EGR680 High Level Implementation on FPGA

Laboratory 07

Python Functions, File I/O, and Object-Oriented Programming

Professor: Dr. C. Parikh

Student: Dimitri Häring

October 24, 2018

Contents

1 Introduction **3**

2 Design **3**
2.1 SDK 3
2.2 ATM machine 3

3 Conclusion **4**

4 Appendix **5**
4.1 Python code Part III 5

1 Introduction

The goal of laboratory six is to familiarize the student with the programming language Python. Therefore, in the first part the required software is installed. Second part provides a simple code to test the installation. Third part is a task for the student to program an ATM machine.

2 Design

In this section the design and decisions that where made to achieve the laboratory are discussed.

2.1 SDK

As software development kit (SDK) the Spyder IDE is used that is part of the anaconda navigator which provides the Python 3.7 interpreter and default packages. This installation makes it very easy for a beginner to start with python programming. The Spyder IDE is shown in Figure 1. Highlighted are the python version on top, the debug run button to execute a script in console, and the console to make input and outputs to the script. Notice tis is not a program it is a script.

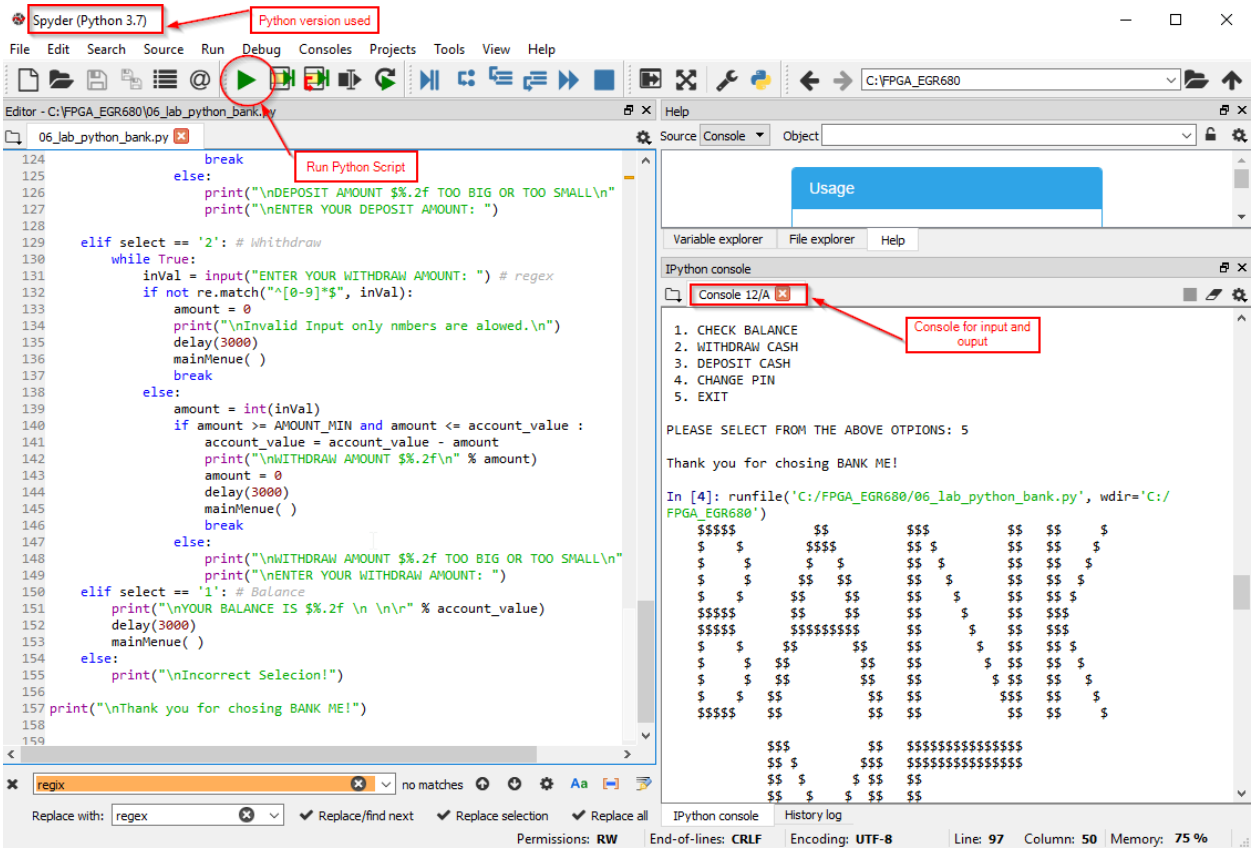

Figure 1: Shows the Spyder IDE with the most important information highlighted.

2.2 ATM machine

Part III states to program an ATM machine with input and output in the command line or python IDLE environment. The user input validation is done with with regex so that each input is restricted to a specific set of characters in combination with a while loop, as shown in Listing 1. The function used for regex is re.match() and is imported with the package re. To build and test a regex a vast amount of on-line tools are available as the one the author used was regex101.com.

```python
# PIN validation
while True:
  inVal = input("Please enter your PIN: ") # regex
  if not re.match("^[0-9]{4}$", inVal):
    print ("Error! Make sure you only use numbers from 0-9 in PIN")
    inVal = 'Z'
  else:
    if inVal == PIN :
      #print("\nCorect PIN") # debug only
      inVal = 'Z'
      break
    else:
      print("\nInvalid PIN!")
      inVal = 'Z'
```

Listing 1: Python code for input validation.

Python allows to make functions as in C, a function definition is shown in Listing 2. First the time package was imported but with the function sleep the problem is it stop's the thread before the print output to the console was made. therefore a simple for loop was used and approximated to one second which is not precious but does the job. Furthermore a variable in python can be referenced to any data type.

```python
def delay( msec ):
  cnt = 0
  while cnt < msec:
    cnt += 0.0001
    return
```

Listing 2: Python code for delay.

The ATM machine is first welcoming the user with the name of the Bank and a Welcome message. Below the user is ask to identify himself with his personal identification number (PIN) which will allow the user access to his account if entered correctly. Followed by prompt of the Main Menu as shown in Figure 2. The user can use the keyboard to make a selection of the task he wants do to. The figure shows highlighted where the user made an input selection of no. five and the output generated by the script exiting the users individual secure area. As long as the user does not exit he can proceed with all actions presented in the main menu. A demonstration of the functionality will be given to the instructor in class.

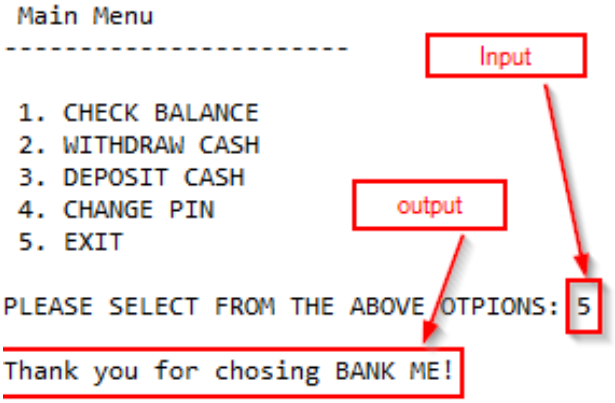

Figure 2: Main menu with highlighted user input and program output.

3 Conclusion

The lab demonstrates the use of the python as simple and fast scripting language that allows access to vast number of packages that allows an decreased development time. The syntax is easy to learn but it is possible to lose the overview by having too many continuations of statements.

4 Appendix

The appendix contains code listening and other large information parts that contain partial or complete relevance to the reports topic.

4.1 Python code Part III

```
# -*- coding: utf-8 -*-
"""
Spyder Editor

This is a temporary script file.
"""
import time
import re

AMOUNT_MIN = 0
AMOUNT_MAX = 1000

user = 0
PIN  = "1234"
newPIN = "99999999999999"
select = 0
account_value = 0
amount = 0
inVal = 'Z'

def mainMenue( ):
    print(" Main Menu")
    print("---------------------- \n")
    print(" 1. CHECK BALANCE")
    print(" 2. WITHDRAW CASH")
    print(" 3. DEPOSIT CASH")
    print(" 4. CHANGE PIN")
    print(" 5. EXIT ")
    return

def delay( msec ):
    cnt = 0
    while cnt < msec:
        cnt += 0.0001
    return

# Star program
print("\n     $$$$$           $$           $$$           $$    $$      $   ")
print("     $     $         $$$$          $$ $          $$    $$     $    ")
print("     $      $        $    $        $$  $         $$    $$    $     ")
print("     $      $       $$    $$       $$   $        $$    $$   $      ")
print("     $     $       $$      $$      $$    $       $$    $$ $        ")
print("     $$$$$        $$      $$      $$     $      $$    $$$         ")
print("     $$$$$        $$$$$$$$$$      $$      $     $$    $$$         ")
print("     $     $     $$        $$     $$       $    $$    $$ $        ")
print("     $      $    $$          $$    $$        $   $$    $$   $      ")
print("     $      $    $$          $$    $$          $ $$    $$    $     ")
print("     $     $    $$            $$   $$           $$$    $$     $    ")
print("     $$$$$     $$            $$   $$             $$    $$      $   \n")
print("               $$$           $$   $$$$$$$$$$$$$$   ")
print("               $$ $         $$$   $$$$$$$$$$$$$$    ")
print("               $$  $       $ $$   $$       ")
print("               $$   $     $  $$   $$      ")
print("               $$    $   $   $$   $$      ")
print("               $$     $$     $$   $$$$$$$$$$$$$$$        ")
print("               $$            $$   $$$$$$$$$$$$$$$        ")
print("               $$            $$   $$       ")
print("               $$            $$   $$       ")
print("               $$            $$   $$       ")
```

```
print("              $$            $$    $$$$$$$$$$$$$$$   ")
print("              $$            $$    $$$$$$$$$$$$$$$   ")
print("-------------------------------------------------------------")
print("                     Welcome to Bank Me  ")
print("=============================================================")

# PIN validation
while True:
    inVal = input("Please enter your PIN: ") # regex
    if not re.match("^[0-9]{4}$", inVal):
        print ("Error! Make sure you only use numbers from 0-9 in PIN")
        inVal = 'Z'
    else:
        if inVal == PIN :
            #print("\nCorect PIN") # debug only
            print("\n")
            mainMenue()
            inVal = 'Z'
            break
        else:
            print("\nInvalid PIN!")
            inVal = 'Z'

# Main menu
while  select != "5":

    inVal = input("PLEASE SELECT FROM THE ABOVE OTPIONS: ")
    if not re.match("^[1-5]*$", inVal):
        print ("Error! Make sure you only use numbers from 1-5 in selecion")
    else:
        select = inVal
#
    if select == '5': # exit
        exit
    elif select == '4': # change PIN
        while True:
            inVal = input("ENTER YOUR NEW PIN: ") # regex
            if not re.match("^[0-9]{4}$", inVal):
                print ("Error! Make sure you only use numbers from 0-9 in PIN")
                inVal = 'Z'
            else:
                PIN = inVal
                inVal = 'Z'
                print("\nYOUR NEW PIN IS", PIN, "\n")
                delay(3000)
                mainMenue()
                break
    elif select == '3': # deposit
        while True:
            inVal = input("ENTER YOUR DEPOSIT AMOUNT: ") # regex
            if not re.match("^[0-9]*$", inVal):
                amount = 0
                print("\nInvalid Input only nmbers are alowed.\n")
                delay(3000)
                mainMenue( )
                break
            else:
                amount = int(inVal)
                if amount > AMOUNT_MIN and amount <= AMOUNT_MAX :
                    account_value = account_value + amount
                    print("\nDEPOSIT AMOUNT $%.2f\n" % amount)
                    amount = 0
                    delay(3000)
                    mainMenue( )
                    break
                else:
                    print("\nDEPOSIT AMOUNT $%.2f TOO BIG OR TOO SMALL\n" % amount)
                    print("\nENTER YOUR DEPOSIT AMOUNT: ")
```

```
128
129     elif select == '2': # Whithdraw
130         while True:
131             inVal = input("ENTER YOUR WITHDRAW AMOUNT: ") # regex
132             if not re.match("^[0-9]*$", inVal):
133                 amount = 0
134                 print("\nInvalid Input only nmbers are alowed.\n")
135                 delay(3000)
136                 mainMenue( )
137                 break
138             else:
139                 amount = int(inVal)
140                 if amount >= AMOUNT_MIN and amount <= account_value :
141                     account_value = account_value - amount
142                     print("\nWITHDRAW AMOUNT $%.2f\n" % amount)
143                     amount = 0
144                     delay(3000)
145                     mainMenue( )
146                     break
147                 else:
148                     print("\nWITHDRAW AMOUNT $%.2f TOO BIG OR TOO SMALL\n" % amount)
149                     print("\nENTER YOUR WITHDRAW AMOUNT: ")
150     elif select == '1': # Balance
151         print("\nYOUR BALANCE IS $%.2f \n \n\r" % account_value)
152         delay(3000)
153         mainMenue( )
154     else:
155         print("\nIncorrect Selecion!")
156
157 print("\nThank you for chosing BANK ME!")
```

Listing 3: Python code for an ATM.