

Project Objectives

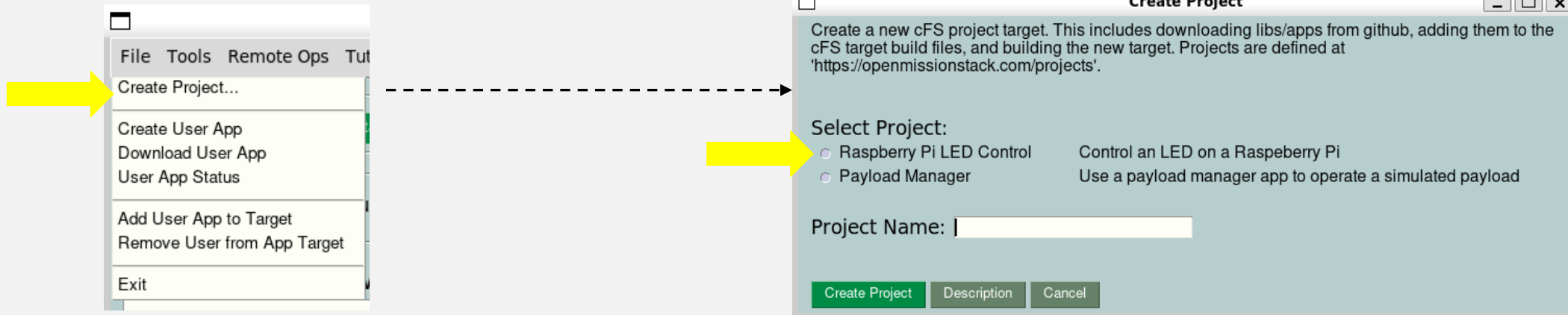
In this project you will connect an LED to a Raspberry Pi's General-Purpose Input/Output (GPIO) header and control the LED using a cFS app. It allows cFS developers to gain experience with using a library and app to interface and control an external hardware component using low-cost materials.

Detailed project instructions with videos can be found at

<https://spacesteps.com/2024/10/12/cfs-raspberry-pi-led-control/>

Software Installation

1. Create RPI LED Control project using the Create Project tool



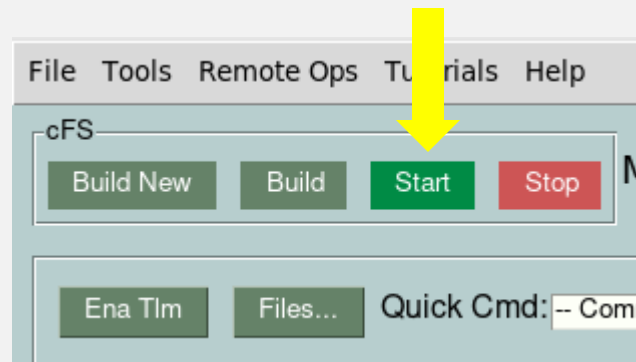
2. Before restarting Basecamp GUI, set basecamp.ini configurations to start the cFS with elevated permissions. The password needs to match your Raspberry Pi password

```
[CFS_TARGET]
# Define the default c
# in cfs-basecamp/cfe-
MISSION_EDS_NAME = bas
CPU_EDS_NAME = cpu1
BASE_PATH = ../../cfe-
# SUDO_START_CFS - Tru
SUDO_START_CFS = True

[APP]
DEBUG = False
VERSION = 2.0
EVENT_LOGS = logs/ever
DEFAULT_DOC = basecamp
# User password of acc
PASSWORD = cfs
```

Yellow arrows point to the `SUDO_START_CFS = True` line and the `PASSWORD = cfs` line in the configuration file.

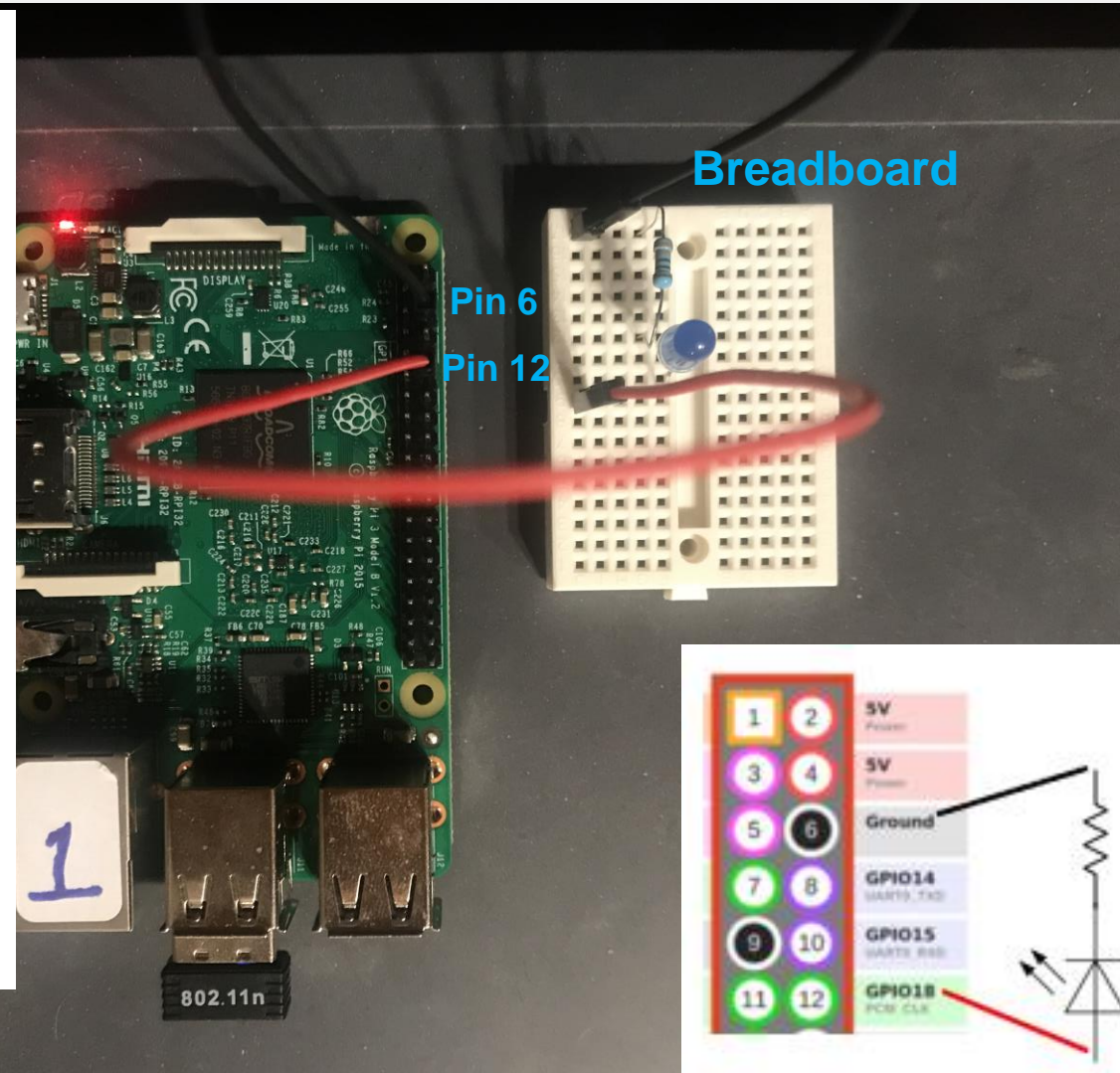
3. Start the cFS



Hardware Assembly

1. Connect the resistor and LED on a breadboard
2. Connect the breadboard to the GPIO pins as shown
 - Note physical pin 12 is logical GPIO pin 18 which is the identifier used by the FSW

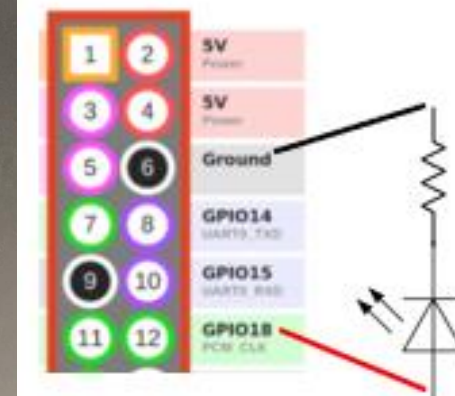
Pin#	NAME	NAME	Pin#
01	3.3v DC Power	DC Power 5v	02
03	GPIO02 (SDA1 , I2C)	DC Power 5v	04
05	GPIO03 (SCL1 , I2C)	Ground	06
07	GPIO04 (GPIO_GCLK)	(TXD0) GPIO14	08
09	Ground	(RXD0) GPIO15	10
11	GPIO17 (GPIO_GEN0)	(GPIO_GEN1) GPIO18	12
13	GPIO27 (GPIO_GEN2)	Ground	14
15	GPIO22 (GPIO_GEN3)	(GPIO_GEN4) GPIO23	16
17	3.3v DC Power	(GPIO_GEN5) GPIO24	18
19	GPIO10 (SPI_MOSI)	Ground	20
21	GPIO09 (SPI_MISO)	(GPIO_GEN6) GPIO25	22
23	GPIO11 (SPI_CLK)	(SPI_CE0_N) GPIO08	24
25	Ground	(SPI_CE1_N) GPIO07	26
27	ID_SD (I2C ID EEPROM)	(I2C ID EEPROM) ID_SC	28
29	GPIO05	Ground	30
31	GPIO06	GPIO12	32
33	GPIO13	Ground	34
35	GPIO19	GPIO16	36
37	GPIO26	GPIO20	38
39	Ground	GPIO21	40



220 Ohm Resistor
(red, red, brown)

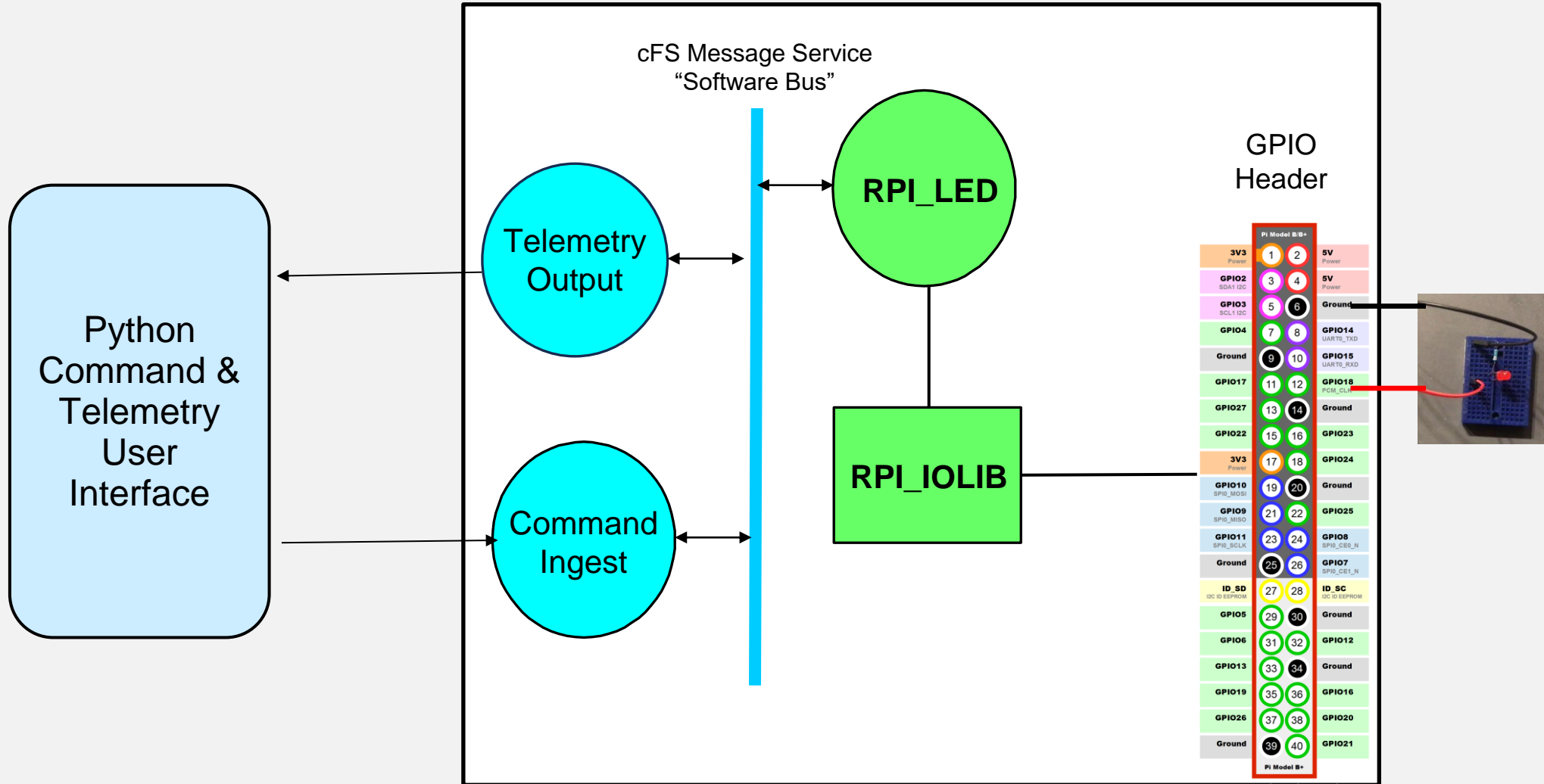
LED

- Connect shorter leg Cathode (-) to resistor
- Connect longer leg Anode (+) to GPIO pin 12



Project Architecture

cFS Target



 = Project Apps and Library

Project Library and App Summary

RPI_IOLIB

- This library adapts the “minimalistic peripheral access” (MIPEA) library for the Raspberry Pi, <https://github.com/jasLogic/mipea> so it can serve as a cFS library.
- It provides an interface to the Raspberry Pi’s memory mapped General Purpose I/O registers
- The config.h configuration file allows you to select your Broadcom processor chip

RPI_LED

- The main app manages the ground command and telemetry interface
- A child task controls turning on and off the LED
- Ground commands can be sent to set the LED on and off time durations
- JSON initialization table defines the GPIO LED control pin and the default LED on/off time durations

RPI_LED App Operations

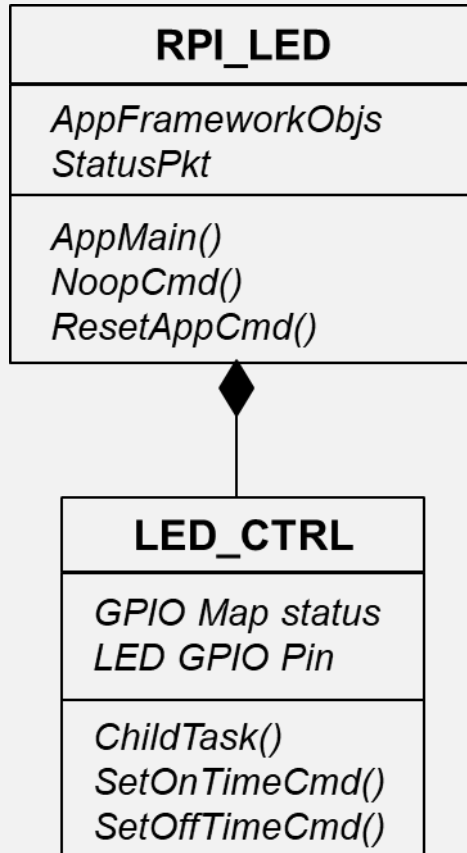
Commands

- Set LED On Time
- Set LED Off Time

Telemetry

```
HkTlm.Payload.ValidCmdCnt : 0  
HkTlm.Payload.InvalidCmdCnt: 0  
HkTlm.Payload.CtrlIsMapped : FALSE  
HkTlm.Payload.CtrlOutPin : 18  
HkTlm.Payload.CtrlLedOn : FALSE  
HkTlm.Payload.CtrlSpare : 0  
HkTlm.Payload.CtrlOnTime : 3000  
HkTlm.Payload.CtrlOffTime : 6000
```

RPI_LED App Object Design



RPI_LED

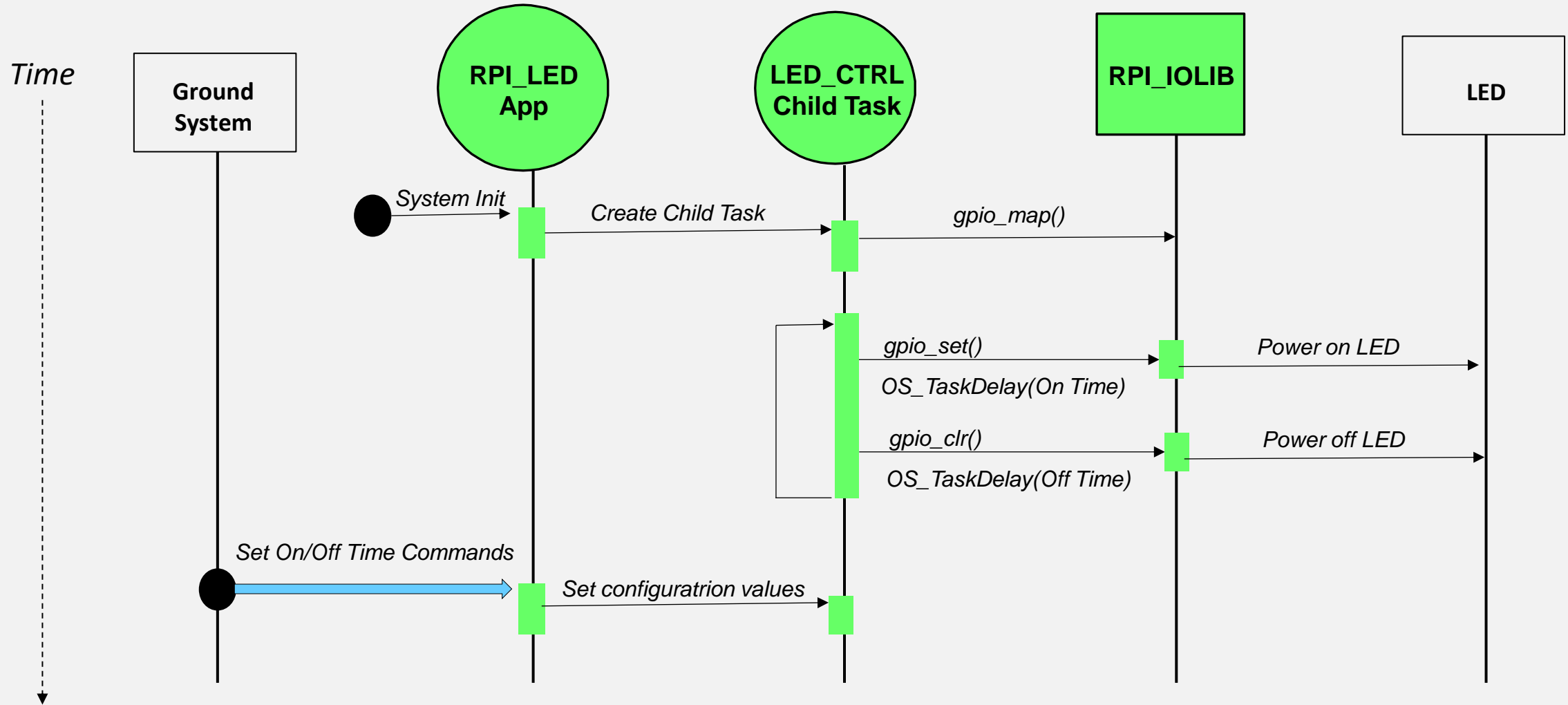
- Manages app initialization, main runtime loop, and status telemetry
- Dispatches commands to objects

cFS

LED_CTRL

- Manages GPIO and LED interface
- Contains child task that turns the LED on and off
- Calls to RPI_IOLIB functions should be limited to this object to localize coupling

Control Flows



● Start Event

➡ Software Bus Message

➡ Library Call

RPI_LED Control Flow

App

Initialize App

Subscribe to Command and
'Send Status' messages
Create child task

Pend on SB message pipe

**Send
Status?**

Yes

Send status telemetry

No

Cmd?

Yes

Process Command

No

Send error event

Child Task

Initialize Child Task

Read ini table LED pin and
on/off times
Map GPIO

**GPIO
Mapped?**

No

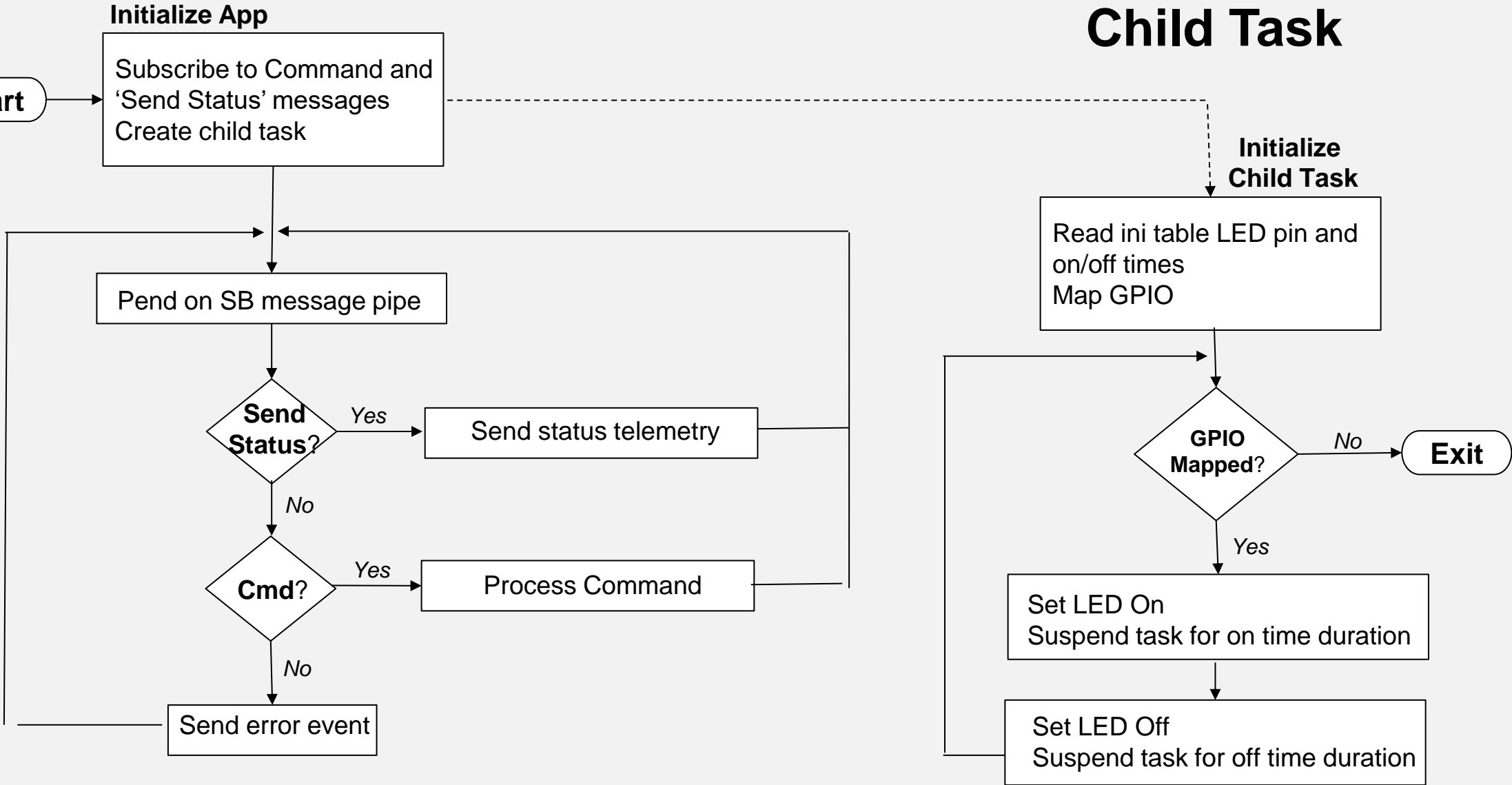
Exit

Yes

Set LED On
Suspend task for on time duration

Set LED Off
Suspend task for off time duration

Start



Optional Manual Software Installation

