**GPIOs Related Programs**

**;ARM ALP to display sum on port0**

area reset,data,readonly

export \_\_Vectors

\_\_Vectors

dcd 0

dcd Reset\_Handler

area mycode,code,readonly

FIO0DIR equ 0x2009c000

FIO0MASK equ 0x2009c010

FIO0PIN equ 0x2009c014

FIO0SET equ 0x2009c018

FIO0CLR equ 0x2009c01c

entry

export Reset\_Handler

Reset\_Handler

ldr r0,=0x12345678

ldr r1,=0x11111111

adds r2,r0,r1 ; r2 = r0 + r1

rev r3,r2

ldr r4,=sum

str r3,[r4]

ldr r5,=FIO0DIR

ldr r6,=0xffffffff ; port0 is configured as o/p port

str r6,[r5]

ldr r7,=FIO0PIN

str r2,[r7] ; send sum to port0 through FIOPIN register

stop b stop

area mydata,data,readwrite

sum space 0

end

**;ARM ALP to demonstrate the led blink operation**

area reset,data,readonly

export \_\_Vectors

\_\_Vectors

dcd 0

dcd Reset\_Handler

area ports,code,readonly

FIO0DIR EQU 0x2009c000

FIO0MASK EQU 0x2009c010

FIO0PIN EQU 0x2009c014

FIO0SET EQU 0x2009c018

FIO0CLR EQU 0x2009c01c

entry

export Reset\_Handler

Reset\_Handler

start ldr r0,=FIO0DIR

ldr r1,=0xffffffff

str r1,[r0]

ldr r2,=FIO0SET

str r1,[r2]

bl delay

ldr r3,=FIO0CLR

str r1,[r3]

bl delay

b start

stop b stop

delay ldr r4,=0x2ffff

loop subs r4,r4,#1

bne loop

bx lr

end

**;ARM ALP to demonstrate bit led blink operation**

area reset,data,readonly

export \_\_Vectors

\_\_Vectors

dcd 0

dcd Reset\_Handler

area ports,code,readonly

FIO0DIR EQU 0x2009c000

FIO0MASK EQU 0x2009c010

FIO0PIN EQU 0x2009c014

FIO0SET EQU 0x2009c018

FIO0CLR EQU 0x2009c01c

entry

export Reset\_Handler

Reset\_Handler

ldr r0,=FIO0DIR

orr r1,#1<<31

str r1,[r0]

ldr r2,=FIO0PIN

start eor r3,#1<<31

str r3,[r2]

bl delay

b start

stop b stop

delay ldr r4,=0x2ffff

loop subs r4,r4,#1

bne loop

bx lr

end

**;ARM ALP to demonstrate the alternate led blink operation**

area reset,data,readonly

export \_\_Vectors

\_\_Vectors

dcd 0

dcd Reset\_Handler

area ports,code,readonly

FIO0DIR EQU 0x2009c000

FIO0MASK EQU 0x2009c010

FIO0PIN EQU 0x2009c014

FIO0SET EQU 0x2009c018

FIO0CLR EQU 0x2009c01c

entry

export Reset\_Handler

Reset\_Handler

start ldr r0,=FIO0DIR

ldr r1,=0xffffffff

str r1,[r0]

ldr r2,=FIO0PIN

ldr r5,=0xaaaaaaaa

str r5,[r2]

bl delay

ldr r6,=0x55555555

str r6,[r2]

bl delay

b start

stop b stop

delay ldr r4,=0x2ffff

loop subs r4,r4,#1

bne loop

bx lr

end

**;ARM ALP to demonstrate led walking operation**

area reset,data,readonly

export \_\_Vectors

\_\_Vectors

dcd 0

dcd Reset\_Handler

area ports,code,readonly

FIO0DIR EQU 0x2009c000

FIO0MASK EQU 0x2009c010

FIO0PIN EQU 0x2009c014

FIO0SET EQU 0x2009c018

FIO0CLR EQU 0x2009c01c

entry

export Reset\_Handler

Reset\_Handler

ldr r0,=FIO0DIR

ldr r1,=0xffffffff

str r1,[r0]

start2 ldr r2,=FIO0PIN

ldr r3,=0x00000001

start str r3,[r2]

bl delay

lsl r3,r3,#1

cmp r3,#0x80000000

bne start

start1 str r3,[r2]

bl delay

lsr r3,r3,#1

cmp r3,#0x00000001

bne start1

b start2

stop b stop

delay ldr r4,=0x2ffff

loop subs r4,r4,#1

bne loop

bx lr

end

**;ARM ALP to demonstrate ring operation**

area reset,data,readonly

export \_\_Vectors

\_\_Vectors

dcd 0

dcd Reset\_Handler

area ports,code,readonly

FIO0DIR EQU 0x2009c000

FIO0MASK EQU 0x2009c010

FIO0PIN EQU 0x2009c014

FIO0SET EQU 0x2009c018

FIO0CLR EQU 0x2009c01c

entry

export Reset\_Handler

Reset\_Handler ;proc

ldr r0,=FIO0DIR

ldr r1,=0xffffffff

str r1,[r0]

ldr r2,=FIO0PIN

start1 ldr r3,=0x00000001

start str r3,[r2]

bl delay

lsl r3,r3,#1

cmp r3,#0

bne start

b start1

stop b stop

delay ldr r4,=0x2ffff

loop subs r4,r4,#1

bne loop

bx lr

end

**;ARM ALP to demonstrate twisted ring operation**

area reset,data,readonly

export \_\_Vectors

\_\_Vectors

dcd 0

dcd Reset\_Handler

area ports,code,readonly

FIO0DIR EQU 0x2009c000

FIO0MASK EQU 0x2009c010

FIO0PIN EQU 0x2009c014

FIO0SET EQU 0x2009c018

FIO0CLR EQU 0x2009c01c

entry

export Reset\_Handler

Reset\_Handler

ldr r0,=FIO0DIR

ldr r1,=0xffffffff

str r1,[r0]

ldr r2,=FIO0PIN

ldr r3,=0x80000000

start str r3,[r2]

bl delay

eor r3,r3,#0x00000001

ror r3,r3,#1

b start

stop b stop

delay ldr r4,=0x2ffff

loop subs r4,r4,#1

bne loop

bx lr

end

**;ARM ALP to demonstrate switch status makes LED blinking**

area reset,data,readonly

export \_\_Vectors

\_\_Vectors

dcd 0

dcd Reset\_Handler

area ports,code,readonly

FIO0DIR EQU 0x2009c000

FIO0MASK EQU 0x2009c010

FIO0PIN EQU 0x2009c014

FIO0SET EQU 0x2009c018

FIO0CLR EQU 0x2009c01c

FIO1DIR EQU 0x2009c020

FIO1MASK EQU 0x2009c030

FIO1PIN EQU 0x2009c034

FIO1SET EQU 0x2009c038

FIO1CLR EQU 0x2009c03c

entry

export Reset\_Handler

Reset\_Handler

start

ldr r0,=FIO0DIR

ldr r1,=0xffffffff

str r1,[r0]

ldr r2,=FIO1DIR

ldr r3,=0x00000000

str r3,[r2]

ldr r4,=FIO1PIN

ldr r5,=0x00000000

str r5,[r4]

ldr r6,[r4]

ldr r7,=0x80000000

cmp r6,r7

bne stop

ldr r8,=0xaaaaaaaa

stop ldr r9,=FIO0SET

str r8,[r9]

bl delay

ldr r10,=FIO0CLR

str r8,[r10]

bl delay

b start

delay ldr r11,=0x2ffff

loop subs r11,r11,#1

bne loop

bx lr

end

**;ARM ALP to demonstrate bit led blinking when bit switch is open/close**

area reset,data,readonly

export \_\_Vectors

\_\_Vectors

dcd 0

dcd Reset\_Handler

area ports,code,readonly

FIO0DIR EQU 0x2009c000

FIO0MASK EQU 0x2009c010

FIO0PIN EQU 0x2009c014

FIO0SET EQU 0x2009c018

FIO0CLR EQU 0x2009c01c

FIO1DIR EQU 0x2009c020

FIO1MASK EQU 0x2009c030

FIO1PIN EQU 0x2009c034

FIO1SET EQU 0x2009c038

FIO1CLR EQU 0x2009c03c

entry

export Reset\_Handler

Reset\_Handler ;proc

start

ldr r0,=FIO0DIR

ldr r1,=0x80000001

str r1,[r0]

ldr r2,=FIO1DIR

and r3,#1<<31

str r3,[r2]

start2 ldr r4,=FIO1PIN

and r5,#1<<31

str r5,[r4]

ldr r6,[r4]

;and r7,#1<<31

cmp r6,#1<<31

bhi stop

ldr r8,=FIO0PIN

eor r9,#1<<31

str r9,[r8]

bl delay

b start2

stop ldr r8,=FIO0PIN

eor r10,#1<<0

str r10,[r8]

bl delay

b start2

delay ldr r11,=0x2ffff

loop subs r11,r11,#1

bne loop

bx lr end