

# 「情報工学実験 3」 進捗レポート

学生番号： 2031133

氏名： 増田瑞樹

題目：

KCS を用いた 8080 マイコンの音声データインプッタ

詳しいデータ置き場 <https://github.com/nex-finger/jikken3>

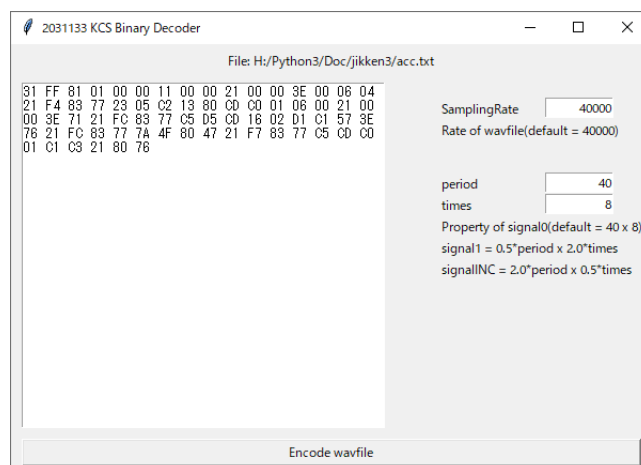
前回から行った作業内容：

前回では作業内容が簡単そうなものから、というスタンスだったが、論理的に考えてスタートから考えた方が良かったと思った。例えば、先週出力回路をつくったが、わざわざ Arduino でテストプログラムをくまなければならなかった(これは実装するものとは全然形式が違うので移植したりはできない)。よって、今週は機械語を wav ファイルに変換するソフトをつくることにした。

実装に processing は使わない。Processing は Java に似たコードを書けらしいが、Java ではない。クラス内の静的変数は認められておらず、ユーザが明示的に純粋 java モードに切り替える必要がある。自分のプールを広げるためにも、今まで 10 日でわかる系の参考書を一冊しか読まなかった python を利用した。GUI の実装には王道の tkinter を利用した。

ソースコードはデータ置き場(<https://github.com/nex-finger/jikken3>)にあるが、スクリーンショットだけで説明すると以下ようになる。(写真 1)

写真 1 KCS Binary Decoder の様子



ウィンドウ左側に変換前のソースを表示し、ここでコードの編集もできる。課題だったビットレートについても、逐次変更できるようにした。写真 1 の場合はサンプリングを 1 秒間に 40000 回行っており、電圧が高いのを 40 サンプル、低いのを 40 サンプル行い、それを 8 回繰り返して 1 ビットの情報を持つので、

$$\frac{1}{40000} * (40 + 40) * 8 = 0.016s$$

となり, 0.016 秒で 1 ビット, 逆数を考えると 62.5bps になっている.

次回までの作業予定:

- ・wav ファイル作成プログラムの作成  
→ Python でつくる
- ・音声入力装置の作成  
秋月で必要部品を購入、ユニバーサル基板うえお重ねて実装する
- ・Arduino の割り込み処理の理解, 実装  
MsTimer2 というライブラリを使う予定
- ・Arduino での信号読み取りと訂正  
なるべく短いクロック数で実装したい
- ・比較プログラムの完成  
なるべく小さいサイズでの実装

要検討事項 (To Do リスト):

- ・python おそ  
インタプリタの性質上, 30 秒の wav ファイルを出力に 3.6GHz の CPU で 20 分から 25 分かかる. → PulInstaller を用いて exe 化?