

Design (E) 314

Technical Report

Digital Multimeter and Signal Generator

Author: Student Number:

Joshua Du Plooy Joshua Du Plooy

May 21, 2022

**Plagiaatverklaring / Plagiarism Declaration**

1. Plagiaat is die oorneem en gebruik van die idees, materiaal en ander intellektuele eiendom van ander persone asof dit jou eie werk is.

*Plagiarism is the use of ideas, material and other intellectual property of another’s work and to present is as my own.*

1. Ek erken dat die pleeg van plagiaat ’n strafbare oortreding is aangesien dit ’n vorm van diefstal is.

*I agree that plagiarism is a punishable offence because it constitutes theft.*

1. Ek verstaan ook dat direkte vertalings plagiaat is.

*I also understand that direct translations are plagiarism.*

1. Dienooreenkomstig is alle aanhalings en bydraes vanuit enige bron (ingesluit die internet) volledig verwys (erken). Ek erken dat die woordelikse aanhaal van teks sonder aanhalingstekens (selfs al word die bron volledig erken) plagiaat is.

*Accordingly all quotations and contributions from any source whatsoever (including the internet) have been cited fully. I understand that the reproduction of text without quotation marks (even when the source is cited) is plagiarism.*

1. Ek verklaar dat die werk in hierdie skryfstuk vervat, behalwe waar anders aangedui, my eie oorspronklike werk is en dat ek dit nie vantevore in die geheel of gedeeltelik ingehandig het vir bepunting in hierdie module/werkstuk of ’n ander module/werkstuk nie.

*I declare that the work contained in this assignment, except where otherwise stated, is my original work and that I have not previously (in its entirety or in part) submitted it for grading in this module/assignment or another module/assignment.*

|  |  |  |
| --- | --- | --- |
|  |  | 19862245 |
| Handtekening / *Signature* |  | Studentenommer / *Student number* |
| J Du Plooy |  | May 21, 2022 |
| Voorletters en van / *Initials and surname* |  | Datum / *Date* |

Abstract

Stellenbosch University has conscripted engineering students to design, build, test and evaluate a multi-use instrument that provide two main functionalities: No. 1) To operate as a digital multimeter capable of measuring DC and AC voltage and current. No. 2) To operate as an arbitrary waveform generator capable of adjustable amplitude, frequency, and shape. The purpose of this project is to educate students about the full process of electronic system design, with regards to circuit and software design, implementation and debugging and possible pitfalls one might come across during this process.

In order to streamline the process for students, certain design elements were pre-determined in order to manage procurement of parts and standardizing the overall design amongst all students. Students would be required to make design choices in circuitry and software based on calculations, measurements, testing and simulations wherever possible.

This project resulted in a design that can perform all system requirements excluding the digital multimeter measurement of DC and AC current. The system performs well under pressure however, performing specific tasks in a specific order result in the digital multimeter not functioning correctly.

Contents

[1 Introduction 1](#_Toc104561939)

[1.1 Introduction 1](#_Toc104561940)

[1.2 Methods Used in the Project 1](#_Toc104561941)

[1.3 Project Results 1](#_Toc104561942)

[1.4 Discussion 1](#_Toc104561943)

[1.5 Report Overview 1](#_Toc104561944)

[2 System description 1](#_Toc104561945)

[2.1 User Requirements 1](#_Toc104561946)

[2.2 High Level System Description 2](#_Toc104561947)

[3 Hardware design and implementation 3](#_Toc104561948)

[3.1 Hardware Block Diagram and Description of each interaction 3](#_Toc104561949)

[3.1.1 9V Battery interactions 4](#_Toc104561950)

[3.1.2 Power Supply Interactions 4](#_Toc104561951)

[3.1.3 Microcontroller 4](#_Toc104561952)

[3.1.4 Test Interface Connector 5](#_Toc104561953)

[3.2 Power Supply 5](#_Toc104561954)

[3.3 Buttons 6](#_Toc104561955)

[3.4 LEDs 6](#_Toc104561956)

[3.5 ADC (input stage) 6](#_Toc104561957)

[3.6 DAC (output stage) 7](#_Toc104561958)

[3.7 LCD 8](#_Toc104561959)

[3.8 Current Sensor 8](#_Toc104561960)

[4 Software design and implementation 8](#_Toc104561961)

[4.1 Software Concept Design 8](#_Toc104561962)

[4.2 Software Detailed Design 8](#_Toc104561963)

[4.3 LCD Driver and initialization functions. 10](#_Toc104561964)

[4.4 Button Debounce handling 11](#_Toc104561965)

[4.5 UART Communications protocol 12](#_Toc104561966)

[5 Measurements and Results 12](#_Toc104561967)

[5.1 Power Supply 12](#_Toc104561968)

[5.2 UART Communications 13](#_Toc104561969)

[5.3 Buttons 13](#_Toc104561970)

[5.4 Debug LEDs 13](#_Toc104561971)

[5.5 ADC 13](#_Toc104561972)

[5.6 DAC 14](#_Toc104561973)

[6 Conclusions 14](#_Toc104561974)

[Appendix A References 15](#_Toc104561975)

[Appendix B Complete Schematic 15](#_Toc104561976)

[Appendix C Microcontroller Pinout, Peripheral and Configuration 15](#_Toc104561977)

[Appendix D Figures and Tables 19](#_Toc104561978)

[Appendix E Communications log of Demonstration 4 30](#_Toc104561979)

List of Figures

1. [Hardware Block Diagram 3](#_Toc104561986)
2. [LED equivalent circuit 6](#_Toc104561987)
3. [unity gain operational amplifier circuit 7](#_Toc104561988)
4. [Software Concept Design 8](#_Toc104561989)
5. [menu Tree 12](#_Toc104561990)
6. [High-Level hardware concept design 20](#_Toc104561991)
7. [Spice simulation: Output Voltage of L7805CV against input voltage and current through power supply on/off LED (LD1) 21](#_Toc104561992)
8. [Suggested circuit for L7805CV [1] 21](#_Toc104561993)
9. [5V regulator current spike at 1.8Ω load 22](#_Toc104561994)
10. [Suggested circuit for LM2950 22](#_Toc104561995)
11. [current output to student board 23](#_Toc104561996)
12. [Time taken between UART message received and beginning of transmitting required output message 24](#_Toc104561997)
13. [Difference between 0x01 and 0x02 24](#_Toc104561998)
14. [The oscilloscope output of 0x99 or 1001 1001 in binary. 25](#_Toc104561999)
15. [Button Debouncing observed from oscilloscope 26](#_Toc104562000)
16. [Rise time of LD2 27](#_Toc104562001)
17. [Time taken for system to respond after button debouncing in software 28](#_Toc104562002)
18. [LED Brightness can easily be viewed while LCD is displaying DC voltage measured and a duty cycle measurement while scrolling. 28](#_Toc104562003)
19. [The voltage output of the DAC after filter and scaling when in DC, Pulse and sinusoidal modes. 29](#_Toc104562004)
20. [The rise and fall time of the microcontroller’s GPIO output pin. 29](#_Toc104562005)
21. [Spice simulation of LED equivalent circuit showing current through each LED, total current draw and LED voltage drop 29](#_Toc104562006)

List of Tables

1. [Global Variables used in software 9](#_Toc104561980)
2. [Multimeter measurements on power supply 13](#_Toc104561981)
3. [LED measurements from multimeter 13](#_Toc104561982)
4. [Project user requirements as stated in the PDD [1] 19](#_Toc104561983)
5. [Components made available to all students for the project 20](#_Toc104561984)
6. [Menu Logic 30](#_Toc104561985)

List of Abbreviations

**AC** Alternating Current

**DC** Direct Current

**PDD** Project Definition Document

**ADC** Analog to digital Converter

**DAC** Digital to Analog Converter

**I2C** Inter-Integrated Circuit

**UART** Universal Asynchronous Receiver/Transmitter

**V** Volt

**A** Ampere

**mA** milliampere

**GPIO** General Purpose Input Output

**PDD** Project Definition Document

**LED** Light Emitting Diode

**LCD** Liquid Crystal Display

**RX** Receive

**TX** Transmit

**UR** User Requirement

**SDA** Serial Data

**SCL** Serial Clock

**E** Enable

**RS** Register Select

**R/W** Read/Write

**MSB** Most Significant Bit

**DMA** Direct Memory Access

List of Symbols

**Ω** Ohm

**%** Percent (when used after a number)

**%** Modulo division (when used as an operator between two numbers)

# 

# Introduction

## Introduction

Stellenbosch University has conscripted engineering students to design, build, test and evaluate a multi-use instrument that provide two main functionalities: No. 1) To operate as a digital multimeter capable of measuring DC and AC voltage and current. No. 2) To operate as an arbitrary waveform generator capable of adjustable amplitude, frequency, and shape. The purpose of this project is to educate students about the full process of electronic system design, with regards to circuit and software design, implementation and debugging and possible pitfalls one might come across during this process.

## Methods Used in the Project

In order to streamline the process for students, certain design elements were pre-determined in order to manage procurement of parts and standardizing the overall design amongst all students. Students would be required to make design choices in circuitry and software based on calculations, measurements, testing, and simulations wherever possible.

## Project Results

This project resulted in a design that can perform all system requirements excluding the digital multimeter measurement of DC and AC current. The system performs well under pressure however, performing specific tasks in a specific order result in the digital multimeter not functioning correctly.

## Discussion

Although the design implemented contains its faults (small error resulting in nonfunctioning measurements and current measurement not correctly implemented), minor changes in software could result in a fully functional system within the system requirements.

## Report Overview

This report covers:

* An introduction detailing its background, methodology used in design, the final project outcome, and a discussion of that outcome.
* A high-level system description detailing a hardware concept design and how its various components can achieve the desired result.
* Hardware design and implementation that covers detailed full system design and detailed hardware design in various subsections outlined in the full system design.
* Software design and implementation detailing a high-level software concept design and detailed design of program sections outlined in the concept design.
* Measurements and results that covers important measurements for each hardware subsystem and a discussion pertaining to the system requirements.
* A conclusion containing discussions on various non-compliances, design shortcomings and possible improvements.

# System description

## User Requirements

From the user requirements, stated in Table 4 as is in the PDD [1], the following subsystems will need to be designed and implemented:

* From UR1 a 5V and a 3.3V regulated power supply circuits needs to be designed and implemented.
* From UR2, UR3, UR4, UR5, UR6 and UR7 a microcontroller needs to be selected to handle the following:
  + 1 dedicated pin for ADC handling
  + 1 dedicated pin for DAC handling
  + 2 dedicated pins for I2C handling
  + 16 dedicated pins for GPIO 5, of which need separate interrupt channels for buttons, debug LEDs and LCD screen
  + UART connection protocol capable of handling 8 data bits, no parity bits and a single stop bit at a data rate of 115200 baud transmitting and receiving.
* From UR2 DC/AC voltage measurements, between 0.1V and 2V (initially 3.3V, limits lowered), via microcontroller internal ADC. Generally, microcontrollers have a low current sink and draw limit, therefore a protection circuit needs to be designed and implemented.
* From UR2 DC/AC current measurements (up to 9mA) a current sensor is to be used and implemented, interfacing with the selected microcontroller via I2C protocol
* From UR3 signal generation (DC/AC and pulse waveforms with values ranging between 0.1V and 3.2V up to 5kHz and 100% duty cycle), due to the volatile nature of microcontrollers, DAC outputs signals usually have a fair amount of noise and filtering of this noise will require the design and implementation of a noise filtering circuit.
* UR4 will need to be handled internally by the selected microcontroller and state machines will need to be designed and implemented in software to handle the information to be transmitted and received.
* From UR5 and LCD screen will need to be used for displaying information pertaining to system settings, multimeter measurements, and signal generation. Therefore, circuitry surrounding the LCD screen will need to be designed and implemented. A state machine will need to be designed and implemented to keep track of LCD display. LCD screen driver functions will need to be designed and implemented in order to correctly start and use the LCD screen.
* From UR6 5 push buttons need to be implemented in order to change the system settings (measurement modes and signal generator parameters). A state machine will need to be designed and implemented in order to decode various button presses.
* From UR7 4 generic LEDs is needed to display various system modes. Surrounding circuitry needs to be designed and implemented in order to protect the LEDs. A state machine will need to be designed and implemented to keep track of various system modes.

## High Level System Description

In order for all students to have some standardization and integral components for the implementation of the project, the integral components, listed in Table 5 were bought in bulk and made available for all students. Starting from the microcontroller as a base, it will need 5V regulated power from the power supply.

Once powered the microcontroller can send and receive data via UART protocol to the interface connector or computer USB port for debugging purposes. This can be done through the TX and RX pins or the USB mini port on the STM32F303RE’s ST-link (a debugger/programmer probe interface for STMicroelectronics microcontrollers). A state machine will need to be designed to handle all data transmitted and received by the microcontroller.

The microcontroller can interface with an LCD screen through seven GPIO pins (4 data pins, an enable pin, read/write pin and register select pin). The LCD screen will need 5V regulated power from the power supply, and circuitry will need to be designed for the LCD screen contrast and backlight. LCD Screen drivers will need to be design in order to correctly interface with the LCD Screen. A State machine will need to be designed to handle what must be displayed on the LCD at any given time.

The microcontroller can measure voltage between 0 and 3.6V via an ADC pin, however due to the microcontroller’s low current and voltage sink limit a protection circuit will need to be designed. This can be achieved through the use of one channel on the available operational amplifier (which has high impedance) supplied for the project. Therefore, an unknown voltage (between 0.1V and 2V) is connected between the interface connector and microcontroller ADC pin via an operational amplifier.

Similarly, the microcontroller can output a voltage between 2.4V and 3.6V. This analog signal will need to be filtered for noise and scaled to between 0V and 3.2V. This can again be achieved via 1 channel on the supplied operational amplifier. The circuitry surrounding both channels of the operational amplifier will need to be designed. The microcontroller DAC pin (2.4V-3.6V) is connected to the operational amplifier’s second channel input and its output is connected to the interface connector. The operational amplifier will need to be powered through the power supply

In order to measure current, a current sensor is used and connected to the microcontroller through I2C (1 SDC pin and 1 SCL pin). Due to pin limitations on the interface connector, only 1 pin is available for connection to the current sensor. Therefore, a standard 220Ω resistor is to be connect with the current sensor and variable voltage (0V to 2 V) is applied to generate current. This is the same pin on the interface connector connected to the microcontroller ADC pin via the operational amplifier.

In order to interface with the multi-use tool, five push buttons are used to change its settings. The push buttons surrounding circuitry will need to be designed. These push buttons are connected to GPIO pins on the microcontroller and are supplied with 3.3V regulated voltage from the power supply. Due to a phenomenon known as button bouncing, a state machine in order to hand debouncing will need to be designed.

In order to show what mode, the multi-use tool is in, 4 LEDs are used. These LEDs are connected to 4 GPIO pins on the microcontroller. Surrounding circuitry for the LEDs will need to be designed

# Hardware design and implementation

## Hardware Block Diagram and Description of each interaction

Diagram, schematic

Description automatically generated

Figure 1: Hardware Block Diagram

### 9V Battery interactions

A 9V to 12 V battery is to power the system and connect to the power supply. Therefore, the power supply needs to handle a variable input of at least 9V to 12V

### Power Supply Interactions

* Microcontroller: The power supply is to power the microcontroller externally. Therefore, the power supply needs to supply a 5V regulated signal to the microcontroller. For the microcontroller to accept external power, the PWR needs to be bridged on the E5V side only.
* LCD Screen: The power supply is to power the LCD screen and its backlight with 5V regulated power. The current needs to be limited to the LCD screen backlight. A voltage divider circuit needs to be designed and tuned to get a suitable contrast
* Current Sensor: The current sensor requires 3.3V regulated power from the power supply.
* ADC Input Buffer and DAC Output Filter and Scaling: Because a 2-channel operational amplifier is used. Both the ADC input buffer and DAC Output Filter and scaling can make use of the same operational amplifier. Therefore, only a single positive and negative rail voltage is needed for the operational amplifier is needed. Due to the user requirements 3.3V and 0V regulated power, respectively, is sufficient to power the operational amplifier to function as needed (this is further explored in 3.5ADC (input stage) and 3.6DAC (output stage)) Therefore, the power supply needs to be able to supply 3.3V regulated power to the operational amplifier.
* Buttons: The buttons are used to interface with the microcontroller. Therefore, the buttons require a constant 3.3V regulated power from the power supply.
* Test Interface Connector: When the multi-use tool’s test interface connecter is connected to its counterpart device, the multi-use tool tester’s test interface connecter, it is supplied with a varying voltage signal between 9V and 12V. This is to act as a battery and a 9V battery is no longer needed to power the multi-use tool. The multi-use tool also measures the 5V and 3.3V lines that the power supply provides in order to ensure that the lines are within a 5% tolerance.

### Microcontroller

* LCD Screen: The LCD screen is to display various measurements and a settings menu when needed. The microcontroller is to control what is displaying on the LCD at any given time. This is done through the use of 7 GPIO pins. An enable pin to control when data or instructions is sent to the LCD screen from the microcontroller. A register select pin that choose if either an instruction or data is sent to the LCD screen from the microcontroller. A read/write pin that controls which direction data is sent (LCD to microcontroller or microcontroller to LCD). Four Data pins that can send half a byte of information at a time. The microcontroller will require driver functions in software to ensure that the LCD runs correctly. The LCD will need to be sent a serries of start up protocols that the microcontroller will have to send through these 7 GPIO pins before the LCD can be used in the correct mode of operation.
* Current Sensor: The current sensor interfaces with the microcontroller through I2C protocol. Meaning that the current sensor is connected to the microcontroller via two pins: a clock pin and a data pin. The current sensor has an address that the microcontroller needs to interface with in software to send instructions and receive data to and from the current sensor. The current sensor will need to be sent a serries of start up protocols through the I2C interface to ensure that the current sensor is used in the correct mode of operation. The current sensor will measure the current through a dedicated resistor. Due to the tolerance of the resistor, the value the current sensor reports will need to be slightly scaled as resistors available for use to students are only tolerant to 5%. This will be handled in software.
* LEDs: The microcontroller will need to switch on four LEDs separately when is required, depending on what mode of operation the multi-use tool is in. The LEDs will be powered directly from the microcontrollers GPIO pins therefore current limiting resistors are needed for each LED to ensure that the current does not exceed that of the LED’s or microcontroller’s limit.
* ADC Input Buffer: The microcontroller will receive voltage signal from the ADC input buffer between 0V and 2V via and ADC pin on the microcontroller. The microcontroller will need to sample the ADC at an appropriate rate in order to get an accurate value from the ADC input buffer. The microcontroller will need to interpret the ADC measurements in order to ascertain the correct offset, frequency, duty cycle and amplitude.
* DAC Output Filter and Scaling: The microcontroller will send a voltage signal between 2.4V and 3.6V to the DAC output filer and scaling sub-circuit via the microcontroller DAC pin. The sub-circuit will need to filter out noise and scale the voltage to an appropriate level. The microcontroller output will need to change its value at an appropriate rate in order to supply a clean DC or AC voltage. This will be handled in software
* Buttons: The buttons are used as input devices for the multi-use tool. 5 buttons are connected to the microcontrollers GPIO pins in pull up mode. Meaning that the voltage on the microcontroller pins is 3.3V when not being pressed an 0V when being pressed. Due to the microcontrollers low current sink limit. Pull up resistors will be needed for the buttons to operate correctly. The buttons are only meant have an effect when they are released (i.e., on positive edge trigger.). Therefore, the GPIO pins will be used in interrupt mode breaking away from normal functionality in software. Additionally, button debouncing will need to be handled in software.
* Test Interface Connector: The microcontroller will send and receive data in a predetermined format via UART protocol operating at 115200 baud/s with 8 data bits no parity and a single stop bit. The UART will make use of 2 pins on the microcontrollers ST-link and 2 pins on the test interface connecter respectively. One for sending data to the microcontroller and one for sending data to the testing station.

### Test Interface Connector

* ADC Input Buffer and Current Sensor: A single pin on the test interface connecter is used to send a voltage signal (0V to 2V) to the ADC input buffer and Current Sensor. The current sensor will need surrounding circuitry with a dedicated resistor to generate current.
* DAC Output Filter and Scaling: The test interface connecter is connected to the output of the DAC output filter and scaling operational amplifier. This allows the signal being sent to the test interface connecter to be between 0V and 3.2V instead of 2.4V and 3.6V.
* LEDs: The test interface connector is connected to the positive terminal on LD2 to ensure that the multi-use to is in the correct mode of operation.
* Buttons: Each button is connected to the test interface connector (on the same pins connected to the microcontroller GPIO pins) to allow the test interface connector to simulate button presses. Button bouncing will also be sent through the test interface connector to ensure that button debouncing is handled in the microcontroller software.

## Power Supply

Using the available 5V voltage regulator (L7805CV) and suggested circuit (Figure 8: Suggested circuit for L7805CV) a SPICE simulation was done against a variable input voltage range from 7V to 13V, surpassing the range required of the multi-use tool (Figure 7: Spice simulation: Output Voltage of L7805CV against input voltage and current through power supply on/off LED (LD1)).

The output voltage varied between 5.0028462V and 5.0028477V, a mere 1.4305115µV difference with no external load. When input is kept at 9V and a variable load is applied. The internal limits of the voltage regulator limits current and a minimum of 1.8Ω is required for the regulator to function at or near 5V (±0.2V). Below 2Ω the output voltage of the regulator increases linearly from 0.001Ω (simulation not possible at perfect short circuit) with a slope of 2.7A. Additionally, the voltage regulators output current spikes at 1.8Ω with a peak current of 1.8A and decays according to ohms law. This shows that the peak current output of the regulator is 2.7A.

The 3.3V voltage regulator’s (LM7950-3.3) proposed circuit can be seen in Figure 10 as seen in its datasheet. [2] According to the datasheet it has a maximum input voltage of 30V and a maximum current draw of 100mA. This needs to be noted for the design of button pull up resistors and current sensor.

## Buttons

With the microcontroller set to pull up mode, an internal pull up resistor between 25kΩ and 55kΩ and a typical value of 40kΩ [3]. This gives a typical current of 82µA when the pin is pulled to ground by the button. Adding any extra resistors will lower this resistance as the added resistor would be in parallel. The lower the resistance the faster the pin can be pulled to ground or pulled up to 3.3V. However, Figure 20 shows that the fall time of the pin is 58ns and the rise time is 11.2µs. This is far below the required response time of the system, and a parallel resistor is not required.

## LEDs

A generic datasheet for 3mm green LEDs [4] has a recommended current of 20mA. However, the stm32F303RE has a pin limit of 25mA and total limit of 80mA [3]. A maximum of 3 LEDs should be on at a time would give a total of 75mA not mentioning the ADC, DAC, 2 I2C and 7 LCD pins that will need current of their own. The ADC and DAC will not need much current due to the high impedance of the operational amplifier and however, the I2C and LCD pins could require a substantial amount of current. Therefore, it is best to limit the current as much as possible and acquire noticeable light when powered. It is recommended that LEDs have a current between 10mA and 20mA. Therefore, using the circuit shown in Equation 1, it is recommended that 130Ω resistors are used. However, due to availability of resistors, the closest resistor available was 120Ω. This gives a current of 10.08mA. With a total current usage of 32.5mA if all three LEDs are on. A Spice simulation, Figure 21, shows that the projected total current draw on three LEDs is 31.42mA.

Diagram, schematic, box and whisker chart

Description automatically generated

Figure 2: LED equivalent circuit

Text

Description automatically generated with medium confidence

Equation 1: Recommended Resistor for LEDs

## ADC (input stage)

The ADC was initially required to measure a voltage range between 0V and 3.3V. This was later changed to 2V. Therefore, a unity gain operational amplifier was assumed to be sufficient. The following unity gain amplifier was implemented:

Diagram

Description automatically generated

Figure 3: unity gain operational amplifier circuit

## DAC (output stage)

Following the operational amplifier recommended design criteria [5] the following calculations was completed:

Graphical user interface, application, table

Description automatically generated

## LCD

## Current Sensor

# Software design and implementation

## Software Concept Design

Diagram

Description automatically generated

Figure 4: Software Concept Design

## Software Detailed Design

From the above Pseudocode, the general flow of the software implemented can be seen. Starting from the beginning. At the very top of the software are a number of global variables that various functions and interrupts interact with.

Table 1: Global Variables used in software

|  |  |  |
| --- | --- | --- |
| **Variable Type** | **Variable Name** | **Variable Use** |
| typedef | enum {  up, down, left, right, middle, off  } button; | A Type definition named button that allows the use of button type variables |
| button | ButtonInt = off; | A button type variable used to indicate when a button interrupt occurs and for what button simultaneously |
| button | ButtonState = off; | A button type variable used to indicated if a button is currently being pressed and which button simultaneously |
| button | ButtonPressed = off; | A button type variable used to indicate if a button has been pressed and released and which button simultaneously |
| uint32\_t | DebounceTim = 0; | An integer variable used to save the systick timer when a button interrupt occurs |
| uint8\_t | menuLayer = 0; | An integer variable used to determine and set what layer the menu is in currently |
| uint8\_t | menunumber = 0; | An integer variable used to determine and set what which menu item in that layer the menu currently is in |
| typedef | enum {Menu, MeasureAndOutput  } lcd\_state; | A Type definition named LCD\_state that allows the use of LCD\_state type variables |
| lcd\_state | LCDState = MeasureAndOutput; | An LCD state used to determine if the LCD is in Menu state or in measure and output state |
| uint8\_t | MeasureMode = 0; | Used to determine and set what measurement mode the system is in and which measurement display mode the LCD is in. 0=DCV;1=DCC;2=ACV;3=ACC |
| uint8\_t | SigGenType = 0; | Used to determine and set what Signal generation type the system is in and which signal gen display mode the LCD is in. 0=DC;1=Sinusoidal;2=Pulse |
| uint8\_t | SigGenOutput = 8; //7=On;8=Off | Used to determine and set if Signal generation output is off or on and if the LCD is in output off mode or signal gen display mode |
| uint16\_t | SigGenAmp = 1000; | The amplitude in mV the signal generator would generate if in the correct mode and in an output is needed |
| uint16\_t | SigGenOffset = 1200; | The DC offset of any signal generator type in mV |
| uint16\_t | SigGenFreq = 1000; | The frequency of Duty cycle or sinusoidal signal in Hz |
| uint8\_t | SigGenDuty = 25; | The duty cycle of the signal % |
| uint16\_t | SigGenAmpSelect = 1000; | The amplitude of a in mV the true amplitude would be if selected. This is displayed on the LCD and can be changed without setting the true amplitude |
| uint16\_t | SigGenOffsetSelect = 1200; | The DC offset of a in mV the true offset would be if selected. This is displayed on the LCD and can be changed without setting the true offset |
| uint16\_t | SigGenFreqSelect = 1000; | The frequency of a in mV the true frequency would be if selected. This is displayed on the LCD and can be changed without setting the true frequency |
| uint8\_t | SigGenDutySelect = 25; | The Duty cycle of a in mV the true Duty cycle would be if selected. This is displayed on the LCD and can be changed without setting the true Duty cycle |
| uint8\_t | RXByte[1]; | The single byte that the UART receive interrupt fills when there is an interrupt |
| uint8\_t | RXBuffer[20]; | The Buffer variable that RXByte is moved to when an interrupt occurs |
| uint8\_t | RXBufferCounter = 0; | A counter used to count up when UART interrupts occur. When RXByte receives the char ‘\n’ A full instruction has been received and a flag is set to true |
| uint8\_t | uartInstructionFlag = 0; | The uart instruction flag. When set to true, the uart decode function is called and the flag is set back to 0 |
| uint16\_t | measAmp = 0; | The measured Amplitude after decoding from the ADC |
| uint16\_t | measOffset = 0; | The measured Offset after decoding from the ADC |
| uint16\_t | measFreq = 0; | The measured Frequency after decoding from the ADC |
| uint16\_t | VoltageCounter = 0; | A counter that counts the number of times the top of a sinusoidal signal is measured |
| uint16\_t | VoltageBuffer[1024]; | The Voltage Array that the ADC fills |
| uint32\_t | DACOutputData[128]; | The DAC output array that the DAC DMA uses to output a voltage signal |
| uint8\_t | scroller = 0; | A counter that counts through the stages of scrolling on the LCD |
| uint32\_t | freqencytimer = 0; | The timer that measures takes the systick time to measure the period of time it takes to fill the ADC buffer |
| uint32\_t | freqencytimerprev = 0; | The timer that saves the previoussi systick time to measure the period of time it takes to fill the ADC buffer |

Following the global variables are all functions that either drive the LCD, initialize the system, or contain state machines.

## LCD Driver and initialization functions.

The LCD needs various driver functions in order to work:

* void lcdScreenUpdate(char message, uint8\_t RS): This function first takes the 4 most significant bits of the message variable and sets the 4 data pins on the led as such and the RS variable is used to set the RS pin on the LCD. Next the enable pin is set and immediately Timer is started in counter mode, when the counter reaches 100, the enable pin reset to 0 and the timer is started again and waits until the counter reaches 100 before exiting the function. This timer is a microsecond counter and allows the LCD to wait under the millisecond range ensuring that minimal code blocking occurs when instructions and characters are sent to the LCD.
* void lcdScreenCommand(char command): A function that takes a byte and splits it in half and calls the LCD Screen update function. With RS set to instruction mode
* void lcdScreenPlaceCurser(uint8\_t row, uint8\_t column) : A function that uses the command function to place the LCD curser where ever needed
* void lcdScreenData(char data) A function that takes a byte and splits it in half and calls the LCD Screen update function. With RS set to read mode
* void lcdScreenDysplayString(char \*string): a function that steps through a string and calls LCDScreen Data function at each character.
* void initializeBoard(): A function that is run at the start of the program’s main function.
  + First it sets LD2 on
  + Next it transmits the student number Uart protocol. (“@,19862245,!\n”)
  + Then a 5ms delay function is used(since this is outside the main while loop, code blocking is not a concern yet)
  + The program sends 0x30 shifter right 4 bits through the LCD screen update command . This is the go to 8 bit mode command.
  + Then a 5ms delay function is used.
  + The program sends 0x30 shifter right 4 bits through the LCD screen update command
  + Then a 5ms delay function is used
  + The program sends 0x30 shifter right 4 bits through the LCD screen update command
  + Then a 1ms delay function is used
  + The program sends 0x30 shifter right 4 bits through the LCD screen update command
  + Then a 10ms delay function is used
  + The program sends 0x20 shifter right 4 bits through the LCD screen update command. This returns the curser to the home position
  + Then a 5ms delay function is used
  + The program sends 0x28 through the LCD screen update command.
  + Then a 1ms delay function is used
  + The program sends 0x08 through the LCD screen update command.
  + Then a 1ms delay function is used
  + The program sends 0x01 through the LCD screen update command.
  + Then a 1ms delay function is used
  + The program sends 0x06 through the LCD screen update command.
  + Then a 1ms delay function is used
  + The program sends 0x0C through the LCD screen update command.
  + Uses place curser function to set the curser to the top left
  + “Start message” is sent through the LCD display string function
  + LCDmenumode variable is set to menu
  + The generate DAC output function if call to set the signal generator to 0V;

## Button Debounce handling

The Buttons are connected to GPIO pins in Pull Up mode and rising and falling edge trigger detection enabled. Generate global interrupt is also enabled for these pins. As seen in Figure 4, when an interrupt is generated (rising or falling) the button interrupt variable is set to its respective enum value. Also inside the interrupt handler, the systick time is saved in a variable called DebounceTim. Inside the main loop a function called Button debounce is called. If the systick timer 20 more than the debounce time variable, the button’s pin that caused the interrupt is checked if it is set or reset if. If it is currently set then the button pressed variable is set to that button. In the main loop, the function menu logic is called when the button pressed variable is set to anything but off.

**Error! Reference source not found.** Shows the menu tree as seen in the PDD. The variable menulayer refers to the layer on this tree. And the variable menu number refers to the number counting from the left on that layer.

**Error! Reference source not found.** shows how each menu item is referred to in software. Inside the Function menulogic, there is 2 layers of switch case statements. First the menu layer then the menu number. Inside each case are 5 if statements asking if that button has been pressed. if the down or up button is pressed the menu layer is increased or decreased by one and its respective menu number is updated. If you cannot go further down or up the branch in that branch the button does nothing similarly to the left and right buttons increasing and decreasing the menu number if possible. If at the top of the tree, and the middle button is pressed, its if statement will turn off the menu LEDs LD2 off and LD3 on. When at the bottom of the tree the and the middle button is pressed, its respective mode, type or parameter will update to that value. If the left and write buttons are pressed while in the Amplitude, offset frequency or duty cycle value select layer, the respective variable have a duel variable with the same name underscore select is displayed on the LCD if left or right button is pressed this value decrease by 100 or 10, if in duty cycle and similarly, if this value reaches its limit, the button will do nothing. At this stage if the middle button is pressed, the respective counterpart is set to the select versions value and the menu layer is reduced and the menu number is set to its correct value. If any type parameter or output is changed the DAC Output function is called to update the signal generator. Inside the if statements when a button is pressed and it is possible for something to happen the button pressed variable is also set to off and the function menumatrix is called.

Menu matrix is a function that checks what menu layer and menu number the system is at and sends its corresponding strings to the function LCD String display.

Diagram

Description automatically generatedDiagram

Description automatically generatedDiagram

Description automatically generated

Figure 5: menu Tree

## UART Communications protocol

Before the main loop starts, the UART receive mode is started in interrupt mode waiting for a single byte. Inside the interrupt handler, when a byte is received, it is inserted into a buffer array at the placement of a counter. This counter immediately counts up. If the received value is the character ‘\n’ then a flag is set to true. Outside the if statement, the received by is then set to null. In the main loop if this flag is true it is immediately set back to 0 and the function Decode Instruction is called.

Inside the function a switch case statement is used on the third value in the buffer array. This determines what kind of instruction it is if a the character ‘#’ is found it decodes the rest of the buffer and calls the LCD String display or LCD command function depending on what the fifth character in the array is. If the third character is a '$' the measure mode is updated along with their respective LEDs. If a the third character is a '\*' then either status or measure request is require, the signal generator output is update if necessary. And the correct transmission is given using the corresponding system variables where necessary. When a measurement request is given, the following method is used to turn the integers into ascii values. If X1X2X3X4 represents the number one thousand two hundred and thirty four then:

* X1=1234/1000 +48
* X2=(1234/100)%10 +48
* X3=(1234%100)/10 +48
* X4=1234%10 +48

# Measurements and Results

## Power Supply

Power supply measurements can be seen in Table 2 and meet the user requirements of within 5% of 5V and 3.3V. Additionally student board uses an average of 50mA according to bench power supply.()

Table 2: Multimeter measurements on power supply

|  |  |  |
| --- | --- | --- |
| **Measurement** | **Value** | **SI Unit** |
| L7805CV output @ input=9V | 4.93 | V |
| L7805CV output @ input=12V | 4.95 | V |
| L7805CV input current @ input=12V | 50 | mA |
| LM7850 output @ input=12V | 3.31 | V |

## UART Communications

From the communications log provided by demonstration 4(Appendix EAppendix E) it is clear that UART functionality works as intended, responding to set messages within the appropriate format. Additionally, Figure 12 shows the time taken between the end of a message received and the start of a message sent (32µs). Figure 13 shows the difference between 0x01 and0x02. Figure 14 shows that the baud period is 69.6µs. From this, the true baud rate can be determined to be 115942 bits/s which is close enough to 115200 bits/s. This ensures that the UART communication protocol for the multi-use tool works as required.

## Buttons

By physically pushing a button, button bouncing can be observed on an oscilloscope (Figure 15). From personal testing (on this system) button bouncing takes place in the microsecond range(±250µs). However, from research [6], button bouncing can take up to 6,2ms. However, this system will be tested by an automated system that will induce artificial button bounce, according to the client, of up to 50ms [7]. This seemed a bit excessive and decided to make a design choice of 20ms.

## Debug LEDs

The Debug LEDs respond as instructed according to the project definition document. Additionally, The LEDs take approximately 18µs to rise in voltage (Figure 16). From a generic green LED datasheet, the LEDs peak current for the assumed LED used is 30mA has a forward voltage typically between 2V and 2.4V. However, the LED is sufficiently bright enough when running at an average of 10.625mA(Figure 18).

Table 3: LED measurements from multimeter

|  |  |  |  |
| --- | --- | --- | --- |
| **LED** | **Measurement** | **Value** | **SI Unit** |
| LD2 | LED voltage drop | 2 | V |
|  | Resister voltage drop | 1.2 | V |
|  | Current through the LED | 10 | mA |
| LD3 | LED voltage drop | 1.8 | V |
|  | Resister voltage drop | 1.4 | V |
|  | Current through the LED | 11.7 | mA |
| LD4 | LED voltage drop | 1.9 | V |
|  | Resister voltage drop | 1.2 | V |
|  | Current through the LED | 10 | mA |
| LD5 | LED voltage drop | 2 | V |
|  | Resister voltage drop | 1.3 | V |
|  | Current through the LED | 10.8 | mA |

## ADC

The ADC has been shown to be not fully reliable. The voltage input on the operational amplifier unity gain circuit has been shown to be floating when no or low voltage is applied. The pin floats as high as 2V which is the full range of measurement required for the multi-use tool. However, when a voltage of at least 0.1V is applied, DC voltage measurements are reliable up to 2.6V, above what is required.

## DAC

The DAC output performs as required. Figure 19 show the DAC output at the output of the operational amplifier. In pulse mode, with required frequency of 1kHz the output is 1kHz almost exactly, with a set duty cycle of 25%, the DAC outputs 25.7%. The amplitude in is set to be 1V and measures 1.08V and a offset set to 1.2V and measures 1.2V.

In sinusoidal mode, the offset is set to be 1.2V and measures 1.4V. The frequency is set to be 1kHz and is measured at 992.1Hz. The amplitude is set to be 1V and measures 1.04V

In DC mode, the offset is set to be 1.2 but measures 1.24V.

All outputs measured by the DAC are within the required 5% tolerance.

# Conclusions

1. References

|  |  |
| --- | --- |
| [1] | A. Barnard, L. Grootboom, U. Louw and D. Klink, "Design (E) 314 Project Definition," Stellenbosch University, Stellenbosch, 2022. |
| [2] | HTC, "100mA LOW DROPOUT VOLTAGE REGULATORS," December 2015. [Online]. Available: https://learn.sun.ac.za/pluginfile.php/3397348/mod\_resource/content/3/LM2950.pdf. [Accessed 04 May 2022]. |
| [3] | ST Microelectronics, "NUCLEO-F303RE," [Online]. Available: https://www.st.com/en/microcontrollersmicroprocessors/stm32f303re.html. [Accessed 26 05 2022]. |
| [4] | CHINA YOUNG SUN LED TECHNOLOGY CO., LTD., "Model No.: YSL-R341K3D-D2," All About Circuits, [Online]. Available: https://www.sparkfun.com/datasheets/Components/LED/YSL-R341K3D-D2.pdf. [Accessed 04 May 2022]. |
| [5] | B. Carter, "Designing Gain and Offset in Thirty Seconds," February 2002. [Online]. Available: https://learn.sun.ac.za/pluginfile.php/3434727/mod\_resource/content/0/sloa097-Designing%20Gain%20and%20Offset%20in%20Thirty%20Seconds.pdf. [Accessed 18 May 2022]. |
| [6] | M. Maxfield, "Ultimate Guide to Switch Debounce (Part 4)," Engineering Journal, 19 February 2020. [Online]. Available: https://www.eejournal.com/article/ultimate-guide-to-switch-debounce-part-4/#:~:text=As%20Jack%20Ganssle%20reported%20in,microseconds%20(or%206.2%20ms).. [Accessed 01 May 2022]. |
| [7] | A. Barnard, "Official module forum + Questions and Answers," University of Stellenbosch, 2 March 2022. [Online]. Available: https://learn.sun.ac.za/mod/forum/discuss.php?d=349035. [Accessed 1 May 2022]. |

1. Complete Schematic
2. Microcontroller Pinout, Peripheral and Configuration

|  |  |  |  |
| --- | --- | --- | --- |
| Stm32 Pin | Peripheral | Configuration | Connection |
| PA5 | GPIO | * External interrupt mode with rising/falling edge trigger detection (interrupt 0x0020) * Pull Up Mode | Right push button |
| PA6 | GPIO | * External interrupt mode with rising/falling edge trigger detection (interrupt 0x0040) * Pull Up Mode | Middle push button |
| PA7 | GPIO | * External interrupt mode with rising/falling edge trigger detection (interrupt 0x0080) * Pull Up Mode | Up push button |
| PA8 | GPIO | * External interrupt mode with rising/falling edge trigger detection (interrupt 0x0100) * Pull Up Mode | Left push button |
| PA9 | GPIO | * External interrupt mode with rising/falling edge trigger detection (interrupt 0x0200) * Pull Up Mode | Down push button |
| PB11 | GPIO | * Low output level * Output push pull * No Pull up no pull down * Low maximum output speed | LD3 (Menu display state LED) |
| PB12 | GPIO | * Low output level * Output push pull * No Pull up no pull down * Low maximum output speed | LD2 (Measurement & Output display state LED) |
| PB13 | GPIO | * Low output level * Output push pull * No Pull up no pull down * Low maximum output speed | LD4 (Current measurement (DI, or AI) mode LED) |
| PB14 | GPIO | * Low output level * Output push pull * No Pull up no pull down * Low maximum output speed | LD5 (output signal is active LED) |

|  |  |  |  |
| --- | --- | --- | --- |
| PC6 | GPIO | * Low output level * Output push pull * No Pull up no pull down * Low maximum output speed | LCD Data pin 4 |
| PC7 | GPIO | * Low output level * Output push pull * No Pull up no pull down * Low maximum output speed | LCD Data pin 5 |
| PC8 | GPIO | * Low output level * Output push pull * No Pull up no pull down * Low maximum output speed | LCD Data pin 6 |
| PC9 | GPIO | * Low output level * Output push pull * No Pull up no pull down * Low maximum output speed | LCD Data pin 7 |
| PC10 | GPIO | * Low output level * Output push pull * No Pull up no pull down * Low maximum output speed | LCD Register Select pin |
| PC11 | GPIO | * Low output level * Output push pull * No Pull up no pull down * Low maximum output speed | LCD Read/Write Pin |
| PC12 | GPIO | * Low output level * Output push pull * No Pull up no pull down * Low maximum output speed | LCD Enable Pin |
| PC13 | GPIO | * External interrupt mode with rising/falling edge trigger detection (interrupt 0x2000) * No Pull up no pull down | B1 (microcontroller onboard user push button) |

|  |  |  |  |
| --- | --- | --- | --- |
| PB1 | ADC | * IN1 Single Ended * Independent Mode * Clock prescaler: ADC Asynchronous clock mode * 12-bit resolution * Right alignment * Scan conversion disabled * Continuous conversion disabled * Discontinuous conversion disabled * End of conversion selection: End of single conversion * DMA continuous requests disabled * Overrun data overwritten * Low power auto wait disabled * Regular conversions enabled * Number of conversions: 1 * External Trigger Conversion Source: Regular Conversion launched by software * No external trigger conversion edge * Rank: 1   + Channel 1   + Sampling time: 1.5 cycles   + No Offset   + Offset: 0 * Injected conversions enabled * Number of conversions: 0 | Operational Amplifier VINA+ Pin |
| PA4 | DAC | * OUT1 Configuration * Output Buffer Enabled * Timer 4 trigger out event * Wave generation mode disabled | Operational Amplifier VINA+ Pin |
|  | DMA | * DAC1\_CH1 * DMA1 Channel 3 * memory to peripheral * low priority |  |
|  | GPIO NVIC | * EXTI line[9:5] interrupts enabled |  |
|  | UART | * Mode: Asynchronous * Hardware flow control disabled * Baud rate: 115200Bits/s * Word length:8 bit * No parity * Stop Bits: 1 * Data Direction: Send and Receive * Over sampling: 16 samples * Single sample disabled * Auto baud rate disabled * TX and RX pin active level inversion disabled * Data inversion disabled * TX and RX Pins swapping disabled * Overrun Enabled * DMA on RX error Enabled * MSB First Disabled |  |
|  | TIM2 | Internal Clock source  Prescaler: 300  Counter Mode: up  Counter Period (ARR): 72000  Internal clock division: none  Auto-reload preload enabled  Master/slave mode disabled  Trigger event selection: Reset | Update measurements |
|  | TIM3 | Internal Clock source  Prescaler: 10  Counter Mode: up  Counter Period (ARR): 72  Internal clock division: none  Auto-reload preload enabled  Master/slave mode disabled  Trigger event selection: Reset | Counter for under µs delay |
|  | TIM4 | Internal Clock source  Prescaler: 0  Counter Mode: up  Counter Period (ARR): 561  Internal clock division: none  Auto-reload preload disabled  Master/slave mode disabled  Trigger event selection: update event | DMA1 Channel 3 |
|  | TIM7 | Internal Clock source  Prescaler: 6000  Counter Mode: up  Counter Period (ARR): 7200  Auto-reload preload disabled  Trigger event selection: Reset | Update LCD scroller |

1. Figures and Tables

Table 4: Project user requirements as stated in the PDD [1]

Table

Description automatically generated with medium confidence

Diagram

Description automatically generated

Figure 6: High-Level hardware concept design

Table 5: Components made available to all students for the project

|  |  |
| --- | --- |
| **Component** | **Description** |
| STM32F303RE | ARM Cortex M4- 32-bit RISC core microcontroller [3] |
| L7805CV | 5V Regulator |
| LM2950 | 3.3V Regulator |
| MCP602 | 2channel operational amplifier |
| LCD-1602A CA | Generic 16-character 2-line LCD Screen |
| INA219 | Current Sensor |
| Push Button | 4 pin push buttons |
| Green 3mm LED | Generic green 3mm LED |
| Standard gold band Resistors | Standard value 5% tolerance resistors |
| Standard ceramic capacitors | Standard value ceramic capacitors |
| Standard electrolytic capacitors | Standard value electrolytic capacitors |
| 1N4007 | 1A 1000V standard diode |

A screenshot of a computer screen

Description automatically generated with medium confidence

Figure 7: Spice simulation: Output Voltage of L7805CV against input voltage and current through power supply on/off LED (LD1)

Diagram, schematic

Description automatically generated

Figure 8: Suggested circuit for L7805CV [1]

A screenshot of a computer

Description automatically generated with medium confidence

Figure 9: 5V regulator current spike at 1.8Ω load

Chart

Description automatically generated

Figure 10: Suggested circuit for LM2950

A picture containing text, clock, indoor, device

Description automatically generated

Figure 11: current output to student board

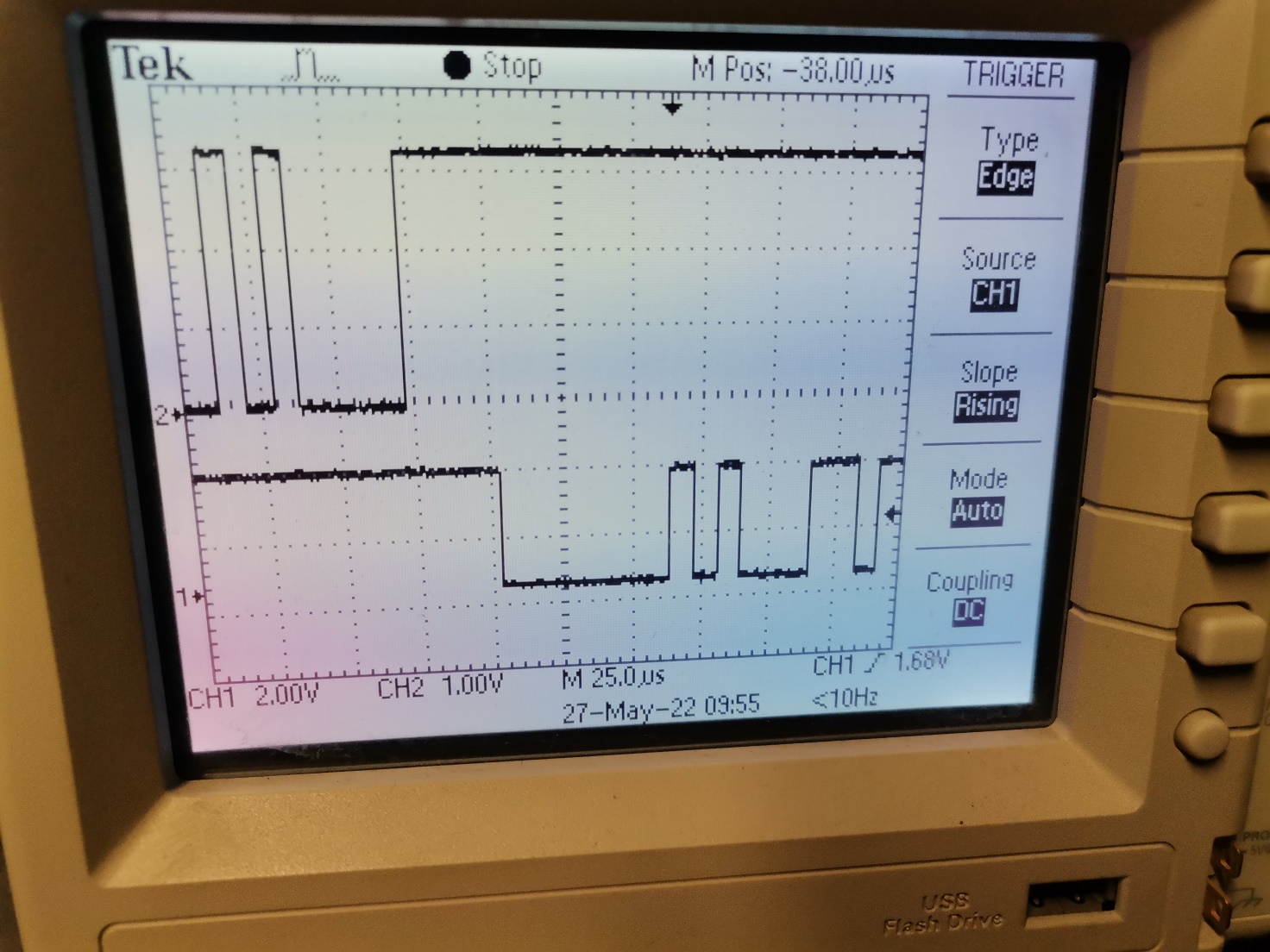


Figure 12: Time taken between UART message received and beginning of transmitting required output message

A screen shot of a computer

Description automatically generated with low confidenceGraphical user interface

Description automatically generated with low confidence

Figure 13: Difference between 0x01 and 0x02

Text, calendar

Description automatically generated

Figure 14: The oscilloscope output of 0x99 or 1001 1001 in binary.

A screen shot of a computer

Description automatically generated with low confidence

Figure 15:Button Debouncing observed from oscilloscope

A picture containing text, electronics, display

Description automatically generated

Figure 16: Rise time of LD2

A picture containing text, electronics, display

Description automatically generated

Figure 17: Time taken for system to respond after button debouncing in software

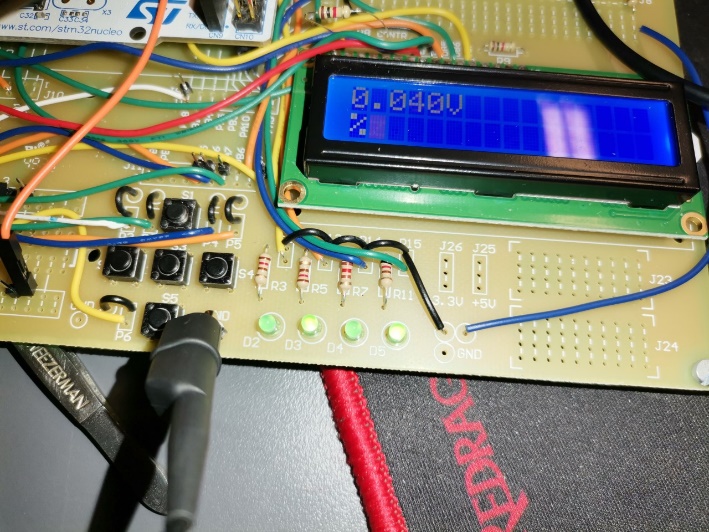
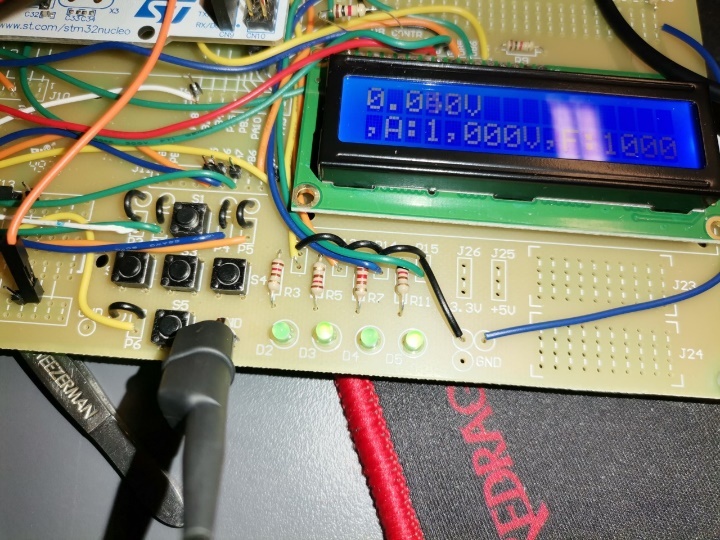


Figure 18: LED Brightness can easily be viewed while LCD is displaying DC voltage measured and a duty cycle measurement while scrolling.

A picture containing text, indoor, microwave

Description automatically generatedA picture containing text, indoor, wall, computer

Description automatically generatedA picture containing text, indoor, oven, microwave

Description automatically generated

Figure 19: The voltage output of the DAC after filter and scaling when in DC, Pulse and sinusoidal modes.

A picture containing graphical user interface

Description automatically generatedA picture containing graphical user interface

Description automatically generated

Figure 20: The rise and fall time of the microcontroller’s GPIO output pin.

Graphical user interface

Description automatically generated with medium confidence

Figure 21: Spice simulation of LED equivalent circuit showing current through each LED, total current draw and LED voltage drop

Table 6:Menu Logic

|  |  |  |
| --- | --- | --- |
| **LCD screen display** | **Menu layer** | **Menu Number** |
| Start Message | 0 | 0 |
| Measure Mode | 1 | 0 |
| Sig Gen Mode | 1 | 1 |
| Measure ModeDC-Voltage | 2 | 0 |
| Measure ModeDC-Current | 2 | 1 |
| Measure ModeAC-Voltage | 2 | 2 |
| Measure ModeAC-Current | 2 | 3 |
| Type | 2 | 4 |
| Parameter | 2 | 5 |
| Output | 2 | 6 |
| TypeDC | 3 | 0 |
| TypeSinusoidal | 3 | 1 |
| TypePulse | 3 | 2 |
| Amplitude | 3 | 3 |
| Offset | 3 | 4 |
| Frequency | 3 | 5 |
| DutyCycle | 3 | 6 |
| Output-On | 3 | 7 |
| Output-Off | 3 | 8 |
| Amplitude-1234 | 4 | 0 |
| Offset1234 | 4 | 1 |
| Frequency1234 | 4 | 2 |
| DutyCycle1234 | 4 | 3 |

1. Communications log of Demonstration 4

|  |
| --- |
| 0,000,Out,0x02, |
| 1,000,Out,0x01, |
| 1,086,In,0x40,@ |
| 1,086,In,0x2C,, |
| 1,086,In,0x31,1 |
| 1,087,In,0x39,9 |
| 1,087,In,0x38,8 |
| 1,088,In,0x36,6 |
| 1,088,In,0x32,2 |
| 1,088,In,0x32,2 |
| 1,088,In,0x34,4 |
| 1,088,In,0x35,5 |
| 1,088,In,0x2C,, |
| 1,088,In,0x21,! |
| 1,088,In,0x0A, |
|  |
| 3,001,Out,0x18, |
| 3,703,In,0x7F, |
| 3,703,In,0x80,? |
| 3,703,In,0xFF,? |
| 3,703,In,0x00, |
| 3,703,In,0x31,1 |
| 3,703,In,0x04, |
| 3,703,In,0x49,I |
| 3,703,In,0x04, |
| 3,703,In,0x00, |
| 3,703,In,0x00, |
| 4,003,Out,0x1A, |
| 4,004,In,0x7F, |
| 4,004,In,0x80,? |
| 4,004,In,0xFF,? |
| 4,004,In,0x03, |
| 4,004,In,0xB2,? |
| 4,005,In,0x09, |
| 4,005,In,0x00, |
| 4,005,In,0x00, |
| 4,005,In,0x00, |
| 4,005,In,0x00, |
| 6,003,Out,0x19, |
| 6,005,In,0x40,@ |
| 6,005,In,0x2C,, |
| 6,005,In,0x44,D |
| 6,005,In,0x56,V |
| 6,005,In,0x2C,, |
| 6,006,In,0x64,d |
| 6,006,In,0x2C,, |
| 6,006,In,0x00, |
| 6,006,In,0x2C,, |
| 6,006,In,0x21,! |
| 6,006,In,0x0A, |
|  |
| 6,504,Out,0x1F,­ |
| 6,504,In,0x7F, |
| 6,504,In,0x80,? |
| 6,504,In,0xFF,? |
| 6,504,In,0x03, |
| 6,504,In,0xB2,? |
| 6,504,In,0x09, |
| 6,504,In,0x00, |
| 6,504,In,0x00, |
| 6,505,In,0x00, |
| 6,505,In,0x00, |
| 6,505,In,0x40,@ |
| 6,505,In,0x2C,, |
| 6,505,In,0x6D,m |
| 6,505,In,0x2C,, |
| 6,505,In,0x6F,o |
| 6,505,In,0x2C,, |
| 6,506,In,0x30,0 |
| 6,506,In,0x30,0 |
| 6,506,In,0x30,0 |
| 6,506,In,0x30,0 |
| 6,506,In,0x2C,, |
| 6,506,In,0x21,! |
| 6,506,In,0x0A, |
|  |
| 7,005,Out,0x1B, |
| 7,007,In,0x7F, |
| 7,007,In,0x80,? |
| 7,007,In,0xFF,? |
| 7,007,In,0x03, |
| 7,007,In,0xE1,? |
| 7,007,In,0x03, |
| 7,007,In,0xCA,? |
| 7,007,In,0x06, |
| 7,007,In,0xC4,? |
| 7,007,In,0x09, |
| 9,006,Out,0x19, |
| 9,008,In,0x40,@ |
| 9,008,In,0x2C,, |
| 9,008,In,0x44,D |
| 9,009,In,0x49,I |
| 9,009,In,0x2C,, |
| 9,009,In,0x64,d |
| 9,009,In,0x2C,, |
| 9,009,In,0x00, |
| 9,009,In,0x2C,, |
| 9,009,In,0x21,! |
| 9,009,In,0x0A, |
|  |
| 9,506,Out,0x1F,­ |
| 9,506,In,0x7F, |
| 9,506,In,0x80,? |
| 9,506,In,0xFF,? |
| 9,506,In,0x03, |
| 9,506,In,0xE1,? |
| 9,506,In,0x03, |
| 9,507,In,0xCA,? |
| 9,507,In,0x06, |
| 9,508,In,0xC4,? |
| 9,508,In,0x09, |
| 9,508,In,0x40,@ |
| 9,508,In,0x2C,, |
| 9,508,In,0x6D,m |
| 9,508,In,0x2C,, |
| 9,509,In,0x6F,o |
| 9,509,In,0x2C,, |
| 9,509,In,0x30,0 |
| 9,509,In,0x30,0 |
| 9,509,In,0x30,0 |
| 9,509,In,0x30,0 |
| 9,509,In,0x2C,, |
| 9,509,In,0x21,! |
| 9,509,In,0x0A, |
|  |
| 10,007,Out,0x20, |
| 10,007,In,0x7F, |
| 10,007,In,0x80,? |
| 10,007,In,0xFF,? |
| 10,007,In,0x03, |
| 10,007,In,0xE1,? |
| 10,007,In,0x03, |
| 10,007,In,0xCA,? |
| 10,007,In,0x06, |
| 10,007,In,0xC4,? |
| 10,007,In,0x09, |
| 10,008,In,0x40,@ |
| 10,008,In,0x2C,, |
| 10,008,In,0x6D,m |
| 10,008,In,0x2C,, |
| 10,009,In,0x61,a |
| 10,009,In,0x2C,, |
| 10,009,In,0x30,0 |
| 10,009,In,0x30,0 |
| 10,009,In,0x30,0 |
| 10,009,In,0x30,0 |
| 10,009,In,0x2C,, |
| 10,009,In,0x21,! |
| 10,009,In,0x0A, |
|  |
| 10,508,Out,0x21,! |
| 10,508,In,0x7F, |
| 10,508,In,0x80,? |
| 10,508,In,0xFF,? |
| 10,508,In,0x03, |
| 10,508,In,0xE1,? |
| 10,508,In,0x03, |
| 10,509,In,0xCA,? |
| 10,509,In,0x06, |
| 10,509,In,0xC4,? |
| 10,509,In,0x09, |
| 10,509,In,0x40,@ |
| 10,509,In,0x2C,, |
| 10,509,In,0x6D,m |
| 10,509,In,0x2C,, |
| 10,509,In,0x66,f |
| 10,509,In,0x2C,, |
| 10,509,In,0x30,0 |
| 10,510,In,0x30,0 |
| 10,510,In,0x30,0 |
| 10,510,In,0x30,0 |
| 10,510,In,0x2C,, |
| 10,510,In,0x21,! |
| 10,510,In,0x0A, |
|  |
| 11,009,Out,0x02, |
| 12,009,Out,0x01, |
| 12,093,In,0x40,@ |
| 12,094,In,0x2C,, |
| 12,094,In,0x31,1 |
| 12,094,In,0x39,9 |
| 12,094,In,0x38,8 |
| 12,094,In,0x36,6 |
| 12,094,In,0x32,2 |
| 12,095,In,0x32,2 |
| 12,095,In,0x34,4 |
| 12,095,In,0x35,5 |
| 12,095,In,0x2C,, |
| 12,095,In,0x21,! |
| 12,095,In,0x0A, |
|  |
| 14,010,Out,0x24,$ |
| 14,010,In,0x7F, |
| 14,011,In,0x80,? |
| 14,011,In,0xFF,? |
| 14,011,In,0x03, |
| 14,011,In,0xD1,? |
| 14,011,In,0x05, |
| 14,011,In,0x00, |
| 14,011,In,0x00, |
| 14,011,In,0x00, |
| 14,011,In,0x00, |
| 14,012,In,0x40,@ |
| 14,012,In,0x2C,, |
| 14,013,In,0x44,D |
| 14,013,In,0x56,V |
| 14,013,In,0x2C,, |
| 14,013,In,0x64,d |
| 14,013,In,0x2C,, |
| 14,013,In,0x00, |
| 14,013,In,0x2C,, |
| 14,013,In,0x21,! |
| 14,013,In,0x0A, |
|  |
| 16,516,In,0x40,@ |
| 16,516,In,0x2C,, |
| 16,516,In,0x44,D |
| 16,516,In,0x56,V |
| 16,516,In,0x2C,, |
| 16,516,In,0x64,d |
| 16,516,In,0x2C,, |
| 16,516,In,0x01, |
| 16,516,In,0x2C,, |
| 16,516,In,0x21,! |
| 16,516,In,0x0A, |
|  |
| 17,343,In,0x7F, |
| 17,343,In,0x80,? |
| 17,343,In,0xFF,? |
| 17,343,In,0x04, |
| 17,343,In,0x00, |
| 17,343,In,0x80,? |
| 17,343,In,0x00, |
| 17,343,In,0x00, |
| 17,343,In,0x00, |
| 17,343,In,0x01, |
| 18,847,In,0x7F, |
| 18,847,In,0x80,? |
| 18,847,In,0xFF,? |
| 18,847,In,0x04, |
| 18,847,In,0x00, |
| 18,847,In,0x80,? |
| 18,847,In,0x00, |
| 18,847,In,0x00, |
| 18,847,In,0x00, |
| 18,847,In,0x02, |
| 19,515,Out,0x19, |
| 19,517,In,0x40,@ |
| 19,517,In,0x2C,, |
| 19,517,In,0x44,D |
| 19,517,In,0x56,V |
| 19,517,In,0x2C,, |
| 19,517,In,0x64,d |
| 19,517,In,0x2C,, |
| 19,518,In,0x01, |
| 19,518,In,0x2C,, |
| 19,518,In,0x21,! |
| 19,518,In,0x0A, |
|  |
| 20,017,Out,0x24,$ |
| 20,017,In,0x7F, |
| 20,017,In,0x80,? |
| 20,017,In,0xFF,? |
| 20,018,In,0x03, |
| 20,018,In,0xD1,? |
| 20,018,In,0x05, |
| 20,018,In,0xD9,? |
| 20,018,In,0x04, |
| 20,018,In,0xE8,? |
| 20,018,In,0x03, |
| 20,019,In,0x40,@ |
| 20,019,In,0x2C,, |
| 20,019,In,0x44,D |
| 20,019,In,0x56,V |
| 20,020,In,0x2C,, |
| 20,020,In,0x64,d |
| 20,020,In,0x2C,, |
| 20,020,In,0x00, |
| 20,021,In,0x2C,, |
| 20,021,In,0x21,! |
| 20,021,In,0x0A, |
|  |
| 22,523,In,0x40,@ |
| 22,523,In,0x2C,, |
| 22,523,In,0x44,D |
| 22,523,In,0x56,V |
| 22,523,In,0x2C,, |
| 22,523,In,0x73,s |
| 22,523,In,0x2C,, |
| 22,523,In,0x01, |
| 22,523,In,0x2C,, |
| 22,523,In,0x21,! |
| 22,524,In,0x0A, |
|  |
| 23,350,In,0x7F, |
| 23,350,In,0x80,? |
| 23,350,In,0xFF,? |
| 23,350,In,0x04, |
| 23,350,In,0x00, |
| 23,350,In,0x80,? |
| 23,350,In,0x00, |
| 23,350,In,0x00, |
| 23,350,In,0x00, |
| 23,351,In,0x01, |
| 24,854,In,0x7F, |
| 24,854,In,0x80,? |
| 24,854,In,0xFF,? |
| 24,854,In,0x04, |
| 24,854,In,0x00, |
| 24,854,In,0x80,? |
| 24,854,In,0x00, |
| 24,854,In,0x00, |
| 24,854,In,0x00, |
| 24,854,In,0x02, |
| 25,523,Out,0x19, |
| 25,525,In,0x40,@ |
| 25,525,In,0x2C,, |
| 25,525,In,0x44,D |
| 25,525,In,0x56,V |
| 25,525,In,0x2C,, |
| 25,525,In,0x73,s |
| 25,526,In,0x2C,, |
| 25,526,In,0x01, |
| 25,526,In,0x2C,, |
| 25,526,In,0x21,! |
| 25,526,In,0x0A, |
|  |
| 26,026,Out,0x24,$ |
| 26,027,In,0x7F, |
| 26,027,In,0x80,? |
| 26,027,In,0xFF,? |
| 26,027,In,0x03, |
| 26,027,In,0xD1,? |
| 26,027,In,0x05, |
| 26,027,In,0xD9,? |
| 26,027,In,0x04, |
| 26,027,In,0xE8,? |
| 26,027,In,0x03, |
| 26,027,In,0x7F, |
| 26,027,In,0x80,? |
| 26,027,In,0xFF,? |
| 26,027,In,0x05, |
| 26,027,In,0x19, |
| 26,027,In,0x00, |
| 26,027,In,0x00, |
| 26,027,In,0x00, |
| 26,027,In,0x00, |
| 26,027,In,0x00, |
| 26,029,In,0x40,@ |
| 26,029,In,0x2C,, |
| 26,029,In,0x44,D |
| 26,029,In,0x56,V |
| 26,029,In,0x2C,, |
| 26,029,In,0x73,s |
| 26,029,In,0x2C,, |
| 26,029,In,0x00, |
| 26,030,In,0x2C,, |
| 26,030,In,0x21,! |
| 26,030,In,0x0A, |
|  |
| 28,532,In,0x40,@ |
| 28,532,In,0x2C,, |
| 28,532,In,0x44,D |
| 28,532,In,0x56,V |
| 28,533,In,0x2C,, |
| 28,533,In,0x70,p |
| 28,533,In,0x2C,, |
| 28,533,In,0x01, |
| 28,533,In,0x2C,, |
| 28,533,In,0x21,! |
| 28,533,In,0x0A, |
|  |
| 29,359,In,0x7F, |
| 29,359,In,0x80,? |
| 29,359,In,0xFF,? |
| 29,359,In,0x04, |
| 29,359,In,0x00, |
| 29,359,In,0x80,? |
| 29,359,In,0x00, |
| 29,359,In,0x00, |
| 29,359,In,0x00, |
| 29,360,In,0x01, |
| 30,873,In,0x7F, |
| 30,873,In,0x80,? |
| 30,873,In,0xFF,? |
| 30,873,In,0x04, |
| 30,873,In,0x00, |
| 30,873,In,0x80,? |
| 30,873,In,0x00, |
| 30,874,In,0x00, |
| 30,874,In,0x00, |
| 30,874,In,0x02, |
| 31,531,Out,0x19, |
| 31,533,In,0x40,@ |
| 31,534,In,0x2C,, |
| 31,534,In,0x44,D |
| 31,534,In,0x56,V |
| 31,534,In,0x2C,, |
| 31,534,In,0x70,p |
| 31,534,In,0x2C,, |
| 31,534,In,0x01, |
| 31,534,In,0x2C,, |
| 31,534,In,0x21,! |
| 31,534,In,0x0A, |
|  |
| 32,035,Out,0x24,$ |
| 32,537,In,0x7F, |
| 32,537,In,0x80,? |
| 32,537,In,0xFF,? |
| 32,538,In,0x03, |
| 32,538,In,0x7C,| |
| 32,538,In,0x00, |
| 32,538,In,0x00, |
| 32,538,In,0x00, |
| 32,538,In,0x00, |
| 32,538,In,0x00, |
| 32,539,In,0x40,@ |
| 32,539,In,0x2C,, |
| 32,539,In,0x44,D |
| 32,539,In,0x56,V |
| 32,539,In,0x2C,, |
| 32,539,In,0x70,p |
| 32,540,In,0x2C,, |
| 32,540,In,0x00, |
| 32,540,In,0x2C,, |
| 32,540,In,0x21,! |
| 32,540,In,0x0A, |
|  |
| 35,040,In,0x40,@ |
| 35,040,In,0x2C,, |
| 35,040,In,0x44,D |
| 35,041,In,0x56,V |
| 35,041,In,0x2C,, |
| 35,041,In,0x64,d |
| 35,041,In,0x2C,, |
| 35,041,In,0x01, |
| 35,041,In,0x2C,, |
| 35,042,In,0x21,! |
| 35,042,In,0x0A, |
|  |
| 35,867,In,0x7F, |
| 35,869,In,0x80,? |
| 35,869,In,0xFF,? |
| 35,869,In,0x04, |
| 35,869,In,0x00, |
| 35,869,In,0x80,? |
| 35,869,In,0x00, |
| 35,869,In,0x00, |
| 35,869,In,0x00, |
| 35,869,In,0x01, |
| 37,375,In,0x7F, |
| 37,375,In,0x80,? |
| 37,375,In,0xFF,? |
| 37,375,In,0x04, |
| 37,375,In,0x00, |
| 37,375,In,0x80,? |
| 37,375,In,0x00, |
| 37,375,In,0x00, |
| 37,375,In,0x00, |
| 37,375,In,0x02, |
| 38,040,Out,0x19, |
| 38,042,In,0x40,@ |
| 38,042,In,0x2C,, |
| 38,042,In,0x44,D |
| 38,042,In,0x56,V |
| 38,042,In,0x2C,, |
| 38,042,In,0x64,d |
| 38,042,In,0x2C,, |
| 38,042,In,0x01, |
| 38,042,In,0x2C,, |
| 38,042,In,0x21,! |
| 38,043,In,0x0A, |
|  |
| 38,543,Out,0x24,$ |
| 39,046,In,0x7F, |
| 39,046,In,0x80,? |
| 39,046,In,0xFF,? |
| 39,046,In,0x03, |
| 39,046,In,0x83,? |
| 39,046,In,0x0F, |
| 39,046,In,0x00, |
| 39,046,In,0x00, |
| 39,046,In,0x00, |
| 39,046,In,0x00, |
| 39,048,In,0x40,@ |
| 39,048,In,0x2C,, |
| 39,048,In,0x44,D |
| 39,048,In,0x56,V |
| 39,048,In,0x2C,, |
| 39,048,In,0x64,d |
| 39,048,In,0x2C,, |
| 39,048,In,0x00, |
| 39,048,In,0x2C,, |
| 39,048,In,0x21,! |
| 39,048,In,0x0A, |
|  |
| 41,550,In,0x40,@ |
| 41,550,In,0x2C,, |
| 41,550,In,0x44,D |
| 41,550,In,0x56,V |
| 41,550,In,0x2C,, |
| 41,550,In,0x64,d |
| 41,550,In,0x2C,, |
| 41,550,In,0x01, |
| 41,550,In,0x2C,, |
| 41,550,In,0x21,! |
| 41,550,In,0x0A, |
|  |
| 42,375,In,0x7F, |
| 42,376,In,0x80,? |
| 42,376,In,0xFF,? |
| 42,376,In,0x04, |
| 42,376,In,0x00, |
| 42,376,In,0x80,? |
| 42,376,In,0x00, |
| 42,377,In,0x00, |
| 42,377,In,0x00, |
| 42,377,In,0x01, |
| 43,878,In,0x7F, |
| 43,878,In,0x80,? |
| 43,879,In,0xFF,? |
| 43,879,In,0x04, |
| 43,879,In,0x00, |
| 43,879,In,0x80,? |
| 43,879,In,0x00, |
| 43,879,In,0x00, |
| 43,879,In,0x00, |
| 43,879,In,0x02, |
| 44,550,Out,0x19, |
| 44,552,In,0x40,@ |
| 44,552,In,0x2C,, |
| 44,552,In,0x44,D |
| 44,552,In,0x56,V |
| 44,553,In,0x2C,, |
| 44,553,In,0x64,d |
| 44,553,In,0x2C,, |
| 44,553,In,0x01, |
| 44,553,In,0x2C,, |
| 44,553,In,0x21,! |
| 44,553,In,0x0A, |
|  |
| 45,052,Out,0x24,$ |
| 46,559,In,0x7F, |
| 46,559,In,0x80,? |
| 46,559,In,0xFF,? |
| 46,559,In,0x03, |
| 46,559,In,0x45,E |
| 46,559,In,0x07, |
| 46,559,In,0xA3,? |
| 46,559,In,0x03, |
| 46,559,In,0x94,? |
| 46,559,In,0x11, |
| 46,561,In,0x40,@ |
| 46,561,In,0x2C,, |
| 46,561,In,0x44,D |
| 46,561,In,0x56,V |
| 46,561,In,0x2C,, |
| 46,561,In,0x64,d |
| 46,561,In,0x2C,, |
| 46,561,In,0x00, |
| 46,561,In,0x2C,, |
| 46,561,In,0x21,! |
| 46,561,In,0x0A, |
|  |
| 49,064,In,0x40,@ |
| 49,064,In,0x2C,, |
| 49,064,In,0x44,D |
| 49,064,In,0x56,V |
| 49,064,In,0x2C,, |
| 49,064,In,0x73,s |
| 49,064,In,0x2C,, |
| 49,064,In,0x01, |
| 49,064,In,0x2C,, |
| 49,064,In,0x21,! |
| 49,064,In,0x0A, |
|  |
| 49,890,In,0x7F, |
| 49,891,In,0x80,? |
| 49,891,In,0xFF,? |
| 49,891,In,0x04, |
| 49,891,In,0x00, |
| 49,891,In,0x80,? |
| 49,891,In,0x00, |
| 49,891,In,0x00, |
| 49,891,In,0x00, |
| 49,891,In,0x01, |
| 51,400,In,0x7F, |
| 51,400,In,0x80,? |
| 51,401,In,0xFF,? |
| 51,401,In,0x04, |
| 51,401,In,0x00, |
| 51,401,In,0x80,? |
| 51,401,In,0x00, |
| 51,401,In,0x00, |
| 51,401,In,0x00, |
| 51,401,In,0x02, |
| 52,060,Out,0x19, |
| 52,062,In,0x40,@ |
| 52,062,In,0x2C,, |
| 52,062,In,0x44,D |
| 52,062,In,0x56,V |
| 52,062,In,0x2C,, |
| 52,063,In,0x73,s |
| 52,063,In,0x2C,, |
| 52,063,In,0x01, |
| 52,063,In,0x2C,, |
| 52,063,In,0x21,! |
| 52,063,In,0x0A, |
|  |
| 52,562,Out,0x24,$ |
| 54,068,In,0x7F, |
| 54,068,In,0x80,? |
| 54,068,In,0xFF,? |
| 54,068,In,0x03, |
| 54,068,In,0x45,E |
| 54,068,In,0x07, |
| 54,068,In,0xA3,? |
| 54,069,In,0x03, |
| 54,069,In,0x19, |
| 54,069,In,0x00, |
| 54,070,In,0x40,@ |
| 54,070,In,0x2C,, |
| 54,070,In,0x44,D |
| 54,070,In,0x56,V |
| 54,070,In,0x2C,, |
| 54,070,In,0x73,s |
| 54,070,In,0x2C,, |
| 54,070,In,0x00, |
| 54,070,In,0x2C,, |
| 54,071,In,0x21,! |
| 54,071,In,0x0A, |
|  |
| 56,573,In,0x40,@ |
| 56,573,In,0x2C,, |
| 56,573,In,0x44,D |
| 56,573,In,0x56,V |
| 56,573,In,0x2C,, |
| 56,573,In,0x73,s |
| 56,573,In,0x2C,, |
| 56,573,In,0x01, |
| 56,573,In,0x2C,, |
| 56,573,In,0x21,! |
| 56,573,In,0x0A, |
|  |
| 57,399,In,0x7F, |
| 57,400,In,0x80,? |
| 57,400,In,0xFF,? |
| 57,400,In,0x04, |
| 57,400,In,0x00, |
| 57,400,In,0x80,? |
| 57,401,In,0x00, |
| 57,401,In,0x00, |
| 57,401,In,0x00, |
| 57,401,In,0x01, |
| 58,920,In,0x7F, |
| 58,920,In,0x80,? |
| 58,920,In,0xFF,? |
| 58,920,In,0x04, |
| 58,920,In,0x00, |
| 58,920,In,0x80,? |
| 58,920,In,0x00, |
| 58,920,In,0x00, |
| 58,920,In,0x00, |
| 58,920,In,0x02, |
| 59,570,Out,0x19, |
| 59,572,In,0x40,@ |
| 59,572,In,0x2C,, |
| 59,572,In,0x44,D |
| 59,572,In,0x56,V |
| 59,573,In,0x2C,, |
| 59,573,In,0x73,s |
| 59,573,In,0x2C,, |
| 59,573,In,0x01, |
| 59,573,In,0x2C,, |
| 59,573,In,0x21,! |
| 59,573,In,0x0A, |
|  |
| 60,073,Out,0x02, |
| 61,074,Out,0x01, |
| 61,160,In,0x40,@ |
| 61,160,In,0x2C,, |
| 61,160,In,0x31,1 |
| 61,160,In,0x39,9 |
| 61,160,In,0x38,8 |
| 61,160,In,0x36,6 |
| 61,161,In,0x32,2 |
| 61,161,In,0x32,2 |
| 61,161,In,0x34,4 |
| 61,161,In,0x35,5 |
| 61,161,In,0x2C,, |
| 61,161,In,0x21,! |
| 61,161,In,0x0A, |
|  |
| 64,175,Out,0x07, |
| 64,376,Out,0x08, |
| 66,478,Out,0x1C, |
| 66,479,In,0x7F, |
| 66,479,In,0x80,? |
| 66,479,In,0xFF,? |
| 66,479,In,0x03, |
| 66,479,In,0xB2,? |
| 66,479,In,0x09, |
| 66,480,In,0x00, |
| 66,480,In,0x00, |
| 66,480,In,0x00, |
| 66,480,In,0x00, |
| 67,580,Out,0x24,$ |
| 67,580,In,0x7F, |
| 67,581,In,0x80,? |
| 67,581,In,0xFF,? |
| 67,581,In,0x03, |
| 67,581,In,0xD1,? |
| 67,581,In,0x05, |
| 67,581,In,0x00, |
| 67,581,In,0x00, |
| 67,581,In,0x00, |
| 67,581,In,0x00, |
| 67,582,In,0x40,@ |
| 67,583,In,0x2C,, |
| 67,583,In,0x41,A |
| 67,583,In,0x56,V |
| 67,583,In,0x2C,, |
| 67,583,In,0x64,d |
| 67,583,In,0x2C,, |
| 67,583,In,0x00, |
| 67,583,In,0x2C,, |
| 67,583,In,0x21,! |
| 67,583,In,0x0A, |
|  |
| 70,083,In,0x40,@ |
| 70,084,In,0x2C,, |
| 70,084,In,0x41,A |
| 70,084,In,0x56,V |
| 70,084,In,0x2C,, |
| 70,084,In,0x64,d |
| 70,084,In,0x2C,, |
| 70,084,In,0x01, |
| 70,084,In,0x2C,, |
| 70,084,In,0x21,! |
| 70,084,In,0x0A, |
|  |
| 70,581,Out,0x02, |
| 71,581,Out,0x01, |
| 71,667,In,0x40,@ |
| 71,667,In,0x2C,, |
| 71,667,In,0x31,1 |
| 71,667,In,0x39,9 |
| 71,667,In,0x38,8 |
| 71,667,In,0x36,6 |
| 71,667,In,0x32,2 |
| 71,667,In,0x32,2 |
| 71,668,In,0x34,4 |
| 71,668,In,0x35,5 |
| 71,668,In,0x2C,, |
| 71,668,In,0x21,! |
| 71,668,In,0x0A, |
|  |
| 73,582,Out,0x0B, |
| 73,783,Out,0x0C, |
| 74,284,Out,0x0B, |
| 74,485,Out,0x0C, |
| 74,985,Out,0x09, |
| 75,186,Out,0x0A, |
|  |
| 75,687,Out,0x09, |
| 75,888,Out,0x0A, |
|  |
| 77,889,Out,0x07, |
| 78,090,Out,0x08, |
| 80,090,Out,0x19, |
| 80,092,In,0x40,@ |
| 80,092,In,0x2C,, |
| 80,092,In,0x44,D |
| 80,092,In,0x49,I |
| 80,092,In,0x2C,, |
| 80,093,In,0x64,d |
| 80,093,In,0x2C,, |
| 80,093,In,0x00, |
| 80,093,In,0x2C,, |
| 80,093,In,0x21,! |
| 80,093,In,0x0A, |
|  |
| 80,591,Out,0x02, |
| 81,591,Out,0x01, |
| 81,676,In,0x40,@ |
| 81,676,In,0x2C,, |
| 81,676,In,0x31,1 |
| 81,676,In,0x39,9 |
| 81,676,In,0x38,8 |
| 81,677,In,0x36,6 |
| 81,677,In,0x32,2 |
| 81,677,In,0x32,2 |
| 81,677,In,0x34,4 |
| 81,677,In,0x35,5 |
| 81,677,In,0x2C,, |
| 81,677,In,0x21,! |
| 81,677,In,0x0A, |
|  |
| 83,592,Out,0x0B, |
| 83,793,Out,0x0C, |
| 84,294,Out,0x09, |
| 84,495,Out,0x0A, |
|  |
| 84,996,Out,0x0B, |
| 85,196,Out,0x0C, |
| 85,697,Out,0x09, |
| 85,898,Out,0x0A, |
|  |
| 86,399,Out,0x0B, |
| 86,600,Out,0x0C, |
| 87,100,Out,0x09, |
| 87,301,Out,0x0A, |
|  |
| 87,802,Out,0x0B, |
| 88,003,Out,0x0C, |
| 88,503,Out,0x09, |
| 88,704,Out,0x0A, |
|  |
| 89,205,Out,0x09, |
| 89,406,Out,0x0A, |
|  |
| 89,907,Out,0x09, |
| 90,108,Out,0x0A, |
|  |
| 92,108,Out,0x07, |
| 92,309,Out,0x08, |
| 92,810,Out,0x03, |
| 93,011,Out,0x04, |
| 93,512,Out,0x03, |
| 93,712,Out,0x04, |
| 94,213,Out,0x09, |
| 94,414,Out,0x0A, |
|  |
| 94,915,Out,0x09, |
| 95,116,Out,0x0A, |
|  |
| 95,616,Out,0x0B, |
| 95,817,Out,0x0C, |
| 96,318,Out,0x07, |
| 96,519,Out,0x08, |
| 97,020,Out,0x25,% |
| 97,348,In,0x7F, |
| 97,348,In,0x80,? |
| 97,348,In,0xFF,? |
| 97,348,In,0x04, |
| 97,348,In,0x00, |
| 97,348,In,0x80,? |
| 97,348,In,0x00, |
| 97,348,In,0x00, |
| 97,348,In,0x00, |
| 97,348,In,0x01, |
| 98,837,In,0x7F, |
| 98,837,In,0x80,? |
| 98,837,In,0xFF,? |
| 98,837,In,0x04, |
| 98,837,In,0x00, |
| 98,837,In,0x80,? |
| 98,837,In,0x00, |
| 98,837,In,0x00, |
| 98,837,In,0x00, |
| 98,837,In,0x02, |
| 99,521,Out,0x19, |
| 99,523,In,0x40,@ |
| 99,523,In,0x2C,, |
| 99,523,In,0x44,D |
| 99,523,In,0x56,V |
| 99,523,In,0x2C,, |
| 99,524,In,0x64,d |
| 99,524,In,0x2C,, |
| 99,524,In,0x00, |
| 99,524,In,0x2C,, |
| 99,524,In,0x21,! |
| 99,524,In,0x0A, |
|  |
| 100,525,Out,0x02, |
| 101,526,Out,0x01, |
| 101,611,In,0x40,@ |
| 101,612,In,0x2C,, |
| 101,612,In,0x31,1 |
| 101,612,In,0x39,9 |
| 101,612,In,0x38,8 |
| 101,613,In,0x36,6 |
| 101,613,In,0x32,2 |
| 101,614,In,0x32,2 |
| 101,614,In,0x34,4 |
| 101,614,In,0x35,5 |
| 101,614,In,0x2C,, |
| 101,614,In,0x21,! |
| 101,614,In,0x0A, |
|  |
| 103,527,Out,0x1C, |
| 103,528,In,0x7F, |
| 103,528,In,0x80,? |
| 103,528,In,0xFF,? |
| 103,528,In,0x03, |
| 103,528,In,0xB2,? |
| 103,528,In,0x09, |
| 103,528,In,0x00, |
| 103,528,In,0x00, |
| 103,528,In,0x00, |
| 103,528,In,0x00, |
| 105,527,Out,0x19, |
| 105,529,In,0x40,@ |
| 105,529,In,0x2C,, |
| 105,529,In,0x41,A |
| 105,529,In,0x56,V |
| 105,530,In,0x2C,, |
| 105,530,In,0x64,d |
| 105,530,In,0x2C,, |
| 105,530,In,0x00, |
| 105,530,In,0x2C,, |
| 105,530,In,0x21,! |
| 105,530,In,0x0A, |
|  |
| 106,028,Out,0x1F,­ |
| 106,028,In,0x7F, |
| 106,028,In,0x80,? |
| 106,028,In,0xFF,? |
| 106,028,In,0x03, |
| 106,028,In,0xB2,? |
| 106,028,In,0x09, |
| 106,028,In,0x00, |
| 106,029,In,0x00, |
| 106,029,In,0x00, |
| 106,029,In,0x00, |
| 106,029,In,0x40,@ |
| 106,029,In,0x2C,, |
| 106,029,In,0x6D,m |
| 106,029,In,0x2C,, |
| 106,029,In,0x6F,o |
| 106,030,In,0x2C,, |
| 106,030,In,0x30,0 |
| 106,030,In,0x30,0 |
| 106,030,In,0x30,0 |
| 106,031,In,0x30,0 |
| 106,031,In,0x2C,, |
| 106,031,In,0x21,! |
| 106,031,In,0x0A, |
|  |
| 106,529,Out,0x20, |
| 106,530,In,0x7F, |
| 106,530,In,0x80,? |
| 106,530,In,0xFF,? |
| 106,530,In,0x03, |
| 106,530,In,0xB2,? |
| 106,530,In,0x09, |
| 106,530,In,0x00, |
| 106,530,In,0x00, |
| 106,530,In,0x00, |
| 106,530,In,0x00, |
| 106,531,In,0x40,@ |
| 106,531,In,0x2C,, |
| 106,531,In,0x6D,m |
| 106,531,In,0x2C,, |
| 106,531,In,0x61,a |
| 106,531,In,0x2C,, |
| 106,531,In,0x30,0 |
| 106,531,In,0x30,0 |
| 106,531,In,0x30,0 |
| 106,531,In,0x30,0 |
| 106,532,In,0x2C,, |
| 106,532,In,0x21,! |
| 106,532,In,0x0A, |
|  |
| 107,029,Out,0x21,! |
| 107,029,In,0x7F, |
| 107,029,In,0x80,? |
| 107,030,In,0xFF,? |
| 107,030,In,0x03, |
| 107,030,In,0xB2,? |
| 107,030,In,0x09, |
| 107,030,In,0x00, |
| 107,031,In,0x00, |
| 107,031,In,0xE8,? |
| 107,031,In,0x03, |
| 107,031,In,0x40,@ |
| 107,031,In,0x2C,, |
| 107,031,In,0x6D,m |
| 107,032,In,0x2C,, |
| 107,032,In,0x66,f |
| 107,032,In,0x2C,, |
| 107,032,In,0x30,0 |
| 107,032,In,0x30,0 |
| 107,032,In,0x30,0 |
| 107,032,In,0x30,0 |
| 107,032,In,0x2C,, |
| 107,032,In,0x21,! |
| 107,032,In,0x0A, |
|  |
| 107,530,Out,0x1D, |
| 107,531,In,0x7F, |
| 107,531,In,0x80,? |
| 107,531,In,0xFF,? |
| 107,531,In,0x03, |
| 107,532,In,0xD9,? |
| 107,532,In,0x04, |
| 107,532,In,0xBA,? |
| 107,532,In,0x08, |
| 107,532,In,0x96,? |
| 107,532,In,0x00, |
| 109,530,Out,0x19, |
| 109,532,In,0x40,@ |
| 109,532,In,0x2C,, |
| 109,532,In,0x41,A |
| 109,532,In,0x49,I |
| 109,532,In,0x2C,, |
| 109,533,In,0x64,d |
| 109,533,In,0x2C,, |
| 109,533,In,0x00, |
| 109,533,In,0x2C,, |
| 109,533,In,0x21,! |
| 109,533,In,0x0A, |
|  |
| 110,031,Out,0x1F,­ |
| 110,031,In,0x7F, |
| 110,031,In,0x80,? |
| 110,031,In,0xFF,? |
| 110,032,In,0x03, |
| 110,032,In,0xD9,? |
| 110,032,In,0x04, |
| 110,032,In,0xBA,? |
| 110,033,In,0x08, |
| 110,033,In,0x96,? |
| 110,033,In,0x00, |
| 110,033,In,0x40,@ |
| 110,033,In,0x2C,, |
| 110,033,In,0x6D,m |
| 110,033,In,0x2C,, |
| 110,033,In,0x6F,o |
| 110,033,In,0x2C,, |
| 110,033,In,0x30,0 |
| 110,033,In,0x30,0 |
| 110,033,In,0x30,0 |
| 110,033,In,0x30,0 |
| 110,033,In,0x2C,, |
| 110,033,In,0x21,! |
| 110,034,In,0x0A, |
|  |
| 110,532,Out,0x20, |
| 110,533,In,0x7F, |
| 110,533,In,0x80,? |
| 110,533,In,0xFF,? |
| 110,533,In,0x03, |
| 110,533,In,0xD9,? |
| 110,533,In,0x04, |
| 110,533,In,0xBA,? |
| 110,533,In,0x08, |
| 110,533,In,0x96,? |
| 110,533,In,0x00, |
| 110,534,In,0x40,@ |
| 110,534,In,0x2C,, |
| 110,534,In,0x6D,m |
| 110,534,In,0x2C,, |
| 110,534,In,0x61,a |
| 110,534,In,0x2C,, |
| 110,534,In,0x30,0 |
| 110,534,In,0x30,0 |
| 110,534,In,0x30,0 |
| 110,534,In,0x30,0 |
| 110,534,In,0x2C,, |
| 110,534,In,0x21,! |
| 110,534,In,0x0A, |
|  |
| 111,032,Out,0x21,! |
| 111,032,In,0x7F, |
| 111,032,In,0x80,? |
| 111,032,In,0xFF,? |
| 111,032,In,0x03, |
| 111,032,In,0xD9,? |
| 111,033,In,0x04, |
| 111,033,In,0xBA,? |
| 111,033,In,0x08, |
| 111,033,In,0x96,? |
| 111,033,In,0x00, |
| 111,033,In,0x40,@ |
| 111,033,In,0x2C,, |
| 111,033,In,0x6D,m |
| 111,034,In,0x2C,, |
| 111,034,In,0x66,f |
| 111,034,In,0x2C,, |
| 111,034,In,0x30,0 |
| 111,034,In,0x30,0 |
| 111,034,In,0x30,0 |
| 111,034,In,0x30,0 |
| 111,034,In,0x2C,, |
| 111,034,In,0x21,! |
| 111,034,In,0x0A, |
|  |
| 111,533,Out,0x02, |
| 112,533,Out,0x01, |
| 112,618,In,0x40,@ |
| 112,618,In,0x2C,, |
| 112,619,In,0x31,1 |
| 112,619,In,0x39,9 |
| 112,619,In,0x38,8 |
| 112,619,In,0x36,6 |
| 112,619,In,0x32,2 |
| 112,619,In,0x32,2 |
| 112,619,In,0x34,4 |
| 112,619,In,0x35,5 |
| 112,620,In,0x2C,, |
| 112,620,In,0x21,! |
| 112,620,In,0x0A, |
|  |
| 114,534,Out,0x1B, |
| 114,535,In,0x7F, |
| 114,536,In,0x80,? |
| 114,536,In,0xFF,? |
| 114,536,In,0x03, |
| 114,536,In,0xE1,? |
| 114,536,In,0x03, |
| 114,536,In,0xCA,? |
| 114,536,In,0x06, |
| 114,536,In,0xC4,? |
| 114,536,In,0x09, |
| 116,535,Out,0x19, |
| 116,537,In,0x40,@ |
| 116,537,In,0x2C,, |
| 116,537,In,0x44,D |
| 116,537,In,0x49,I |
| 116,538,In,0x2C,, |
| 116,538,In,0x64,d |
| 116,538,In,0x2C,, |
| 116,538,In,0x00, |
| 116,538,In,0x2C,, |
| 116,538,In,0x21,! |
| 116,538,In,0x0A, |
|  |
| 117,035,Out,0x1F,­ |
| 117,035,In,0x7F, |
| 117,035,In,0x80,? |
| 117,035,In,0xFF,? |
| 117,035,In,0x03, |
| 117,035,In,0xE1,? |
| 117,035,In,0x03, |
| 117,035,In,0xCA,? |
| 117,035,In,0x06, |
| 117,035,In,0xC4,? |
| 117,035,In,0x09, |
| 117,036,In,0x40,@ |
| 117,036,In,0x2C,, |
| 117,036,In,0x6D,m |
| 117,036,In,0x2C,, |
| 117,036,In,0x6F,o |
| 117,037,In,0x2C,, |
| 117,037,In,0x30,0 |
| 117,037,In,0x30,0 |
| 117,037,In,0x30,0 |
| 117,037,In,0x30,0 |
| 117,037,In,0x2C,, |
| 117,037,In,0x21,! |
| 117,037,In,0x0A, |
|  |
| 117,536,Out,0x20, |
| 117,536,In,0x7F, |
| 117,536,In,0x80,? |
| 117,536,In,0xFF,? |
| 117,536,In,0x03, |
| 117,537,In,0xE1,? |
| 117,537,In,0x03, |
| 117,537,In,0xCA,? |
| 117,537,In,0x06, |
| 117,537,In,0xC4,? |
| 117,537,In,0x09, |
| 117,537,In,0x40,@ |
| 117,537,In,0x2C,, |
| 117,537,In,0x6D,m |
| 117,538,In,0x2C,, |
| 117,538,In,0x61,a |
| 117,538,In,0x2C,, |
| 117,538,In,0x30,0 |
| 117,538,In,0x30,0 |
| 117,538,In,0x30,0 |
| 117,538,In,0x30,0 |
| 117,538,In,0x2C,, |
| 117,538,In,0x21,! |
| 117,539,In,0x0A, |
|  |
| 118,037,Out,0x21,! |
| 118,038,In,0x7F, |
| 118,038,In,0x80,? |
| 118,038,In,0xFF,? |
| 118,038,In,0x03, |
| 118,038,In,0xE1,? |
| 118,038,In,0x03, |
| 118,038,In,0xCA,? |
| 118,038,In,0x06, |
| 118,038,In,0xC4,? |
| 118,038,In,0x09, |
| 118,039,In,0x40,@ |
| 118,039,In,0x2C,, |
| 118,039,In,0x6D,m |
| 118,039,In,0x2C,, |
| 118,039,In,0x66,f |
| 118,039,In,0x2C,, |
| 118,039,In,0x30,0 |
| 118,039,In,0x30,0 |
| 118,039,In,0x30,0 |
| 118,039,In,0x30,0 |
| 118,039,In,0x2C,, |
| 118,039,In,0x21,! |
| 118,040,In,0x0A, |
|  |
| 122,538,Out,0x1A, |
| 122,538,Out,0x24,$ |
| 122,540,In,0x7F, |
| 122,540,In,0x80,? |
| 122,540,In,0xFF,? |
| 122,540,In,0x03, |
| 122,540,In,0xB2,? |
| 122,540,In,0x09, |
| 122,540,In,0x00, |
| 122,540,In,0x00, |
| 122,540,In,0x00, |
| 122,540,In,0x00, |
| 122,540,In,0x7F, |
| 122,540,In,0x80,? |
| 122,540,In,0xFF,? |
| 122,541,In,0x03, |
| 122,541,In,0xD1,? |
| 122,541,In,0x05, |
| 122,541,In,0x00, |
| 122,541,In,0x00, |
| 122,541,In,0x00, |
| 122,541,In,0x00, |
| 122,541,In,0x40,@ |
| 122,541,In,0x2C,, |
| 122,541,In,0x44,D |
| 122,542,In,0x56,V |
| 122,542,In,0x2C,, |
| 122,542,In,0x64,d |
| 122,542,In,0x2C,, |
| 122,542,In,0x00, |
| 122,542,In,0x2C,, |
| 122,542,In,0x21,! |
| 122,542,In,0x0A, |
|  |
| 125,040,Out,0x02, |