**Universidad Autónoma de Guadalajara**

Ingeniería Electrónica Biomédica

System designing with Microprocessors

*“*Practice 8: I2C Communication with EEPROM”

Jesús Arnoldo Zerecero Núñez

2885993

Andrea Alejandra Mondragón Olivos

2915351

Practice 8. I2C Communication with EEPROM

**Introduction**

I2C (Inter-Integrated Circuit) is a widely used serial synchronous communication protocol. It’s mostly used to communicate in very short distances, most of the time between devices of the same circuit. It’s a simple, dual-duplex communication with a huge advantage: only 2 cables need to be used: SCL (the clk set by the master) and SDA (data line and addresses). A single or multiple masters may communicate with a single or multiple slave devices. Both the master and the slave may act as transmitters or receivers.

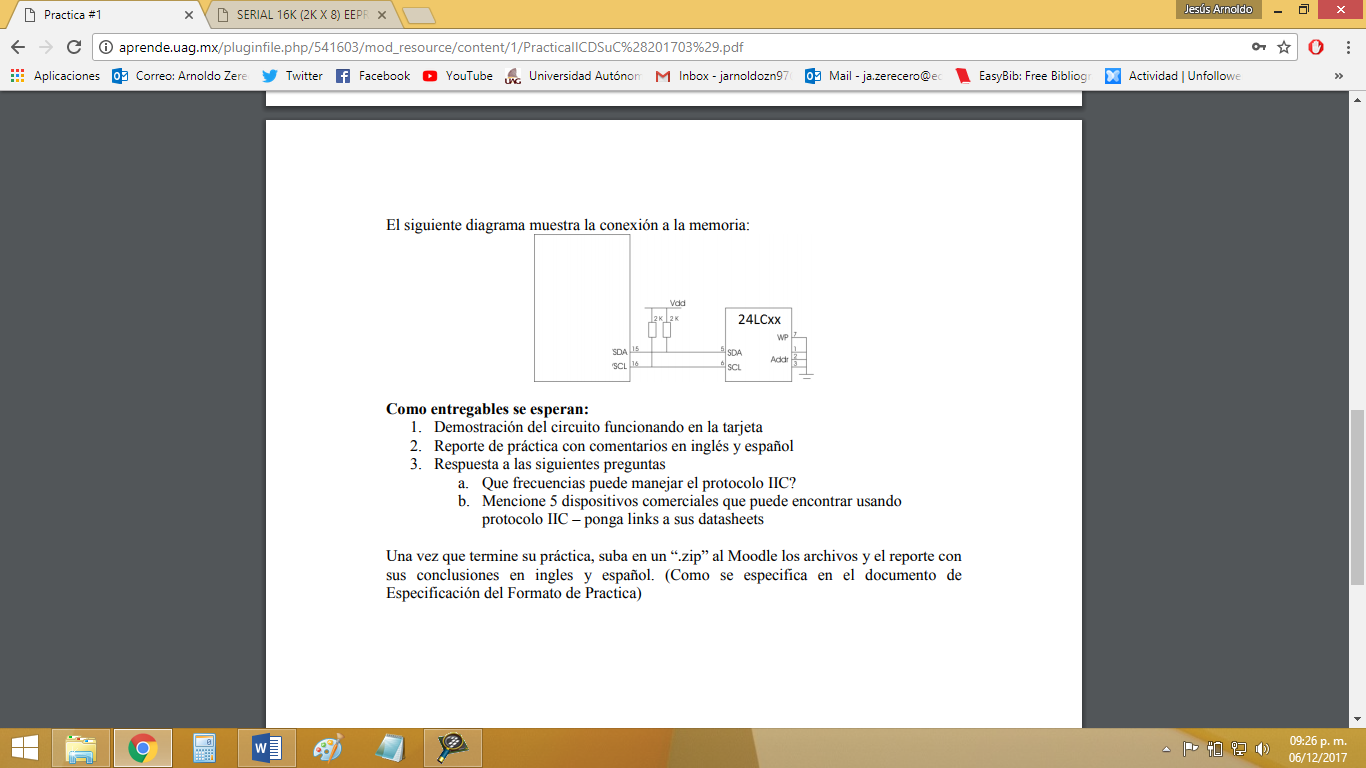
The I2C protocol works in this way:

1. A start condition is sent by one master (SDA up to low transition).
2. An 8-bit address is sent to choose which slave to talk to.
   * The slave with matching address sends an ACK signal back through the SDA and gets prepared to receive/transmit.
   * The last bit of the address indicates the operation. 0 to write, 1 to read.
3. In case of that slave being an EEPROM (electrically erasable programmable memory) an 8-bit address is sent to specify which part of the memory to read/write.
4. In case of writing the memory, the master sends up to 8 bytes of data, each needed to be acknowledge by the slave, one by one.
   * At the end of the transfer, the master sends a stop signal (SDA down to high transition) , initiating the slave’s write cycle, which can take up to 20ms to complete, time in which the slave won’t respond to any other operation.
5. In case of reading the memory, the master must send a repeated start condition, send once again the EEPROM address and then put itself in receiver (slave) mode.
   * The slave then takes control of the SDA line and sends data bytes to the master.
   * If the master acknowledges a byte, the slave proceeds to send the byte stored in its next address.
   * The slave keeps sending bytes until the master instead of acknowledging, sends a stop condition.

In this practice, the KL25z communicates with the ST24C16 16k EEPROM via the I2C module. The user may input commands through a terminal (which communicates to the KL25z through the UART module) to search for slaves, write multiple bytes to memory or read multiple bytes from memory. The terminal shows the result of each operation, and in case of a failed transaction, shows a FAIL message.

**Development.**

**Circuit connection:**



**Conclusion:**

The I2C protocol usually manages clk frequencies of 100KHz or 400KHz. So, the KL25z’s bus clock of 21MHz needed to be divided 192 times.

I2C is widely used to allow communication between small and simple devices, these are some examples:

* MCP9808 High Accuracy I2C Temperature Sensor: <https://www.adafruit.com/product/1782>
* 96 segment LCD driver: <https://www.nxp.com/docs/en/data-sheet/PCF8566.pdf>
* 16 Bit I/O port expander: <http://www.ti.com/lit/ds/symlink/pcf8575.pdf>
* Clock/calendar: <http://www.picmicrolab.com/wp-content/uploads/2014/05/PCF8573.pdf>
* Voice synthesizer: <http://www.alldatasheet.es/datasheet-pdf/pdf/168161/PHILIPS/PCF8200.html>

**Video demonstration link:** [**https://photos.app.goo.gl/5Qrs6c0xtQt8rwso1**](https://photos.app.goo.gl/5Qrs6c0xtQt8rwso1)