2DX4: Digital Signals

Lab #2

Instructor: Drs. Bruce, Haddara, Hranilovic, and Shira

Lab TAs: Abdallah Ghazy

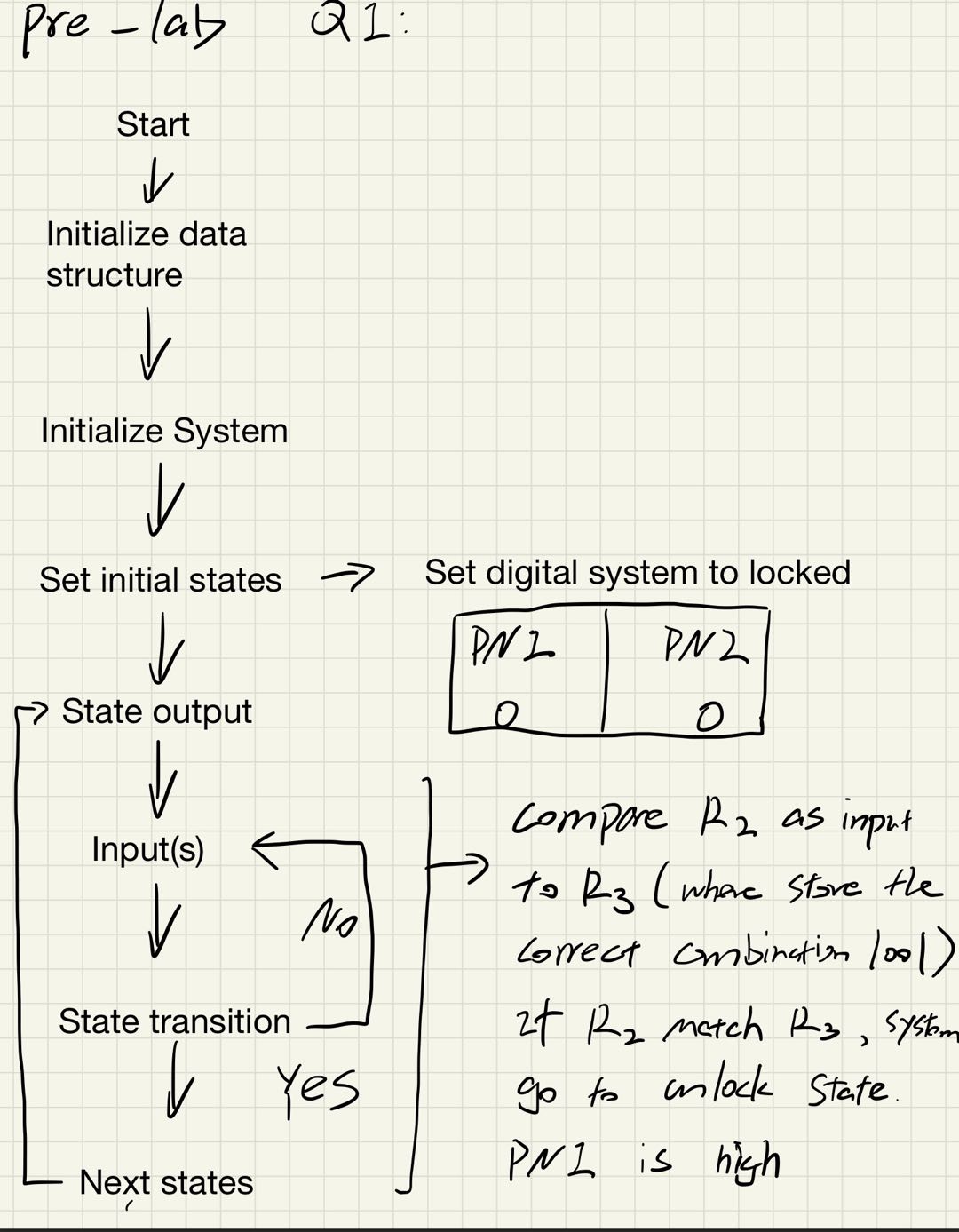
(TA MACID1), Tau Rasethuntsa

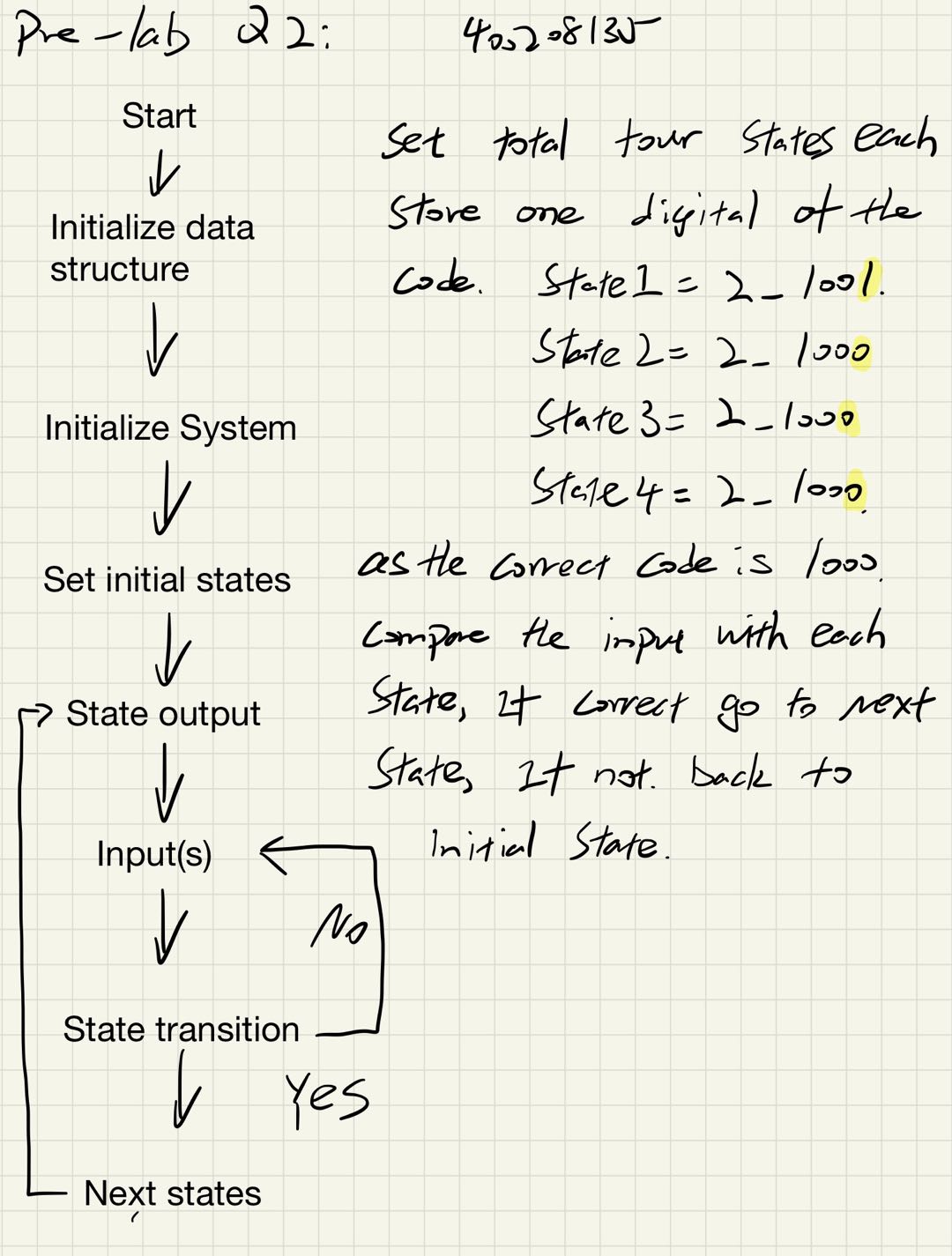
(TA MACID2)

Tianze Zhang L01 – zhant22

As a future member of the engineering profession, the student is responsible for performing he required work in an honest manner, without plagiarism and cheating. Submitting this work with my name and student number is a statement and understanding that this work is my own and adheres to the Academic Integrity Policy of McMaster University and the Code of Conduct of the Professional Engineers of Ontario. Submitted by [Tianze Zhang, zhant22, 400208135]

# Pre lab





# Milestone1

; Name: tianze zhang(Dylan zhang)

; Student Number: 400208135

; Lab Section: L01

; Description of Code: using 1000 as combination code to turn on D2 as successful unlock.

; Original: Copyright 2014 by Jonathan W. Valvano, valvano@mail.utexas.edu

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

;ADDRESS DEFINTIONS

;The EQU directive gives a symbolic name to a numeric constant, a register-relative value or a program-relative value

SYSCTL\_RCGCGPIO\_R EQU 0x400FE608 ;General-Purpose Input/Output Run Mode Clock Gating Control Register (RCGCGPIO Register)

GPIO\_PORTN\_DIR\_R EQU 0x40064400 ;GPIO Port N Direction Register address

GPIO\_PORTN\_DEN\_R EQU 0x4006451C ;GPIO Port N Digital Enable Register address

GPIO\_PORTN\_DATA\_R EQU 0x400643FC ;GPIO Port N Data Register address

GPIO\_PORTM\_DIR\_R EQU 0x40063400 ;GPIO Port M Direction Register Address (Fill in these addresses)

GPIO\_PORTM\_DEN\_R EQU 0x4006351C ;GPIO Port M Direction Register Address (Fill in these addresses)

GPIO\_PORTM\_DATA\_R EQU 0x400633FC ;GPIO Port M Data Register Address (Fill in these addresses)

COMBINATION EQU 2\_1000 ;the 2\_ here means binary, if we dont want our code to be 000,we put 0001, the last 1 is for the load button.

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

;Do not alter this section

AREA |.text|, CODE, READONLY, ALIGN=2 ;code in flash ROM

THUMB ;specifies using Thumb instructions

EXPORT Start

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

;Function PortN\_Init

PortN\_Init

;STEP 1

LDR R1, =SYSCTL\_RCGCGPIO\_R

LDR R0, [R1]

ORR R0,R0, #0x1000

STR R0, [R1]

NOP

NOP

;STEP 5

LDR R1, =GPIO\_PORTN\_DIR\_R

LDR R0, [R1]

ORR R0,R0, #0x3

STR R0, [R1]

;STEP 7

LDR R1, =GPIO\_PORTN\_DEN\_R

LDR R0, [R1]

ORR R0, R0, #0x3

STR R0, [R1]

BX LR

PortM\_Init

;STEP 1

LDR R1, =SYSCTL\_RCGCGPIO\_R

LDR R0, [R1]

ORR R0,R0, #0x800 ;port M is R11

STR R0, [R1]

NOP

NOP

;STEP 5

LDR R1, =GPIO\_PORTM\_DIR\_R

LDR R0, [R1]

ORR R0,R0, #0x00

STR R0, [R1]

;STEP 7

LDR R1, =GPIO\_PORTM\_DEN\_R

LDR R0, [R1]

ORR R0, R0, #0xF ;which means the last 4 bit, also can be wtitten in binary 2\_001111;

STR R0, [R1]

BX LR

State\_Init LDR R5,=GPIO\_PORTN\_DATA\_R ;Locked is the Initial State

MOV R4,#2\_0000010 ; the initial state is locked, so D1 ON

STR R4,[R5]

BX LR

Start

BL PortN\_Init

BL PortM\_Init

BL State\_Init ;call stste init to lock

LDR R0, = GPIO\_PORTM\_DATA\_R ; Inputs set pointer to the input

LDR R3, =COMBINATION ;R3 stores our combination

Loop

LDR R1,[R0] ;load the cotent of data register into r1

AND R2,R1,#2\_001111 ; we only care about the 1111 bits . and store into r2. bit masking.

CMP R2,R3 ; we compare r2 and r3, r3 store the combination to unlock,

BEQ Unlocked\_State ;if R2 = R3, we go to unlock state

BNE Locked\_State ; if not, we go to lockd state

Locked\_State

LDR R5,=GPIO\_PORTN\_DATA\_R

MOV R4,#2\_00000010 ; if licked, we light up 01 in prot n D1 still ON

STR R4,[R5]

B Loop ;back to loop

Unlocked\_State

LDR R5, =GPIO\_PORTN\_DATA\_R

MOV R4,#2\_00000001 ; if unlock, we light up 10 in port n D2 ON.

STR R4, [R5]

B Loop

ALIGN

END

# Milestone 2

; Name: tianze zhang(Dylan zhang)

; Student Number: 400208135

; Lab Section: L01

; Description of Code: using 1000 as combination code to turn on D2 as successful unlock.

; Original: Copyright 2014 by Jonathan W. Valvano, valvano@mail.utexas.edu

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

;ADDRESS DEFINTIONS

;The EQU directive gives a symbolic name to a numeric constant, a register-relative value or a program-relative value

SYSCTL\_RCGCGPIO\_R EQU 0x400FE608 ;General-Purpose Input/Output Run Mode Clock Gating Control Register (RCGCGPIO Register)

GPIO\_PORTN\_DIR\_R EQU 0x40064400 ;GPIO Port N Direction Register address

GPIO\_PORTN\_DEN\_R EQU 0x4006451C ;GPIO Port N Digital Enable Register address

GPIO\_PORTN\_DATA\_R EQU 0x400643FC ;GPIO Port N Data Register address

GPIO\_PORTM\_DIR\_R EQU 0x40063400 ;GPIO Port M Direction Register Address (Fill in these addresses)

GPIO\_PORTM\_DEN\_R EQU 0x4006351C ;GPIO Port M Direction Register Address (Fill in these addresses)

GPIO\_PORTM\_DATA\_R EQU 0x400633FC ;GPIO Port M Data Register Address (Fill in these addresses)

counter EQU 0x8888

COMBINATION EQU 2\_1001 ;the 2\_ here means binary, if we dont want our code to be 1000,we put0001

digit1 EQU 2\_1001

digit2 EQU 2\_1010

digit3 EQU 2\_1100

digit4 EQU 2\_1000

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

;Do not alter this section

AREA |.text|, CODE, READONLY, ALIGN=2 ;code in flash ROM

THUMB ;specifies using Thumb instructions

EXPORT Start

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

;Function PortN\_Init

PortN\_Init

;STEP 1

LDR R1, =SYSCTL\_RCGCGPIO\_R

LDR R0, [R1]

ORR R0,R0, #0x1000

STR R0, [R1]

NOP

NOP

;STEP 5

LDR R1, =GPIO\_PORTN\_DIR\_R

LDR R0, [R1]

ORR R0,R0, #2\_011111

STR R0, [R1]

;STEP 7

LDR R1, =GPIO\_PORTN\_DEN\_R

LDR R0, [R1]

ORR R0, R0, #2\_011111

STR R0, [R1]

BX LR

PortM\_Init

;STEP 1

LDR R1, =SYSCTL\_RCGCGPIO\_R

LDR R0, [R1]

ORR R0,R0, #0x800 ;port M is R11

STR R0, [R1]

NOP

NOP

;STEP 5

LDR R1, =GPIO\_PORTM\_DIR\_R

LDR R0, [R1]

ORR R0,R0, #0xC0

STR R0, [R1]

;STEP 7

LDR R1, =GPIO\_PORTM\_DEN\_R

LDR R0, [R1]

ORR R0, R0, #2\_011111

STR R0, [R1]

BX LR

State\_Init

LDR R5,=GPIO\_PORTN\_DATA\_R ;Locked is the Initial State

MOV R4,#2\_0000010 ; the initial state is locked, so D1 ON

STR R4,[R5]

BX LR

IsZero

LDR R0, =GPIO\_PORTM\_DATA\_R

LDR R1, [R0]

AND R2, R1, #2\_00010000

CMP R2, #2\_00000000

BNE IsZero

BX LR

IsOne

LDR R0, =GPIO\_PORTM\_DATA\_R

LDR R1, [R0]

AND R2, R1, #2\_00010000

CMP R2, #2\_00010000

BNE IsOne

BX LR

Start

BL PortN\_Init ;The BL instruction is like a function call

BL PortM\_Init ;The BL instruction is like a function call

BL State\_Init ;The BL instruction is like a function call

State\_0

BL IsZero

BL IsOne

MOV R3, #2\_00010001 ;Load the COMBINATION code into register 3 (R1), R3 = 0x400643FC

AND R2,R1,#2\_00010001 ;Perform a bitwise OR operation with the contents of R0 with 0x10 and put the contents into R0 , R0 = 0x3

CMP R2,R3 ;Perform the comparesion between the input code and combination code.

BEQ State\_1 ;If R2 = R3, begin function Unlocked\_State

BNE State\_0 ;If R2 != R3, begin function Locked\_State

State\_1

BL IsZero

BL IsOne

MOV R3, #2\_00010000 ;LDR R3, = Digit\_11 ;Load the COMBINATION code into register 3 (R1), R3 = 0x400643FC

AND R2,R1,#2\_00010000 ;Perform a bitwise OR operation with the contents of R0 with 0x10 and put the contents into R0 , R0 = 0x3

CMP R2,R3 ;Perform the comparesion between the input code and combination code.

BEQ State\_2 ;If R2 = R3, begin function Unlocked\_State

BNE State\_0 ;If R2 != R3, begin function Locked\_State

State\_2

BL IsZero

BL IsOne

MOV R3, #2\_00010000 ;LDR R3, = Digit\_11 ;Load the COMBINATION code into register 3 (R1), R3 = 0x400643F

AND R2,R1,#2\_00010000 ;Perform a bitwise OR operation with the contents of R0 with 0x10 and put the contents into R0 , R0 = 0x3

CMP R2,R3 ;Perform the comparesion between the input code and combination code.

BEQ State\_3 ;If R2 = R3, begin function Unlocked\_State

BNE State\_0 ;If R2 != R3, begin function Locked\_State

State\_3

BL IsZero

BL IsOne

MOV R3, #2\_00010000 ;LDR R3, = Digit\_11 ;Load the COMBINATION code into register 3 (R1), R3 = 0x400643F

AND R2,R1,#2\_00010000 ;Perform a bitwise OR operation with the contents of R0 with 0x10 and put the contents into R0 , R0 = 0x3

CMP R2,R3 ;Perform the comparesion between the input code and combination code.

BEQ Unlocked\_State ;If R2 = R3, begin function Unlocked\_State

BNE State\_0 ;If R2 != R3, begin function Locked\_State

Unlocked\_State

LDR R5, =GPIO\_PORTN\_DATA\_R ;Load the memory addess of the GPIO Port N DATA Register into register 5 (R5), R5 = 0x400643FC

MOV R4,#0x1 ;Move 0x1 to R4.

STR R4, [R5] ;Stores R4 contents into contents of the address located in R5; GPIO Port N DATA Register now has 0x2 stored in it

ALIGN

END

# Milestone3

; Name: tianze zhang(Dylan zhang)

; Student Number: 400208135

; Lab Section: L01

; Description of Code: using 1000 as combination code to turn on D2 as successful unlock.

; Original: Copyright 2014 by Jonathan W. Valvano, valvano@mail.utexas.edu

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

;ADDRESS DEFINTIONS

;The EQU directive gives a symbolic name to a numeric constant, a register-relative value or a program-relative value

SYSCTL\_RCGCGPIO\_R EQU 0x400FE608 ;General-Purpose Input/Output Run Mode Clock Gating Control Register (RCGCGPIO Register)

GPIO\_PORTN\_DIR\_R EQU 0x40064400 ;GPIO Port N Direction Register address

GPIO\_PORTN\_DEN\_R EQU 0x4006451C ;GPIO Port N Digital Enable Register address

GPIO\_PORTN\_DATA\_R EQU 0x400643FC ;GPIO Port N Data Register address

GPIO\_PORTM\_DIR\_R EQU 0x40063400 ;GPIO Port M Direction Register Address (Fill in these addresses)

GPIO\_PORTM\_DEN\_R EQU 0x4006351C ;GPIO Port M Direction Register Address (Fill in these addresses)

GPIO\_PORTM\_DATA\_R EQU 0x400633FC ;GPIO Port M Data Register Address (Fill in these addresses)

GPIO\_PORTF\_DIR\_R EQU 0x4005D400 ;GPIO Port F Direction Register address

GPIO\_PORTF\_DEN\_R EQU 0x4005D51C ;GPIO Port F Digital Enable Register address

GPIO\_PORTF\_DATA\_R EQU 0x4005D3FC ;GPIO Port F Data Register address

counter EQU 0x8888

COMBINATION EQU 2\_1001 ;the 2\_ here means binary, if we dont want our code to be 1000,we put0001

digit1 EQU 2\_1001

digit2 EQU 2\_1010

digit3 EQU 2\_1100

digit4 EQU 2\_1000

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

;Do not alter this section

AREA |.text|, CODE, READONLY, ALIGN=2 ;code in flash ROM

THUMB ;specifies using Thumb instructions

EXPORT Start

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

;Function PortN\_Init

PortN\_Init

;STEP 1

LDR R1, =SYSCTL\_RCGCGPIO\_R

LDR R0, [R1]

ORR R0,R0, #0x1000

STR R0, [R1]

NOP

NOP

;STEP 5

LDR R1, =GPIO\_PORTN\_DIR\_R

LDR R0, [R1]

ORR R0,R0, #2\_011111

STR R0, [R1]

;STEP 7

LDR R1, =GPIO\_PORTN\_DEN\_R

LDR R0, [R1]

ORR R0, R0, #2\_011111

STR R0, [R1]

BX LR

PortM\_Init

;STEP 1

LDR R1, =SYSCTL\_RCGCGPIO\_R

LDR R0, [R1]

ORR R0,R0, #0x800 ;port M is R11

STR R0, [R1]

NOP

NOP

;STEP 5

LDR R1, =GPIO\_PORTM\_DIR\_R

LDR R0, [R1]

ORR R0,R0, #0xC0

STR R0, [R1]

;STEP 7

LDR R1, =GPIO\_PORTM\_DEN\_R

LDR R0, [R1]

ORR R0, R0, #2\_011111

STR R0, [R1]

BX LR

PortF\_Init

; STEP 1

LDR R1, =SYSCTL\_RCGCGPIO\_R ;Loads the memory address of RCGCGPIO into register 1(R1); R1 = 0x400FE608

LDR R0, [R1] ;Put the contents of the memory address of RCGCGPIO into register 0 (R0), R0 = 0x00000000

ORR R0,R0, #0x20 ;Performs a bitwise OR operation with the contents of R0 and 0x20 and stores it back into R0, R0 = 0x20

STR R0, [R1] ;Stores R0 contents into contents of the address located in R1,RCGCGPIO now has Ox20 stored in it

NOP ;Waiting for GPIO Port F to be enabled

NOP ;Waiting for GPIO Port F to be enabled

; STEP 5

LDR R1, =GPIO\_PORTF\_DIR\_R ;Load the memory address of the GPIO Port F DIR Register into register 1 (R1), R1 = 0x4005D400

LDR R0, [R1] ;Put the contents of the memory address of GPIO Port F DIR Register in R0, R0 = 0x00000000

ORR R0,R0, #2\_011111 ;Perform a bitwise OR operation with the contents of R0 with 0x10 and put the contents into R0 , R0 = 0x10

STR R0, [R1] ;Stores R0 contents into contents of the address located in R1; GPIO Port F Direction Register now has 0x10 stored in it

; STEP 7

LDR R1, =GPIO\_PORTF\_DEN\_R ;Load the memory addess of the GPIO Port F DEN Register into register 1 (R1), R1 = 0x4005D51C

LDR R0, [R1] ;Put the contents of the memory address of GPIO Port F DEN Register in register 0 (R0,), R0 = 0x00000000

ORR R0, R0, #2\_011111 ;Perform a bitwise OR operation with the contents of R0 with 0x10 and put the contents into R0, R0 = 0x10

STR R0, [R1] ;Stores R0 contents into contents of the address located in R1; GPIO Port F DEN Register now has 0x10 stored in it

BX LR

State\_Init

LDR R5,=GPIO\_PORTN\_DATA\_R ;Locked is the Initial State

MOV R4,#2\_0000010 ; the initial state is locked, so D1 ON

STR R4,[R5]

BX LR

IsZero

LDR R0, =GPIO\_PORTM\_DATA\_R

LDR R1, [R0]

AND R2, R1, #2\_00010000

CMP R2, #2\_00000000

BNE IsZero

BX LR

IsOne

LDR R0, =GPIO\_PORTM\_DATA\_R

LDR R1, [R0]

AND R2, R1, #2\_00010000

CMP R2, #2\_00010000

BNE IsOne

BX LR

Start

BL PortN\_Init ;The BL instruction is like a function call

BL PortM\_Init ;The BL instruction is like a function call

BL State\_Init ;The BL instruction is like a function call

State\_0

BL IsZero

BL IsOne

MOV R3, #2\_00010001 ;Load the COMBINATION code into register 3 (R1), R3 = 0x400643FC

AND R2,R1,#2\_00010001 ;Perform a bitwise OR operation with the contents of R0 with 0x10 and put the contents into R0 , R0 = 0x3

CMP R2,R3 ;Perform the comparesion between the input code and combination code.

BEQ State\_1 ;If R2 = R3, begin function Unlocked\_State

BNE State\_0 ;If R2 != R3, begin function Locked\_State

State\_1

LDR R5, =GPIO\_PORTN\_DATA\_R

MOV R4,#0x3

STR R4, [R5]

BL IsZero

BL IsOne

MOV R3, #2\_00010000 ;LDR R3, = Digit\_11 ;Load the COMBINATION code into register 3 (R1), R3 = 0x400643FC

AND R2,R1,#2\_00010000 ;Perform a bitwise OR operation with the contents of R0 with 0x10 and put the contents into R0 , R0 = 0x3

CMP R2,R3 ;Perform the comparesion between the input code and combination code.

BEQ State\_2 ;If R2 = R3, begin function Unlocked\_State

BNE State\_0 ;If R2 != R3, begin function Locked\_State

State\_2

LDR R5, =GPIO\_PORTN\_DATA\_R

MOV R4,#0x1

STR R4, [R5]

BL IsZero

BL IsOne

MOV R3, #2\_00010000 ;LDR R3, = Digit\_11 ;Load the COMBINATION code into register 3 (R1), R3 = 0x400643F

AND R2,R1,#2\_00010000 ;Perform a bitwise OR operation with the contents of R0 with 0x10 and put the contents into R0 , R0 = 0x3

CMP R2,R3 ;Perform the comparesion between the input code and combination code.

BEQ State\_3 ;If R2 = R3, begin function Unlocked\_State

;BEQ state\_3\_LightOn

BNE State\_0 ;If R2 != R3, begin function Locked\_State

State\_3

;LDR R5, =GPIO\_PORTF\_DATA\_R ;Load the memory addess of the GPIO Port F DATA Register into register 1 (R1), R1 = 0x4005D3FC

;Put the contents of the memory address of GPIO Port F DATA Register 0 (R0), R0 = 0x00000000

;MOV R4, #0x1 ;Perform a bitwise OR operation with the contents of R0 with 0x10 and put the contents into R0, R0 = 0x10

;STR R4, [R5]

LDR R5, =GPIO\_PORTN\_DATA\_R

MOV R4,#0x3

STR R4, [R5]

BL IsZero

BL IsOne

MOV R3, #2\_00010000 ;LDR R3, = Digit\_11 ;Load the COMBINATION code into register 3 (R1), R3 = 0x400643F

AND R2,R1,#2\_00010000 ;Perform a bitwise OR operation with the contents of R0 with 0x10 and put the contents into R0 , R0 = 0x3

CMP R2,R3 ;Perform the comparesion between the input code and combination code.

BEQ Unlocked\_State ;If R2 = R3, begin function Unlocked\_State

BNE State\_0 ;If R2 != R3, begin function Locked\_State

Unlocked\_State

LDR R5, =GPIO\_PORTN\_DATA\_R ;Load the memory addess of the GPIO Port N DATA Register into register 5 (R5), R5 = 0x400643FC

MOV R4,#0x1 ;Move 0x1 to R4.

STR R4, [R5] ;Stores R4 contents into contents of the address located in R5; GPIO Port N DATA Register now has 0x2 stored in it

state\_2\_LightOn

BL PortF\_Init ;The BL instruction is like a function call

;STEP 8

LDR R1, =GPIO\_PORTF\_DATA\_R ;Load the memory addess of the GPIO Port F DATA Register into register 1 (R1), R1 = 0x4005D3FC

LDR R0,[R1] ;Put the contents of the memory address of GPIO Port F DATA Register 0 (R0), R0 = 0x00000000

ORR R0,R0, #0x1 ;Perform a bitwise OR operation with the contents of R0 with 0x10 and put the contents into R0, R0 = 0x10

STR R0, [R1]

state\_3\_LightOn

BL PortF\_Init ;The BL instruction is like a function call

;STEP 8

LDR R1, =GPIO\_PORTF\_DATA\_R ;Load the memory addess of the GPIO Port F DATA Register into register 1 (R1), R1 = 0x4005D3FC

LDR R0,[R1] ;Put the contents of the memory address of GPIO Port F DATA Register 0 (R0), R0 = 0x00000000

ORR R0,R0, #0x10 ;Perform a bitwise OR operation with the contents of R0 with 0x10 and put the contents into R0, R0 = 0x10

STR R0, [R1]

ALIGN

END

# Video Demo: