Fpt Software

Macro and Bit, Byte Operations

LanBT/MinhNQ2

Objectives

- Macro
- Bit Operations
- Quiz

Macro

- Macro definition
- Object-like Macros
- Function-like Macros
- Stringification and Concatenation
- Undefining and Redefining Macros

Macro Definition

What is macro

A *macro* is a fragment of code which has been given a name. Whenever the name is used, it is replaced by the contents of the macro

Macro is defined using #define preprocessor directive in the C language.

When to use

When creating constants that represent numbers, strings or expressions.

Macro classification

- Predefined macro
- User-defined macro

Object-like macros

#define MACRO_NAME macro's body

- Give symbolic names to numeric constants
- The macro's body end at the end of the #define line
- Single line macro:

```
#define SIZE 10
```

- Multiple line macro:

```
#define NUMBERS        1, \
                       2
```

Function-like macros

#define macro_name(list of parameters) macro's body

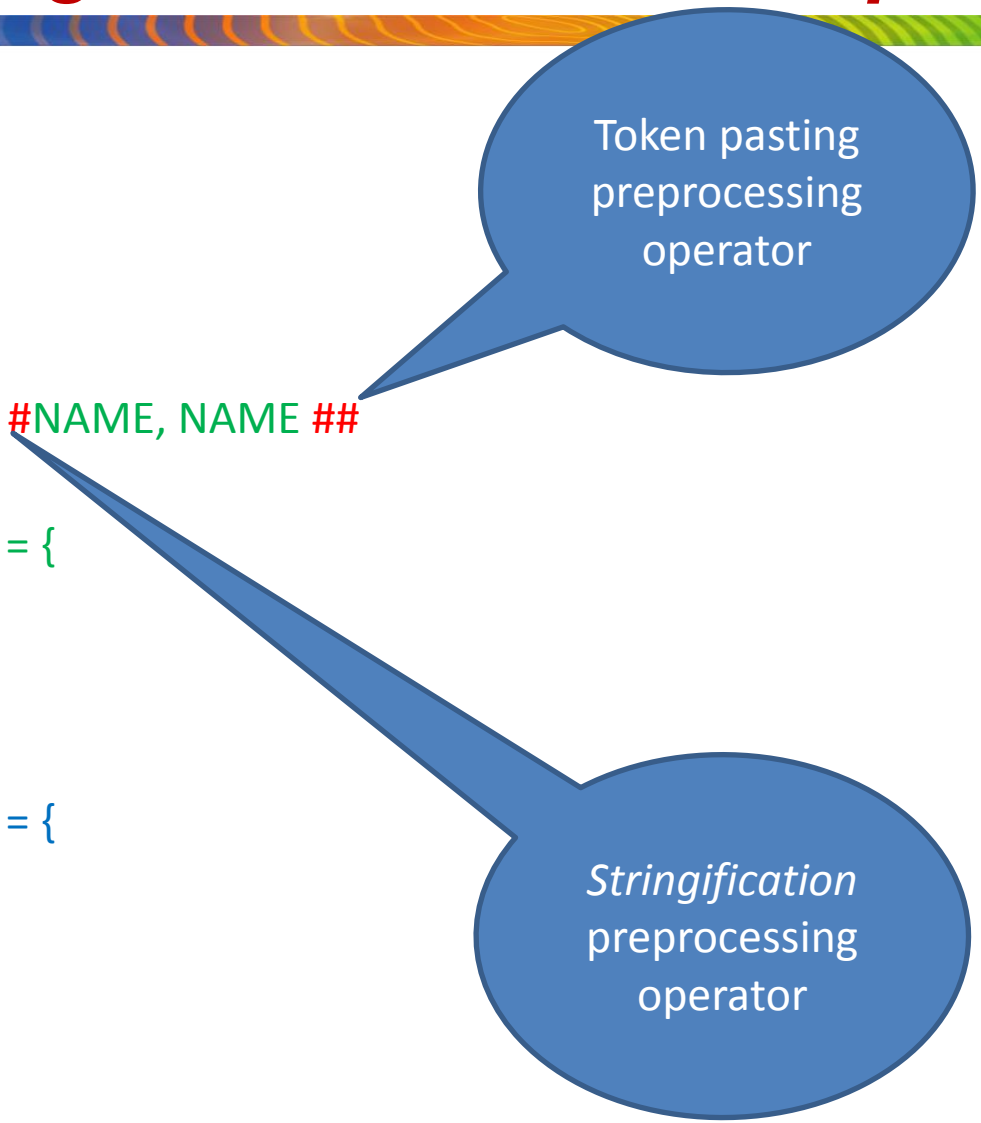

```
#define min(X, Y) ((X) < (Y) ? (X) : (Y))
x = min(a, b); ==> x = ((a) < (b) ? (a) : (b));
y = min(1, 2); ==> y = ((1) < (2) ? (1) : (2));
```

?

```
extern void foo(void);
#define foo() /* optimized inline version */
#define f           ()      callback()
...
foo();  → ?
funcptr = foo;  → ?
f() → ?
```

Stringification and *token pasting*

```
struct command {
    char *name;
    void (*function) (void);
};
#define COMMAND(NAME)  { #NAME, NAME ##
_command }
struct command commands[] = {
    COMMAND (quit),
    COMMAND (help),
};
struct command commands[] = {
    { "quit", quit_command },
    { "help", help_command },
};
```



Undefining and Redefining Macros

```
#ifdef  TRUE
#undef  TRUE
#define TRUE 1
#endif
```

BITWISE OPERATION (1)

Symbol	Operation
&	bitwise AND
|	bitwise inclusive OR
^	bitwise XOR (eXclusive OR)
<<	Left shift
>>	Right shift
~	bitwise NOT (one's complement) (unary)

bit a	bit b	a & b	a | b	a ^ b
0	0	0	0	0
0	1	0	1	1
1	0	0	1	1
1	1	1	1	0

bit a	~a
0	1
1	0

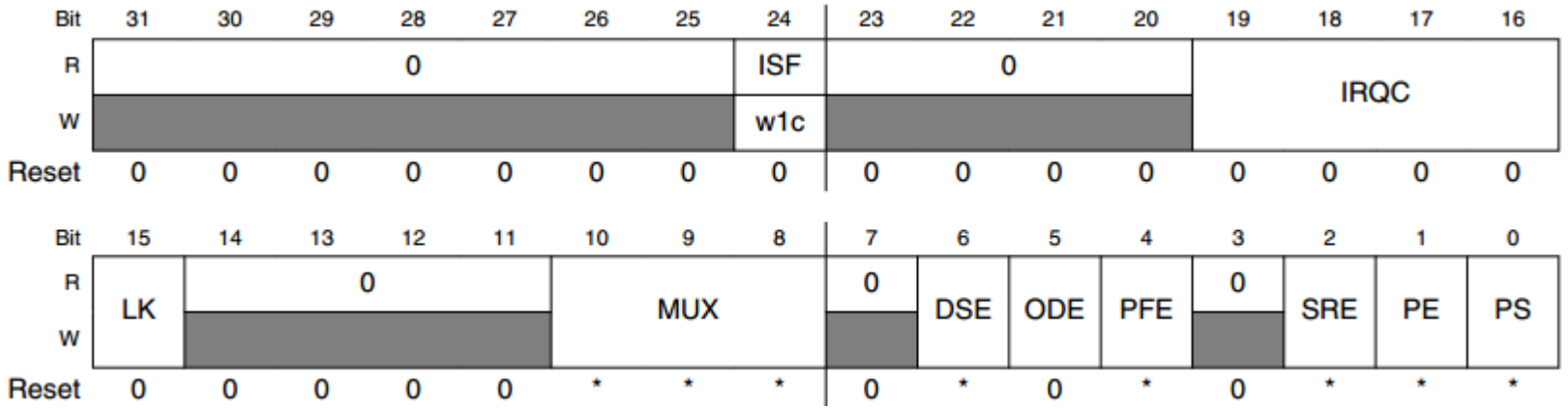
BITWISE OPERATION (2)

Shift operations

A <<3

A >> 3

Quiz(1)



Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0							ISF	0				IRQC			
W								w1c								
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	LK	0				MUX			0	DSE	ODE	PFE	0	SRE	PE	PS
W																
Reset	0	0	0	0	0	*	*	*	0	*	0	*	0	*	*	*

The figure show the description of register PCR.

1. Write macros to define MASK and SHIFT location of each bit field.
2. Write macro to set IRQC to 3

Quiz(2)

- Write macro to convert 32bit value from big endian to little endian form

Thank you!