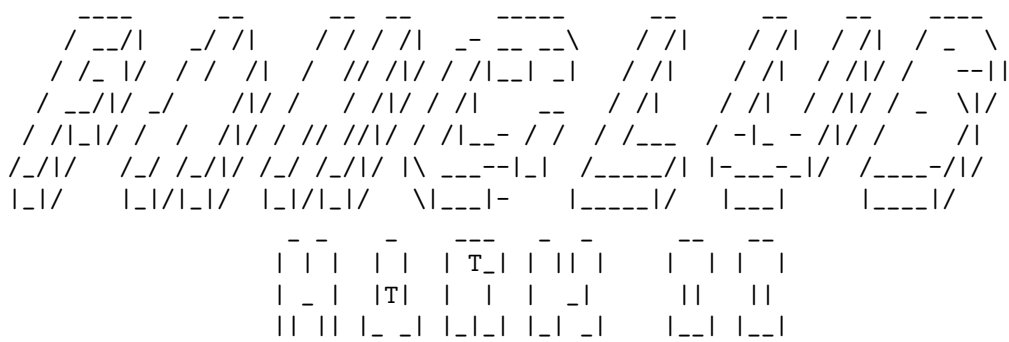

"FANCLUB MkII" USER MANUAL AND DESIGN SPECIFICATION (0)

Revised January 27, 2018

Alejandro A. Stefan Zavala

alestefanz@hotmail.com

Institution:

California Institute of Technology
Graduate Aerospace Laboratory &
Center for Autonomous Systems and
Technologies
1200 East California Boulevard
Pasadena, California 91125

Supervisors:

Christopher J. Dougherty
cdougher@caltech.edu
Marcel Veismann
mveisman@caltech.edu

Contents

INTRODUCTION

[Leave the introduction for last!] [In short, the purpose of the Fanclubproject is the operation of Fan Array Wind Tunnels. More specifically...]

NAMING CONVENTIONS

[By the time I found out the correct spelling would be "Fan Club," it was too late.]

Part I

USER MANUAL

Part II

DESIGN SPECIFICATION

Part II of this report describes the design process behind Fanclub MkII; this includes the underlying objectives, precedents, design choices made, and the reasoning behind them.

1 PRECEDENTS

[We stand on the shoulders of giants. We inherit the fruits of their greatness --- and the consequences of their sins.]

2 OBJECTIVES

2.1 PRIMARY OBJECTIVES

These objectives[1] are critical to the use of Fanclub MkII and need be fulfilled. FCMkII *must* :

- Given no ‘‘external" failures (e.g misconfigured firewall, broken hardware), reliably secure a connection between the given Master and Slaves in the network (as chosen by the user), and, furthermore
  - Maintain said connection until it is terminated either by the user or an external failure
  - Detect any external failures and adjust accordingly, whether by automatically re-connecting or alerting the user
  - In the event of an unexpected disconnection, affected Slaves with active fans must immediately shut down said fans. This is to say that *no fan can be ‘‘on" if it cannot be turned off by the user.*
- Low level fan control (I/O)
- Feedback loop
- Master control
- Display feedback at Master
- Slave feedback by Slave
- Detect bad fans

2.2 SECONDARY OBJECTIVES

These objectives are not critical to the use of Fanclub MkII , but are a significant enough improvement on the usefulness of the software to be pursued. Unlike the aforementioned objectives, failure to meet any of these objectives does not render the software unfit for its mission. FCMkII may:

- GUI
- Graphing
- All parameters
- Save profiles and logs
- Graph measurements

[1] Additional requirements that are specific to communications can be found under EXPECTED BEHAVIOR.

2.3 ADDITIONAL OBJECTIVES

Lastly, although these objectives are not pursued in the current design, they are worth documenting as potential for future work.

- ‘‘Offline" full-tunnel operation, possibly w/ LCD
- USB mode
- ‘‘Online" software loading onto Slaves

3 GENERAL SPECIFICATION

3.1 GENERAL NETWORKING

One of the defining tasks of Fanclub MkII is that of conveying information from the user to the system's microcontroller units (commands...) and vice versa (replies, measurements...). This is achieved through networking --- in particular, this is achieved using low-level socket programming in a form of [...]

3.1.1 EXPECTED BEHAVIOR

The networking capability of Fanclub MkII must satisfy the following:

- Both Master and Slaves can handle irresponsiveness from the other party, by verifyin the connection, attempting reconnection, or issuing disconnection alerts.
- Slaves can detect "irresponsive" networks --- defined here as one in which no IP address can be obtained using the Mbed *EthernetInterface* class, which in most cases implies the Slave is physically disconnected. Furthermore:
  - Slaves in such irresponsive networks will automatically resume normal execution when the network is fixed, without the need to be restarted by the user, no matter at what point of execution the network becomes irresponsive.
- Once a connection is established, the user will receive (in Master) periodic updates on the state of the connected Slaves without the need for explicit request. These updates include the current task, if any, and most recent RPM reading of all active fans; the period of these updates can be configured, but is expected to be of less than a second.
- The "relevant" information of a Slave can be accessed (and modified, if applicable) through Master once a connection is established. The information deemed "relevant" is defined in [#&SLAVE PROFILE].

3.1.2 SUMMARY

The flow chart below summarizes the communications process between Master and Slave. Note that this is a simplified illustration of the actual algorithm.

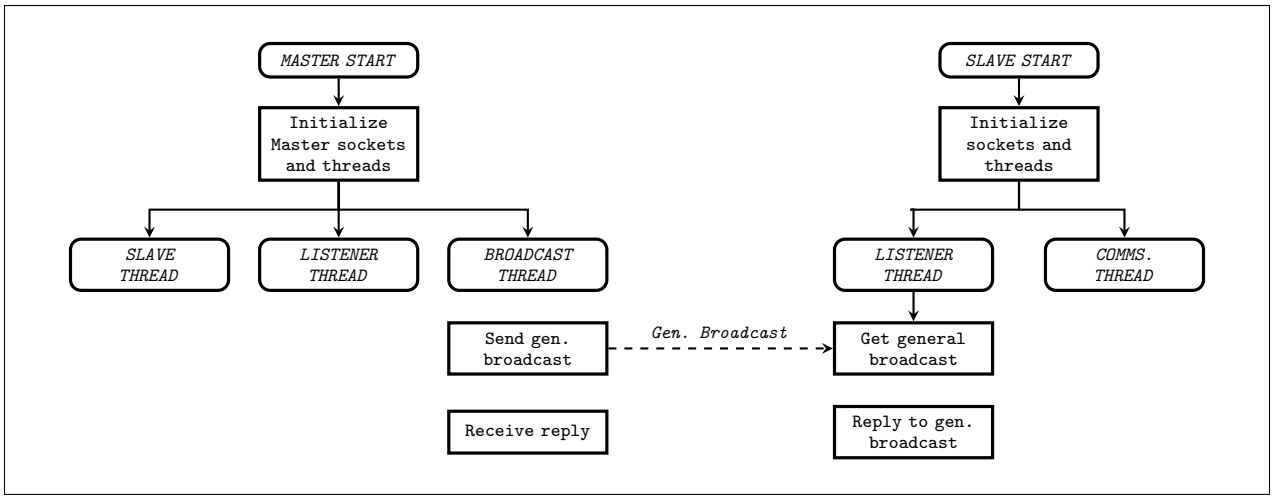

Figure 1: Illustrated summary of communications between Master and one Slave

4 "MASTER" SPECIFICATION

5 "SLAVE" SPECIFICATION

5.1 MODULAR BREAKDOWN

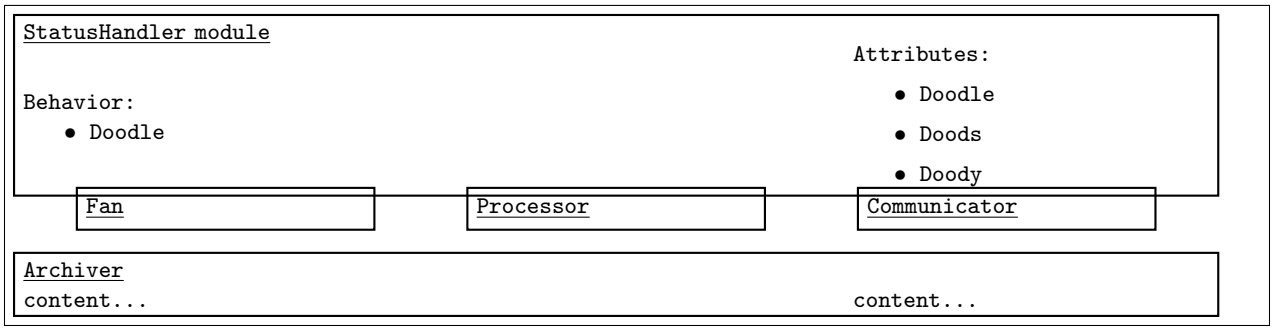

Figure 2: Modular breakdown of the Slave program

5.2 ‘‘SLAVE" NETWORKING

In a Slave, all networking functions (e.g receive a command, send a reply, secure a connection) are handled by its Communicator module.

References