

2017年6月10日 関西言語学会ワークショップ 参考資料

文書作成者：中野陽子（関西学院大学）

## I. DMDX を使用する準備

I. DMDX を下記のサイトからダウンロードして、インストールする。

<http://www.u.arizona.edu/~kforster/dmdx/dmdx.htm>

スタートメニューから DmDX のファイルに DmDX、DmDX Aotomode と TimeDX が入っているかをチェックする。Windows7 は DMDX, Windows10 では DmDX Automode を使用する。

## II. モニターとソフトの調整

DMDX を起動し、Help をクリックする。

左の目次から Using TimeDX を選ぶと、モニターの調整や音声を使うための調整のため、Step 0. から Step 11 までについての案内がある。

音声を使用しない場合は取りあえず、Step 5 まで行っておく。（下記の1と2を実施しておかないと、プログラムがうまく動かない。）

Start メニューから DMDX のファイルを探して、その中に入っている TimeDX を起動する。

### 1 File/ Select Video Mode

「メニューバーの File/ Select Video Mode / Select Desktop Video Mode / Just Select it をクリックし、 Do Test をクリックし、結果を見る。

Done をクリックして閉じる。

### 2 Advanced Tests/Time Video Mode

メニューバーの Advanced Tests から Time Video Mode を選んで、クリックすると“Refresh Rate ##”と表示される。しばらくすると Video Mode と先ほど選んだ解像度が一番上に表示されたボックスが現れる。Save Last Used values in Registry をクリックして、Registry の4つ数値を保存する。Do Test をクリックし、適当なところで Esc か Enter をクリックする。Done を押して終了する。

### 3 Basic Tests/Refresh Rate

コマンドを書くとき、使用する機器のリフレッシュレートから、刺激の提示時間などを計算することがあるため、確認しておく方がよい。

メニューバーの Basic Tests から Refresh Rate を選び、Do Test をクリックする。

解像度の下に refresh rate が表示される。ラップトップは約 16.7ms、デスクトップは約 11.8ms。

例： fd 30

<fd N> Set Frame Duration 単語などを呈示する時間を指定するコマンド

N はミリ秒ではなく、tick 数またはリフレッシュレートに基づいて表される。たとえば、単語を 500ms 呈示したい場合、リフレッシュレートが 16.68ms なら N は  $500\text{ms} \div 16.7\text{ms} \approx 29.94$  約 30 ticks になる。

## II. プログラムの作成

マイクロソフトワードのリッチテキストで書く。保存するときに、ファイル形式 .rtf を選択する。

単語のフォントサイズや種類は、リッチテキストのフォントサイズや種類がそのまま反映される。

### 語彙性判断課題

凝視点 (500ms) → 空白 (500ms) → プライム語 (500ms) → ターゲット語 (500ms) でターゲット語に対する語彙性判断の反応時間を測定する場合。

プログラムの例

Windows 7 は 1 行目に <vm 1366, 768, 768, 32, 59>が入っているが、Windows10 は入れない。(vm : Video Mode モニターの属性に関する記述, 下記のコマンドの説明を参照して下さい。)

ラップトップコンピューター使用

Windows 7 のとき

```
<id "キーボード"> <mr "+Enter"> <mpr "+M"> <mnr "+V"> <vm 1366, 768, 768, 32, 59> <dbc 000000000> <dwc 255255255> <cr> <n 5> <msfd 500> <d 60> <t 4000> <nfb>
```

0 "start";

-1 "+" / / <ms% 500> "にがい" / <ms% 500> \* "らでい"/;

+2 "+" / / <ms% 500> "かくまう" / <ms% 500> \* "うるおす"/;

+3 "+" / / <ms% 500> "あつがる" / <ms% 500> \* "さむがる"/;

+4 "+" / / <ms% 500> "さした" / <ms% 500> \* "けあた"/;

```
+5 "+" / / <ms% 500> "いたわる" / <ms% 500> * "なみだ"/;
0 "以上です。";
```

Windows 10 のとき

```
<id "キーボード"> <mr "+Enter"> <mpr "+M"> <mnr "+V"> <dbc 00000000> <dwc
255255255> <cr> <n 5><msfd 500> <d 0> <t 4000> <nfb>
```

```
0 "start";
```

```
-1 "+" / / <ms% 500> "にがい" / <ms% 500> * "らでい"/;
+2 "+" / / <ms% 500> "かくまう" / <ms% 500> * "うるおす"/;
+3 "+" / / <ms% 500> "あつがる" / <ms% 500> * "さむがる"/;
+4 "+" / / <ms% 500> "さした" / <ms% 500> * "けあた"/;
+5 "+" / / <ms% 500> "いたわる" / <ms% 500> * "なみだ"/;
0 "以上です。";
```

## コマンドの説明

id: Select and Input Device.

キーボードや外付けの機器、各機器のボタンなどの名前を指定する。名前は TimeDX で調べる。TimeDX の中にある Basic Tests/Input Test を起動させる。上に機器名が表示される。機器名を選び Test をクリックする。使用機器に付属しているボタンなどのリストが表示される。キーボード上のスペースキーなど、実験に使用するキーをクリックしてみると、該当する名前（たとえば、Enter キーを押すと、#28, Enter と表示される。番号や名前は各機器によって異なる。ゲームパッドでも製品番号が異なれば、名前も異なる。）がハイライトされる。機器の名前、使用するキーの名前を控えておく。

機器の名前：キーボード

```
<id "キーボード"> のように書く。
```

""でくくった場合は 使用するキーの名前も<>の中のものを使用する。

```
<id "キーボード"><mr "+Enter">
```

<id #キーボード>のように書いたら、#のついた番号の方を使用する。

```
<id #キーボード><mr +#28>
```

mr Map Button to Request +はキーなどを押す、-は引くことを表す  
<mr "+Enter"> Enter キーを押すと実験が始まる。

mpr Map button to Positive Response  
<mpr "+M"> Yes 反応には M のキーを押す。

mnr Map button to Negative Response  
<mnr "+V"> No 反応には V のキーを押す。

vm Video Mode モニターの属性  
<vm 1366, 768, 768, 32, 59>

N<sub>1</sub> the horizontal size,  
N<sub>2</sub> is the optionally reprogrammed vertical size,  
N<sub>3</sub> is the real vertical DirectX video size that is re-programmed to N<sub>2</sub>'s size,  
N<sub>4</sub> is the color depth  
and N<sub>5</sub> is the refresh rate

dbc Default Background Color 背景色を指定する。  
<dbc 00000000> 背景色を黒にする。

dwc Default writing color 文字の色  
<dwc 255255255> 文字の色を白にする

cr continuous running 継続して実験を実施する。

n 試行数  
<n5>5 試行

fd Set frame duration デフォルトの提示時間を指定する (tick 数またはレトレースの間隔で TimeDX の Refresh Rate で調べる)  
<fd 30> 刺激を 30ticks 分呈示

6 行目以降の / で区切られた部分が frame1 つ分で、各 frame に適用される。  
Frame 内に呈示時間を指定しておく、そちらが優先される。  
msfd はミリ秒で指定する場合。

ms Frame Duration in Milliseconds

<ms% 500> ms% ミリ秒で指定したい場合。

d Delay from request to next item's display request (ボタンを押すなど) があってから、次の item を何 tick 遅らせるか。

<d 60> ボタンを押したら、1秒遅れて次の刺激が出る。上記の例ではボタンを押すのは実験開始前に Start の画面が出たときと、各試行の最後で語彙性判断をするとき。Start の画面を見てボタンを押して、1秒後に最初の凝視点が呈示される。各試行の終わりにボタンを押すと、その1秒後に次の試行の凝視点が呈示される。

t Subject response Time

<t 4000> 最大4秒待つ

<nfb> Disable feedback display フィードバックなし。

0 “ ”に書かれたことを表示する。

; コマンドの終了

0 “Start”; Start と表示される。

-1 "+" / / <ms% 500> "にがい" / <ms% 500> \* "らでい" / ;

- 語彙性判断の正解が negative という意味。

+ の場合は positive という意味。

1 は任意の番号。

“+” / : 凝視点 +

/ : frame の区切り

/ / : 空白の frame

<ms% 500> "にがい":にがい を 500 ミリ秒呈示

\* : Turn RT Clock On ここから反応時間の測定を開始する。

<ms% 500> \* "らでい" :らでい を 500 ミリ秒呈示し、その onset から反応時間の測定を開始する。

プログラムは MsWord などに書き、リッチテキストで保存する。その際、呈示したい種類のフォントとサイズにしておく。たとえば、プログラムを Courier New の 36point で書いて保存すると、その種類とサイズで呈示される。

# コマンドは DmDX の Help 機能に掲載されている。

### III. DmDX での実験実施

1. Windows 7 は DmDX を、Windows 10 は DmDX Automode を起動する。
2. ブラウザーから保存したリッチテキストを探して開く。
3. Syntax Check をクリックすると、リッチテキストで入力したコマンド等が表示される。
4. subject の欄に任意の番号を入力する。(たとえば、被験者番号の 100\_1A)
5. Run をクリックするとプログラムで指定した最初の frame の Start が呈示される。
6. 上記の例では Enter を押すと実験が開始される。

終了したら、Esc キーを押す。データを Save するかどうか尋ねてくる。このとき、「はい」を押せばデータが保存されるが、「いいえ」を選ぶとデータが残らないので注意する。

#### Masked Priming Task のプログラムの例

ラップトップコンピューター使用

Windows 7 で USB 接続のゲームパッドを使用するとき

```
<id "Elecom Wired Gamepad"> <mr "+C"> <mpr "+F"> <mnr "+E"> <vm 1366, 768, 768, 32, 59><dbc 00000000> <dwc 255255255> <cr> <n 5> <d 60><fd 30> <t 4000> <nfb>
```

```
0 "start";
```

```
-1 "+" / /"#####" / <ms% 50> "にがい" / <ms% 500> * "らでい"/;
```

```
+2 "+" / /"#####" / <ms% 50> "かくまう" / <ms% 500> * "うるおす"/;
```

```
+3 "+" / /"#####" / <ms% 50> "あつがる" / <ms% 500> * "さむがる"/;
```

```
+4 "+" / /"#####" / <ms% 50> "さした" / <ms% 500> * "けあた"/;
```

```
+5 "+" / /"#####" / <ms% 50> "いたわる" / <ms% 500> * "なみだ"/;
```

```
0 "以上です。";
```