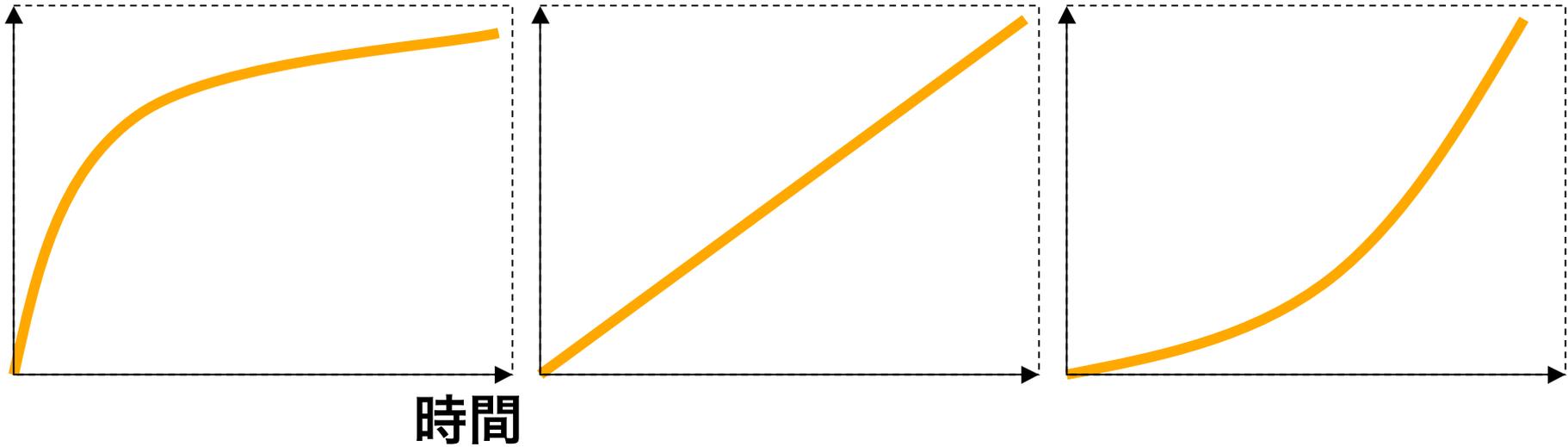
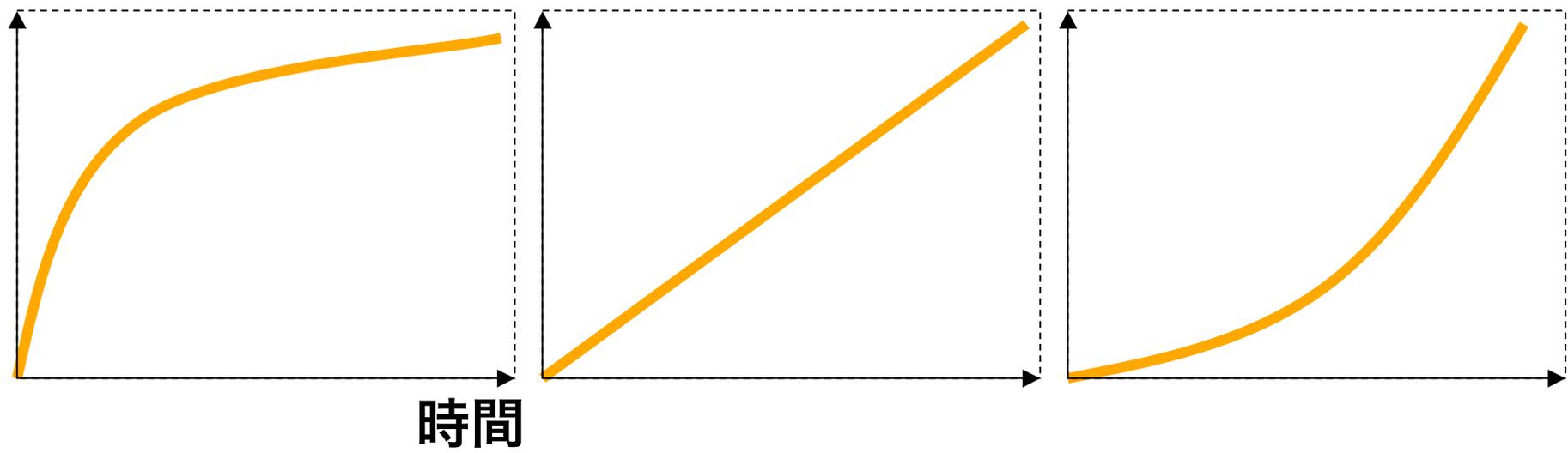


難しさ



難しさ



苦勞が大きいほど、達成時の感動も大きい。

ソースファイルの分割(1)

- ソースプログラムを複数のファイルに分割してもよい
 - 例) `gcc -o lego main.c legolib.c`
 - `main`関数の定義
 - `lego_read`と`lego_write`の定義
 - 関数呼び出しより前にプロトタイプ宣言が必要 (ヘッダファイルを使うと便利)
 - `int lego_read (int fd, int command, unsigned char arg[], size_t nbytes);`

ソースファイルの分割(2)

□ (狭義の)コンパイルとリンクを別に行ってもよい

- 例) `gcc -c main.c` `main.o`を生成
 `gcc -c legolib.c` `legolib.o`を生成
 `gcc -o lego main.o legolib.o`

□ makeの接尾辞ルールを使うと便利

- `.SUFFIXES: .c .o`
 `.c.o:`
 `gcc -c -o $@ $<`

S-Record



2007.11.26
情報システム工学実験2

S-Recordファイルの生成

□ % cat reset.S

```
                .section .text
                .p2align 1
                .global  _start
_start:         mov.w   @0,r0
                jsr    @r0
                .section .data
                .string "Do you byte, when I knock?"
                .end
```

```
% h8300hms-elf-as -o reset.o reset.S
```

```
% h8300hms-elf-ld -o reset.mot reset.o -Ttext 0x8000 --oformat srec
```

```
% cat reset.mot
```

```
S00C000072657365742E6D6F7452
```

```
S10980006B0000005D00AE
```

```
S1138006446F20796F7520627974652C20776865D2
```

```
S10F80166E2049206B6E6F636B3F00000E
```

```
S90380007C
```

S-Record

- バイナリデータを16進数文字列で表現。
- 課題30でダウンロードした機械語プログラムをS-Recordで表すと:

S00C000072657365742E6D6F7452

S10980006B0000005D00AE

S1138006446F20796F7520627974652C20776865D2

S10F80166E2049206B6E6F636B3F00000E

S90380007C

タイプ以外は
2文字=1バイト
を1組に扱う

タイプ

番地(2~4バイト)

データ

チェックサム

レコード長
(1バイト)

(レコード長以降の合計の
下位1バイトの補数)