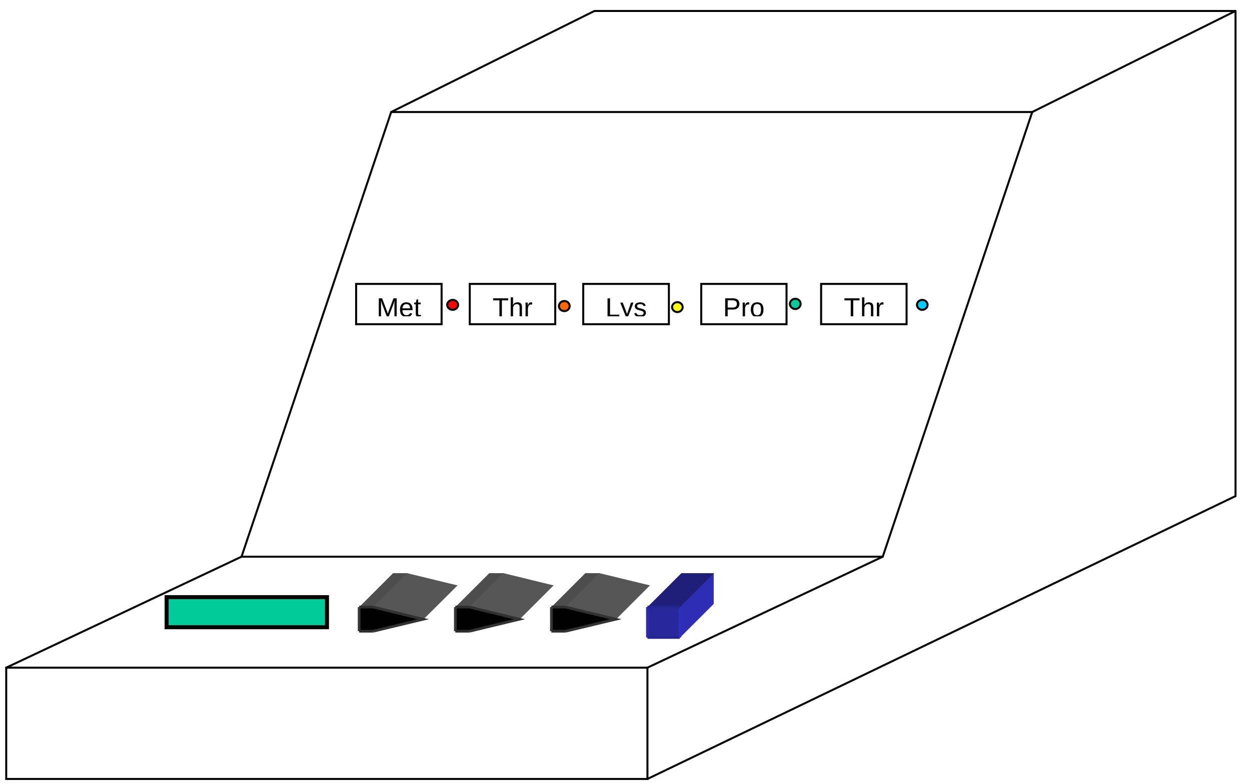
The Codon Decoder

A SMART 2005 Project



Submitted by:
Vera E. Mihalcik
Marta Valle Secondary School
and
Cameron Jahn
Midwood High School

1. INTRODUCTION

In each of our cells, proteins can be considered the building blocks of life. From hormones to antibodies to enzymes, proteins are involved in almost every basic function occurring within an organism. Genes, located along the strands of an organism's DNA, instruct the cells how to produce these proteins. These genes instruct the ribosomes, also known as the cell's protein "factories," which proteins to create and how to create them (see Fig. 1). From a simple list of twenty amino acids, tens of thousands of proteins are created, as each protein has its own unique sequence of amino acids. The goal of this project is to visualize the process of protein synthesis and create an exciting mode of realizing a process that is often intangible to students because they cannot see it occurring.

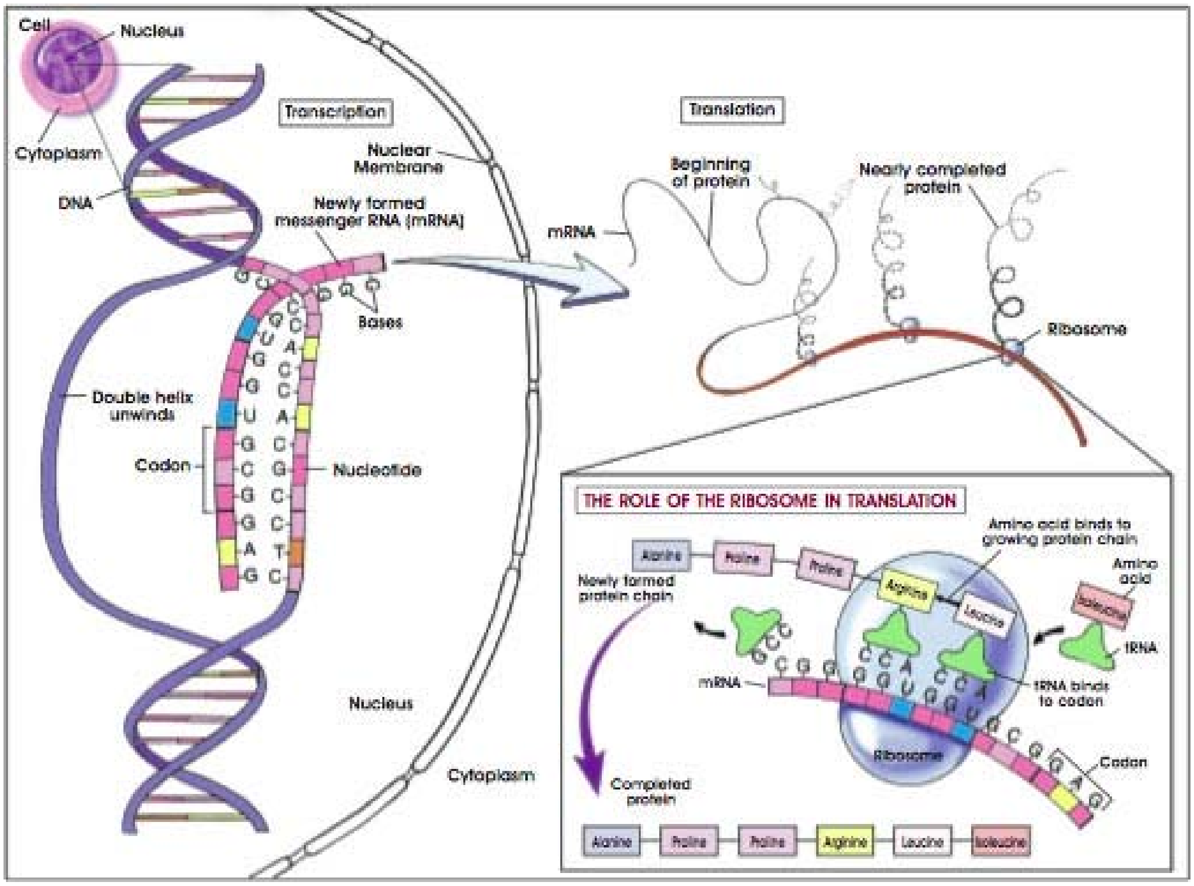


Figure 1: Overview of protein synthesis.

2. BACKGROUND

DNA, deoxyribonucleic acid, can be found inside the cell of every living organism. Inside this tiny, yet complicated molecule, is the blueprint for life. DNA is a long chain made up of nucleotide subunits; these subunits consist of three components, a five-carbon sugar (deoxyribose), a phosphate, and a nitrogenous base, adenine, guanine, cytosine, thymine (found only in the DNA), and uracil (found only in the RNA). The sugars and phosphates act as the “backbone” to the DNA structure, holding the molecule together and giving it a sturdy, rugged support; whereas the nucleic acids act as the “library” of information that can be found within the strand. The sequence of the nucleotides determines a code. This code, made up of what are known as genes, has many functions, such as controlling cell activity.

To make use of the genetic information found within the DNA molecule, organisms must convert the information into proteins. Protein synthesis, the process through which proteins are created from amino acids, occurs in the ribosomes, tiny organelles found within the cytoplasm of the cell. The genetic information must be transported from the DNA molecule out of the nuclear membrane through the cytoplasm to the ribosome where the information is read, decoded, and realized. The end result is a chain of amino acids that when linked in the correct order form a protein (see Fig. 2).

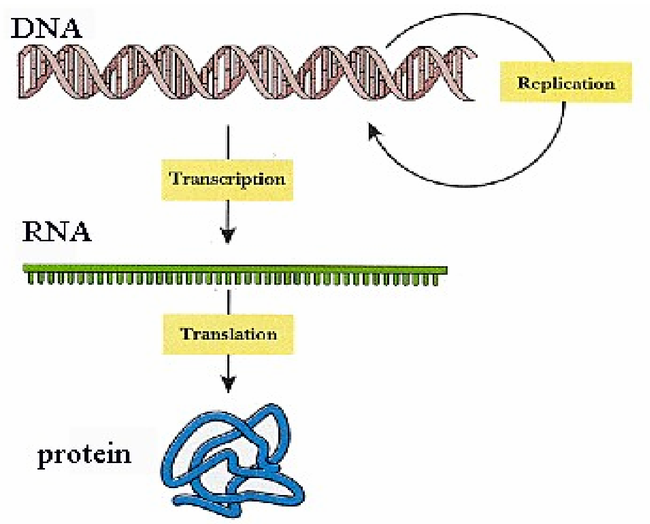


Figure 2: Overview of protein synthesis.

Because the instructions for the construction of each protein are contained in the DNA, the information must be transported from the DNA to the ribosome. This transfer is completed with the help messenger ribonucleic acid molecule (mRNA). This molecule “copies” a specific sequence of nitrogenous bases on the DNA through a process called transcription. This length of bases, known as a gene, contains the information needed to make a protein. As mentioned before, each protein is coded for by a specific gene found on the DNA; each mRNA molecule transcribes one gene at a time.

After the information is copied onto the mRNA, the molecule travels through the cell cytoplasm to the ribosome. At the ribosome, a second ribonucleic acid molecule, the transfer RNA (tRNA), helps to ensure that the information is read and decoded properly. The tRNA assists in the process known as translation, the synthesis of a protein using the DNA code found in the mRNA as a template. During this process, the nucleotide sequence, the “language” of the DNA, must be translated into a protein “language”, the sequence of amino acids that will eventually create the molecules. The sequence of the different nitrogenous bases determines, or code for, specific amino acids.

At the ribosome, the sequence is "read" in groups of three nitrogenous bases. These triplet patterns are known as codons (see Fig. 3).

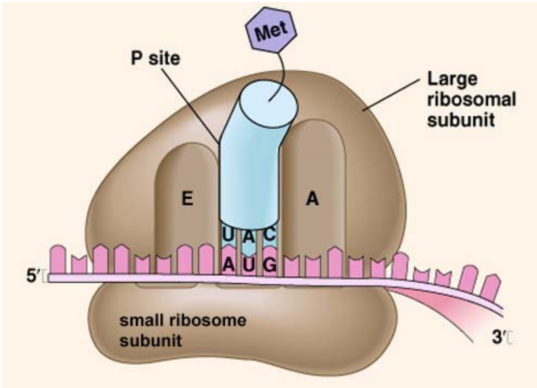


Figure 2: The ribosome translates the mRNA into amino acids with the help of the tRNA.

For example, the sequence adenine-uracil-guanine (AUG), codes for the amino acid Methionine (MET). (For a complete list of codons, refer to Appendix A). As the sequence of codons passes through the ribosome, a chain of amino acids is created; this is the start of a protein. The tRNA molecules help to ensure that the correct amino acids are linked as the codons are translated. Moreover, found within the chain are "start" and "stop" codons which essentially tell the ribosome when to start protein production and when to cease linking the proteins together. Thus, the genes found on the DNA give the specific instructions as to the start, sequence, and stop of the creation of the amino acid chains.

3. EXPERIMENTAL PROCEDURE

3.a. Goals

The Codon Decoder serves as a learning tool for students studying protein synthesis and the interactions between different cellular molecules. It enables the students to learn model abstract biological concepts by making them tangible and engage. The students can use this game to better understand the construction of amino acids from the DNA code delivered by the RNA to the ribosomes. As the ribosome reads the mRNA sequence, it creates amino acids that will later be linked together to form proteins. *The Codon Decoder* quizzes the students on their knowledge of the amino acids by having the students act as tRNA molecules. This allows the student take part in hands-on activities while budding their interest in an often mundane topic. *The Codon Decoder* also promotes the use of technology in the classroom and will increase student interest in science and engineering.

3.b. Procedure

The Codon Decoder consists of a control box (see Fig. 3) composed of three knobs connected to potentiometers. The voltage (reading) passing through the potentiometers is

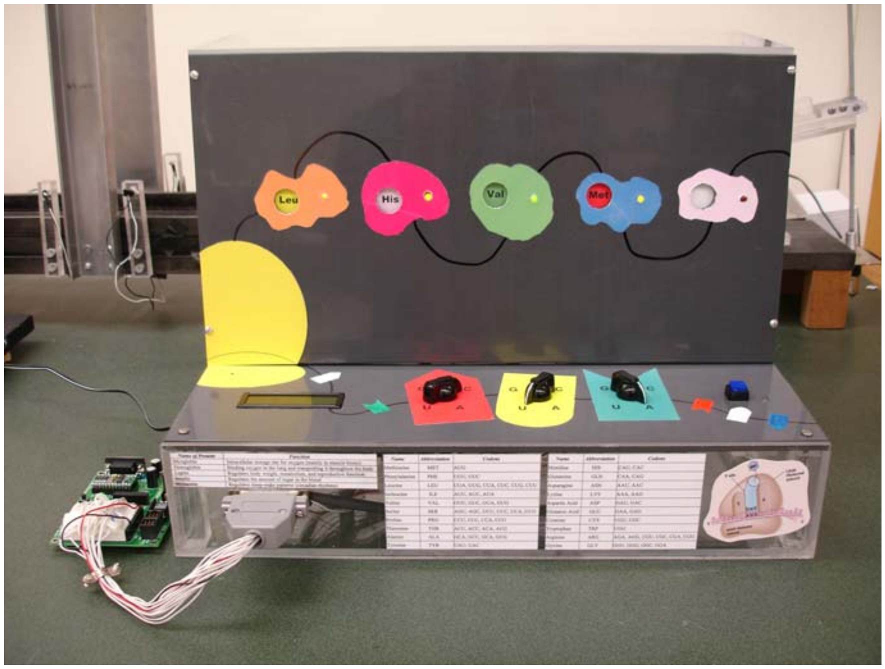


Figure 3: *The* (completed) *Codon Decoder*

recognized by the microcontroller and is assigned a specific nitrogenous base (A, U, C, or G), and the combination of three nitrogenous bases comprises a specific amino acid (See Fig. 4). At the start of the game, the program selects a random protein to be created. The LCD then reads the name of the first amino acid in the sequence. The student must then turn the knobs (of the potentiometers) to the correct sequence of nitrogen bases. With the press of a button, the program (see Appendix C) checks to see if the student identified the accurate combination of bases that code for the specific amino acid.

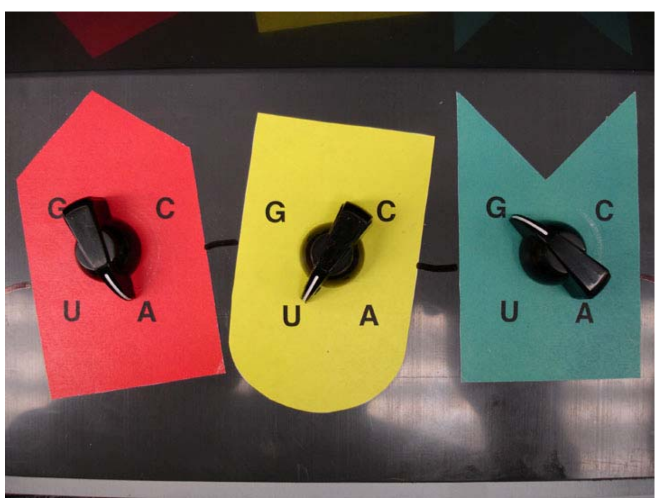


Figure 4: The potentiometer knobs used to “code” for amino acids.

With each correct response, a pulse is sent to a servo motor that turns a wheel that displays the amino acid that the student has just created (see Fig. 5); an LED is illuminated; and a buzzer makes a “happy sound” (high frequency). With each correct response, the amino acid sequence grows- another wheel turns, another LED illuminates, and a buzzer sounds each time. After the student completes the fifth correct amino acid in a row, the buzzer sounds three times, the LEDs blink, and the LCD reads, “Congratulations! You have started (the name of the protein).” At that time, the program resets, and the student can play again with a different protein. If the student incorrectly creates a protein at any time, then the program resets. The buzzer makes a groan (low frequency), the LEDs turn off, and the motor reset to a blank start position. For a complete representation of the circuitry diagrams, see Appendix B.

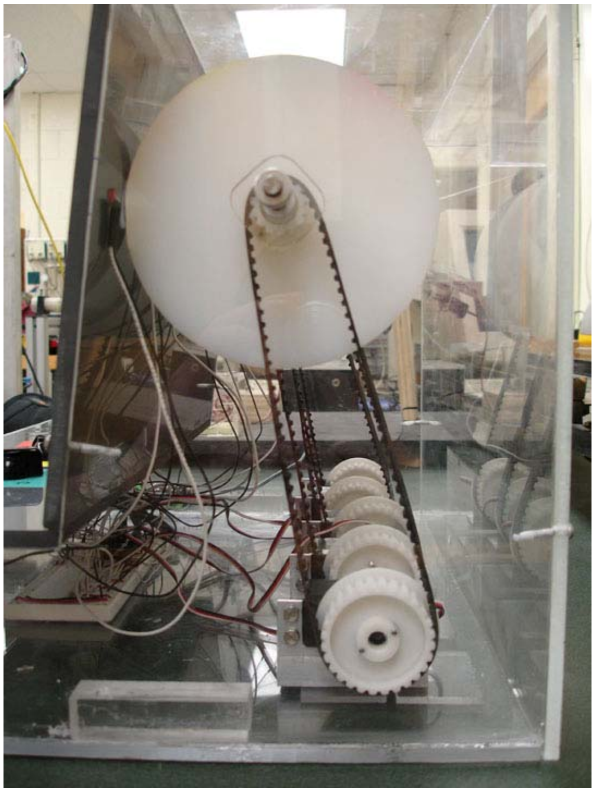


Figure 5: The gear and wheel mechanism of *The Codon Decoder*.

3.c. Discussion

The Codon Decoder can be used as an assessment tool in the classroom. Instead of giving a worksheet asking students to write the codes, they can use the knobs and memorize the codes while having fun. Teachers can also have students use the internet to research the different amino acid sequences that comprise each protein; this can get students motivated to learn not only the structure but the function of each protein (see Fig. 6).

Name of Protein	*Sequence of Amino Acids*	*Function*
Myoglobin	Met-Leu-Phe-Lys-Lys	Intracellular storage site for oxygen (mainly in muscle tissue)
Hemoglobin	Met-Val-His-Leu-Thr	Binding oxygen in the lung and transporting it throughout the body
Leptin	Met-Asp-Thr-Lys-Thr	Regulates body weight, metabolism, and reproductive function
Insulin	Met-Thr-Lys-Pro-Thr	Regulates the amount of sugar in the blood
Melatonin	Met-Val-Phe-Val-Val	Regulates sleep-wake patterns (circadian rhythms)

Figure 6: Protein Information

With some modifications to the program, *The Codon Decoder* can also be used as a reference tool. The student could turn the knobs to a specific codon (triplet), and the program would recognize and then display the amino acid on the LCD screen.

In terms of the actual mechanism, ideally, the program would have a large library of proteins from which to randomly select. However, because of space limitations on the BS2 (and the number of conditional statements in the program), *The Codon Decoder* only stores five proteins and has a limited number of sounds. In addition, to make a more realistic model, the number of amino acids in the chain could also be lengthened, requiring more LEDs, motors, and wheels.

4. REFERENCES

Online: http://pir.georgetown.edu/pirwww/search/textpsd.shtml (amino acid sequences)

Hallman, Rick. The Living Environment Biology. Amsco School Publications, Inc. New York : 2000.

Appendix A: Amino acid triplet codes.

Name	*Abbreviation*	*Codons*
Methionine	MET	AUG
Phenylalanine	PHE	UUU, UUC
Leucine	LEU	UUA, UUG, CUA, CUC, CUG, CUU
Isoleucine	ILE	AUU, AUC, AUA
Valine	VAL	GUU, GUC, GUA, GUG
Serine	SER	AGU, AGC, UCU, UCC, UCA, UCG
Proline	PRO	CCU, CCC, CCA, CCG
Threonine	THR	ACU, ACC, ACA, ACG
Alanine	ALA	GCA, GCC, GCA, GCG
Tyrosine	TYR	UAU, UAC
Histidine	HIS	CAU, CAC
Glutamine	GLN	CAA, CAG
Asparagine	ASN	AAU, AAC
Lysine	LYS	AAA, AAG
Aspartic Acid	ASP	GAU, GAC
Glutamic Acid	GLU	GAA, GAG
Cysteine	CYS	UGU, UGC
Tryptophan	TRP	UGG
Arginine	ARG	AGA, AGG, CGU, CGC, CGA, CGG
Glycine	GLY	GGU, GGG, GGC, GGA

Appendix B: Circuitry Diagrams

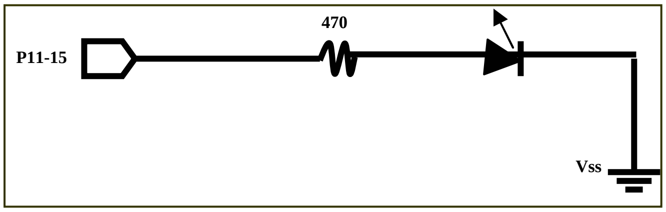


Figure 7: LED Circuit

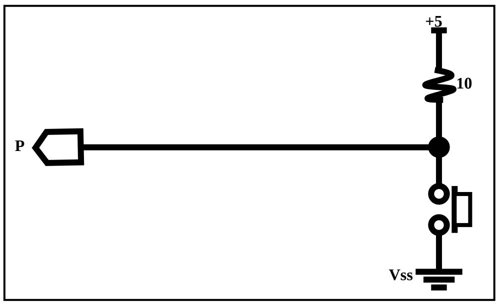


Figure 8: Button Circuit

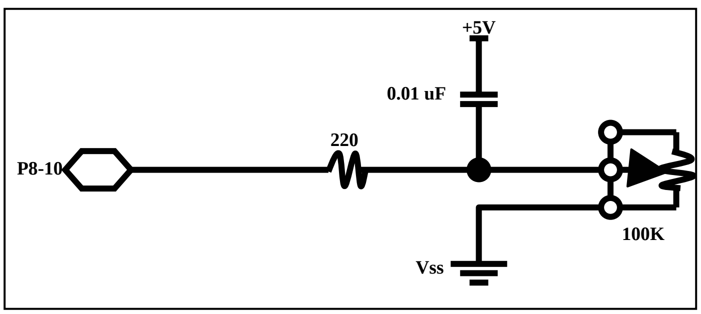


Figure 9: Potentiometer Circuit

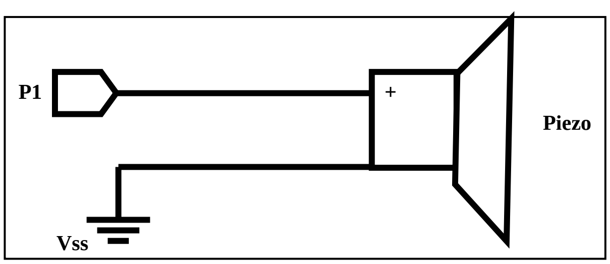


Figure 10: Speaker Circuit

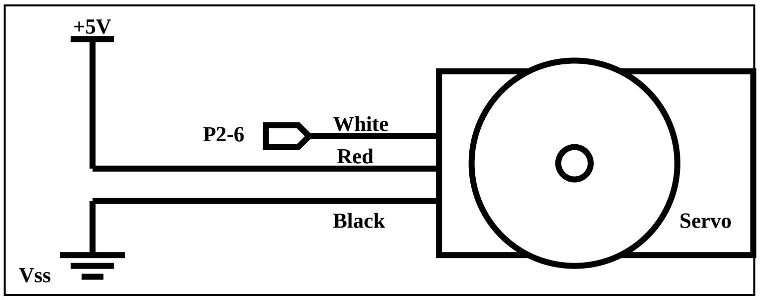


Figure 11: Servo Motor Circuit

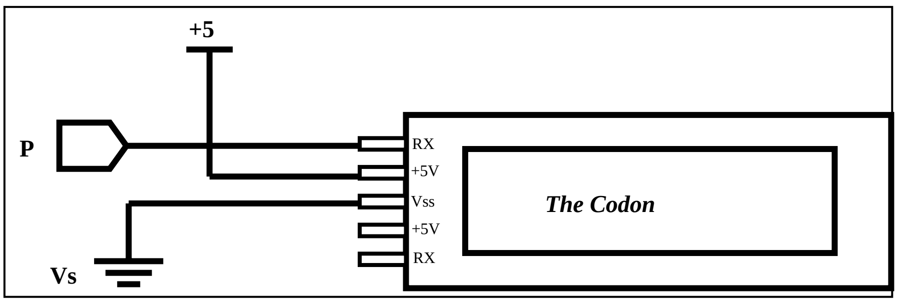


Figure 12: LCD Circuit

Appendix C: Program Code

```
' {$STAMP BS2}
' {$PBASIC 2.5}

' The Codon Decoder

' Define Constants

A     CON 1
C     CON 2
G     CON 3
U     CON 4
Blank CON 500
Met   CON 550
Phe   CON 600
Leu   CON 650
Val   CON 700
Pro   CON 750
Thr   CON 800
His   CON 850
Lys   CON 900
Asp   CON 950
TxPin CON 0
LcdBLoff      CON     $12
LcdOn1        CON     $16
LcdCls        CON     $0C
LcdCR         CON     $0D
Baud19200 CON 32
Speaker CON 1

' Define Variables

PickProtein VAR Word
tone VAR Word
x VAR Word
Amino1 VAR Word
Amino2 VAR Word
Amino3 VAR Word
Amino4 VAR Word
Amino5 VAR Word
Base1 VAR Nib
Base2 VAR Nib
Base3 VAR Nib
ChargeTime1 VAR Word
ChargeTime2 VAR Word
ChargeTime3 VAR Word
LEDs VAR OUTS

' Defin I/O Pins

OUTPUT 2
OUTPUT 3
OUTPUT 4
OUTPUT 5
OUTPUT 6

' Initial Reset

GOSUB Reset2

' Main Program

Main:

HIGH TxPin
  PAUSE 250
  SEROUT TxPin, BAUD19200, ["Push button", $0D, "to begin."]
```

```
PickProtein=11000
RandomLoop:
RANDOM PickProtein
IF IN7=0 THEN Protein
GOTO RandomLoop

Protein:
  IF (PickProtein < 13107) THEN GOSUB Insulin
  IF (PickProtein >=13107 AND PickProtein < 26214) THEN GOSUB Hemoglobin
  IF (PickProtein >=26214 AND PickProtein < 39321) THEN GOSUB Melatonin
  IF (PickProtein >=39321 AND PickProtein < 52428) THEN GOSUB Myoglobin
  IF (PickProtein >=52428 AND PickProtein <= 65535) THEN GOSUB Leptin

Insulin:

PAUSE 1000

Amino1 = Met ' Set protein variables.
Amino2 = Thr
Amino3 = Lys
Amino4 = Pro
Amino5 = Thr

GOSUB Methionine

GOSUB Threonine
GOSUB Correct2

GOSUB Lysine
GOSUB Correct3

GOSUB Proline
GOSUB Correct4

GOSUB Threonine
GOSUB Correct5

GOSUB Congrats
  SEROUT TxPin, BAUD19200, ["Insulin."]
  PAUSE 3000
GOSUB Reset2

Hemoglobin:

PAUSE 1000

Amino1 = Met
Amino2 = Val
Amino3 = His
Amino4 = Leu
Amino5 = Thr

GOSUB Methionine

GOSUB Valine
GOSUB Correct2

GOSUB Create
  SEROUT TxPin, BAUD19200, ["Histidine."]
  GOSUB PotReadings
IF ((Base1=C AND Base2=A AND Base3=U) OR (Base1=C AND Base2=A AND Base3=C)) THEN GOSUB Correct3 ELSE
GOSUB Reset2

GOSUB Leucine
GOSUB Correct4

GOSUB Threonine
GOSUB Correct5

GOSUB Congrats
```

```
  SEROUT TxPin, BAUD19200, ["Hemoglobin."]
  PAUSE 3000
GOSUB Reset2

Melatonin:

PAUSE 1000

Amino1 = Met
Amino2 = Val
Amino3 = Phe
Amino4 = Val
Amino5 = Val

GOSUB Methionine

GOSUB Valine
GOSUB Correct2

GOSUB Phenylalanine
GOSUB Correct3

GOSUB Valine
GOSUB Correct4

GOSUB Valine
GOSUB Correct5

GOSUB Congrats
  SEROUT TxPin, BAUD19200, ["Melatonin."]
  PAUSE 3000
GOSUB Reset2

Myoglobin:

PAUSE 1000
Amino1 = Met
Amino2 = Leu
Amino3 = Phe
Amino4 = Lys
Amino5 = Lys

GOSUB Methionine

GOSUB Leucine
GOSUB Correct2

GOSUB Phenylalanine
GOSUB Correct3

GOSUB Lysine
GOSUB Correct4

GOSUB Lysine
GOSUB Correct5

GOSUB Congrats
  SEROUT TxPin, BAUD19200, ["Myoglobin."]
  PAUSE 3000
GOSUB Reset2

Leptin:

PAUSE 1000

Amino1 = Met
Amino2 = Asp
Amino3 = Thr
Amino4 = Lys
Amino5 = Thr
```

```
GOSUB Methionine

GOSUB Aspartic_Acid
GOSUB Correct2

GOSUB Threonine
GOSUB Correct3

GOSUB Lysine
GOSUB Correct4

GOSUB Threonine
GOSUB Correct5

GOSUB Congrats
  SEROUT TxPin, BAUD19200, ["Leptin."]
  PAUSE 3000
GOSUB Reset2



' Subroutines

Reset2:
HIGH TxPin
  PAUSE 100
  SEROUT TxPin, BAUD19200, [LcdBLoff, LcdOn1, LcdCls]
FREQOUT Speaker, 1000, 150 ' Buzzer
FOR x=1 TO 40 ' Reset wheels
  PULSOUT 2, 500
  PAUSE 20
  PULSOUT 3, 500
  PAUSE 20
  PULSOUT 4, 500
  PAUSE 20
  PULSOUT 5, 500
  PAUSE 20
  PULSOUT 6, 500
  PAUSE 20
NEXT
DIRS=%0000000000000000
GOTO Main

PotReadings:
DO
HIGH 8 ' discharge cap
PAUSE 3 ' for 1 millisecond
RCTIME 8, 1, ChargeTime1 ' read the Pot
HIGH 9 ' discharge cap
PAUSE 3 ' for 1 millisecond
RCTIME 9, 1, ChargeTime2 ' read the Pot
HIGH 10 ' discharge cap
PAUSE 3 ' for 1 millisecond
RCTIME 10, 1, ChargeTime3
IF (ChargeTime1<54) THEN Base1=U
IF (ChargeTime1>=54 AND ChargeTime1<261) THEN Base1=G
IF (ChargeTime1>=261 AND ChargeTime1<486) THEN Base1=C
IF (ChargeTime1>=486 AND ChargeTime1<700) THEN Base1=A
IF (ChargeTime2<42) THEN Base2=U
IF (ChargeTime2>=42 AND ChargeTime2<307) THEN Base2=G
IF (ChargeTime2>=307 AND ChargeTime2<580) THEN Base2=C
IF (ChargeTime2>=580 AND ChargeTime2<700) THEN Base2=A
IF (ChargeTime3<34) THEN Base3=U
IF (ChargeTime3>=34 AND ChargeTime3<247) THEN Base3=G
IF (ChargeTime3>=247 AND ChargeTime3<515) THEN Base3=C
IF (ChargeTime3>=515 AND ChargeTime3<700) THEN Base3=A
IF IN7=0 THEN RETURN
LOOP
```

```
Correct1:
FREQOUT Speaker, 150, 3000
PAUSE 75
FREQOUT Speaker, 500, 3000
  FOR x=1 TO 40
    PULSOUT 2, Amino1
    PAUSE 20
  NEXT
HIGH 11
RETURN

Correct2:
FREQOUT Speaker, 150, 3000
PAUSE 75
FREQOUT Speaker, 500, 3000
  FOR x=1 TO 40
    PULSOUT 3, Amino1
    PAUSE 20
    PULSOUT 2, Amino2
    PAUSE 20
  NEXT
HIGH 12
RETURN

Correct3:
FREQOUT Speaker, 150, 3000
PAUSE 75
FREQOUT Speaker, 500, 3000
  FOR x=1 TO 40
    PULSOUT 4, Amino1
    PAUSE 20
    PULSOUT 3, Amino2
    PAUSE 20
    PULSOUT 2, Amino3
    PAUSE 20
  NEXT
  HIGH 13
RETURN

Correct4:
FREQOUT Speaker, 150, 3000
PAUSE 75
FREQOUT Speaker, 500, 3000
  FOR x=1 TO 40
    PULSOUT 5, Amino1
    PAUSE 20
    PULSOUT 4, Amino2
    PAUSE 20
    PULSOUT 3, Amino3
    PAUSE 20
    PULSOUT 2, Amino4
    PAUSE 20
  NEXT
HIGH 14
RETURN

Correct5:
FREQOUT Speaker, 150, 3000
PAUSE 75
FREQOUT Speaker, 500, 3000

  FOR x=1 TO 40
    PULSOUT 6, Amino1
    PAUSE 20
    PULSOUT 5, Amino2
    PAUSE 20
    PULSOUT 4, Amino3
    PAUSE 20
    PULSOUT 3, Amino4
    PAUSE 20
```

```
    PULSOUT 2, Amino5
    PAUSE 20
  NEXT

HIGH 15

FOR x=1 TO 3
FREQOUT Speaker, 200, 3500
PAUSE 75
NEXT



DelayTime CON 250
DIRS = %1111100000000000 ' make pins outputs
LEDs = %0000100000000000 ' start with one LED on (pin 11)

Go_Forward:
PAUSE DelayTime
LEDs = LEDs << 1
IF (LEDs = %1000000000000000) THEN Go_Reverse
GOTO Go_Forward
Go_Reverse:
PAUSE DelayTime
LEDs = LEDs >> 1
IF (LEDs = %0000100000000000) THEN RETURN
GOTO Go_Reverse

Methionine:

HIGH TxPin
  GOSUB Create
  SEROUT TxPin, BAUD19200, ["Methionine."]
  GOSUB PotReadings
  IF (Base1=A AND Base2=U AND Base3=G) THEN GOSUB Correct1 ELSE GOSUB Reset2
RETURN

Threonine:

GOSUB Create
  SEROUT TxPin, BAUD19200, ["Threonine."]
  GOSUB PotReadings
  IF (Base1=A AND Base2=C) THEN RETURN ELSE GOSUB Reset2

Lysine:

GOSUB Create
  SEROUT TxPin, BAUD19200, ["Lysine."]
  GOSUB PotReadings
  IF ((Base1=A AND Base2=A AND Base3=A) OR (Base1=A AND Base2=A AND Base3=G)) THEN RETURN ELSE GOSUB
Reset2

Phenylalanine:

GOSUB Create
  SEROUT TxPin, BAUD19200, ["Phenylalanine."]
  GOSUB PotReadings
  IF ((Base1=U AND Base2=U AND Base3=U) OR (Base1=U AND Base2=U AND Base3=C)) THEN RETURN ELSE GOSUB
Reset2

Proline:

GOSUB Create
  SEROUT TxPin, BAUD19200, ["Proline."]
  GOSUB PotReadings
  IF (Base1=C AND Base2=C) THEN RETURN ELSE GOSUB Reset2

Valine:

GOSUB Create
```

```
  SEROUT TxPin, BAUD19200, ["Valine."]
  GOSUB PotReadings
  IF (Base1=G AND Base2=U) THEN RETURN ELSE GOSUB Reset2

Leucine:

GOSUB Create
  SEROUT TxPin, BAUD19200, ["Leucine."]
  GOSUB PotReadings
  IF ((Base1=C AND Base2=U) OR (Base1=U AND Base2=U AND Base3=A) OR (Base1=U AND Base2=U AND Base3=G))
THEN RETURN ELSE GOSUB Reset2

Aspartic_Acid:

GOSUB Create
  SEROUT TxPin, BAUD19200, ["Aspartic Acid."]
  GOSUB PotReadings
  IF ((Base1=G AND Base2=A AND Base3=U) OR (Base1=G AND Base2=A AND Base3=C)) THEN RETURN ELSE GOSUB
Reset2

Create:
SEROUT 0, 32, [LcdBLoff, LcdOn1, LcdCls]
SEROUT 0, 32, ["Create ", $0D]
RETURN

Congrats:
SEROUT TxPin, BAUD19200, [LcdBLoff, LcdOn1, LcdCls]
  SEROUT TxPin, BAUD19200, ["Congratulations!"]
  PAUSE 2000
  SEROUT TxPin, BAUD19200, [LcdBLoff, LcdOn1, LcdCls]
  SEROUT TxPin, BAUD19200, ["You've started", $0D]
RETURN
```