EXAMPLE 8

Description:
Using interrupt functions.

Notes:

GCC syntax for interrupt function is very different from IAR.
GCC interrupt function can be refenced by other functions - even Thumb. That is very dangerous.

IAR relies on __irq and __fiq extended keywords. It also requires to use __arm extended keyword
for Thumb code. An interrupt function cannot call Thumb function directly.

Danger warning for IAR code: LOW

GCC code example:

```
void IRQ_Routine (void)    __attribute__ ((interrupt("IRQ")));
void Spurious_IRQ_Routine (void)    __attribute__ ((naked));
void FIQ_Routine (void)    __attribute__ ((interrupt("FIQ")));
void UNDEF_Routine (void) __attribute__ ((interrupt("UNDEF")));

void SendInfo(char *Ptr)
{
  int Len;

  while (*Ptr++){}
}

void IRQ_Routine (void) {
  SendInfo("IRQ");
}

void FIQ_Routine (void)  {
  SendInfo("FIQ");
}

void UNDEF_Routine (void) {
  SendInfo("UNDEF");
}
```

GCC assembler output:

```
          .align    2
          .global   SendInfo
          .type     SendInfo,function
SendInfo:
          @ args = 0, pretend = 0, frame = 0
          @ frame_needed = 0, current_function_anonymous_args = 0
          @ link register save eliminated.
.L17:
          ldrb      r3, [r0], #1
          cmp       r3, #0
          bne       .L17
          mov       pc, lr
.Lfe6:
          .size     SendInfo,.Lfe6-SendInfo
          .section .rodata
          .align    2
```

```
.LC0:
            .ascii      "IRQ\000"
            .text
            .align      2
            .global     IRQ_Routine
            .type       IRQ_Routine,function
IRQ_Routine:
            @ Interrupt Service Routine.
            @ args = 0, pretend = 0, frame = 0
            @ frame_needed = 1, current_function_anonymous_args = 0
            mov         ip, sp
            stmfd       sp!, {r0, fp, ip, lr, pc}
            sub         fp, ip, #4
            ldr         r0, .L19
            bl          SendInfo
            ldmea       fp, {r0, fp, sp, lr}
            subs        pc, lr, #4
.L20:
            .align      2
.L19:
            .word       .LC0
.Lfe7:
            .size       IRQ_Routine,.Lfe7-IRQ_Routine
            .section .rodata
            .align      2
.LC1:
            .ascii      "FIQ\000"
            .text
            .align      2
            .global     FIQ_Routine
            .type       FIQ_Routine,function
FIQ_Routine:
            @ Fast Interrupt Service Routine.
            @ args = 0, pretend = 0, frame = 0
            @ frame_needed = 1, current_function_anonymous_args = 0
            mov         ip, sp
            stmfd       sp!, {fp, ip, lr, pc}
            sub         fp, ip, #4
            ldr         r0, .L22
            bl          SendInfo
            ldmea       fp, {fp, sp, lr}
            subs        pc, lr, #4
.L23:
            .align      2
.L22:
            .word       .LC1
.Lfe8:
            .size       FIQ_Routine,.Lfe8-FIQ_Routine
            .section .rodata
            .align      2
.LC2:
            .ascii      "UNDEF\000"
            .text
            .align      2
            .global     UNDEF_Routine
            .type       UNDEF_Routine,function
UNDEF_Routine:
            @ ARM Exception Handler.
            @ args = 0, pretend = 0, frame = 0
            @ frame_needed = 1, current_function_anonymous_args = 0
            mov         ip, sp
            stmfd       sp!, {fp, ip, lr, pc}
            sub         fp, ip, #4
            ldr         r0, .L25
            bl          SendInfo
            ldmea       fp, {fp, sp, lr}
            movs        pc, lr
```

```
.L26:
            .align     2
.L25:
            .word      .LC2
.Lfe9:
            .size      UNDEF_Routine,.Lfe9-UNDEF_Routine
```

Equivalent IAR C code:

```
__irq __arm void IRQ_Routine (void);
__irq __arm void Spurious_IRQ_Routine (void);
__fiq __arm void FIQ_Routine (void);
__irq __arm void UNDEF_Routine (void);

__arm void SendInfo(char *Ptr)
{
            while (*Ptr++){}
}

__irq __arm void IRQ_Routine (void) {
    SendInfo("IRQ");
}

__fiq __arm void FIQ_Routine (void)  {
    SendInfo("FIQ");
}

__irq __arm void UNDEF_Routine (void) {
    SendInfo("UNDEF");
}
```

IAR assembler output:

```
    134           __irq __arm void IRQ_Routine (void);
    135           __irq __arm void Spurious_IRQ_Routine (void);
    136           __fiq __arm void FIQ_Routine (void);
    137           __irq __arm void UNDEF_Routine (void);
    138

  \                                 In section .text, align 4, keep-with-next
    139           __arm void SendInfo(char *Ptr)
    140           {
    141                 while (*Ptr++){}
  \                     SendInfo:
  \                     ??SendInfo_0:
  \   00000000   0110D0E4          LDRB      R1,[R0], #+1
  \   00000004   000051E3          CMP       R1,#+0
  \   00000008   FCFFFF1A          BNE       ??SendInfo_0
    142           }
  \   0000000C   1EFF2FE1          BX        LR                ;; return
    143

  \                                 In section .text, align 4, keep-with-next
    144           __irq __arm void IRQ_Routine (void) {
  \                     IRQ_Routine:
  \   00000000   04E04EE2          SUB       LR,LR,#+4
  \   00000004   0F502DE9          PUSH      {R0-R3,R12,LR}
    145                 SendInfo("IRQ");
  \   00000008   00008FE2          ADR       R0,??IRQ_Routine_0  ;; "IRQ"
  \   0000000C   ........          B         ?Subroutine0
  \                     ??IRQ_Routine_0:
  \   00000010   49525100          DC8       "IRQ"
    146           }
```

```
\                                 In section .text, align 4, keep-with-next
\                    ?Subroutine0:
\    00000000   ........          BL      SendInfo
\    00000004   0F90FDE8          LDM     SP!,{R0-R3,R12,PC}^  ;; return
\                    ??Subroutine0_0:
\    00000008   ........          DC32    `?<Constant "UNDEF">`
 147

\                                 In section .text, align 4, keep-with-next
 148          __fiq __arm void FIQ_Routine (void)  {
\                    FIQ_Routine:
\    00000000   04E04EE2          SUB     LR,LR,#+4
\    00000004   0F402DE9          PUSH    {R0-R3,LR}
\    00000008   04D04DE2          SUB     SP,SP,#+4
 149            SendInfo("FIQ");
\    0000000C   08008FE2          ADR     R0,??FIQ_Routine_0  ;; "FIQ"
\    00000010   ........          BL      SendInfo
 150          }
\    00000014   04D08DE2          ADD     SP,SP,#+4          ;; stack cleaning
\    00000018   0F80FDE8          LDM     SP!,{R0-R3,PC}^  ;; return
\                    ??FIQ_Routine_0:
\    0000001C   46495100          DC8     "FIQ"
 151

\                                 In section .text, align 4, keep-with-next
 152          __irq __arm void UNDEF_Routine (void) {
\                    UNDEF_Routine:
\    00000000   04E04EE2          SUB     LR,LR,#+4
\    00000004   0F502DE9          PUSH    {R0-R3,R12,LR}
 153            SendInfo("UNDEF");
\    00000008   ........          LDR     R0,??Subroutine0_0  ;; `?<Constant
"UNDEF">`
\    0000000C                     REQUIRE ?Subroutine0
\    0000000C                     ;; // Fall through to label ?Subroutine0
 154          }

\                                 In section .rodata, align 4
\    00000000   49525100          DC8 "IRQ"

\                                 In section .rodata, align 4
\    00000000   46495100          DC8 "FIQ"

\                                 In section .rodata, align 4
\                    `?<Constant "UNDEF">`:
\    00000000   554E44454600      DC8 "UNDEF"
\    00000006   0000              DC8 0, 0
```