Project F192 - Servo Digitizer

The Servo Digitizer is developed to channel control input from a model airplane radio remote control, the Futaba Skysport 4, to a flight simulator program running in a PC. It translates a pulse-width modulated signal output from the remote control into ASCII data and sends them to the PC via an RS-232 serial port. A flight simulator program in PC can use the data from the RS-232 port to control a simulated airplane in the PC. The software component of this project demonstrates the use of a source code modularity technique with a microcontroller that has limited resources.

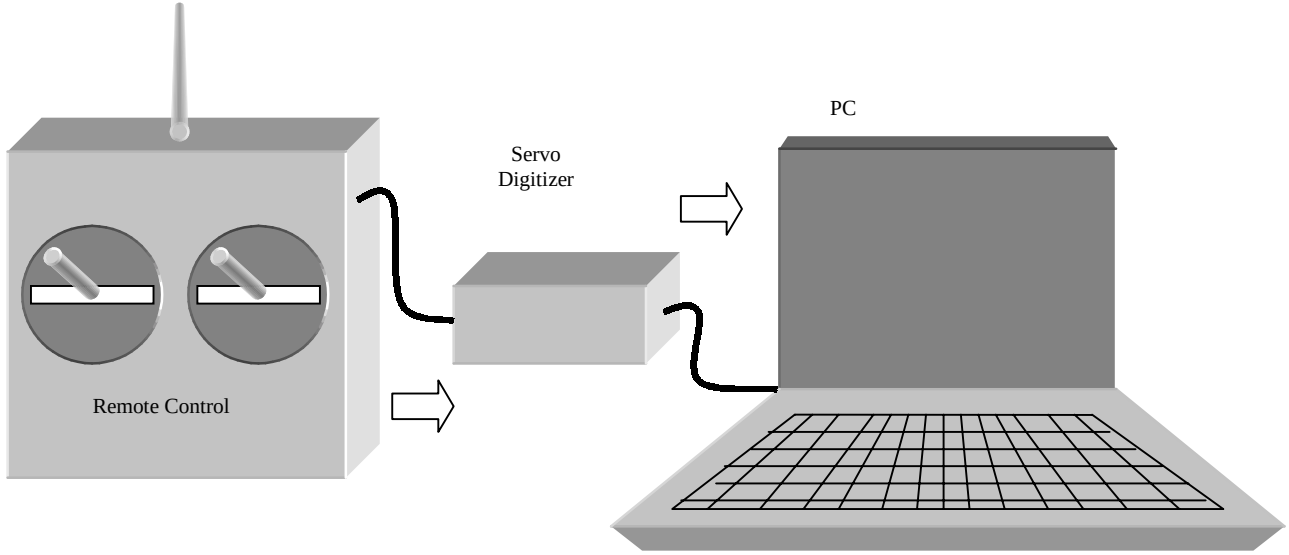
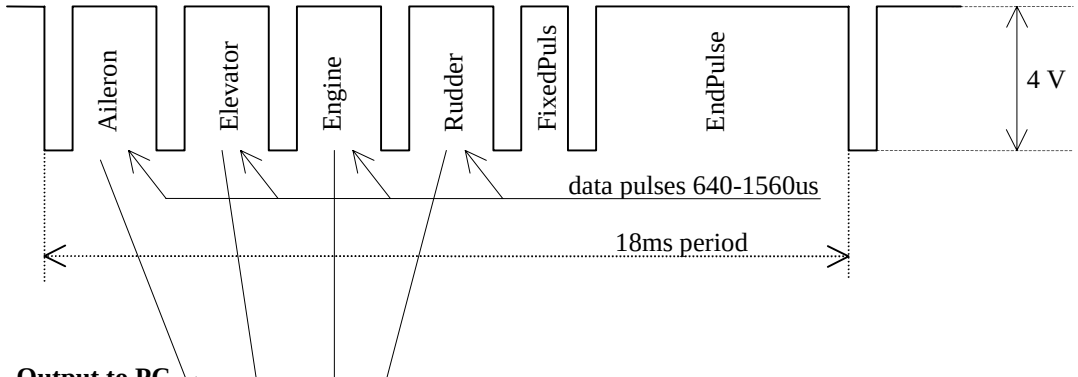


Diagram 1: Overall Connection

The Futaba Skysport 4 remote control has two X-Y joysticks that allow a user to change four flight parameters on a model airplane: alleron, elevator, engine, and rudder. A user can change a flight parameter by moving the corresponding joystick to the left, right, up or down. The remote control then sends signals via a radio frequency to a receiver on a model airplane. The Futaba Skysport 4 also has a DIN-6 connector at the back. One of the pins on this connector sends an output signal that corresponds to the joystick positions. This output signal becomes the input signal for Servo Digitizer.

Input Signal



The Servo Digitizer uses input capture feature of the microcontroller to measure widths of the input pulses. A serial port is simulated using output compare feature of the microcontroller. The Servo Digitizer must transmit the output ASCII text to the PC within 18ms, therefore selecting a correct baud rate is crucial.

The hardware consists of a power supply, the MC68HC908QT4 microcontroller, and the RS-232 transceivers. A DIN6 connector connects the board to Futaba model airplane remote control, which provides 0-4V input signal and a 9V DC battery to power the board. A linear regulator provides 5V DC power to the rest of the system. The system clock comes from a 24MHz crystal. A female DB9 connector is used to send the output to a PC.

A special software framework is created for this project. The framework is an integral part of the software as it organizes the source code into layers of code module. The software is written entirely in C language using Metrowerks CodeWarrior development environment. The source code is organized into three abstraction layers. The three layers are Startup, Dispatcher, and Operation layers. A layer consists of one or more modules. Each module has dedicated task and only refers to the other modules, in the same or different layers, in a predefined way. When possible, the framework encourages the separation of code and data into different files. When the modules need to be changed for a new environment, most or all of the needed parameters reside in the data file (.h file), while there is little or no change in source file (.c).

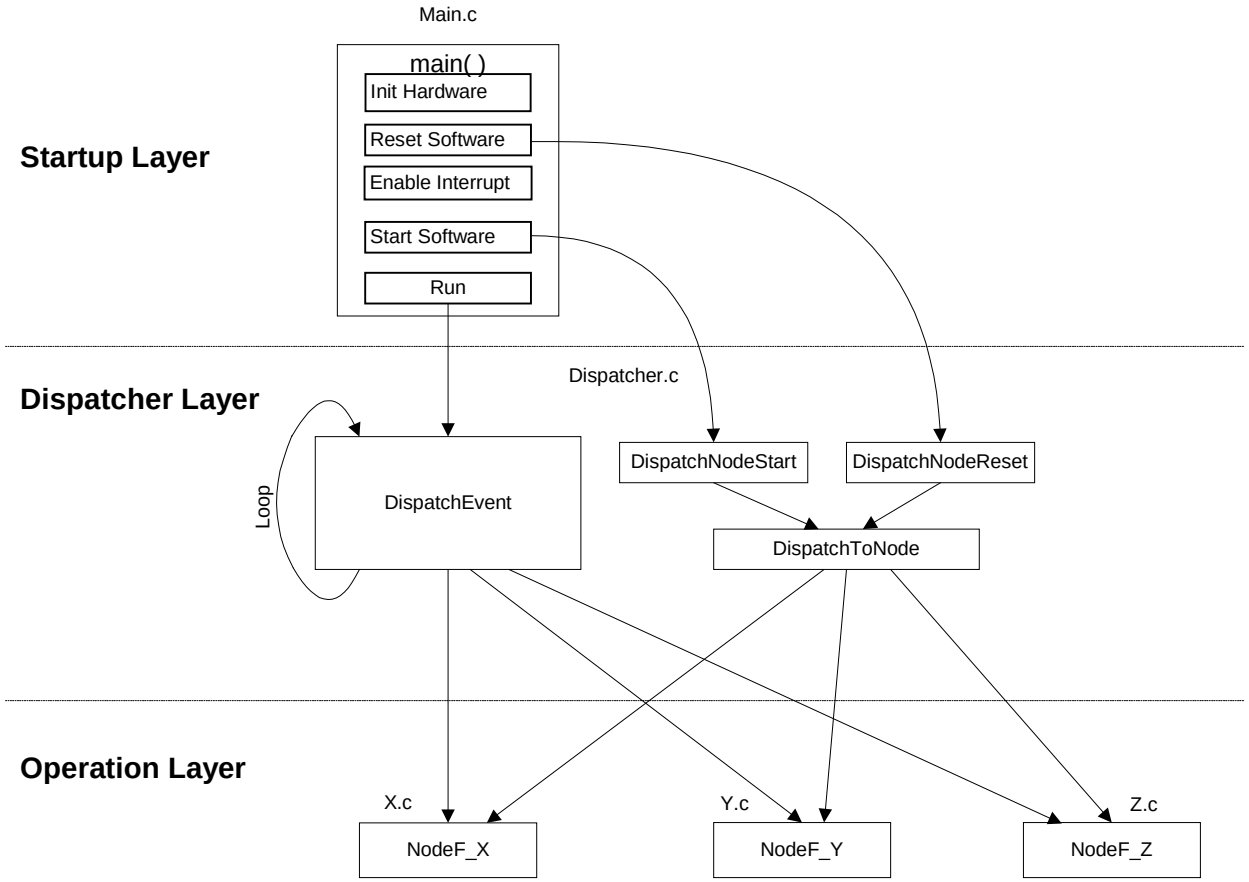
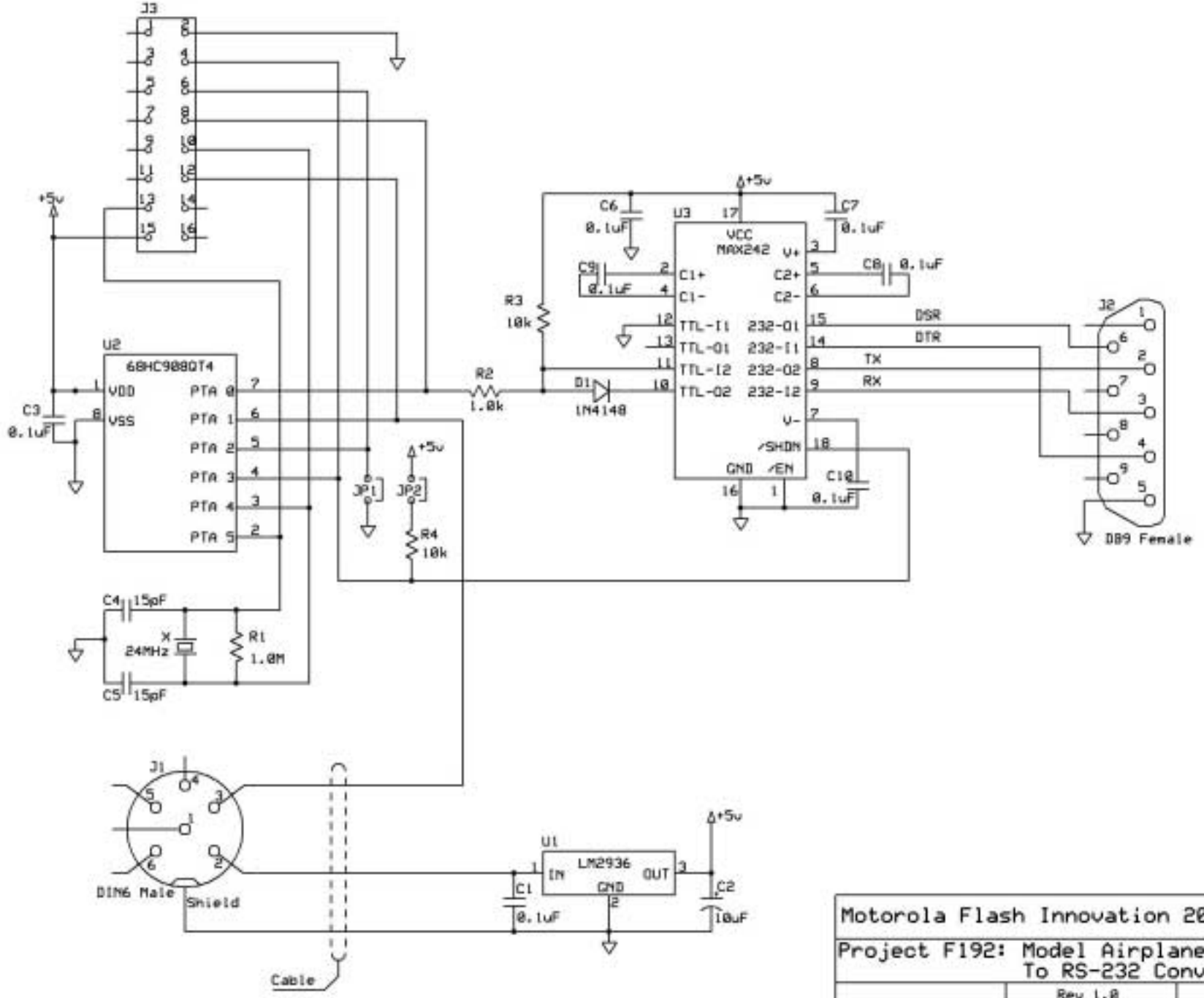


Diagram 3: Execution Flow

In addition, the software can be easily modified for use with the MC68HC908QT/Y microcontroller in other applications by adding new modules and updating the parameters in the header files.



Motorola Flash Innovation 2003 Contest		
Project F192: Model Airplane Controller To RS-232 Convertor		
	Rev 1.0 6/3/2003	1 OF 1


Futaba
SKYSPORT 4