

Битовый процессор

i8051

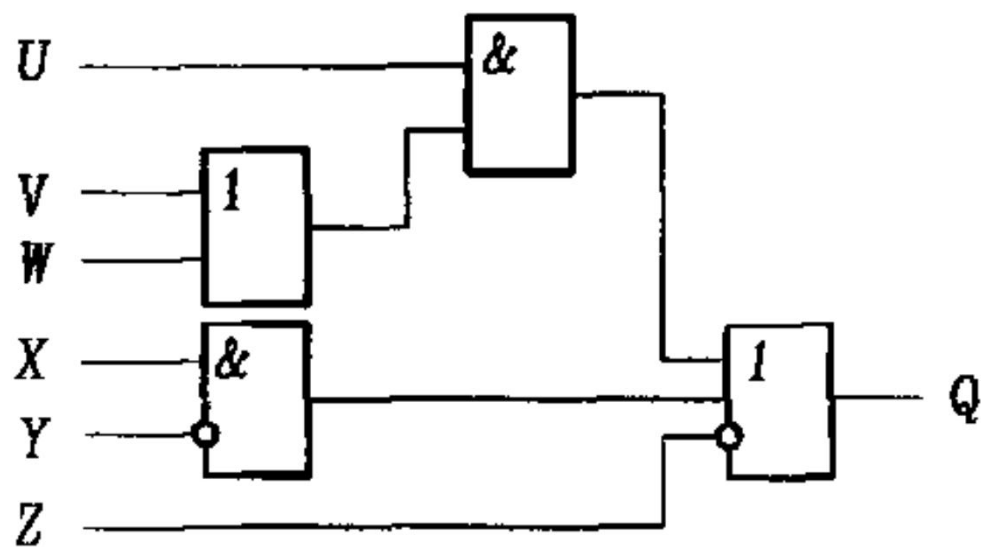
(примеры)

Битовый процессор

- 1 Аккумулятор битовый – бит переноса (C)**
- 2 Набор битовых команд микроконтроллера →**
- 3 Побитовоадресуемая область памяти данных**
(20h – 2Fh)
- 4 Побитовоадресуемые регистры специальных функций (SFR)**
- 5 Битовые адреса областей RAM и SFR**
(20H.1, 01h, -); (D0H.1, D1H, PSW.1, F1).

CLR	C	clear bit to zero	1	12
CLR	bit		2	12
SETB	C	set bit to one	1	12
SETB	bit		2	12
CPL	C	complement bit	1	12
CPL	bit		2	12
ANL	C,bit	AND bit with C	2	24
ANL	C,/bit	...NOTbit with C	2	24
ORL	C,bit	OR bit with C	2	24
ORL	C,/bit	...NOTbit with C	2	24
MOV	C,bit	move bit to bit	2	12
MOV	bit,C		2	24
JC	rel	jump if C set	2	24
JNC	rel	jmp if C not set	2	24
JB	bit,rel	jump if bit set	3	24
JNB	bit,rel	jmp if bit not set	3	24
JBC	bit, rel	jmp&clear if set	3	24

Битовый процессор



$$Q = (U * (V + W)) + (X * \bar{Y}) + \bar{Z}$$

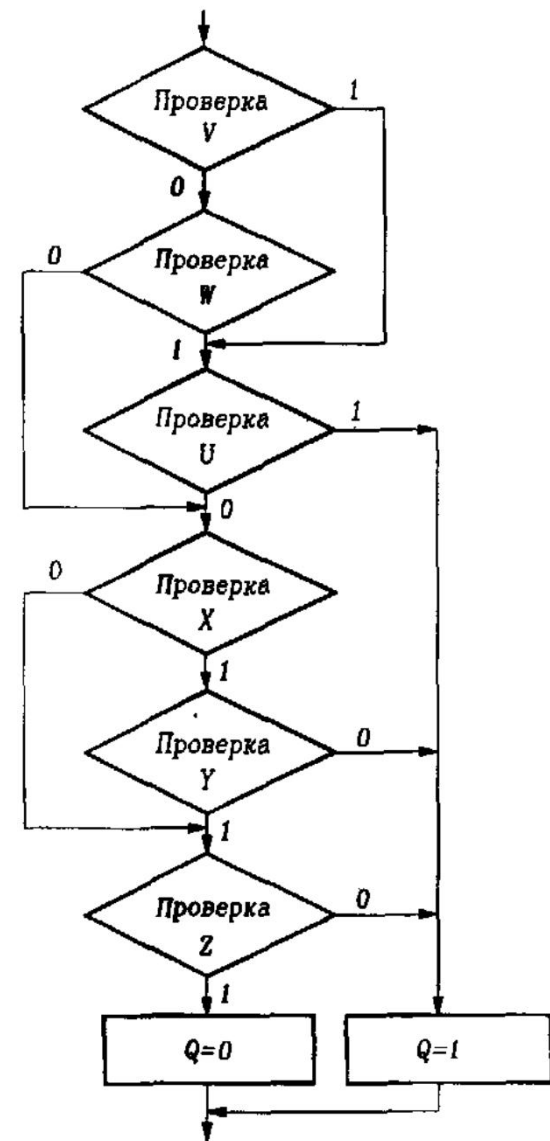
Битовый процессор

$$Q = (U * (V + W)) + (X * \bar{Y}) + \bar{Z}$$

```

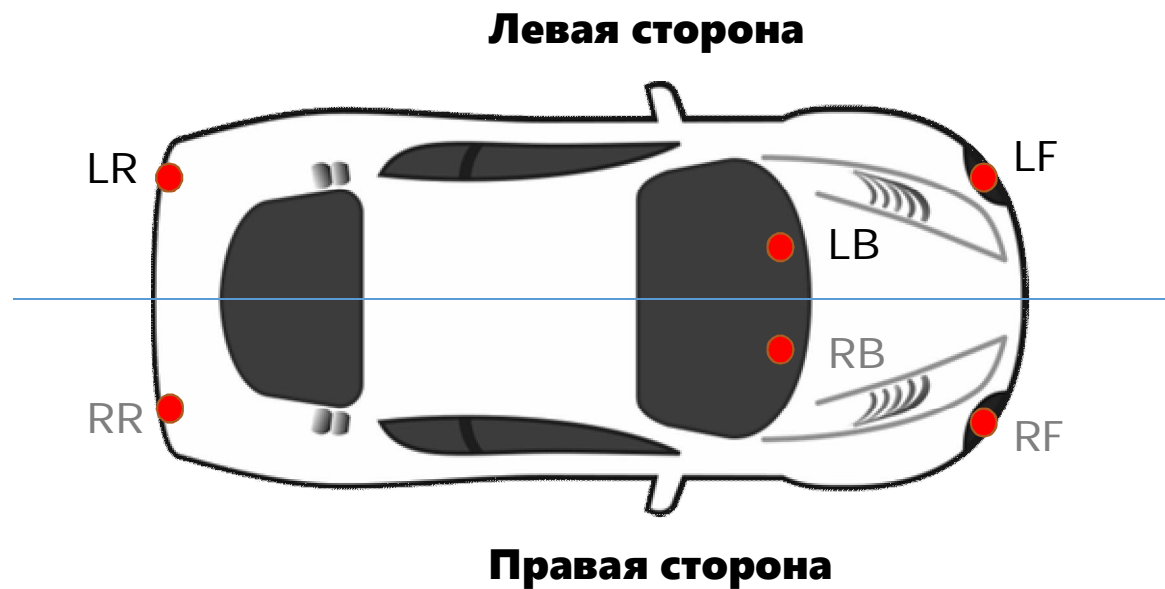
U      BIT    P1.1
V      BIT    P2.2
W      BIT    TF0
X      BIT    IE1
Y      BIT    20H.0
Z      BIT    21H.1
Q      BIT    P3.3
;
TEST_V: JB     V, TEST_U
        JNB    W, TEST_X
TEST_U: JB     U, SET_Q
TEST_X: JNB    X, TEST_Z
        JNB    Y, SET_Q
TEST_Z: JNB    Z, SET_Q
CLR_Q:  CLR    Q
        JMP    NXTTST
SET_Q:  SETB   Q
NXTTST: ...
    
```

; продолжение программы



Битовый процессор

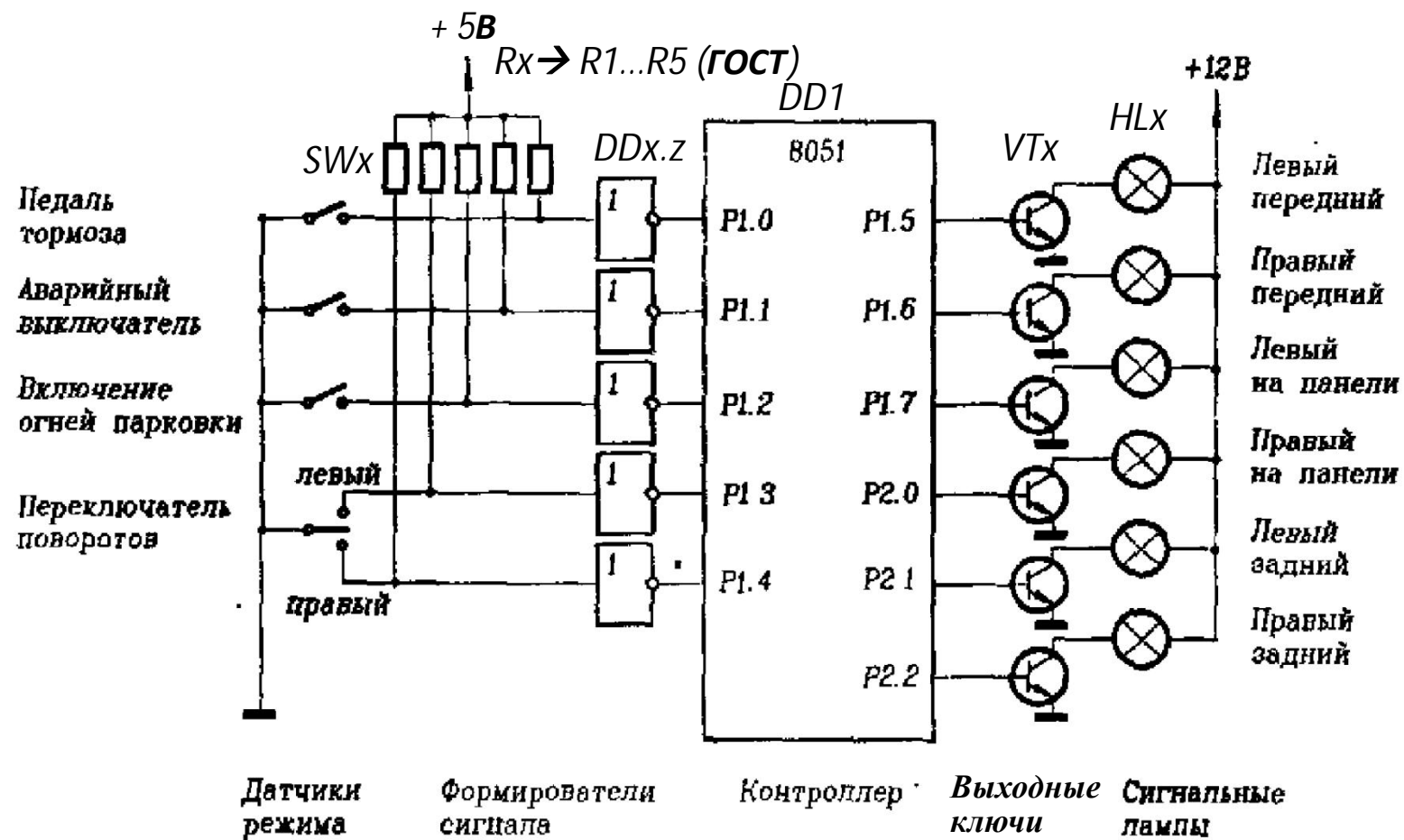
Управление сигналами автомобиля



Битовый процессор

Входные сигналы				Выходные сигналы			
Педаль тормоза	Аварийный переключатель	Переключатель левого поворота	Переключатель правого поворота	Левая передняя лампа и указатель на перед панели	Правая передняя лампа и указатель на перед. панели	Левая задняя лампа	Правая задняя лампа
0 0 0	0 0 0	0 0 1	0 1 0	выкл выкл мигание	выкл мигание выкл	выкл выкл мигание	выкл мигание выкл
0 0 0	1 1 1	0 0 1	0 1 0	мигание мигание мигание	мигание мигание мигание	мигание мигание мигание	мигание мигание мигание
1 1 1	0 0 0	0 0 1	0 1 0	выкл выкл мигание	выкл мигание выкл	вкл вкл мигание	вкл мигание вкл
1 1 1	1 1 1	0 0 1	0 1 0	мигание мигание мигание	мигание мигание мигание	вкл вкл мигание	вкл мигание вкл

Битовый процессор



Битовый процессор

	Описание входов	(Положительная логика)
;		
;		
BRAKE	BIT P1.0	; Педаль тормоза нажата
EMERG	BIT P1.1	; Аварийный сигнал включен
PARK	BIT P1.2	; Включение габаритных огней
L_TURN	BIT P1.3	; Включен левый поворот
R_TURN	BIT P1.4	; Включен правый поворот
;		
;	Описание выходов	(Положительная логика)
;		
L_FRNT	BIT P1.5	; Передний указатель левого поворота
R_FRNT	BIT P1.6	; Передний указатель правого поворота
L_DASH	BIT P1.7	; Индикатор левого поворота на панели
R_DASH	BIT P2.0	; Индикатор правого поворота на панели
L_REAR	BIT P2.1	; Задний указатель левого поворота
R_REAR	BIT P2.2	; Задний указатель правого поворота
;

Битовый процессор

```
;
SUB_DIV  DATA  20H      ; Делитель частоты прерываний
HI_FREQ  BIT    SUB_DIV.0 ; Бит генератора высокой частоты
LO_FREQ  BIT    SUB_DIV.7 ; Бит генератора низкой частоты
;
;          ...
;          ORG    0000H
;          JMP    INIT
;
;          ...
INIT:     ORG    100H
MOV       TMOD,#00000001B ; Таймер 0 в режим 1
MOV       TL0,#0          ; Инициализация регистров таймера
MOV       TH0,#-16
MOV       SUB_DIV,#244
SETB      ET0             ; Разрешение прерывания от таймера
SETB      EA              ; Общее разрешение всех прерываний
SETB      TR0             ; Старт таймера
;          ...           ; Продолжение основной программы
```

Битовый процессор

```
ORG    000BH      ; Вектор прерывания таймера 0
MOV     TH0, #-16   ; Обслуживание прерывания таймера 0
PUSH    PSW
PUSH    ACC
PUSH    B
DJNZ    SUB_DIV, T0SERV
MOV     SUB_DIV, #244
```

$$T = (16 \times 256) \times 244 = 4096 \times 244 = 0,999 \text{с}$$

```
DIM      BIT    PSW.1      ; Назначение флага временного хранения
```

```
MOV      C, PARK
ANL      C, HI_FREQ

MOV      DIM, C
```

Битовый процессор

LB - function:

```
MOV    C, L_TURN  
  
ORL    C, EMERG  
ANL    C, LO_FREQ  
  
MOV    L_DASH, C
```

LF - function:

```
MOV    F0, C  
ORL    C, DIM  
  
MOV    L_FRNT, C
```

Битовый процессор

LR - function:

```
MOV    C, BRAKE  
ANL    C, /L_TURN
```

```
ORL    C, FØ  
ORL    C, DIM  
MOV    L_REAR, C
```

Битовый процессор




```
MOV    C, R_TURN  
  
ORL    C, EMERG  
ANL    C, LO_FREQ  
  
MOV    R_DASH, C  
MOV    FØ, C  
ORL    C, DIM  
  
MOV    R_FRNT, C  
  
MOV    C, BRAKE  
ANL    C, /R_TURN  
  
ORL    C, FØ  
ORL    C, DIM  
MOV    R_REAR, C
```

Битовый процессор

POP	B	; Восстановление регистров процессора
POP	ACC	
POP	PSW	
RETI		

Битовый процессор

Биты SUB_DIV								Скважность						
7	6	5	4	3	2	1	0	12.5%	25%	37.5%	50%	62.5%	75%	87.5%
X	X	X	X	X	0	0	0	нет	нет	нет	нет	нет	нет	нет
X	X	X	X	X	0	0	1	нет	нет	нет	нет	нет	нет	да
X	X	X	X	X	0	1	0	нет	нет	нет	нет	нет	да	да
X	X	X	X	X	0	1	1	нет	нет	нет	нет	да	да	да
X	X	X	X	X	1	0	0	нет	нет	нет	да	да	да	да
X	X	X	X	X	1	0	1	нет	нет	да	да	да	да	да
X	X	X	X	X	1	1	0	нет	да	да	да	да	да	да
X	X	X	X	X	1	1	1	да	да	да	да	да	да	да

MOV C, SUB_DIV.1 ;  50%
 ANL C, SUB_DIV.0 ;  25%
 ORL C, SUB_DIV.2 ;  62,5%
 MOV DIM, C ; Результат в DIM