第2章　ネットワークの設定

本章の目標は，ネットワークインタフェースの設定を行い，これにより，PC 間で通信ができることを確認することである．なお，本実験の内容をより理解するため，文献 [1], [2], [3] などが参考になる．

2.1　ネットワークインタフェースの確認

OS は起動時に接続されたネットワークインタフェースを検出する．ここで検出されたネットワークインタフェースを確認するには，図 2.1 に示したように，`ip` に `link` (省略して `l` でも良い) パラメタを付けて実行する．その結果，図 2.1 では，lo, enp1s0, enx0090cce7c748 の 3 つのネットワークインタフェースが検出されたことが分かる．

本実験では，今後，これらのネットワークインタフェース名を使用するが，この内， enp1s0 と enx0090cce7c748 は使用する PC によって名称が変わるため，混乱を招く恐れがある．そこで，本テキストでは，これらのネットワークインタフェースを順に eth0, eth1 と呼ぶ．以下で，この別名が使用された場合は，演習 2.1 の要領で，各自の PC での名称に置き換えること．

```
$ ip link    (ip l でも良い)
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN mode
DEFAULT group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
2: enp1s0: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN
mode DEFAULT group default qlen 1000
    link/ether 00:24:8c:56:07:4a brd ff:ff:ff:ff:ff:ff
3: enx0090cce7c748: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN
mode DEFAULT group default qlen 1000
    link/ether 00:90:cc:e7:c7:48 brd ff:ff:ff:ff:ff:ff
```

図 2.1: ip の実行例

演習 2.1 割当てノート PC の任意の USB ポートに，LAN アダプタを 1 台付け (今後，同じ LAN アダプタを使い続けるため，番号を控えておくこと)，ネットワークインタフェースの情報を，図 2.1 の要領で表示しなさい．

以下では，ここで表示されたネットワークインタフェースの内， enp で始まるものを eth0， enx で始まるものを eth1 と呼ぶ．その対応付けを以下に記録しておくこと．

- 1 台目
  - eth0 =
    (enp で始まるもの)
  - eth1 =
    (enx で始まるもの)
- 2 台目
  - eth0 =
    (enp で始まるもの)
  - eth1 =
    (enx で始まるもの)

□

2.2 ハードウェアアドレスと IP アドレス

演習 2.1 で得られた情報から， eth0 と eth1 はそれぞれリンクの種類が ether (Ethernet) であること分かる (lo については後述する)．その後ろのコロン (:) で区切られた 12 桁の 16 進数は，そのネットワークインタフェースのハードウェアアドレスであり，例えば，演習 2.1 において，図 2.1 の enp1s0 の場合，そのハードウェアアドレスは 00:24:8c:56:07:4a である．このように， (物理的な) ネットワークインタフェースには，固有な 48 ビットのハードウェアアドレスが割り振られている．そして，この 48 ビットの値を 8 ビット毎に分け，それぞれを 16 進数で表したものをコロン (:) などで区切って表記するのが一般的である．ハードウェアアドレスの構成を以下に示す．

- 先頭から 1〜2 ビット目

  先頭の 2 ビットは，それぞれが特別な意味を持つ．課題 2.1 を行いなさい．

- 先頭から 3〜24 ビット目 (22 ビット分)

  先頭の 2 ビットが 00 の場合，続く 22 ビットはメーカ識別子と呼ばれ，IEEE (Institute of Electrical and Electronics Engineers) がネットワークインタフェースを作成するメーカ等に割り当てる番号である．従来，メーカ識別子は OUI (Organizationally Unique Identifier) と呼ばれていたが， MA-L (MAC Address Block Large) に改称された．また，26 ビットのメーカ識別子 MA-M と 34 ビットのメーカ識別子 MA-S も追加された．

- 先頭から 25〜48 ビット目 (24 ビット分)

  先頭の 2 ビットが 00 の場合，残りの 24 ビット (MA-M のときは 22 ビット，MA-S の場合は 20 ビット) はメーカ内識別子である．各メーカは，この値が一意になるように製品に割り当てることで，ハードウェアアドレスが世界で唯一の番号になる．

なお，ハードウェアアドレスのことを MAC (Media Access Control) アドレス，又は，物理アドレスと呼ぶことがある．このハードウェアアドレスは，Wi-Fi や Bluetooth などでも用いられるが，トラッキングを防ぐため，ランダムなハードウェアアドレスが用いらることもある．

課題 2.1 (*) ハードウェアアドレスの先頭 2 ビットが持つ意味を調べなさい． □

課題 2.2 (*) 割当て PC (いずれか 1 台) のネットワークインタフェース eth0 と eth1 を登録したメーカを調べなさい．
ヒント: https://regauth.standards.ieee.org/standards-ra-web/pub/view.html を訪れなさい．ここの「Please select a Product」を「All MAC (MA-L, MA-M, MA-S)」にして虫眼鏡ボタンを押し,「SEARCH RESULTS」に出てきた「Filter results by search text」と書かれたテキストボックスに，検索する MAC アドレスを，必要な桁だけ，コロンを付けずに入力して,「Filter」する． □

一般に，ネットワークを用いて通信するマシン (PC など) では，図 2.2 のような階層構造を構成して，送受信されたデータを処理している．この図は，図 1.1 (p.4) のように，通信する 2 台のマシンが直接接続している場合のネットワークの階層構成を示したものであり，各 PC は第 1 層から第 5 層までの階層を持つ．

そして， pc1 のユーザから送信されたデータは，第 5 層で処理された後，第 4 層に送られ，以下同様に，第 3 層，第 2 層，第 1 層へと下の層に順々に送られる．第 1 層は，最終的なデータを電気などの信号の形に変換し，伝送路やスイッチング Hub などの中継器を通して， pc2 の第 1 層に届けられ，ここで元のデータの形に戻される． pc2 では，第 1 層，第 2 層，… と上の層に順々に送られて，最後に，第 5 層で， pc2 のユーザに届けられる．

また，この図に示すように，それぞれの層は，通信相手の同じ層と論理的 (第 1 層は物理的) なネットワーク接続を持ち，この上で互いの同じ層同士で通信をしている．ここで行われる通信の約束 (どのような順序で，どのような情報をやりとりするかなど) のことをプロトコル (protocol) と呼ぶ.

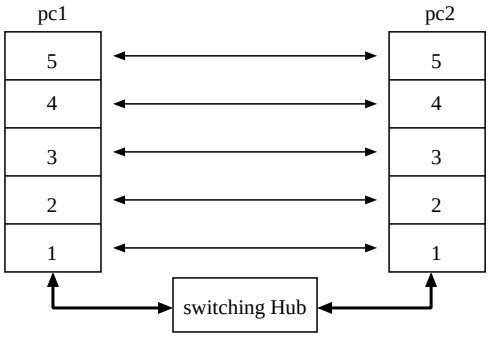

図 2.2: 階層構造 (直結しているとき)

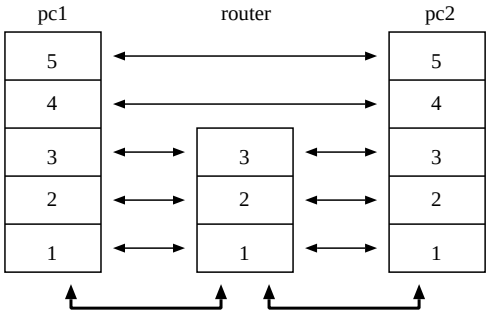

図 2.3: 階層構造 (ルータを経由するとき)

ハードウェアアドレスは，この内の第 2 層 (データリンク層) の通信を行う際に通信相手を区別する為に使用されるアドレスである．データリンク層は直接ケーブルが接続されている範囲 (正確には，送信した情報がそのまま伝わる範囲であれば，スイッチング Hub などの装置を介しても良い) の通信を実現する層である．

しかし，全ての通信相手とケーブルを接続することは出来ないので，図 2.3 のように，通信相手との間に中継機器 (ルータ: router) を 1 台以上介在させることになる．このような場合は，データリンク層だけではなく，第 3 層 (ネットワーク層) の機能がないと通信ができない.

データリンク層と同様に，ネットワーク層でも通信相手を区別するための世界的に一意なアドレスが必要であるが，インターネットで利用されているネットワーク層のプロトコル IP (Internet Protocol) の内，現在最も良く利用されている IP バージョン 4 (IPv4) では，32 ビットのアドレスが使われる．そして，これを 8 ビット

毎に分けて 10 進数で表し，これらをピリオド (.) で区切って表記する．例えば，筑波大学情報学群のウェブサーバの IP アドレスは， 130.158.224.105 である．

IP アドレスは，ネットワークインタフェースに対して固有に付けられてはいないので，ネットワーク上で一意になるように管理者が設定する必要がある． 本実験では，各割り当て PC (3 年実験 A〜P) で使用可能な IP アドレスの範囲を表 2.1 で定める． 特に指示がない限り，**他人に割り当てられた番号を誤って使うことがないように**注意すること．ここで，$\{x,y\}$ は，番号 x と y が使用可能であることを意味する．また， $n = 1$〜254 である． 3 年実験 A と 3 年実験 B を使用している場合は， IP アドレスとして， 192.168.1.1〜254 と 192.168.2.1〜254 を利用する．

使用 PC	割当て IP アドレス
3 年実験 A,B	192.168.$\{1,2\}$.n
3 年実験 C,D	192.168.$\{3,4\}$.n
3 年実験 E,F	192.168.$\{5,6\}$.n
3 年実験 G,H	192.168.$\{7,8\}$.n
3 年実験 I,J	192.168.$\{9,10\}$.n
3 年実験 K,L	192.168.$\{11,12\}$.n
3 年実験 M,N	192.168.$\{13,14\}$.n
3 年実験 O,P	192.168.$\{15,16\}$.n

表 2.1: IP アドレス割当て (3 年実験 J〜P は欠番)

課題 2.3 (*) IPv4 アドレスのクラス構造について調べなさい．クラス A〜クラス C については，ネットワーク部とホスト部が分かるようにしなさい．クラス D については，クラス A〜クラス C と比べて，どのような目的で使うかを明示しなさい．

また，表 2.1 で割り当てられた IP アドレスは，どのクラスに属するか? □

課題 2.4 (*) 表 2.1 の n は 1〜254 であると説明した．しかし， n は 8 ビットの値なので， 0 及び 255 を指定することも可能なはずである．実は，この値が 0 と 255 のときは，特別な IP アドレスであることが規定されており，特定マシン用の IP アドレスとしては使えない．それでは， 0 と 255 のときはどのような用途で用いられるのか． □

なお，ネットワークインタフェース lo はローカルループバックと呼ばれ，自分自身と通信するときに利用される論理的なインタフェースである．このため，ハードウェアアドレスは持たない．また，ローカルループバック用のアドレスには 127.x.y.z ($x,y,z = 0$〜255) のいずれかを使うことになっているが，この内の 127.0.0.1 を使うのが一般的である．

2.3 IPアドレスの設定

ネットワークインタフェースにIPアドレスを割り当てる際も，`ip`コマンドを使用する．

図2.4では，最初に，`ip`コマンドの`link`でEthernetを通信可能なup状態とし，次に，`ip`コマンドの`address`(省略して`a`でも良い)で，ネットワークインタフェースeth0に対してIPアドレス192.168.100.1を割り当てている．ここで，コマンド内のアドレス末尾に`/24`を付ける理由は4.3.4節(p.69)で述べる．なお，以上の設定変更は，管理者権限が必要なので，`sudo`コマンドを使う．そして，`ip`コマンドでeth0の情報を表示させた結果，そのIPアドレス(inet)が192.168.100.1になっていることが分かる．

```
$ sudo ip l set eth0 up
$ sudo ip a add 192.168.100.1/24 dev eth0
$ ip a show dev eth0
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq
state UP group default qlen 1000
    link/ether 00:24:8c:56:07:4a brd ff:ff:ff:ff:ff:ff
    inet 192.168.100.1/24 scope global eth0
       valid_lft forever preferred_lft forever
```

図2.4: IPアドレスの設定

演習2.2 表2.1から割当てPCで利用可能な異なるIPアドレスを2つ選びなさい．但し，選択したIPアドレスの内，3番目の値は同じものを，4番目の値は異なるものを選ぶこと．例えば，割当てPCが3年実験Aと3年実験Bの場合は，3番目の値を1に統一し，4番目の値として1と2を選ぶことにより，2つのIPアドレス192.168.1.1と192.168.1.2を選択することなどが考えられる．

これらのIPアドレスを，それぞれの割当てPCのeth0に設定しなさい(eth1には**設定しないないこと**．誤って設定した場合は，設定時に用いたコマンドの`add`を`del`に変更して実行すると，設定を解除できる)．このとき，eth0ではなく，演習2.1で記録したインタフェース名を指定すること．そして，それぞれの割当てPCでIPアドレスが正しく割当てられたことを`ip`コマンドで確認しなさい． □

2.4 通信実験

演習2.2の設定により，IPアドレスを設定した割当てPC相互間で通信が可能になったはずである．これを`ping`コマンドを用いて検証してみる．図2.5に，192.168.100.1のPCから192.168.100.2に対して`ping`コマンドを行った実行例を示す．

```
$ ping -c 4 192.168.100.2
PING 192.168.100.2 (192.168.100.2) 56(84) bytes of data.
64 bytes from 192.168.100.2: icmp_seq=1 ttl=64 time=3.28 ms
64 bytes from 192.168.100.2: icmp_seq=2 ttl=64 time=1.61 ms
64 bytes from 192.168.100.2: icmp_seq=3 ttl=64 time=1.61 ms
64 bytes from 192.168.100.2: icmp_seq=4 ttl=64 time=1.62 ms

--- 192.168.100.2 ping statistics ---
4 packets transmitted, 4 received, 0% loss, time 3028ms
rtt min/avg/max/mdev = 1.615/2.033/3.280/0.720 ms
```

図 2.5: ping による通信 (成功例)

```
$ ping -c 4 192.168.100.2
PING 192.168.100.2 (192.168.100.2) 56(84) bytes of data.
From 192.168.100.1 icmp_seq=1 Destination Host Unreachable
From 192.168.100.1 icmp_seq=2 Destination Host Unreachable
From 192.168.100.1 icmp_seq=3 Destination Host Unreachable
From 192.168.100.1 icmp_seq=4 Destination Host Unreachable

--- 192.168.100.2 ping statistics ---
4 packets transmitted, 0 received, +4 errors, 100% loss, time 3021ms
pipe 3
```

図 2.6: ping による通信 (失敗例)

ping コマンドでは，指定された IP アドレスに対して特殊なメッセージを送る．これを受け取った相手は，このメッセージに対する応答メッセージを返すので，これらのメッセージが往復した時間 (往復伝搬遅延 (RTT: Round Trip Time) と呼ぶ) を計測し，これを表示する．なお，ping の -c オプションは 応答を要求する特殊なメッセージを送る回数を指定するものであり，この指定がないときは，強制終了するまでメッセージを送付し続ける．図では，メッセージを 4 回送信した結果，192.168.100.2 から順番に (メッセージに付けられた番号 icmp_seq が 1 から順に 4 まで) 応答メッセージを 4 回受信し，それぞれの RTT が 3.28ms, 1.61ms, 1.61ms, 1.62ms であることが分かる．

また，最後の統計情報 (statistics) から，メッセージの損失率 (loss) は 0% であり，RTT の平均 (avg) は 2.033ms であることが分かる．接続に失敗している場合は，図 2.6 のように損失率が 100% と表示される．このため，ping コマンドは，ネットワークの設定や接続に異常がないことを確認する際に，非常に良く用いられる．

演習 2.3 演習 2.2 で IP アドレスを設定した割当て PC の一方から他方に対して，ping コマンドにより通信ができることを確認しなさい． このとき， スイッチング Hub (または Hub) のランプはどのようになっているか．

また，通信相手の方のネットワークケーブルを外す ことによって，通信ができなくなることを確認しなさい． このとき， スイッチング Hub (または Hub) のランプはどのようになっているか.

ネットワークケーブルが正しく接続されているにも関わらず，通信が出来ない場合は，演習 2.1 において記録したネットワークインタフェース名を確認し，IP アドレスを別のネットワークインタフェースに割当てていないか確認すること．誤って割り当ててしまった場合は，設定時に用いたコマンドの `add` を `del` に変更して実行することで，割り当てた IP アドレスを削除できる． □

なお，これ以降の演習や課題も，**割当て PC2 台で行うことを基本**とし，その場合は特に指示を出さない．これに対して，他の人と組となり，3 台以上の PC で実験する場合は，演習や課題ごとに，組む人数を指示する.

課題 2.5 `ping` コマンドを用いて，以下のアドレスに対して通信したときの状況を示しなさい．但し，メッセージの**送信回数は 3 回**とすること.

このとき観測された RTT と スイッチング Hub (または Hub) のランプの挙動を演習 2.3 と比較し，両者の RTT が有意に異なる理由を示しなさい.

1. ローカルループバックアドレス 127.0.0.1
2. eth0 に割り当てた IP アドレス

   (例えば， eth0 に 192.168.1.1 を割り当てたマシンから 192.168.1.1 へ `ping` コマンドにより通信する)

□

課題 2.6 この実験を行う前に，割当て PC **双方**において，以下のコマンドを実行しなさい.

```
$ sudo sh -c "echo 0 > /proc/sys/net/ipv4/icmp_echo_ignore_broadcasts"
```

その後， `ping` コマンド (但し， `-b` オプションをつけること) を用いて，以下のアドレスに対して通信したときの状況を示しなさい．但し，メッセージの**送信回数は 3 回**とすること．このとき，実行結果の先頭に WARNING が表示されており，また，図 2.5 の場合よりも**応答メッセージの回数が多い理由**を，課題 2.4 での調査結果を踏まえて述べなさい.

1. eth0 に割り当てた IP アドレスの 4 番目の値を 255 としたアドレス

   (例えば， eth0 に 192.168.1.1 を割り当てたマシンから 192.168.1.255 へ `ping` コマンドにより通信する)

2. eth0 に割り当てた IP アドレスの 4 番目の値を 0 としたアドレス

後者のアドレスは OS の実装によって解釈が異なることがあり，課題 2.4 での調査結果と食い違う可能性がある．実験終了後に，以下を両方の PC で実行して，設定を元に戻すこと．

```
$ sudo sh -c "echo 1 > /proc/sys/net/ipv4/icmp_echo_ignore_broadcasts"
```

□

2.5 ネットワーク設定の保存

2.3 節では，`ip` コマンドによりネットワークインタフェースに IP アドレスを設定した．しかし，この設定は保存されないため，再起動すると，IP アドレスを設定し直さなくてはならない．

このため，Debian 12.7 ではネットワークの設定を保存する仕組みが用意されている．そして，OS の起動時に，`ifup` コマンドを使って保存された内容を設定する．図 2.7 は，この操作を手動で行った例である．

まず，`/etc/network/interfaces.d/eth0` に eth0 の IP アドレスとネットマスク (4.3.1 節 (p.57) で調べる) を保存した後，`ifup` で eth0 を起動し，これにより通信が可能になったことを確認している．

```
$ sudo ln -s /etc/resolv.conf /etc/resolv.conf
$ sudo emacs /etc/network/interfaces.d/eth0
$ cat /etc/network/interfaces.d/eth0
auto eth0
iface eth0 inet static
 address 192.168.100.1
 netmask 255.255.255.0
iface eth0 inet6 auto
$ sudo ip a del 192.168.100.1/24 dev eth0  ← 手動で設定したIPアドレ
スを取り消す
$ sudo ifdown eth0            ← eth0 を念の為停止
$ sudo ifup eth0              ← eth0 を起動
$ ip a show dev eth0          ← IPアドレスが設定された
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP
 group default qlen 1000
    link/ether 00:0c:29:4a:9c:a7 brd ff:ff:ff:ff:ff:ff
    inet 192.168.100.1/24 brd 192.168.100.255 scope global eth0
       valid_lft forever preferred_lft forever
    inet6 fe80::20c:29ff:fe4a:9ca7/64 scope link
       valid_lft forever preferred_lft forever
$ ping -c 4 192.168.100.2
PING 192.168.100.2 (192.168.100.2) 56(84) bytes of data.
64 bytes from 192.168.100.2: icmp_seq=1 ttl=64 time=3.28 ms
64 bytes from 192.168.100.2: icmp_seq=2 ttl=64 time=1.61 ms
64 bytes from 192.168.100.2: icmp_seq=3 ttl=64 time=1.61 ms
64 bytes from 192.168.100.2: icmp_seq=4 ttl=64 time=1.62 ms

--- 192.168.100.2 ping statistics ---
4 packets transmitted, 4 received, 0% loss, time 3028ms
rtt min/avg/max/mdev = 1.615/2.033/3.280/0.720 ms
```

図 2.7: ifup によるインタフェースの起動

```
$ sudo ifdown eth0      ← eth0 を停止
$ ip a show dev eth0    ← 停止した為，IP アドレスが表示されない
2: eth0: <BROADCAST,MULTICAST> mtu 1500 qdisc mq state DOWN
group default qlen 1000
    link/ether 00:24:8c:56:07:4a brd ff:ff:ff:ff:ff:ff
$ ping -c 4 192.168.100.2
connect: ネットワークに届きません  ← 通信できない
```

図 2.8: ifdown によるインタフェースの停止

次に，OS の終了時には，ifdown コマンドを用いてネットワークインタフェースが停止される．図 2.8 では，同コマンドを使って eth0 を手動で停止し，その後，通信が出来なくなることを確認している．

演習 2.4 割当て PC 双方の /etc/network/interfaces.d/eth0 に，演習 2.2 で割当てた IP アドレスとネットマスクを設定しなさい．但し，ファイル名と，ファイル内で指定するネットワークインタフェース名の eth0 はいずれも，演習 2.1 で記録した名称に変更すること．

そして，ifup コマンドでネットワークインタフェースを起動し，IP アドレスが設定できることを確認しなさい．その後，ifdown コマンドでネットワークインタフェースを停止し，通信が出来なくなることを確認しなさい．以上を確認した後，ifup でネットワークインタフェースを再起動すること．

ifup コマンドでうまく設定できないときは，/etc/network/interfaces.d ディレクトリに #eth0# や eth0~ というファイルがないか確認しなさい．このファイルは emacs が作る一時保存用のファイルで，ファイルの編集途中で emacs を強制終了すると削除されずに残る．特に，ifup は eth0 より #eth0# を先に読み込むため，このファイルがあるとうまく設定できない場合がある． emacs を終了した後に，#eth0# などを削除してから，再度 ifup を行うこと．

□

2.6 カスケード接続

図 1.1 (p.4) では，2 台の PC とスイッチング Hub が接続されているが，PC の接続台数が，1 台のスイッチング Hub で接続できる台数よりも増えたときは，図 2.9 のように，スイッチング Hub 同士を接続して，接続台数を増やすことが出来る．このような接続方法をカスケード接続 (cascading connection) と呼ぶ．カスケード接続は， Hub を用いても可能である．しかし， スイッチング Hub (または Hub) は，PC が接続されることを前提として作られているため，本来は，この接続では正しく通信出来ない．

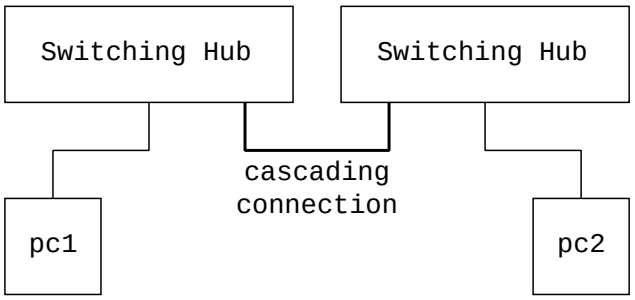

図 2.9: カスケード接続

一般に，ネットワークで用いるケーブルは 4 対 (8 本) の線 (ネットワークケーブルのコネクタ部分を確認せよ) から構成されており，通信速度が 100Mbps 以下の場合は，そのうちの 1 対を送信用に，他の 1 対を受信用に使っている．これらの送信と受信は，図 2.10 左に示すように，PC とスイッチング Hub とでは立場が異なる．PC の送信側のピンはスイッチング Hub から見ると受信側のピンであり，逆もそうである．なお，PC のピン配列のポートを MDI (Medium Dependent Interface)，スイッチング Hub のピン配列のポートを MDI-X (MDI Crossover) と呼ぶ．また，通信速度が 1Gbps 以上の場合は，送信用と受信用で共に 2 対ずつ用いるが，本数が変わるだけで，以上の状況は同じである．

この状況でスイッチング Hub 同士を接続すると図 2.10 中央のようになり，MDI-X のポート同士が接続されることから，一方のスイッチング Hub がデータを送信しても，他方の受信用ケーブルにはデータが渡らない．同様の問題は，PC 同士をスイッチング Hub を介さずに直接接続した場合にも生じる．これらの問題を回避するには，図 2.10 右のように，ネットワークケーブルの結線を変えて，送信と受信が互いの想定と一致するようにする必要がある．ここで，MDI と MDI-X を接続するネットワークケーブル (図 2.10 左) をストレートケーブル，MDI-X (または MDI) 同士を接続するもの (図 2.10 右) をクロスケーブルと呼ぶ．

なお，スイッチング Hub (または Hub) によっては，ストレートケーブルでもスイッチング Hub (または Hub) 同士を接続できるように MDI になるポート (カスケードポート) を準備しているものがあり，実験で用いている Hub (CentreCOM FH708TP) の 8 番ポートがこれに相当する．このポートの右側を見ると切替えスイッチがあり，これを左側の「X PC」にすると同ポートが MDI-X となってストレートケーブルで PC と接続可能となる．逆に右側の「= HUB」にすると，同ポートは MDI となってストレートケーブルで Hub と接続可能となる．前章で PC をネットワークケーブルに接続したとき,「Hub の 8 番ポートは特別なのでランプが点灯しない場合がある」と述べたのは，このことを示している．

このように切替えスイッチによって MDI と MDI-X を変更できると便利であるが，ネットワーク機器が増えるとこれらの設定や管理が大変になる．そのため，こ

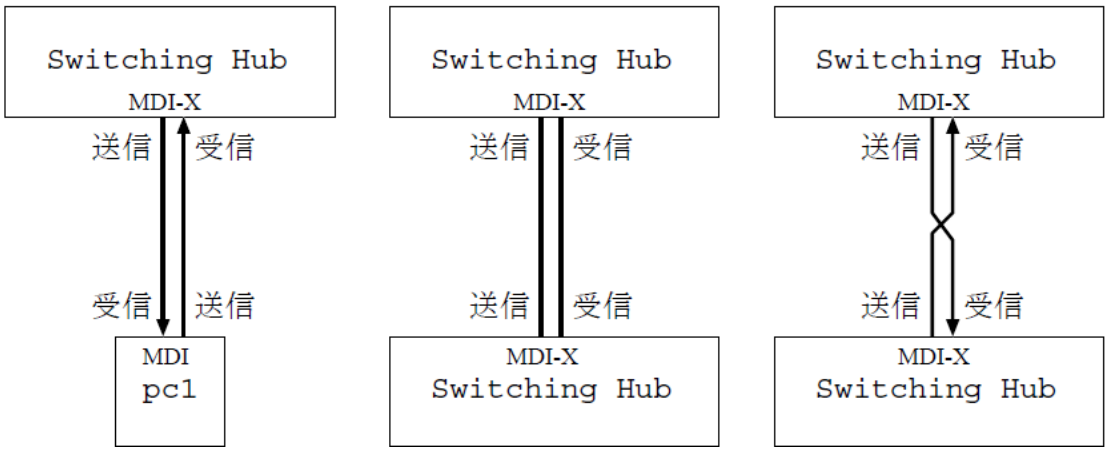

図 2.10: ストレートケーブルとクロスケーブル

の切替を自動的に行う スイッチング Hub (または Hub) がある．実験で用いているスイッチング Hub (CentreCOM FS708EX V1 及び Buffalo LSW-TX-8NS) には「AUTO MDI/MDI-X」と記載されており，これは自動切替え可能であることを示している．

課題 2.7 Hub を 2 台準備し，次の条件でカスケード接続しなさい．このとき，それぞれの Hub を接続したポートのランプはどうなるか．また，2 台の PC を図 2.9 のように接続したとき，`ping` コマンドによる通信の結果を示しなさい．

1. 一方の 1 番ポートを他方の 8 番ポート (切替えスイッチは「X PC」) に繋ぐ
2. 一方の 1 番ポートを他方の 8 番ポート (切替えスイッチは「= HUB」) に繋ぐ

□