Kevin Gilbert and Graham Gilvar

1-31-2012 Lab1

Tuesday 4-5PM Lab Section EE319K

PartA:

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Lab\_1 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

; Program written by: Kevin Gilbert and Graham Gilvar

; Date created: 29 January 2012 4:45 PM

; Last Modified 29 January 2012 10:00 PM

; Section Tuesdays 4:00-5:00 PM

; Lab Number: 1

; Program designed to simulate a digital lock, in which

; a LED represents a lock, and is only on (unlocked)

; when both of the two switches are unpressed (open)

; Hardware Connections:

; PH2 is Switch Input H

; PP2 is Switch Input P

; PT2 is LED Output T (ON = UNLOCKED)

; I/O Port definitions

PTH equ $0260

DDRH equ $0262

PTP equ $0258

DDRP equ $025A

PTT equ $0240

DDRT equ $0242

; \*\*\*\*\*\*\*\*\* EEPROM Begin \*\*\*\*\*\*\*\*\*

main org $4000

ldaa DDRT ;Data Direction Register for the LED (T)Output,Bit2 = 1

clra

oraa #$04

staa DDRT

ldaa DDRH ;Data Direction Register for SwitchH (H)Input,Bit2 = 0

clra

anda #$FB

staa DDRH

ldaa DDRP ;Data Direction Register for SwtichP (P) Input,Bit2 = 0

clra

anda #$FB

staa DDRH

loop ldaa PTH ;Beginning of Testing Loop, load masked PTH and PTP

;Into RegA and RegB, if RegD = 0, LED=On, if not, LEDOFF

anda #$04

ldab PTP

andb #$04

addd #$00 ; Load condition codes with RegD for branches

beq LED\_On

bra LED\_Off

LED\_On ldaa PTT

oraa #$04

staa PTT

bra loop

LED\_Off ldaa PTT

anda #$FB

staa PTT

bra loop

org $FFFE ; Reset Vector

fdb main

PartB:

RegD = 0?

Set condition codes with RegD

Load masked PTP into RegB

Load masked PTH into RegA

Set DDRT to Output, and DDRP/DDRH to input

YES NO

Load RegA with PTT

Flip data bit 2 to 0 to turn off LED

Flip data bit 2 to 1 to activate LED

Load RegA with PTT

PartC:

Main Initialize I/O: Set DDRT to output, DDRH/DDRP to input

Loop Load bitmasked value of PTH into RegA

Load bitmasked value of PTP into RegB

Add zero to RegD to initialize condition codes with RegD

Branch to LED\_ON if RegD = 0

Unconditional branch to LEDOFF

LED\_ON Load RegA with PTT

Bitmask PTT to make bit2 = 1

Store RegA back to PTT

Unconditional branch to Loop

LEDOFF Load RegA with PTT

Bitmask PTT to make bit2 = 0

Store RegA back to PTT

Unconditional branch to Loop

