

# C termios 構造体

[Programming C]

## termios 構造体

- 端末 interface を制御するには、termios 型の構造体に値を設定し、関数呼び出しを使う。
- 構造体と関数呼び出しは、termios.h で定義されている。
- termios.h で定義されている関数呼び出しを使う program では、適切な関数 library と link する必要がある。
- 通常使用するのは、curses library。例えば、-lcurses や -lncurses のように指定する。
- 端末を制御するときに操作する値は、次のような様々な mode に group 化できる。
  - 入力
  - 出力
  - 制御
  - local
  - 特殊制御文字

一般に termios 構造体は、次のように宣言されている

```
#include <termios.h>
struct termios {
    tcflag_t c_iflag;
    tcflag_t c_oflag;
    tcflag_t c_cflag;
    tcflag_t c_lflag;
    cc_t      c_cc[NCCS];
};
```

- termios 構造体の各 member には、上に示した 5 つの parameter の種類に対応する名前がついている。
- tcgetattr 関数を呼び出すと、端末に対応する termios 構造体を初期化することができる。

```
#include <termios.h>
int tcgetattr(int fd, struct termios, *termios_p);
```

- tcgetattr 関数は、端末 interface 変数の現在の値を termios\_p で示される構造体書き込む。
- 端末 interface 変数の値を変更したあと、tcsetattr 関数を呼び出すと、端末 interface を設定しなおすことができる。

```
#include <termios.h>
int tcsetattr(int fd, int actions, const struct termios, *termios_p);
```

- tcsetattr 関数の parameter には、変更の適用方法として、次の 3 つのうちのいずれかを指定する。

actions parameter	説明
TCSANOW	値を直ちに変更
TCSADRAIN	現在の出力が完了した後に値を変更
TCSAFLUSH	現在の出力が完了した後に値を変更するが、現在利用可能な入力のうち、read の呼び出しで返されていないものはすべて破棄する

program の中で端末の設定を変更する場合には、program を終了するときに起動前の端末の値を

復元する。

## 入力 mode

- ・入力 mode は、入力 (serial port または keyboard から端末 driver が受け取った文字) をどのように処理してから program に渡すかを制御
- ・具体的には、termios 構造体の member `c_iflag` で flag を設定する。
- ・flag はいずれも macro として定義されており、これを bit 単位の論理和で指定する。

この指定方法は、入力 mode だけでなく、すべての端末 mode に適用される。

c_iflag で使用できる macro	説明
BRKINT	回線上で break を検出したときに割り込みを生成
IGNBRK	回線上の break を無視
ICRNL	受信した carriage return を開業に変換
IGNCR	受信した carriage return を無視
INLCR	受信した開業を carriage return に変換
IGNPAR	parity error の文字を無視
INPCK	受信した文字に対して parity check を行う
PARMRK	parity error を mark する
ISTRIP	すべての入力文字の 8bit 目を落とす
IXOFF	入力に対する software flow 制御を有効にする
IXON	出力に対する software flow 制御を有効にする

## 出力 mode

- ・出力 mode は、出力文字をどのように処理するか、program から送信された文字をどのように処理してから serial port または画面に転送するかを制御
- ・出力 mode の flag の多くは、入力 mode の flag に対応している。
- ・出力 mode を指定するには、termios 構造体の member `c_oflag` で flag を設定する

c_oflag で使用できる macro	説明
OPOST	出力処理を有効にする
ONLCR	出力される改行を carriage return と line feed の pair に変換する
OCRNL	出力される carriage return を改行に変換する
ONOCR	0 カラム目では carriage return を出力しない
ONLRET	改行が carriage return も行う
OFILL	遅延に対して埋め込み文字を送信する
OFDEL	埋め込み文字として NULL ではなく DEL を使う

NLDLY	改行遅延を選択
CRDLY	carriage return 遅延を選択
TABDLY	tab 遅延を選択
BSDLY	back space 遅延を選択
VTDLY	垂直 tab 遅延を選択
FFDLY	form feed 遅延を選択

## 制御 mode

- ・ 端末 hardware 特性を制御
- ・ 制御 mode を指定するには、termios 構造体 member c\_cflag で flag を指定

c_cflag で指定できる macro	説明
CLOCAL	modem status line を無視する
CREAD	文字の受信を可能にする
CS5	送受信文字に 5bit を使う
CS6	送受信文字に 6bit を使う
CS7	送受信文字に 7bit を使う
CS8	送受信文字に 8bit を使う
CSTOPB	1 文字ごとに、1 つではなく 2 つの stop bit を使う
HUPCL	close 時に modem を hung up する
PARENB	parity の生成と検出を有効にする
PARODD	偶数 parity ではなく、奇数 parity を使う

## local mode

- ・ local mode は、端末のさまざまな特性を制御する。
- ・ local mode を指定するには、termios 構造体 member c\_lflag で flag を設定する。

c_lflag で使用できる macro	説明
ECHO	入力文字の local echo を有効に
ECHOE	ERASE の受信時に、BackSpace、Space、BackSpace の組み合わせを実行する
ECHOK	KILL 文字で行の抹消を実行する
ECHONL	改行文字を echo する
ICANON	canonical 入力処理を有効にする
IEXTEN	実装に依存する特殊機能を有効にする
ISIG	signal を有効にする

NOFLSH	queue の flush を無効にする
TOSTOP	書き込みの試みに対して background process に signal を送信する

- これらの flag のうち、最も重要なのが ECHO と ICANON。
- ECHO flag を使うと、入力された文字が echo されないようにすることができる
- ICANON flag は、受信文字を処理する 2 つの mode を切り替えるときに利用
- ICANON flag が設定されている場合、回線は canonical mode になる
- flag が設定されていない場合、non canonical mode になる

## 特殊制御文字

- 特殊制御文字は、user が入力したときに特別な方法で処理される、Ctrl-C などの文字の集合
- termios 構造体の配列 c\_cc には、特殊制御文字が support する各機能に割り当てられた文字が収められている
- 各文字の位置 (配列の添え字) は macro により定義されているが、各文字が制御文字でなければならないわけではない
- 配列 c\_cc は、端末が canonical mode に設定されているかどうかに応じて 2 つの異なる利用法がある
- 異なる 2 つの mode での配列の添え字は部分的に重複しているため、mode の値を混在させないようにする必要がある。

canonical mode での配列の添え字	説明
VEOF	EOF 文字
VEOL	EOL 文字
VERASE	ERASE 文字
VINTR	INTR 文字
VKILL	KILL 文字
VQUIT	QUIT 文字
VSUSP	SUSP 文字
VSTART	START 文字
VSTOP	STOP 文字

---

この本からの覚書。