SCANNER

Prepared under the able guidance of:
Prof. K Anupama

Group 27

Members:

Sarthak Dalmia (2018B3A70290G)

Yash Bohra (2018B3AA0711G)

Manav Tanwar (2018B3A70629G)

Somrat Dutta (2018B3A70327G)

Sudarshan Sawal (2018B3AA0760G)

Sehaj Pal Sandhu (2018B3A30728G)

Submitted as a part of partial fulfillment of the course

Microprocessors, Programming & Interfacing (CS F241)

Contents

User Requirements & Technical Specifications 2

Assumptions & Justifications 2

- Justification 2
- Assumptions 2

Components used with justification wherever required 3

Address Map 4

- Memory Map 4
- I/O Map 4

Design 5

Flow Chart 6

- Main Program 6
- Flow Chart of ISR 7
- ADC ISR 8

Variations in Proteus Implementation with Justification 9

Firmware 10

List of Attachments 11

User Requirements & Technical Specifications

Design a microprocessor-based scanner which will scan a black and white image and store it as binary data. The user presses a switch labelled ***Start Scan*** when he wants the scanning process to be completed. Once scanning is completed, an LED labelled ***Scan Complete*** will glow. The Technical Specifications are as follows:

- The scanner has two stepper motors for motion along two orthogonal coordinates. The rotational
motion is converted into translational motion through a lead screw mechanism.
- Five paired LED photodiodes intended for B&W image scanning are placed 0.1 centimeter apart.
- The maximum size scannable is 10cm x 10cm.
- The photodiode output is analog signal (between 0 to 5 volts) which is to be digitized.
- Image information is stored sequentially in the RAM.

Design Specifications

The Intel 8086 processor is the central processing unit (CPU) of the image scanner system. Two Intel 8255 Programmable Peripheral Interfacing devices are used to interface the two stepper motors and an Analog-To Digital Converter ADC0808. Each of the stepper motors are connected to the 8255 through the High Voltage High Current Darlington Transistor Array LC2003A. The 5 paired photodiodes are interfaced to the system through the ADC0808. The ALP program which is used to run the device is stored in ROM chips. Four 2716 ROM chips are used for this purpose, two of which work as even banks and the other working as odd banks. For storing the image two RAM chips are used each of size 64 kilo-bytes each. One of the RAM chips is working as an even bank while the other is working as an odd bank. The RAM model used is 61512. The two ROM chips which count a total of 8 kilo-bytes of memory are interfaced to the address lines 00000H to 00FFFH and from FF000H to FFFFFH. Three 74LS373 octal latches are used to demultiplex the AD lines coming from the processor to get address line 0 to 19. Two 74LS245 bidirectional buffers are also used to demultiplex the AD lines coming from the processor to get data lines 0 to 15. Some OR and NOT gates are used to generate the various signal lines for memory and I/O interfacing. The rotational motion of the stepper motors is converted to translational motion by using the lead-screw mechanism.

Assumptions & Justifications

Justification

1. As there are 5 photodiodes 0.1cm apart and given our assumption about the distance between two pixels, at each sampling of the ADC output we get 5 digitized values of pixels each 8 pixels apart. (or 0.1cm apart)

Assumptions

Certain assumptions are made while implementing the project.
These assumptions are:

1. There is no power failure while the system is operating.
2. The 8086 Chip is already programmed with the specified code from an external source.
3. The pixels to be read from the paper are 0.0125 cm apart. Therefore, there are 8 gaps each of 0.0125 cm in 0.1 centimeter. Therefore, there are 801 pixels to be read in each row and 801 such rows.
4. The starting position is from the top-left corner of the page. Motion of the photodiodes would be from left to right
and top to bottom of a page in the read operation.
5. The stepper motor requires a software delay of 1 millisecond while it rotates into position.

Components used with justification wherever required

Name of the component	Number of the component used	Purpose
INTEL 8086	1	It is the main processor of the image scanner. All the devices are controlled directly or indirectly through 8086
INTEL 8255A	2	This device interfaces the ADC, the LED, the Switch and Stepper motors to the 8086.
ADC0808	1	The main purpose of this device is to convert the analog signals that are coming from the photo-diodes to digital signals so that we can store the image as binary data in the memory.
ULN2003A	2	These devices are used for connection of the Stepper motors with 8255. One of the ULN2003A is connected to the horizontal movement while the other is to vertical movement.
ROM2716	2	For storing the ALP program which is used to run the scanner.
RAM61512	2	These ram chips are used to store the binary data of the scanned image.
74LS138	2	The chip is used to interface the ROM and RAM chips. Other chip is used to interface the two 8255 and one 8254
74LS373	3	Octal latches for address lines
74LS245	2	Bidirectional buffer for data lines.
STEPPER MOTORS	2	The stepper motors are responsible for the movement along X and Y direction.
INTEL 8254-5	1	This timer is used to generate the 1MHz clock for ADC0808
LDR	5	Used to detect light.

Components used with justification wherever required

Besides these devices, simple gates like AND, OR and NOT gates are used for various purposes which include the generation of IOR', IOW', MEMR', MEMW' signals, connecting address lines to the peripheral devices and to control RAM and ROM through 74LS138. The image sensor is made up of five photodiodes. Physically, these photodiodes would be connected side by side in a board, which would be connected to stepper motors enabling the structure to scan the entire image.

• 8086 – main computing unit

• LDR– 5 in quantity for pixel detection. This gives an output between 0–5V depending on the light intensity. (higher intensity value close to 5V and lower intensity value close to 0V.)

• Stepper Motors – (2 unipolar motors in full stepping mode) used in a lead screw mechanism for horizontal and vertical movement of the photodiode (or LDR) array.

• ULN2003A – 2 in quantity to drive the two stepper motors.

• ADC 0808 - 5 analog inputs with voltage varying from 0 - 5 V with 8–bit resolution photodiodes output can be directly connected to it (as it directly compatible) - Manual Attached

• 8255 - Interface ADC, stepper motors, 1 switch and 1 LED

• 8254 - to generate ADC Clock

• 2716 - 4 nos. Smallest ROM chip available is 2K and as we need to have even and odd bank and ROM is required at reset address which is at FFFF0H and 00000H – where there is the IVT

• 61512 - 2 nos. of this RAM chip available and we need odd and even banks. We need RAM for stack and temporary storage of data in this case an Image. We have pixels in 0.1cm. So, storing 0.1cm of data requires 1 byte of storage. Thus, for 10cm we need 100 bytes of storage. As there will be 1 odd pixel left at the end, we will have to allocate another byte of storage to store that last pixel. So, each row (0.0125cm apart) requires 101bytes of storage. As there are 801 such rows in 10cms we need a total of 80,901 bytes of storage. Thus, we have used 2 chips of 64k each to get a total of 128k memory.

• LS 138 - 2 decoders (1 for memory interfacing, 1 for IO devices interfacing)

• LS 373 –3 latches (To get address lines)

• LS 245 – 2 octal transceivers (to get the data lines)

• LS 244 – 1 buffer (to get the control signals)

• LED – 1 for indication of completion of scanning process

• Switch – 1 for start of scanning process

Address Map

Memory Map

ROM1 – 00000H – 00FFFH

RAM – 01000H – 20FFFH

ROM2 – FF000H – FFFFFH

I/O Map

8255(1) - 80H – Port A

82H – Port B

84H – Port C

86H – Control Register

8255(2) - 88H – Port A

8AH – Port B

8CH – Port C

8EH – Control Register

8254 - 86H – Counter 0

86H – Counter 1

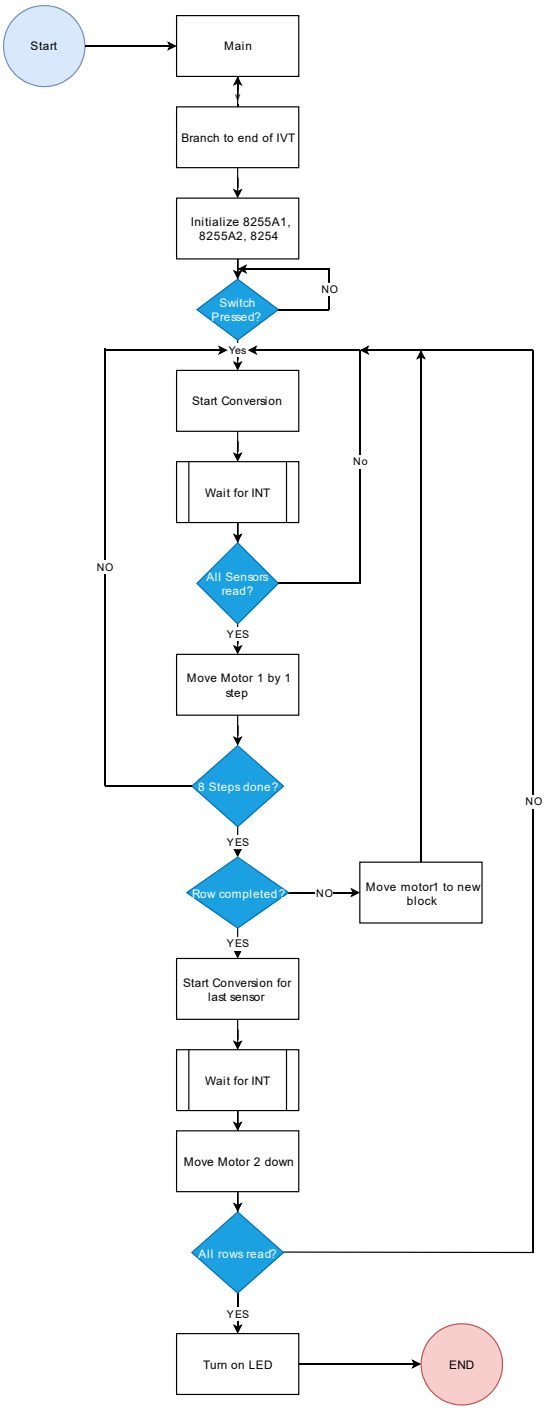
86H – Counter 2

86H – Control Register

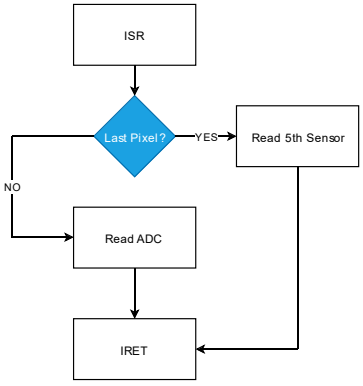
Design Flow Chart

Complete design shown with proper labelling (design attached)

Flow Chart



ad_isr



Variations in Proteus Implementation with Justification

1. ROM is only 00000 - as proteus allows to change reset address.

2. We have not used 8254 (as not there in proteus), to generate the clock for ADC0808, we have directly used a pulse generator to get 1MHz clock.

3. 6116 RAM is used instead of 61512 as this is not available in Proteus. So instead of two chips of 61512 to get 128k of RAM we use 2 chips of 6116 to get 4K of RAM which although does not satisfy our requirements of 81k memory but is enough for simulation purposes.

4. 2732 ROM is used instead of 2716 as its not available in Proteus

5. Using a gate-based circuit for memory - does the same as LS 138 here

6. Used an LDR instead of a photodiode due to availability in Proteus.

7. EOC is used as NMI.

Firmware

Implemented using emu8086 attached.

List of

1. Complete Hardware Real World Design - Scanner.pdf
2. Manuals
    a. ADC 0808
    b. 12-Volt Unipolar Stepper Motor (#27964)
    c. NORP12 (LDR)
    d. ULN2003A
3. Proteus File - design.dsn
4. EMU8086 ASM File - Scanner.asm
5. Binary File after assembly - Scanner.bin