

Design of Foscam Webcam Firmware

UML for Embedded Systems

Simone ROSSI

November 18, 2016

Date Performed: November 18, 2016

Partners: Simone Rossi

1 Assumptions

- **LightSensor_value_rate:** the light sensor provides a valid data once per second
- **Camera_frame_rate:** the camera provides a valid frame every 1/25 sec (40 ms)
- **Request_notification:** each incoming Ethernet request is preceeded by a request notification
- **QR_configuration:** the QR code is the serial number of the camera. During configuration, the serial read by the smartphone is compared with the internal serial number. If equal, then the communication is configured
- **Recording_video:** If the camera has been configured, it always records video and checks for motion objects.

2 Block Diagram

In my design of the Foscam Firmware, the main controller is divided into five different submodules, each of them specialized for one particular task.

To be more realistic, the design includes also the definition of three custom types: Email, Image and Video.

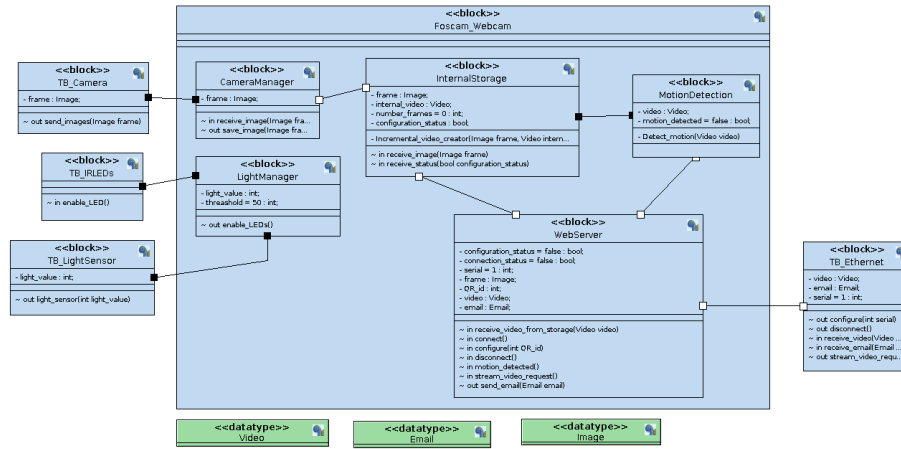
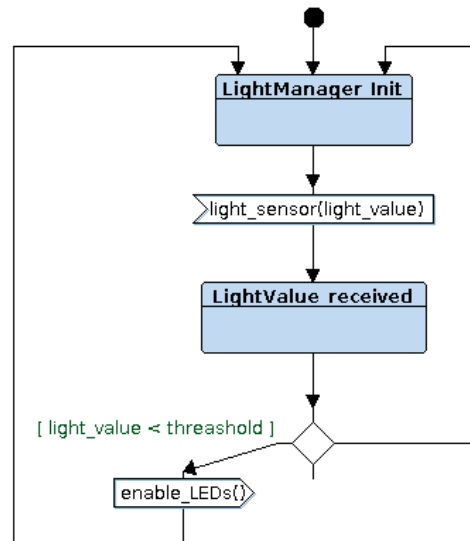


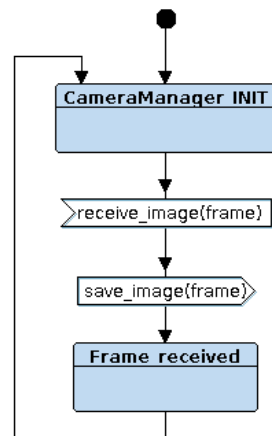
Figure 1: Main blocks with their testbench modules

2.1 Light Manager



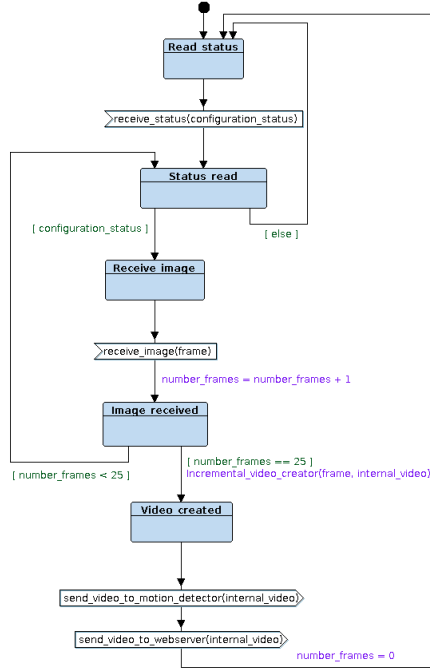
The **Light Manager** simply waits for a new value from the light sensor and if it is [else] higher than a given threshold, it drives the signal **enable_LED**, which will be eventually received by the IR LEDs module.

2.2 Camera Manager



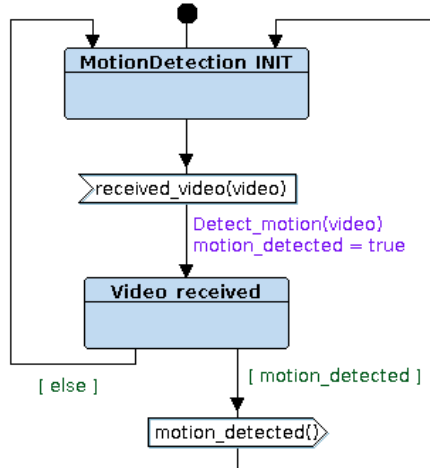
The **Camera Manager** is in charge of receiving the video frame by frame and create a stream for the **Internal Storage**.

2.3 Internal Storage



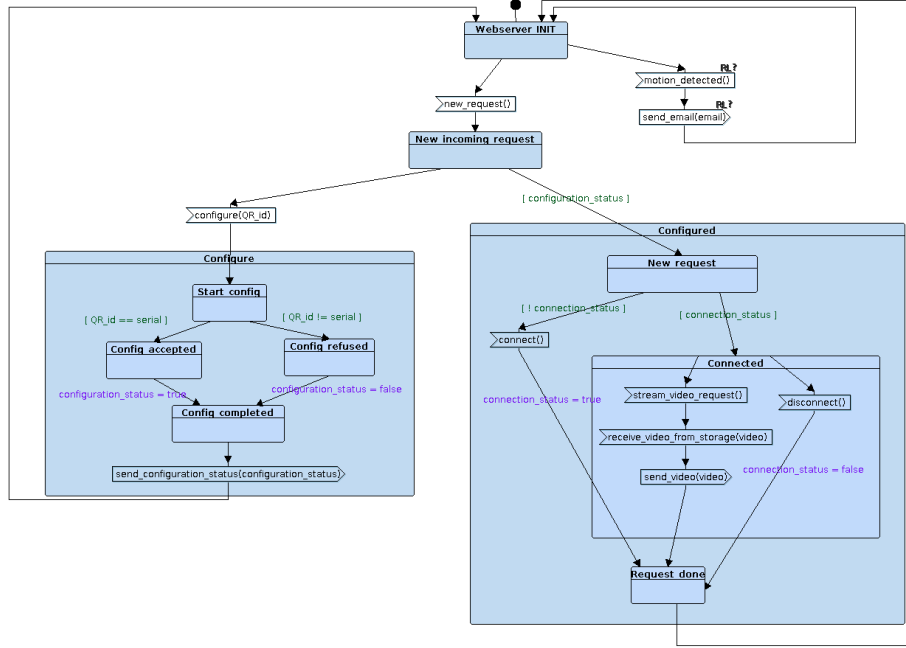
If the camera has been configured, the **Internal Storage** will start receiving images and after 25 frames (e.g. 1 second) the video is created and stored. As soon as a video is created, it will be send both to the **Ethernet Interface** and to the embedded **Motion Detection** engine.

2.4 Motion Detector



The motion detection engine will analyse a one second video to eventually identify some moving objects. If so, the Web-Server will be notified and an alarm will be triggered.

2.5 WebServer



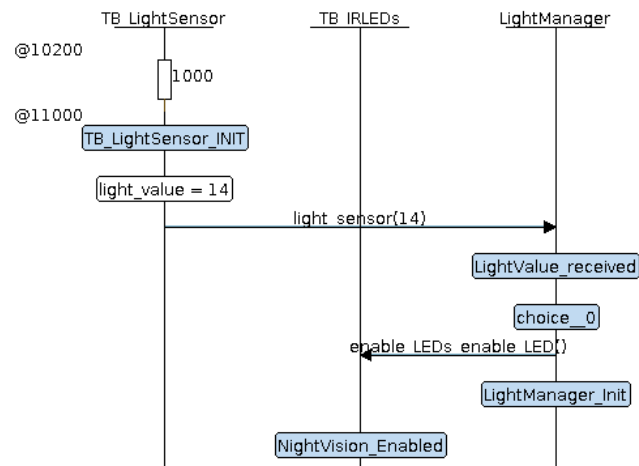
This is the most complex submodule of the firmware. This module is used for:

- **Configuration** and pairing to a new smartphone/computer using a QR code;
- **Connection** and **disconnection** of a configured smartphone/computer;
- **Stream video** if required by the user;
- **Alarm Management** if the detection engine has triggered an object movement.

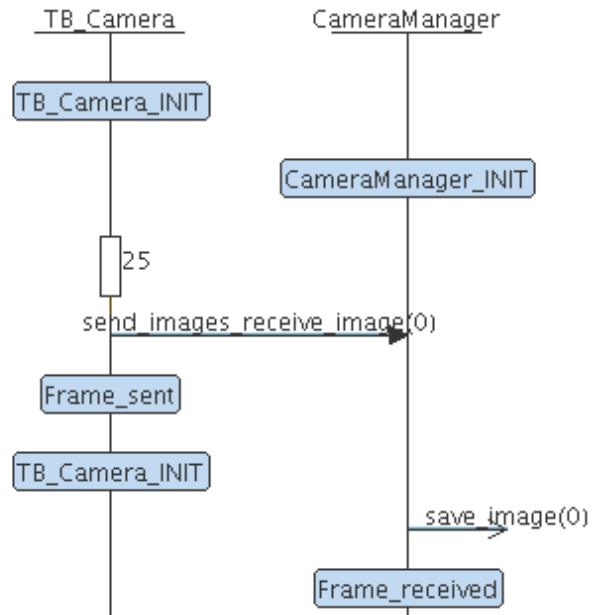
3 Nominal Case Diagrams

These are some nominal case diagrams used to understand whether or not all the subsystems work as stated in the requirements.

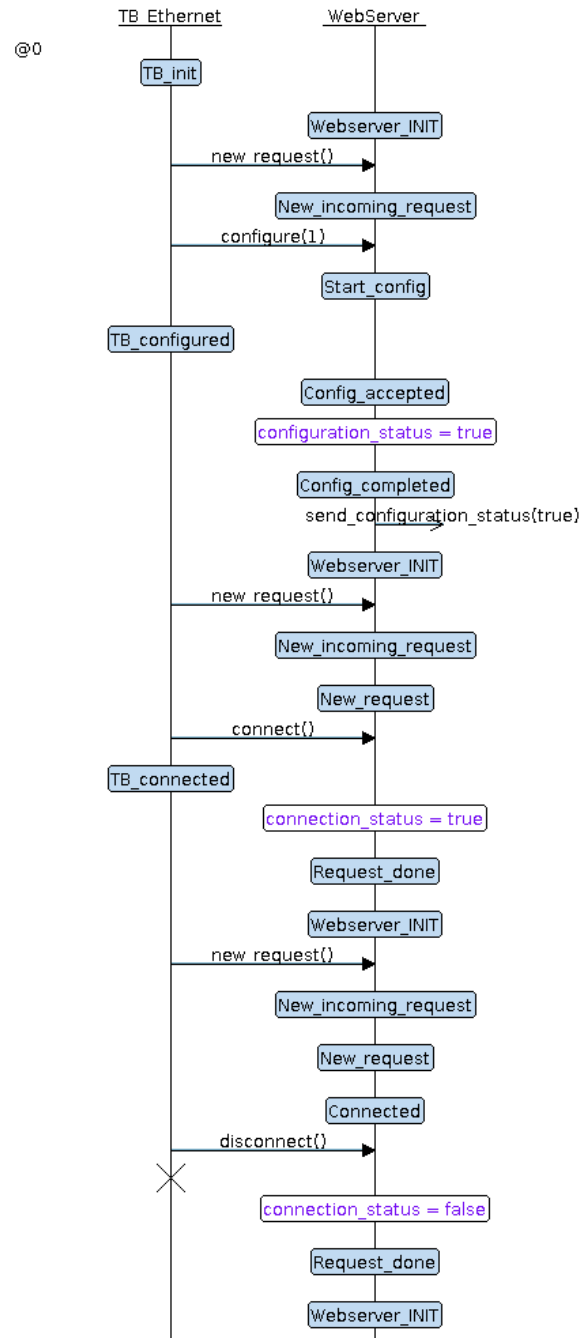
3.0.1 Light Management



3.0.2 Camera Management



3.0.3 WebServer



4 Formal Verification

For the formal verification, I verified that whenever a moving object has been detected, an email will be send. The formal proof has shown that both these states are reachable and the liveness property is guaranteed.

Reachability of: WebServer.Send signal: send_email(email)
-> property is satisfied

Reachability of: WebServer.Receive signal: motion_detected()
-> property is satisfied

Liveness of: WebServer.Send signal: send_email(email)
-> property is satisfied

Liveness of: WebServer.Receive signal: motion_detected()
-> property is satisfied