
}
return true;
}
static void service_app_terminate(void *data)
{
app_data *ad = (app_data *)data;
for (int i = 0; i < PIN_MAX; i++) {
if (ad->getter_timer) {
ecore_timer_del(ad->getter_timer);
}
}
/**
* Releases the resource about connectivity.
*/
Rootstrap
Platform Ver: Wearable 4.0
Architecture: arm
Rootstrap name
Tizen Device 4.0
Description:
Problems
Console
Log
Normal Analysis Warning View
Warnings information view
Search
ction Explorer
position-finder-server
Tizen Studio update available

Just do it! Tizen is behind you!

Whatever you want

It'll be here

- resource
  - resource_illuminance_sensor.c
  - resource_infrared_motion_sensor.c
  - resource_infrared_obstacle_avoidance_sensor.c
  - resource_led.c
  - resource_touch_sensor.c
  - resource_ultrasonic_sensor.c
  - resource_xxx_sensor.c
  - resource_xxx_sensor.c
  - resource_xxx_sensor.c

TIZEN™

Thank you.

Tizen Talks

TIZEN