

A sourcecode for doing a CIA IRQ Timer A interrupt Example: If you want to play a music file with init at \$1000 and play at \$1003 Add jsr \$1000 at "Initcode" comment and jsr \$1003 at "add code here"

```

;Cia Timer B interrupt example by Terric(Anders L) in 2011
;This example shows a stable one frame CIA Timer B interrupt.
;This piece of code fires of at same cycle on PAL, NTSC, NTSCOLD, DREAN
;Revision 4, Compiled with 64tass 1.46

    *=$0801
    .byte $0a,$08,$0a,$00,$9e,$32,$30,$36,$32
    .byte $00,$00,$00,$00 ; a BASIC SYS LINE FOR STARTING CODE AT $080E
    *=$080e
    sei

    jsr $e544    ;Clear screen with a kernel call

    lda #$35
    sta $01
;cia detection routine begins, do not touch if you want cycle perfect
timing.
    lda #$00
    sta $d011
    sta $d015
    lda #<fdcia
    sta $fffa
    lda #>fdcia
    sta $ffffb
    lda #4
    sta $dd04
    lda #$00
    sta $dd05
    sta $40
    lda #$81
    sta $dd0d
    lda $dd0e
    ora #%00011001
    sta $dd0e
    lda $dd0d
    lda $dd0d
    inc $40
    jmp *
fdcia
    lda $dd0d
    pla
    pla
    pla
    sei
    lda #<fdcia2
    sta $fffe

```

```
    lda #>fdcia2
    sta $ffff
    lda #4
    sta $dc04
    lda #$00
    sta $dc05
    sta $44
    lda #$7f
    sta $dc0d
    lda #$81
    sta $dc0d
    lda $dc0e
    ora #%00011001
    sta $dc0e
    lda $dc0d
    cli
    nop
    inc $44
    jmp *
fdcia2
    lda $dc0d
    pla
    pla
    pla

; Determine PAL/NTSC/NTSCOLD and store in $42, 1 ntscold, 3 pal , 6 ntsc, 7
drear
-   lda $d011
    bpl -
-   lda $d012
-   cmp $d012
    beq -
    bmi --
    and #$03
    sta $42
    lsr a
    sta $41
; Is it the DREAN
    lda #$ff
    sta $dc04
    sta $dc05
-   lda $d011
    bpl -
-   lda $d011
    bmi -
    lda $dc0e
    ora #%00011001
    sta $dc0e
    lda #$f0
-   cmp $d012
    bne -
```

```

    lda $dc05
    cmp #$c4
    beq +
    bcs +
; Yes, it is the drean
    lda #$04
    clc
    adc $42
    sta $42
+
here
    sei
    lda #$35
    sta $01

    lda $dc0e
    and #$fe
    sta $dc0e
    lda $dc0f
    and #$fe
    sta $dc0f
    lda $dd0e
    and #$fe
    sta $dd0e
    lda $dd0f
    and #$fe
    sta $dd0f
    lda $dc0d
    lda $dd0d
    lda #$37    ; Select big bank without roms
    sta $01    ; Change to big bank

    ldx $42
    lda timerlo,x    ; cia (6526) 63 cycles per line * 312 lines - 1
    sta $04
    clc
    sbc #44    ; 28 +16 cycles when the sprites are enabled
    sta gnurka+1
    lda timerhi,x
    sta $05
; beginning of stabirq
    cli
    lda #$60
-   cmp $d012
    bne -
-   lda $d011
    bpl -
    ldx #15
-   dex
    bne -
-   lda $d011

```

```

bmi -
sei
lda #$00
sta $d011

lda #<stabirq
sta $0314
lda #>stabirq
sta $0315
lda #$02
sta $d012
sta $1f
lda $d011
and #$7f
sta $d011
lda #$01
sta $d01a
sta $d019
cli
ldx #$04
- dex
bne -
  .for a=0,a<40,a=a+1
  nop
  .next

stabirq
  lda $42      ;3 cycles
  cmp #$03    ;2 cycles
  beq pal     ; on jmp 3 cycles else 2 cycles
  cmp #$06    ; 2
  beq ntsc   ; 2 or 3
  cmp #$07    ; 2
  bne ntscold ; 2 or 3
  nop        ; 2
  jmp drean  ; 3
pal nop     ; 2

ntsc  nop
      lda $1f      ; 3
ntscold
      lda $1f      ;3
drean
      lda $1f      ;3-2
      cmp $d012
      beq +
+
      lda $04
      sta $dc06
      lda $05
      sta $dc07

```

```

lda $dc0f
ora #%00011001
and #%11110111
sta $dc0f ; count o2 pulses and forceload and continious and start
lda #$01
sta $d019
pla
pla
pla
lda #$ff
txs
lda #$35
sta $01
lda #<irq
sta $fffe ; Set low IRQ adress in selected bank
lda #>irq
sta $ffff ; Set high IRQ adress in selected bank
lda #$7f
sta $dc0d ; Clear IRQ interruptmask for CIA
lda #$00
sta $d01a ; Clear VIC interruptmask
lda $d019
and #$81
sta $d019
and #0
sta $d019
lda $d011
and #$7f
sta $d011
lda #$1b
sta $d011
lda #$82
sta $dc0d ; Activate CIA timer B irq
lda #$7f
sta $dd0d
;INITCODE
;You could add init for music here or at the beginning
cli
jmp * ; do nothing while no IRQ, just jump to this line
irq
pha ; 3 cycles
lda $dc0d ; 4 cycles ;Acknowledge CIA timer a interrupt

tya ; 2 cycles
pha ; 4 cycles
txa ; 2 cycles
pha ; 4 cycles
lda $dc06; 4 cycles
clc
gnurka sbc #$c7;set according to system, see above when setting timers
clc

```

```
    eor #$ff
    adc #$01
    lsr a
    bcc +
+
    clc
    sta fakejmp+1
fakejmp bcc fakelocation
fakelocation          ; and accu=7 wait 0 cycles etc.
    .for a=0,a<128,a=a+1
    nop
    .next
    ; Here you can add code
    inc $d020

    dec $d020

    ; End of adding code
    pla
    tax
    pla
    tay
    pla
    rti

timerlo .byte 0,$7f,0,$c7,0,0,$c6,$37
timerhi .byte 0,$41,0,$4c,0,0,$42,$4f
```

From: <https://codebase64.org/> - **Codebase 64 wiki**

Permanent link: <https://codebase64.org/doku.php?id=base:timerinterrupts>

Last update: **2015-04-17 04:34**

