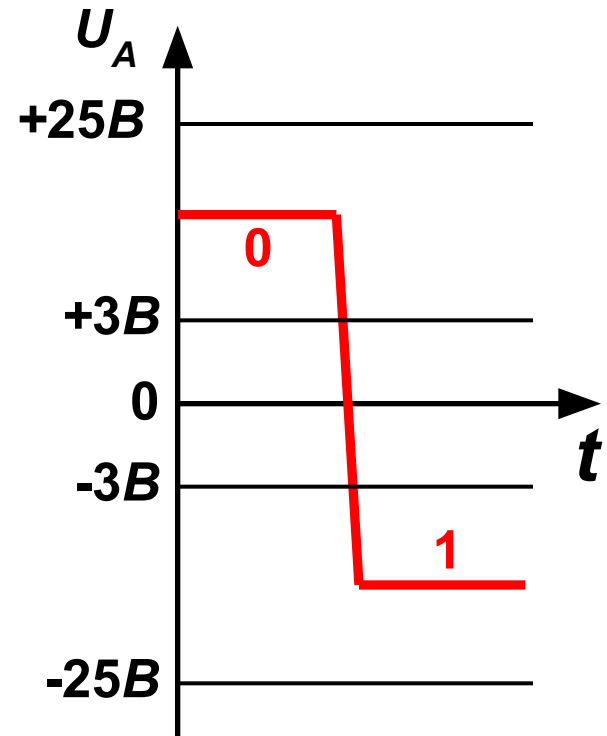
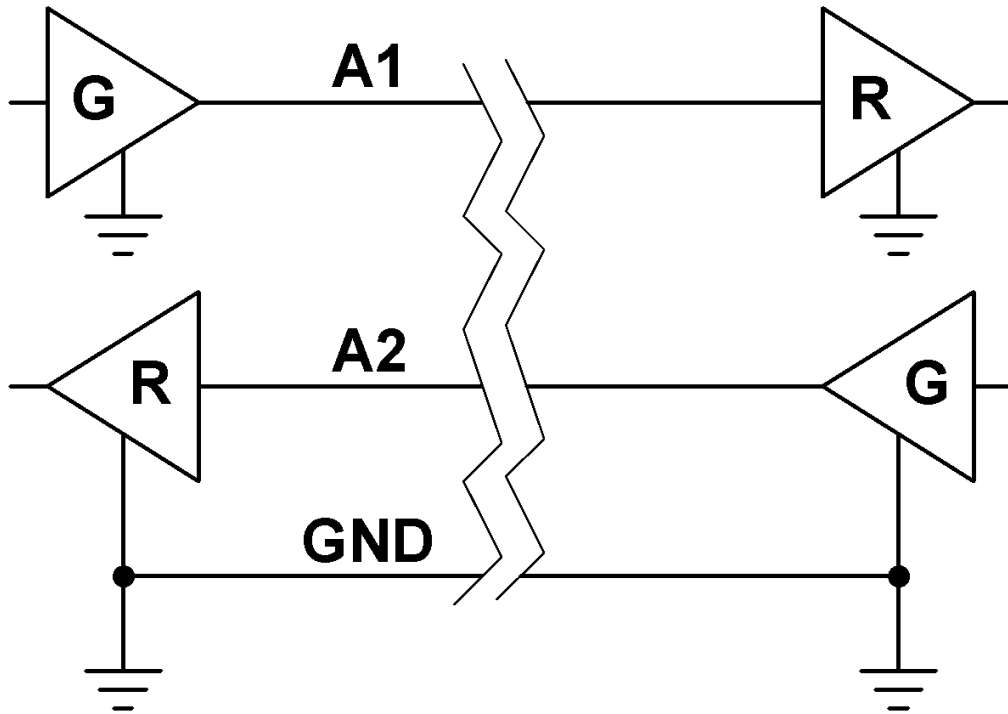


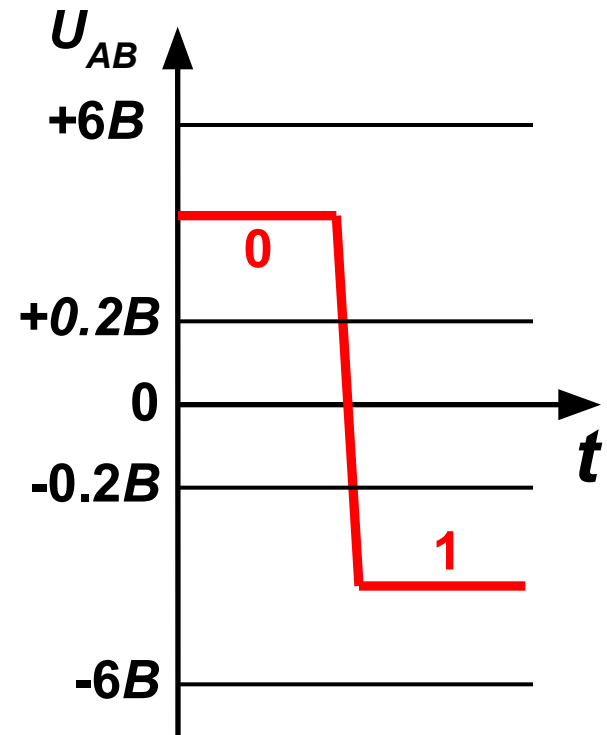
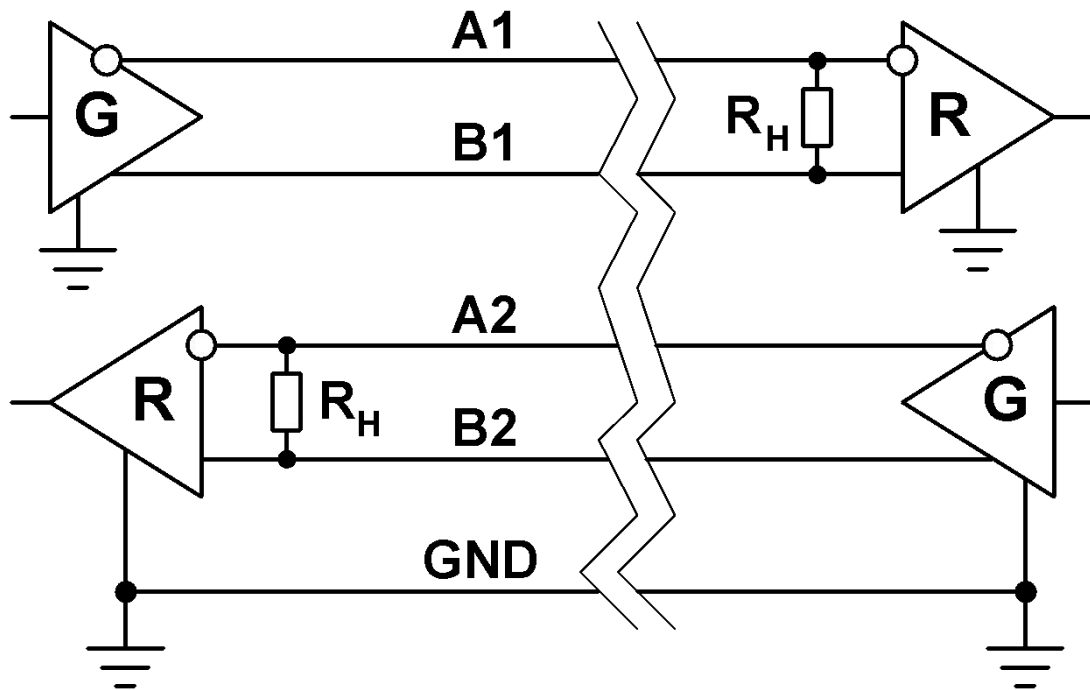
# **Линии последовательной передачи данных**

# RS-232



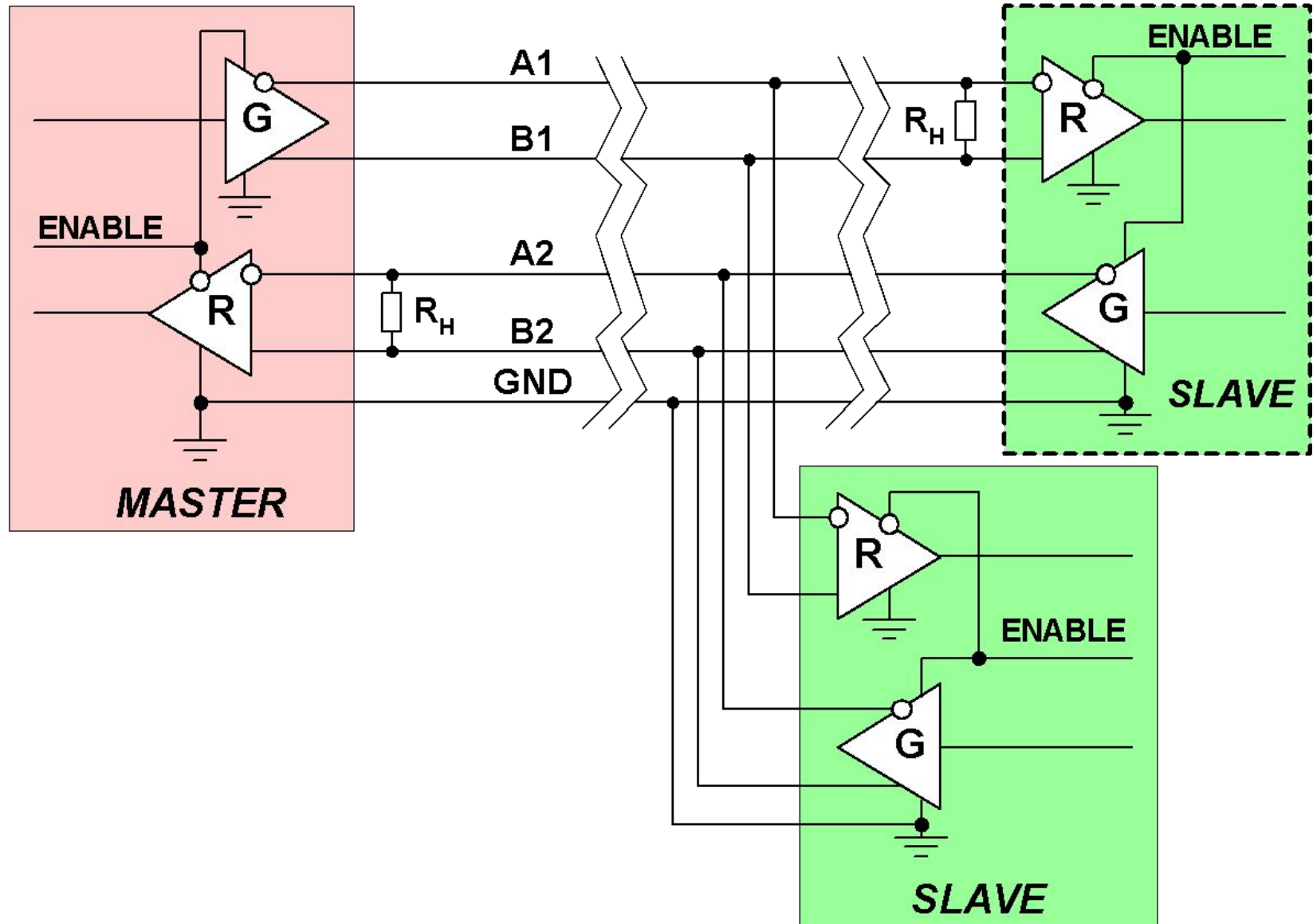
**G** – генератор сигнала (передатчик); **R** – приемник;  
**A1, A2** – сигнальные проводники;  
**GND** – земля

# RS-422

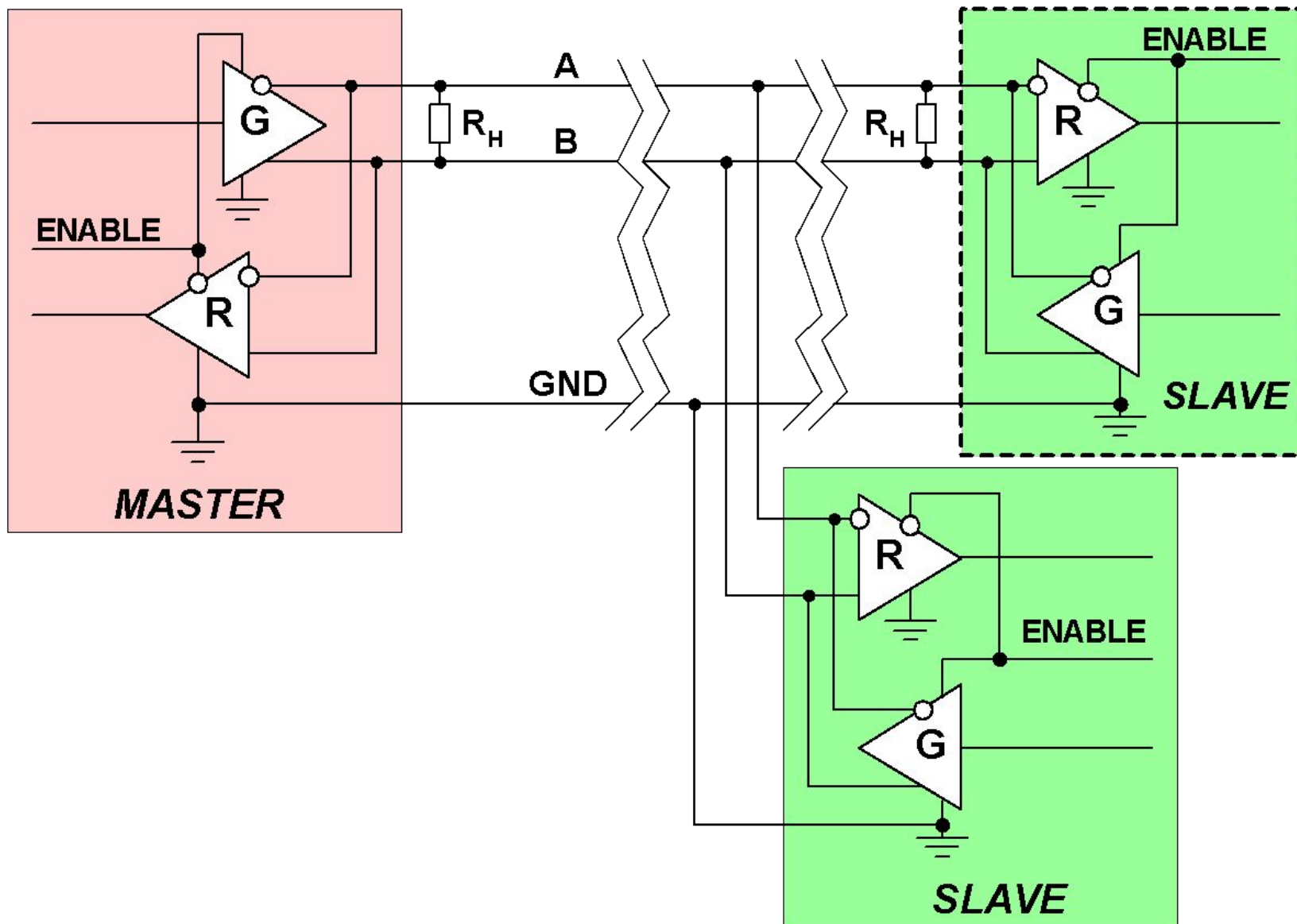


**G** – генератор сигнала (передатчик); **R** – приемник;  
**A1, B1, A2, B2** – сигнальные проводники;  
**GND** – земля;  
**R<sub>H</sub>** – сопротивление нагрузки

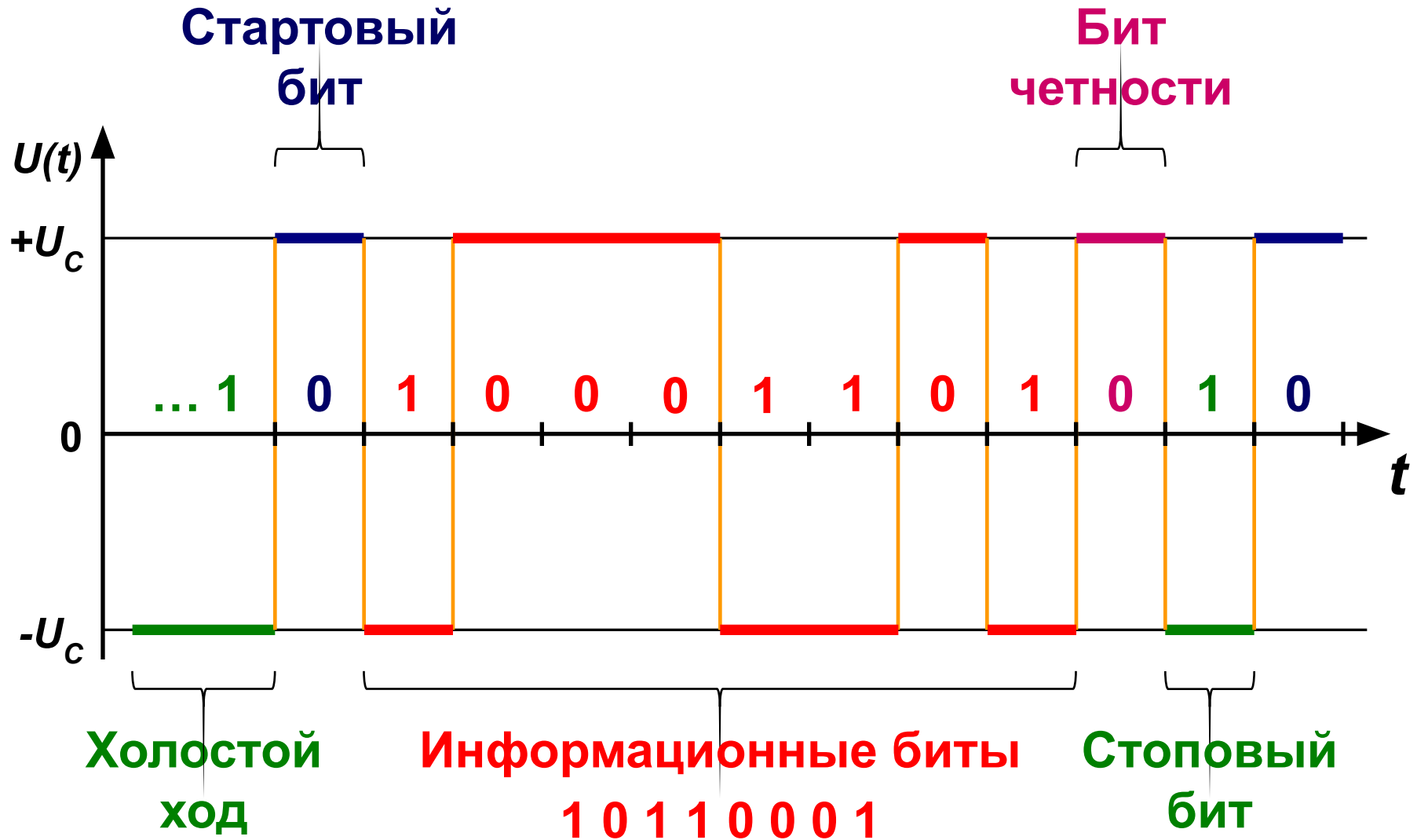
# RS-485 (четырёхпроводная схема)



# RS-485 (двухпроводная схема)



# Кодирование сообщений



# Режимы кодирования

---

<i>Биты</i>	<i>Количество</i>
<b>Стартовый бит</b>	<b>1</b>
<b>Информационные биты</b>	<b>5, 6, 7, 8</b>
<b>Бит четности</b>	<b>0, 1</b>
<b>Стоповые биты</b>	<b>1, 1.5, 2</b>

Режимы бита четности:

- код с проверкой на четность;
- код с проверкой на нечетность;
- бит четности не передается.

# Открытие порта последовательной передачи данных (Windows)

---

```
void* CreateFile(  
    const char *Name,    // "COM1", "COM2", ...  
    long AccessFlags,    // GENERIC_READ|GENERIC_WRITE  
    long ShareMode,      // 0  
    void *Ptr,           // 0  
    long CreateFlags,    // OPEN_EXISTING  
    long Flags,          // FILE_FLAG_WRITE_THROUGH  
    void *TemplateH      // 0  
);  
  
void *Hnd;  
Hnd = CreateFile( "COM1", GENERIC_READ|GENERIC_WRITE,  
    0, 0, OPEN_EXISTING, FILE_FLAG_WRITE_THROUGH, 0 );  
if( Hnd == INVALID_HANDLE_VALUE ) {  
    printf("Open error\n");  
}
```



# Установка режима приема/передачи

---

```
bool SetCommState( void *Hnd, DCB *dcb );
```

```
struct DCB {  
    long DCBlength;           // sizeof(DCB)  
    long BaudRate;           // скорость передачи в бодах  
    long fBinary : 1;        // 1  
    long fParity : 1;        // вкл./откл. бит четности  
    long fDtrControl : 2;    // DTR_CONTROL_DISABLE  
    long fRtsControl : 2;    // RTS_CONTROL_ENABLE  
    char ByteSize;           // 5, 6, 7, 8  
    char Parity;              // 0-нет, 1-нечетн., 2-четн.  
    char StopBits;           // 0-1, 1-1.5, 2-2  
    ...  
};
```

# Установка режима приема/передачи (пример)

---

```
DCB dcb;
memset( &dcb, 0, sizeof(dcb) );
dcb.DCBlength    = sizeof(DCB);
dcb.BaudRate     = CBR_9600;
dcb.fBinary      = true;
dcb.ByteSize     = 8;
dcb.fParity      = true;
dcb.Parity       = EVENPARITY;
dcb.StopBits     = ONESTOPBIT;
dcb.fDtrControl  = DTR_CONTROL_DISABLE;
dcb.fRtsControl  = RTS_CONTROL_ENABLE;
if( !SetCommState(Hnd, &dcb) ) {
    printf("SetCommState error\n");
}
```

# Чтение/запись данных

---

```
bool ReadFile( int Hnd, void *Buf, long N,  
              long *M, void *Ptr );
```

```
bool WriteFile( int Hnd, void *Buf, long N,  
               long *M, void *Ptr );
```

```
unsigned char Buf[100];  
long BytesRead;  
if( !ReadFile( Hnd, Buf, 100, &BytesRead, 0 ) ) {  
    printf("ReadFile error\n");  
}
```