In [1]:
```
print("Hello World")
```

```
Hello World
```

In [4]:
```
from pynq.overlays.base import BaseOverlay
import time
base = BaseOverlay("base.bit")
```

In [ ]:
```
help(base)
```

In [5]:
```
led0 = base.leds[0]
led0.on()
time.sleep(2)
led0.off()
```

In [9]:
```
from pynq.overlays.base import BaseOverlay
import pynq.lib.rgbled as rgbled
import time
base = BaseOverlay("base.bit")
```

In [10]:
```
help(rgbled)
```

```
Help on module pynq.lib.rgbled in pynq.lib:

NAME
    pynq.lib.rgbled

DESCRIPTION
    #   Copyright (c) 2016, Xilinx, Inc.
    #   SPDX-License-Identifier: BSD-3-Clause

CLASSES
    builtins.object
        RGBLED

    class RGBLED(builtins.object)
     |  RGBLED(index, ip_name='rgbleds_gpio', start_index=inf, device=None)
     |
     |  This class controls the onboard RGB LEDs.
     |
     |  Attributes
     |  ----------
     |  index : int
     |      The index of the RGB LED. Can be an arbitrary value.
     |  _mmio : MMIO
     |      Shared memory map for the RGBLED GPIO controller.
     |  _rgbleds_val : int
     |      Global value of the RGBLED GPIO pins.
     |  _rgbleds_start_index : int
     |      Global value representing the lowest index for RGB LEDs
     |
     |  Methods defined here:
     |
     |  __init__(self, index, ip_name='rgbleds_gpio', start_index=inf, device=
None)
     |      Create a new RGB LED object.
     |
     |      Parameters
     |      ----------
     |      index : int
     |          Index of the RGBLED, Can be an arbitrary value.
     |          The smallest index given will set the global value
     |          `_rgbleds_start_index`. This behavior can be overridden by def
ining
     |          `start_index`.
     |      ip_name : str
     |          Name of the IP in  the `ip_dict`. Defaults to "rgbleds_gpio".
     |      start_index : int
     |          If defined, will be used to update the global value
     |          `_rgbleds_start_index`.
     |
     |  off(self)
     |      Turn off a single RGBLED.
     |
     |      Returns
     |      -------
     |      None
     |
     |  on(self, color)
     |      Turn on a single RGB LED with a color value (see color constants).
     |
     |      Parameters
```

```
|      ----------
|      color : int
|         Color of RGB specified by a 3-bit RGB integer value.
|
|      Returns
|      -------
|      None
|
|  read(self)
|      Retrieve the RGBLED state.
|
|      Returns
|      -------
|      int
|          The color value stored in the RGBLED.
|
|  write(self, color)
|      Set the RGBLED state according to the input value.
|
|      Parameters
|      ----------
|      color : int
|          Color of RGB specified by a 3-bit RGB integer value.
|
|      Returns
|      -------
|      None
|
|  ----------------------------------------------------------------------
|  Data descriptors defined here:
|
|  __dict__
|      dictionary for instance variables (if defined)
|
|  __weakref__
|      list of weak references to the object (if defined)

DATA
    RGBLEDS_XGPIO_OFFSET = 0
    RGB_BLUE = 1
    RGB_CLEAR = 0
    RGB_CYAN = 3
    RGB_GREEN = 2
    RGB_MAGENTA = 5
    RGB_RED = 4
    RGB_WHITE = 7
    RGB_YELLOW = 6

FILE
    /usr/local/share/pynq-venv/lib/python3.10/site-packages/pynq/lib/rgbled.py
```

In [12]:
```
led4 = rgbled.RGBLED(4)
led5 = rgbled.RGBLED(5)
```

In [13]:
```
led4.write(0x7)
led5.write(0x4)
```

In [14]:
```
led4.write(0x0)
led5.write(0x0)
```

In [ ]:
```

```