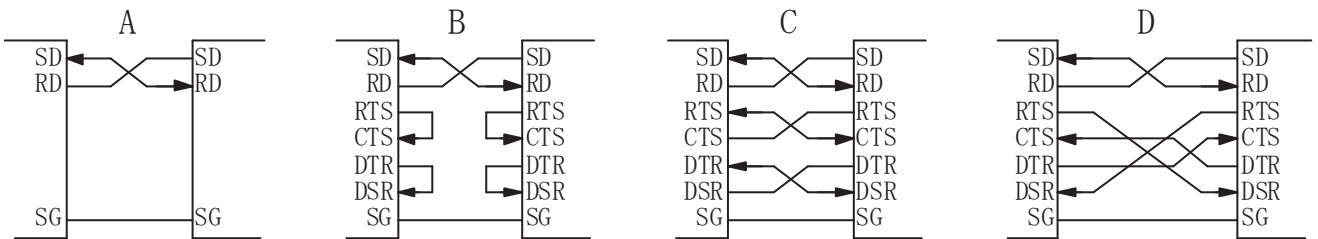


RS232C接続

デバイス相互間のやり取り

**A) 3 線式(その 1)**

やり取りを可能にするためには、先ず、自分の出力を相手が受け取ることが必要です。コネクタを細工することは実用性がありませんから、データ線をクロスさせます。

最も単純には、データ線を介してデータの授受ができれば、それで十分です。制御信号線のやり取りは行いません。制御のためのやり取りが必要なときは、データ線を利用して、制御情報をやり取りします。

B) 3 線式(その 2)

データ線はクロスさせます。制御線は、自分が出したものを自分が受けます。擬似的に、相手とやり取りしているのと、同じやり取りを行うことができます。

本当に相手とやり取りしているのではありませんから、実用上の意味はありません。しかし、相手と制御信号のやり取りをするソフトウェアを、そのまま利用することができます。

C) 7 線式(その 1)

データ線はクロスさせます。制御線も、対応する信号線をクロスさせて接続します。RS232C と同じ意味でのやり取りは不可能ですが、やり取りする信号の種類を最大に活用することができます。各システム毎に、制御の意味を定義して使用します。

D) 7 線式(その 2)

データ線はクロスさせます。RS232C の制御信号の中で、最も汎用性と必要性が高いのは、送信要求(RTS)に対する送信許可(CTS)のやり取りです。この方式では、RTS/CTS のやり取りを行います。送信要求を発行する側が決まっているシステムでは、発行側が、送信要求を出します。両者が対等の場合にも、送信要求が発生した側から、送信要求を出します。

要求側は、RTS を発行します。相手側は、その RTS 信号を、DTR によって受け取ります。そして、DTR から要求側の CTS に対して送信許可を出します