



Universidad de Antioquia
Facultad de Ingenieria
Informatica II

Informe 1. Primer examen

Integrantes:

Juan Jose Baquero Arcila

Jose Alejandro Olivo Petit

Sebastian Bolivar Vanegas

Colombia, Medellín
Febrero 2022

Índice general

1. Objetivos	2
1.1. Objetivo general	2
1.2. Objetivo específico	2
2. Resumen	3
3. Introducción	4
4. Marco Teorico	5
5. Análisis del problema	8
6. Marco experimental	10
7. Resultados	11
Referencias	12

Capítulo 1

Objetivos

1.1. Objetivo general

Adquirir la capacidad para trabajar con Arduino e integrar la programación en C++, además de usar de manera adecuada las funciones de las plataformas explicadas en clase que permiten controlar el puerto serial y los puertos digitales.

1.2. Objetivo específico

- Desarrollar la capacidad de solución de problemas enfrentándonos a problemáticas de la vida cotidiana.
- Adquirir las destrezas y conocimientos fundamentales de la programación con C++, en donde resaltamos estructuras de programación (iterativas, secuenciales y de decisión), tipos de datos, apuntadores, arreglos, funciones y memoria dinámica.

Capítulo 2

Resumen

Espacio para abarcar el Resumen...

Capítulo 3

Introducción

Espacio para abarcar la intro...

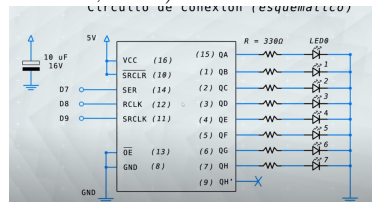
Capítulo 4

Marco Teorico

La paralelizacion es una forma de organizar cierto tipo de cosa, El paralelismo es una forma de computación en la cual varios cálculos pueden realizarse simultáneamente, La computación paralela es un tipo de arquitectura informática en la que varios procesadores ejecutan simultáneamente múltiples cálculos más pequeños desglosados de un problema complejo más grande en general. (Farestrepoca, 2016).

Para el funcionamiento del circuito integrado 74HC595, tuvimos que valernos de la documentación de Texas instrument para conocer el funcionamiento de los 16 pines que tiene el integrado. En la siguiente imagen podremos apreciar el uso de cada uno de los pines con su respectivo nombre. Como los mas relevantes podemos encontrar los 8 pines destinados para la paralelización de datos ubicados del 1 al 7 y el 15, también los pines de voltaje y conexión a tierra y los tres pines para el control del circuito. Una de las tres señales que controlan el funcionamiento es donde le entra la data de forma serial (SER), otra es usada para el registro de entrada del reloj (RCLK) en la cual ya organizo mis salidas con valores y una ultima señal(SRCLK) la cual se encarga de registrar la entrada del reloj del sistema. Con estos pines podemos entender como seria el funcionamiento de este integrado, el cual es el de registro de desplazamiento. Este se da gracias a que recibe datos de forma serial para luego tenerlos separados simultáneamente en los diferentes pines que tiene el circuito.

(intrument, 1982).



I2C es un puerto y protocolo de comunicación serial diseñado por Philips Semiconductors a principios de los 80s, define la trama de datos y las conexiones físicas para transferir bits entre 2 dispositivos digitales. El puerto incluye dos cables de comunicación, SDA y SCL. Además el protocolo permite conectar hasta 127 dispositivos esclavos con esas dos líneas, con hasta velocidades de 100, 400 y 1000 kbits/s. También es conocido como IIC ó TWI – Two Wire Interface.

Descripción de las señales:

o SCL (System Clock) es la línea de los pulsos de reloj que sincronizan el sistema.

o SDA (System Data) es la línea por la que se mueven los datos entre los dispositivos.

o GND (Masa) común de la interconexión entre todos los dispositivos enganchados al bus.

Las líneas SDA y SCL: son del tipo drenaje abierto, es decir, un estado similar al de colector abierto, pero asociadas a un transistor de efecto de campo (o FET). Se deben polarizar en estado alto (conectando a la alimentación por medio de resistores "pull- up") lo que define una estructura de bus que permite conectar en paralelo múltiples entradas y salidas.

El MAESTRO I2C: se encarga de controlar al cable de reloj, por sus siglas en inglés llamada SCL – Serial CLock. Además el MAESTRO se encarga de iniciar y parar la comunicación. La información binaria serial se envía sólo por la línea o cable de datos seriales, en inglés se llama SDA – Serial DAta. Dos Maestros no pueden hacer uso de un mismo puerto I2C. Puede funcionar de dos maneras, como maestro-transmisor o maestro-receptor. Sus funciones principales son:

- Iniciar la comunicación – S
- Enviar 7 bits de dirección – ADDR
- Generar 1 bit de Lectura ó Escritura – R/W
- Enviar 8 bits de dirección de memoria
- Transmitir 8 bits de datos –
- Confirmar la recepción de datos – ACK – ACKnowledged
- Generar confirmación de No-recepción, NACK – No-ACKnowledged
- Finalizar la comunicación

El ESCLAVO I2C, generalmente suele ser un sensor. Este elemento suministra de la información de interés al MAESTRO. Puede actuar de dos formas: esclavo-transmisor ó esclavo-receptor. Un dispositivo I2C esclavo, no puede generar a la señal SCL. Sus funciones principales son:

- Enviar información en paquetes de 8 bits.

- Enviar confirmaciones de recepción, llamadas ACK (Carletti, 2017).

Capítulo 5

Análisis del problema

Transmisión:

En la primera parte de la transmisión, primero recibiremos el arreglo de números enteros proporcionados para nuestro equipo. Para esta parte se creará un algoritmo probado en Qt que reciba cadenas de cualquier tamaño y sea capaz de poder operar con ellas, para luego proceder a hacer la determinada transformación del número de decimal a binario para así poder proporcionar la data que el sistema completo requiere. También en esta parte hay una señal de reloj, que junto a la señal de data que es el número convertido a binario se le pasa al siguiente sistema que paraleliza los datos y al sistema de recepción de información.

Paralelización y descriptación:

En esta parte del sistema, usaremos ya la parte del circuito integrado para poder utilizar la data de forma eficiente, paralelizándola de forma que la tengamos en cada uno de los puertos de nuestro circuito integrado. Cuando ya tengamos nuestra data organizada de la manera que queremos gracias a nuestro circuito, pasaremos a la parte de la lógica del sistema de descriptación que dependerá de la bandera que nos haya tocado y de lo que tengamos que hacer después de tener la bandera. Para la parte de recibir la bandera usaremos la lógica combinatorial de las compuertas más conocidas (and,or,xor) para poder darle lógica a nuestra bandera y empezar la descriptación de los datos y para la parte del procesamiento de los datos podemos usar el reloj del sistema para poder decidir que números del arreglo de la cadena son los que necesitamos.

Recepción:

Después de haber hecho válida la lógica de la bandera en nuestro arreglo de números, procederemos a decirle al sistema de recepción que debe hacer con los datos que le están llegando en serie desde el sistema que genera la información, entonces gracias a esta lógica él puede saber cuáles son los que necesita y así guardarlos para luego poderlos pasar al sistema de visualización y mandarle los

datos clasificados al pc2.

Capítulo 6

Marco experimental

Pruebas de funcionamiento de la transmisión serial.

Para la comunicación la conexión entre los dos Arduino se utilizó el protocolo I2C para ello se utilizaron tres puertos, el SDA , el SCL que en el Arduino uno R3 son los pines A4 y A5 además se realizó la conexión de sus respectivas tierras en el programa realizado se ingresa un carácter por el monitor serial en el Arduino maestro para luego ser transferido al Arduino esclavo y ser impreso en su monitor serial.

Para comprobar el correcto funcionamiento del sistema se hicieron una serie de pruebas:

prueba 1: se ingresó una letra por el monitor serial, dando como resultado una letra en el monitor serial del esclavo.

Prueba 2: se ingresó un numero por el monitor serial, dando como resultado un numero en el monitor serial del esclavo.

Prueba 3: se ingresó una lista de caracteres al mismo tiempo en el monitor serial, dando como resultado la impresión de cada uno de los caracteres de forma individual.

De esta se comprobó satisfactoriamente el correcto funcionamiento del sistema.

link de tinkercad: <https://www.tinkercad.com/things/jMFM9EehBc>

Capítulo 7

Resultados

Referencias

- Carletti, E. J. (2017, 19 de Julio). *Comunicación-bus i2c*. Publicación de un blog. Descargado de <https://blogthinkbig.com/ventajas-latex-editar-documentos>
- Farestrepoca. (2016, 12 de Mayo). *Programación paralela*. Publicación de un blog. Descargado de http://ferestrepoca.github.io/paradigmas-de-programacion/paralela/paralela_teoría/index.html#eleven
- intrument, T. (1982, 22 de Diciembre). *Hoja de referencia circuito integrado*. Publicación de un blog. Descargado de https://www.ti.com/lit/ds/symlink/sn74hc595.pdf?ts=1645444457612&ref_url=https%253A%252F%252Fwww.google.com%252F