รายการอ้างอิง

1. Sonnleitner, B. On-line measurement and validation in biotechnological process. Ini Symp Bioprocess Modelling & Control (JANUARY 1991): 1-14.

2. Papoutsakis, E. T. Equation and Calculation for Fermentations of Butyric Acid Bacteria. Biotech. and Bioeng. 26 (1984): 174-187.

3. McLaughlin, J. K., Meyer, C. L., and Papoutsakis, E. T. Gas Chromatography and Gateway Sensors for On-line State Estimation of Complex Fermentations (Butanol-Acetone Fermentation. Biotech. and Bioeng. 27 (1985): 1246-1257.

4. Votruba, J., Volesky, B., and Yerushalmi, L. Mathematical Model of a Batch Acetone-Butanol Fermentation. Biotech. and Bioeng. 28 (1986): 247-255.

5. PCL-711 PC-Multilab Card User's Manual. by Advantech Co., Ltd. Taiwan: n.p., 1992

6. PCLD-789 Amplifier/Multiplexer Board User's Manual. by Advantech Co., Ltd. Taiwan: n.p., 1992

7. จิรกานต์ เมืองนาโพธิ์, สุวัฒนา พวงเพิกศึก, วรพัฒน์ อรรถยุกติ และ ชัยฤทธิ์ สัตยาประเสริฐ. กระบวนการหมักอาซีโตน-บิวทานอล จากมันสำปะหลัง. โครงการวิจัย เลขที่ 91-IR-2527 คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย, 2529

8. สุทธิศักดิ์ พงศ์ธนาพาณิช. การใช้โปรแกรม Quickbasic. กรุงเทพมหานคร: ซีเอ็ดยูเคชั่น, 2536.

9. ยืน ภู่วรวรรณ และ ไพศาล สงวนหมู่. การสื่อสารข้อมูลและไมโครคอมพิวเตอร์เนตเวอร์ค. กรุงเทพมหานคร: ซีเอ็ดยูเคชั่น, 2536.

10. Bryce, C.F.A. Microcomputers in Biochemistry a Practical Approach. New York: Oxford University Press, 1992

11. Constantinides, A. Applied Numerical Methods with Personal Computers. International ed. Singapore: McGraw-Hill Book Co., 1987

12. Wang, E., Hatanaka, H., Iijima, S., Takebayashi, T., Shi, Z., Shimizu, K., Matsubara, M., and Kobayashi, T. Control of Cell and Lactate concentration in a Hollow-Fiber Bioreactor for Lactic Acid Fermentation. Journal of Chemical Engineering of Japan 20 (1988): 36-40.

13. Model 8700 Gas Chromatographs Reference Manual. 2 Vols. by Perkin-Elmer Ltd. England: n.p., 1989

ภาคผนวก

รายการพิมพ์ออกของโปรแกรม QTHE.BAS

```
' THIS PROGRAM USED FOR ON-LINE MONITORING OF A BATCH-TYPE BIOREACTOR BY COMPUTER
' WRITE BY Mr. JAMROEN  LEOWSEESUK
DECLARE SUB SIMULATE ()
DECLARE SUB PFD ()
DECLARE SUB GAUSS (CON5!, CON6!, CON7!, CON8!, CON9!, CON10!, CON11!, X!())
150 COMMON ARY1%(), ARY2%()
CLS : KEY OFF: SCREEN 9
NARY = 1001: NTIMES = 1000: NDATA = 1001
DIM SHARED X!(11)
DIM PH!(NDATA), ITER%(NDATA), TEMP!(NDATA), GCCb!(1001), PGCCb!(1001)
DIM NAMO!(10 TO 30), VNAME$(10 TO 30)
'****** DEFINE FUNCTION ******
DEF FNBIOMASS! (X!, Y!, B!)
  FNBIOMASS! = (.56 * (Y! - 1) * X!) - (K2 * X! * B!)
END DEF
DEF FNCELLLYSIS! (Y!, S!, B!)
  FNCELLLYSIS! = Y! * (((KL / (KL + B)) * S! * K1) - (.56 * (Y! - 1)))
END DEF
DEF FNSUBSTRATE! (S!, X!)
   FNSUBSTRATE! = ((-K3) * S! * X!) - ((S! / (KS + S!)) * K4 * X!)
END DEF
DEF FNBUTYRATE! (BA!, S!, B!, X!)
   FNBUTYRATE! = ((KL / (KL + B!)) * K5 * S! * X!) - ((BA! / (KBA + BA!)) * K6 * X!)
```

```
END DEF
DEF FNBUTANOL! (B!, S!, X!, BA!)
   FNBUTANOL! = (K7 * S! * X!) - (.841 * FNBUTYRATE!(BA!, S!, B!, X!))
END DEF
DEF FNACETATE! (AA!, S!, B!, X!)
   FNACETATE! = ((S! / (KS + S!)) * (KL / (KL + B!)) * K8 * X!) - ((AA! / (KAA + AA!)) * (S / (KS + S)) * K9
* X)
END DEF
DEF FNACETONE! (AA!, S!, B!, X!)
   FNACETONE! = ((S! / (KS + S!)) * K10 * X!) - (.484 * FNACETATE!(AA!, S!, B!, X!))
END DEF
DEF FNETHANOL! (S!, X!)
   FNETHANOL! = (S! / (KS + S!)) * K11 * X!
END DEF
DEF FNCARBON! (S!, X!)
   FNCARBON! = (S! / (KS + S!)) * K12 * X!
END DEF
DEF FNHYDROGEN! (S!, X!)
   FNHYDROGEN! = ((S! / (KS + S!)) * K13 * X!) + (K14 * S! * X!)
END DEF
'****** END DEFINE ******
MENU1:
' ************ Main Menu **********************
CLS : LOCATE 6, 24: COLOR 12: PRINT "Menu Selection"
LINE (170, 65)-(310, 90), 9, B
LOCATE 11, 1
 COLOR 10: PRINT "              1>  PFD DISPLAY "
 COLOR 11: PRINT "              2>  ON-LINE MONITORING "
```

```
  COLOR 12: PRINT "              3> WRITE CONSTANT (k1-k14) TO FILE "
  LOCATE 18, 15: COLOR 13: PRINT "Select ======>"
  LOCATE 19, 17: Select1$ = INPUT$(1)
  LOCATE 18, 30: PRINT Select1$: FOR Q = 1 TO 400: NEXT Q
  ' ********  End of Main Menu ********
  SELECT CASE Select1$
  ' ---------------------------
  CASE IS = "1"  '**** PFD *******
  ' ---------------------------
 CALL PFD
 GOTO MENU1
 '********* End Case No.1 *****
 ' ---------------------------
  CASE IS = "2" '**** On-line Monitoring *****
 ' ---------------------------
 '***** DATA ACQUISITION PART **********************
 ' clear screen; make sure all files are closed
 CLS : CLOSE : COLOR 15
 PRINT
 PRINT " LOAD 711QB4.QLB FIRST : (Y/N) ";
 CONT$ = INPUT$(1)
 PRINT CONT$: FOR a = 1 TO 30: NEXT a
 IF UCASE$(CONT$) = "N" THEN
   PRINT " PROGRAM WILL STOP NOW "
   PRINT " REMEMBER TO LOAD 711QB4.QLB FIRST "
   PRINT
   PRINT " ANYKEY TO CONTINUE "
 . CONT$ = INPUT$(1)
```

```
    END
  ELSE
    PRINT
    PRINT " INPUT INITIAL CONCENTRATION OF SUBSTRATE : (g/l) "
    INPUT " BIOMASS CONC.     (X) = ", PreCb!
    INPUT " SUBSTRATE CONC.   (S) = ", PreCglu!
    INPUT " BUTANOL CONC.     (B) = ", PreCbl!
    INPUT " ACETONE CONC.     (A) = ", PreCaco!
    INPUT " ETHANOL CONC.     (E) = ", PreCet!
    INPUT " BUTYRIC ACID CONC. (BA) = ", PreCbt!
    INPUT " ACETIC  ACID CONC. (AA) = ", PreCaca!
    GCTIMES = 0
    PRINT
    PRINT " ANYKEY TO CONTINUE "
    CONT$ = INPUT$(1)
  END IF
  MARK1:
  CLS : COLOR 15
  LOCATE 2, 5: PRINT "READ CONSTANT VALUE (k1-k14) FROM FILE"
  LOCATE 3, 5: INPUT "ENTER NAME OF FILE TO READ [DAIVE:NAME] "; FILE2$
  OPEN "FILE2$" FOR INPUT AS #3
  INPUT #3, K1, K2, K3, K4, K5, K6, K7, K8, K9, K10, K11, K12, K13, K14, KL, KS, KBA, KAA
  PRINT
  PRINT " k1 = "; K1; " k2 = "; K2; " k3 = "; K3
  PRINT
  PRINT " k4 = "; K4; " k5 = "; K5; " k6 = "; K6
  PRINT
  PRINT " k7 = "; K7; " k8 = "; K8; " k9 = "; K9
```

```
PRINT
PRINT " k10 = "; K10; " k11 = "; K11; " k12 = "; K12
PRINT
PRINT " k13 = "; K13; " k14 = "; K14; " KL = "; KL
PRINT
PRINT " KS = "; KS; " KBA = "; KBA; " KAA = "; KAA
PRINT
PRINT " THIS CONSTANT IS CORRECT (Y/N)";
CONT$ = INPUT$(1)
PRINT CONT$: FOR a = 1 TO 30: NEXT a
IF UCASE$(CONT$) = "N" THEN GOTO MARK1
CLOSE #3: CLS : COLOR 15
'set communications protocol "serial port 1,9600 bps, even parity, 7 data bits, 1 stop bit,set Clear to Send,
set Data Set Ready with delay upto 65535
COMM$ = "com1:9600,e,7,1,cs0,ds0,cd0"
PRINT " Set Communication parameters for connection between Computer & GC. "
PRINT
PRINT " Communication parameters are  : " + COMM$
LINE INPUT " <Enter> if OK, or change :"; X$
IF X$ <> "" THEN COMM$ = X$
OPEN COMM$ FOR RANDOM AS #1 LEN = 300
PRINT
INPUT " INPUT TIME RANGE TO GET DATA FOR 1 CYCLE : (1-15) MIN. "; TIMEMIN%
GETTIME% = TIMEMIN% * 60
PRINT
INPUT " HOW OFTEN TO GET SAMPLE FOR ANALISE BY G.C. : (Hr) "; TDELTA%
DELTAGC% = TDELTA% * 10
PRINT
```

```
PRINT " NOW WE USE INITIAL CONC. TO SIMULATION "
PRINT
PRINT " ANYKEY TO CONTINUE "
CONT$ = INPUT$(1)
CLS
CALL SIMULATE
FOR P = 1 TO 700
   GCCbl(P) = PGCCbl(P)
NEXT P
CLS : COLOR 15
LOCATE 10, 20: PRINT " REDY TO START ON-LINE : ANYKEY TO CONTINUE "
CONT$ = INPUT$(1)
'****** INITIALIZE PCL-711 BOARD USING FUNC 0 ********
DIM DAT%(4), ARY1%(NARY), ARY2%(NARY)
PORT% = &H220          'SET I/O PORT ADDRESS
DAT%(0) = PORT%        'GET I/O PORT ADDRESS
ER% = 0               'ERROR RETURN CODE
FUN% = 0              'FUNCTION 0
CALL PCL711(FUN%, SEG DAT%(0), SEG ARY1%(0), SEG ARY2%(0), ER%)
IF ER% <> 0 THEN PRINT "DRIVER INITIALIZATION FAILED !": STOP
450 '****** SET SCAN CHANNEL RANGE USING FUNC 1 *****************
START% = 0: STP% = 0: GAIN% = 100
DAT%(0) = START%       'SET START CHANNEL NUMBER
DAT%(1) = STP%         'SET STOP  CHANNEL NUMBER
FUN% = 1              'FUNCTION 1
CALL PCL711(FUN%, SEG DAT%(0), SEG ARY1%(0), SEG ARY2%(0), ER%)
IF ER% <> 0 THEN PRINT "SET SCAN CHANNEL FAILED! RE-ENTER": GOTO 450
```

```
'============> SCAN CHANNEL LOOP
  CLS : COLOR 15
  '*** Event Trapping for RS-232 communication and Jump To SUB GCANAL
  ON COM(1) GOSUB GCANAL
  COM(1) ON
  '*** DEFINE FUNCTION KEY AND TURNING ON KEY
  KEY 1, "PFD": ON KEY(1) GOSUB DIAGRAM
  KEY 2, "SIMULATE": ON KEY(2) GOSUB SIMULATION
  KEY 3, "TIME": ON KEY(3) GOSUB CHANGETIME
  KEY 10, "EXIT": ON KEY(10) GOSUB FINAL
  KEY(1) ON: KEY(2) ON: KEY(3) ON: KEY(10) ON
  KEY ON
  '**** BEGIN SCAN CHANNEL LOOP
 DO
LABEL2:
    '******* STEP 4: DISPLAY DATA ************************************
     PHMIN = 0: PHMAX = 14: TMIN = 10: TMAX = 80: XMIN = 0: XMAX = GETTIME%
LABEL1:
   '**** DRAW SCALE FOR PH
    VIEW (70, 34)-(280, 228): CLS    'DEFINE VIEWPORT OF PH
    WINDOW (XMIN, PHMIN)-(XMAX, PHMAX)   'SCALE DATA TO FIT ENTIRE VIEWPORT
    LINE (XMIN, PHMIN)-(XMAX, PHMAX), 10, B   'PUT BOX AROUND GRAPH
    REM *** calculate number of divisions of x-axis to use
    DELTA = (XMAX - XMIN) / 5
    FOR Z = 0 TO 5
       X2 = Z * DELTA + XMIN
       LINE (X2, PHMIN)-(X2, PHMAX), 11, , &HCCCC  'DRAW TICK MARK
    NEXT Z
```

```
LOCATE 18, 9: COLOR 13: PRINT "0"
LOCATE 18, 34: COLOR 13: PRINT USING "###"; TIMEMIN%
LOCATE 18, 16: COLOR 13: PRINT "TIME (MIN)"
REM *** calculate number of divisions of y-axis to use
DELTA = (PHMAX - PHMIN) / 7
FOR Z = 0 TO 7
  Y2 = Z * DELTA + PHMIN
  LINE (XMIN, Y2)-(XMAX, Y2), 11, , &HCCCC  'DRAW TICK MARK
  LOCATE (8 - Z) * 2 + 1, 6  '*****
  COLOR 13: PRINT USING "##"; Y2     'TEXT LABEL FROM BOTTOM TO TOP
NEXT Z
LOCATE 10, 2: PRINT "pH"
'**** DRAW SCALE FOR TEMPERATURE
VIEW (380, 34)-(600, 228): CLS    'DEFINE VIEWPORT OF TEMP
WINDOW (XMIN, TMIN)-(XMAX, TMAX)   'SCALE DATA TO FIT ENTIRE VIEWPORT
LINE (XMIN, TMIN)-(XMAX, TMAX), 10, B   'PUT BOX AROUND GRAPH
REM *** calculate number of divisions of x-axis to use
DELTA = (XMAX - XMIN) / 5
FOR Z = 0 TO 5
  X2 = Z * DELTA + XMIN
  LINE (X2, TMIN)-(X2, TMAX), 11, , &HCCCC  'DRAW TICK MARK
NEXT Z
LOCATE 18, 48: COLOR 13: PRINT "0"
LOCATE 18, 74: COLOR 13: PRINT USING "###"; TIMEMIN%
LOCATE 18, 56: COLOR 13: PRINT "TIME (MIN)"
REM *** calculate number of divisions of y-axis to use
DELTA = (TMAX - TMIN) / 7
FOR Z = 0 TO 7
```

```
    Y2 = Z * DELTA + TMIN
    LINE (XMIN, Y2)-(XMAX, Y2), 11, , &HCCCC  'DRAW TICK MARK
    LOCATE (8 - Z) * 2 + 1, 44  '*****
    COLOR 13: PRINT USING "###"; Y2      'TEXT LABEL FROM BOTTOM TO TOP
  NEXT Z
  LOCATE 10, 40: PRINT "TEMP"
  LOCATE 11, 41: PRINT "(C)"
'**** DRAW GRAPHICS
  FOR I = 0 TO GETTIME%
  ITER%(I) = I
  '**** FOR PH
    CH% = 0
    GOSUB ADCONVER
    PH!(I) = (VOLTSIN! * 1436.09)
    LOCATE 20, 16: COLOR 10: PRINT "pH = "
    LOCATE 20, 20: COLOR 10: PRINT USING "##.##"; PH!(I)
    VIEW (70, 34)-(280, 228)    'DEFINE VIEWPORT OF PH
    WINDOW (XMIN, PHMIN)-(XMAX, PHMAX)   'SCALE DATA TO FIT ENTIRE VIEWPORT
    IF I = 0 THEN
      PSET (ITER%(I), PH!(I)), 12    'PLOT FIRST POINT
    ELSE
      LINE (ITER%(I - 1), PH!(I - 1))-(ITER%(I), PH!(I)), 12 'DRAW LINE CONNECTING REST OF POINTS
    END IF
    '**** FOR TEMPERATURE
    CH% = 1
    GOSUB ADCONVER
    TEMP!(I) = (VOLTSIN! * 23298) + 31.98
    LOCATE 20, 56: PRINT "TEMP = "
```

```
        LOCATE 20, 63: PRINT USING "##.##"; TEMP!(I)
        VIEW (380, 34)-(600, 228)     'DEFINE VIEWPORT OF TEMP
        WINDOW (XMIN, TMIN)-(XMAX, TMAX)   'SCALE DATA TO FIT ENTIRE VIEWPORT
        IF I = 0 THEN
          PSET (ITER%(I), TEMP!(I)), 14     'PLOT FIRST POINT
        ELSE
          LINE (ITER%(I - 1), TEMP!(I - 1))-(ITER%(I), TEMP!(I)), 14 'DRAW LINE CONNECTING REST OF
 '  POINTS
        END IF
      NEXT I
    LOOP
'****** END CASE " 2 "
'-------------------
CASE IS = "3"   'WRITE CONSTANT VALUE ( k1-k14 ) TO FILE
'-------------------
CLS : COLOR 12
LOCATE 2, 1: INPUT "ENTER NAME OF FILE FOR COLLECT CONSTANT (k1-k14) [ DRIVE:NAME ] ";
FILE1$
OPEN FILE1$ FOR OUTPUT AS #2
LOCATE 3, 5: INPUT " CONSTANT VALUE (k1,k2) = ", K1, K2
LOCATE 4, 5: INPUT " CONSTANT VALUE (k3,k4) = ", K3, K4
LOCATE 5, 5: INPUT " CONSTANT VALUE (k5,k6) = ", K5, K6
LOCATE 6, 5: INPUT " CONSTANT VALUE (k7,k8) = ", K7, K8
LOCATE 7, 5: INPUT " CONSTANT VALUE (k9,k10) = ", K9, K10
LOCATE 8, 5: INPUT " CONSTANT VALUE (k11,k12) = ", K11, K12
LOCATE 9, 5: INPUT " CONSTANT VALUE (k13,k14) = ", K13, K14
LOCATE 10, 5: INPUT " CONSTANT VALUE (KL,KS) = ", KL, KS
LOCATE 11, 5: INPUT " CONSTANT VALUE (KBA,KAA) = ", KBA, KAA
```

```
WRITE #2, K1, K2, K3, K4, K5, K6, K7, K8, K9, K10, K11, K12, K13, K14, KL, KS, KBA, KAA
CLOSE #2
GOTO MENU1
'****** END CASE " 3 "
END SELECT
' #### SUBROUTINE IN MAIN PROGRAM ####
'_______
GCANAL: '******RECEIVE DATA FROM GAS CHROMATOGRAPH ****
'_______
VIEW: CLS 0: COLOR 10
COM(1) OFF: KEY(1) OFF: KEY(2) OFF: KEY(3) OFF: KEY OFF
GCTIMES = GCTIMES + DELTAGC%
'define true and false
FALSE = 0: TRUE = NOT FALSE
'define the XON , XOFF characters
XOFF$ = CHR$(19): XON$ = CHR$(17)
I = 1
PAUSE = FALSE
DO UNTIL EOF(1)
   'if buffer more then 1/2 full, then
   ' set pause flag to say input suspended,
   ' send xoff to host to stop transmission
   DO
      IF LOC(1) > 150 THEN
        PAUSE = TRUE: PRINT #1, XOFF$;
      END IF
      IF I <= 9 THEN
        INPUT #1, TEXT1$
```

```
        END IF
        IF I > 9 AND I <= 14 THEN
         INPUT #1, VNAME$(I)
        END IF
        IF I > 14 THEN
         INPUT #1, TEXT2$
        END IF
        I = I + 1
    LOOP WHILE LOC(1) > 0
    'if transmission suspended by xoff, resume by sending xon
    IF PAUSE THEN PAUSE = FALSE: PRINT #1, XON$;
LOOP
FOR J = 10 TO 14
    RVNAME$ = RIGHT$(VNAME$(J), 25)
    LRVNAME$ = LEFT$(RVNAME$, 6)
    NAMO!(J) = VAL(LRVNAME$)
NEXT J
'Specific initial constant
Cgluin! = PreCglu! / 180.2 'change unit from g/l to mole/l
Ycatp! = 10.5    'dimensionless
MWb! = 103.9     'unit g/mol
'*** TRANFER VALUE OF PRODUCT FROM GC.
PreCb! = GCCb!(GCTIMES)
PreCet! = NAMO!(10): PreCaco! = NAMO!(11): PreCaca! = NAMO!(12)
PreCbl! = NAMO!(13): PreCaci! = 0: PreCbt! = NAMO!(14)
'*** CHANGE UNIT FROM G/L TO MOLE/L
Cb! = PreCb! / MWb!: Cet! = PreCet! / 46.07: Caco! = PreCaco! / 58.08
Caca! = PreCaca! / 60.05: Cbl! = PreCbl! / 74.12: Caci! = 0: Cbt! = PreCbt! / 88.11
```

```
'*** Call GAUSS ELIMINATION Subprogram ***
CALL GAUSS((Cb!), (Cet!), (Caco!), (Caca!), (Cbl!), (Caci!), (Cbt!), X!())
coa! = X!(1): cob! = X!(2): coc! = X!(3): cod! = X!(4): coe! = X!(5)
cof! = X!(6): cog! = X!(7): coh! = X!(8): coj! = X!(9): cok! = X!(10)
col! = X!(11)
'** Calculate fermentation variable
Cglu! = (Cgluin! - (2 * col! + coa!)) * 180.2
ATPtot! = 2 * coa! + coe!
ATPex! = (2 * coa!) + coe! - (103.9 * col! / Ycatp)
Yatp! = PreCb! / ATPtot!
Ys! = PreCb! / ((2 * col! + coa!) * 180.2)
NfF! = cod!
Chy! = coc! * 2
Cco2! = (2 * cok! + cob! + coh!) * 44
'Show result in table form
CLS
TABLE$ = STRING$(80, "*")
COLOR 13: LOCATE 5, 1: PRINT TABLE$
COLOR 10: LOCATE 6, 12: PRINT "PRODUCT"
LOCATE 6, 50: PRINT "FERMENTATION VARIABLE"
COLOR 13: LOCATE 7, 1: PRINT TABLE$
FOR Z = 5 TO 16
    LOCATE Z, 38: PRINT "*"
NEXT Z
COLOR 10
'*** SHOW PRODUCT CONC. FROM GC.
LOCATE 8, 5:  PRINT USING "BIOMASS  ###.####"; PreCb!
LOCATE 9, 5:  PRINT USING "ETHANOL  ###.####"; PreCet!
```

```
LOCATE 10, 5: PRINT USING "ACETONE  ###.####"; PreCacol
LOCATE 11, 5: PRINT USING "ACETIC   ###.####"; PreCaca!
LOCATE 12, 5: PRINT USING "BUTANOL  ###.####"; PreCbl!
LOCATE 13, 5: PRINT USING "ACETOIN  ###.####"; PreCaci!
LOCATE 14, 5: PRINT USING "BUTYRIC  ###.####"; PreCbt!
FOR Z = 8 TO 14
   LOCATE Z, 30: PRINT "g/l"
NEXT Z
'*** SHOW FERMENTATION VARIABLE
LOCATE 8, 42:  PRINT USING "C-glu   ####.####"; Cglu!
LOCATE 9, 42: PRINT USING "C-hy    ####.####"; Chy!
LOCATE 10, 42: PRINT USING "C-co2   ####.####"; Cco2!
LOCATE 11, 42:  PRINT USING "ATP*tot ####.####"; ATPtot!
LOCATE 12, 42: PRINT USING "ATP*ex  ####.####"; ATPex!
LOCATE 13, 42: PRINT USING "NfF*    ####.####"; NfF!
LOCATE 14, 42: PRINT USING "Y-atp   ####.####"; Yatp!
LOCATE 15, 42: PRINT USING "Y-s     ####.####"; Ys!
FOR Z = 8 TO 10
  LOCATE Z, 65: PRINT "g/l"
NEXT Z
FOR Z = 11 TO 13
  LOCATE Z, 65: PRINT "mole/l"
NEXT Z
LOCATE 14, 61: PRINT "g biomass/mol ATP"
LOCATE 15, 61: PRINT "g biomass/g glucose"
COLOR 13: LOCATE 16, 1: PRINT TABLE$
COLOR 12: LOCATE 22, 20: PRINT "PRESS ANY KEY TO CONTINUE"
P$ = INPUT$(1)
```

```
COM(1) ON: KEY(1) ON: KEY(2) ON: KEY(3) ON: CLS : COLOR 15: KEY ON
RETURN LABEL1
'_____
DIAGRAM: '****** SHOW PFD ****
'_____
KEY(1) OFF: KEY(2) OFF: KEY(3) OFF: KEY OFF: VIEW: WINDOW: CLS 0
CALL PFD
KEY(1) ON: KEY(2) ON: KEY(3) ON: CLS : COLOR 15: KEY ON
'**LOCATE 1, 29: PRINT " PRESS <ESC> KEY TO EXIT ": 'COLOR13
RETURN LABEL1
'_____
SIMULATION: '****** SIMULATE ****
'_____
KEY(1) OFF: KEY(2) OFF: KEY(3) OFF: KEY OFF: VIEW: WINDOW: CLS 0
CALL SIMULATE
KEY(1) ON: KEY(2) ON: KEY(3) ON: CLS : COLOR 15: KEY ON
'**LOCATE 1, 29: PRINT " PRESS <ESC> KEY TO EXIT ": 'COLOR13
RETURN LABEL1
'_____
CHANGETIME: '****** CHANGE TIME TO GET DATA FOR ONC CYCLE
'_____
KEY(1) OFF: KEY(2) OFF: KEY(3) OFF: KEY OFF: VIEW: WINDOW: CLS 0
LOCATE 10, 10: COLOR 10: PRINT "NOW TIME RANGE (MIN.) FOR 1 CYCLE = "; TIMEMIN%
LOCATE 12, 10: INPUT "CHANGE TO :(1-15 MIN.) = "; TIMEMIN%
GETTIME% = TIMEMIN% * 60
KEY(1) ON: KEY(2) ON: KEY(3) ON: CLS : COLOR 15: KEY ON
RETURN LABEL2
```

```
'_______
ADCONVER: '****** INPUT ANALOG SIGNAL DATA MANY TIMES AND AVERAGE DATA
'_______
'==========> WRITE DIGITAL OUTPUT TO SELECT CHANNEL USING FUNC 21
FUN% = 21
DAT = CH%
DAT%(1) = DAT \ 256
DAT%(0) = DAT MOD 256
CALL PCL711(FUN%, SEG DAT%(0), SEG ARY1%(0), SEG ARY2%(0), ER%)
IF ER% <> 0 THEN PRINT "PERFORM FUNC 21 FAILED!": STOP
'******* PERFORM N A/D CONVERSIONS USING FUNC 4 ************
'** USE DELAY LOOPS = 150 FOR GET DATA = 1 MIN
DAT%(0) = NARY
DAT%(2) = 150          'DELAY LOOPS BETWEEN SAMPLES
FUN% = 4               'FUNCTION 4
CALL PCL711(FUN%, SEG DAT%(0), SEG ARY1%(0), SEG ARY2%(0), ER%)
IF ER% <> 0 THEN PRINT "PERFORM N A/D CONVERSIONS FAILED!": STOP
' ******* AVERAGE DATA
READIN! = 0
FOR W = 20 TO NTIMES
  READIN! = READIN! + ARY1%(W)
NEXT W
FILREADIN! = READIN! / (W - 20)
VOLTSIN! = FILREADIN! * (10 / 4095) / GAIN%
RETURN
'_______
FINAL: '******* EXIT PROGRAM ****
'_______
```

```
COM(1) OFF: KEY(1) OFF: KEY(2) OFF: KEY(10) OFF: KEY OFF
CLOSE : VIEW: WINDOW
SCREEN 0, 0, 0: WIDTH 80
END
'#### SUBPROGRAM GAUSS ELINIMATION ####
SUB GAUSS (CON5!, CON6!, CON7!, CON8!, CON9!, CON10!, CON11!, X!()) STATIC
   DIM a(11, 12), B(11, 12), NPIVROW(11, 2), NPIVCOL(11, 2)
   N = 11  'Number of equation = 11
   'Specific Coefficient and Constant of equation
   B(1, 1) = 2: B(1, 2) = -1: B(1, 3) = 0: B(1, 4) = 0: B(1, 5) = 0
   B(1, 6) = 0: B(1, 7) = 0: B(1, 8) = 0: B(1, 9) = 0: B(1, 10) = -2
   B(1, 11) = 0: B(1, 12) = 0
   '
   B(2, 1) = 0: B(2, 2) = 1: B(2, 3) = 0: B(2, 4) = 0: B(2, 5) = -1
   B(2, 6) = -1: B(2, 7) = -1: B(2, 8) = -1: B(2, 9) = -1: B(2, 10) = 0
   B(2, 11) = 0: B(2, 12) = 0
   '
   B(3, 1) = 2: B(3, 2) = 0: B(3, 3) = 0: B(3, 4) = 1: B(3, 5) = 0
   B(3, 6) = -2: B(3, 7) = -2: B(3, 8) = 0: B(3, 9) = -2: B(3, 10) = 0
   B(3, 11) = -1.75: B(3, 12) = 0
   '
   B(4, 1) = 0: B(4, 2) = 1: B(4, 3) = -1: B(4, 4) = -1: B(4, 5) = 0
   B(4, 6) = 0: B(4, 7) = 0: B(4, 8) = 0: B(4, 9) = 0: B(4, 10) = 0
   B(4, 11) = 0: B(4, 12) = 0
   '
   B(5, 1) = 0: B(5, 2) = 0: B(5, 3) = 0: B(5, 4) = 0: B(5, 5) = 0
   B(5, 6) = 0: B(5, 7) = 0: B(5, 8) = 0: B(5, 9) = 0: B(5, 10) = 0
   B(5, 11) = 3: B(5, 12) = CON5!
```

```
'
B(6, 1) = 0: B(6, 2) = 0: B(6, 3) = 0: B(6, 4) = 0: B(6, 5) = 0
B(6, 6) = 0: B(6, 7) = 0: B(6, 8) = 0: B(6, 9) = 1: B(6, 10) = 0
B(6, 11) = 0: B(6, 12) = CON6!
'
B(7, 1) = 0: B(7, 2) = 0: B(7, 3) = 0: B(7, 4) = 0: B(7, 5) = 0
B(7, 6) = 0: B(7, 7) = 0: B(7, 8) = 1: B(7, 9) = 0: B(7, 10) = 0
B(7, 11) = 0: B(7, 12) = CON7!
'
B(8, 1) = 0: B(8, 2) = 0: B(8, 3) = 0: B(8, 4) = 0: B(8, 5) = 1
B(8, 6) = -1: B(8, 7) = 1: B(8, 8) = -1: B(8, 9) = 0: B(8, 10) = 0
B(8, 11) = 0: B(8, 12) = CON8!
'
B(9, 1) = 0: B(9, 2) = 0: B(9, 3) = 0: B(9, 4) = 0: B(9, 5) = 0
B(9, 6) = 0: B(9, 7) = 1: B(9, 8) = 0: B(9, 9) = 0: B(9, 10) = 0
B(9, 11) = 0: B(9, 12) = CON9!
'
B(10, 1) = 0: B(10, 2) = 0: B(10, 3) = 0: B(10, 4) = 0: B(10, 5) = 0
B(10, 6) = 0: B(10, 7) = 0: B(10, 8) = 0: B(10, 9) = 0: B(10, 10) = 1
B(10, 11) = 0: B(10, 12) = CON10!
'
B(11, 1) = 0: B(11, 2) = 0: B(11, 3) = 0: B(11, 4) = 0: B(11, 5) = 0
B(11, 6) = 1: B(11, 7) = -1: B(11, 8) = 0: B(11, 9) = 0: B(11, 10) = 0
B(11, 11) = 0: B(11, 12) = CON11!
'
NC = N + 1
EPS = .00001  '** GIVE THE MINIMUM ALLOWABLE VALUE OF THE PIVOT ELEMENT
DET = 1
```

```
    FOR K = 1 TO N
    FOR J = 1 TO NC
    a(K, J) = B(K, J)
    NEXT J: NEXT K
    'Beginning of the Gauss elimination procedure
    FOR K = 1 TO N
    'Apply complete pivoting strategy
    MAXPIVOT = ABS(a(K, K))
    NPIVROW(K, 1) = K: NPIVROW(K, 2) = K
    NPIVCOL(K, 1) = K: NPIVCOL(K, 2) = K
    FOR I = K TO N
    FOR J = K TO N
    IF MAXPIVOT >= ABS(a(I, J)) GOTO 2000
    MAXPIVOT = ABS(a(I, J))
    NPIVROW(K, 1) = K: NPIVROW(K, 2) = I
    NPIVCOL(K, 1) = K: NPIVCOL(K, 2) = J
2000 NEXT J: NEXT I
    IF MAXPIVOT >= EPS GOTO 2010
    GOTO 2050   'Where MAXPIVOT < EPS go back-substitution
2010 IF NPIVROW(K, 2) = K GOTO 2020
    FOR J = K TO NC
    SWAP a(NPIVROW(K, 2), J), a(K, J)
    NEXT J
    DET = DET * (-1)
2020 IF NPIVCOL(K, 2) = K GOTO 2030
    FOR I = 1 TO N
    SWAP a(I, NPIVCOL(K, 2)), a(I, K)
    NEXT I
```

```
    DET = DET * (-1)
 2030 IF K = N THEN GOTO 2040
    FOR I = K + 1 TO N
    MULT = -a(I, K) / a(K, K)
    FOR J = NC TO K STEP -1
    a(I, J) = a(I, J) + MULT * a(K, J)
    NEXT J
    NEXT I
 2040 NEXT K
    'Apply the back-substitution formulas
 2050 RANK = K - 1: NMR = N - RANK
    IF RANK = N THEN X(N) = a(N, N + 1) / a(N, N): NCOUNT = N - 1: GOTO 2060
    FOR JJ = 1 TO NMR: X(N + 1 - JJ) = 1: NEXT JJ
    NCOUNT = RANK
 2060 FOR I = NCOUNT TO 1 STEP -1
    SUM = 0
    FOR J = I + 1 TO N
    SUM = SUM + a(I, J) * X(J)
    NEXT J
    X(I) = (a(I, N + 1) - SUM) / a(I, I)
    NEXT I
    'Interchange the order of the unknowns to correct for the column pivoting
    FOR K = N TO 1 STEP -1
    SWAP X(NPIVCOL(K, 2)), X(NPIVCOL(K, 1))
    NEXT K
END SUB
'#### SUBPROGRAM SHOW PROCESS FLOW DIAGRAM ####
SUB PFD STATIC
```

```
CLS : DIM RUBBER1%(1060), RUBBER2%(1060), RUBBER3%(70), RUBBER4%(1060)
WINDOW SCREEN (0, 0)-(639, 200)
'********** Draw Instrument *********
 LINE (130, 122)-(251, 182), 10, B
 COLOR 10: PSET (40, 145): DRAW "D5 G8 D35 R30  U35 H8 U5 L13 D5 R13"
 COLOR 15
 LINE (46, 170)-(46, 148), 11
 CIRCLE (100, 92), 15, 10: CIRCLE (305, 100), 15, 10
 CIRCLE (136, 109), 14, 14
    BASE1$ = "C10 G8 R40 H8"
    PSET (89, 97): DRAW BASE1$: PSET (294, 105): DRAW BASE1$
 CIRCLE (305, 50), 15, 10: CIRCLE (405, 84), 14, 14
 CIRCLE (410, 152), 15, 10
    PSET (294, 54): DRAW BASE1$: PSET (399, 156): DRAW BASE1$
 LINE (300, 142)-(360, 162), 10, B: LINE (450, 152)-(465, 172), 10, B
 PI = 3.14159: CIRCLE (515, 157), 10, 14, -2 * PI, -PI
 DRAW "C14 D15 F10 U20 G20 U20 F10"
 LINE (564, 112)-(604, 132), 10, B
 LINE (466, 55)-(600, 65), 14, B: LINE (533, 55)-(535, 65), 14, B
 LINE (466, 20)-(600, 35), 14, B
 PSET (350, 110): DRAW "C10 R10 D9 F20 L50 E20 U9"
'********** Link Tube & Cable to Instrument *******
 COLOR 11: LINE (46, 145)-(46, 92): LINE (46, 92)-(100, 92)
 LINE (110, 87)-(155, 87): LINE (155, 87)-(155, 122)
 AD$ = "TA15 NU10 TA345 NU10": DRAW AD$
 LINE (50, 67)-(161, 67): LINE (161, 67)-(161, 145): DRAW AD$
 LINE (355, 122)-(355, 100): LINE (355, 100)-(305, 100)
 LINE (295, 95)-(238, 95): LINE (238, 95)-(238, 130): DRAW AD$
```

```
LINE (232, 145)-(232, 50): LINE (232, 50)-(305, 50)
LINE (315, 45)-(395, 45): AR$ = "TA75 NU10 TA105 NU10": DRAW AR$
LINE (242, 152)-(300, 152): LINE (360, 152)-(413, 152)
DRAW AR$: LINE (420, 147)-(457, 147): LINE (457, 147)-(457, 152)
LINE (457, 172)-(457, 175): LINE (457, 175)-(251, 175)
DRAW "TA255 NU10 TA285 NU10": LINE (465, 166)-(610, 166)
DRAW AR$: COLOR 15
FOR I = 135 TO 235 STEP 20
CIRCLE (I, 135), 5, 11, 0, PI: NEXT I
FOR J = 145 TO 245 STEP 20
CIRCLE (J, 135), 5, 11, PI, 2 * PI: NEXT J
CIRCLE (548, 52), 40, 13, .2 * PI, .8 * PI
CIRCLE (518, 52), 40, 13, .2 * PI, .8 * PI
COLOR 13: AU$ = "TA165 NU10 TA195 NU10 ": LINE (582, 166)-(582, 132)
DRAW AU$: LINE (580, 112)-(580, 65): DRAW AU$: LINE (580, 65)-(580, 42)
LINE (330, 142)-(330, 132): LINE (330, 132)-(550, 132)
LINE (550, 132)-(550, 65): DRAW AU$: LINE (550, 65)-(550, 42)
LINE (242, 139)-(405, 139): LINE (405, 139)-(405, 90): DRAW AU$
LINE (391, 84)-(305, 84): LINE (305, 84)-(305, 94): DRAW AD$
LINE (420, 84)-(567, 84): LINE (567, 84)-(567, 65): DRAW AU$
LINE (567, 65)-(567, 35)
LINE (516, 42)-(516, 55): DRAW AD$: LINE (516, 55)-(516, 152): DRAW AD$
LINE (486, 42)-(486, 55): DRAW AD$: LINE (486, 55)-(486, 70)
LINE (486, 70)-(305, 70): LINE (305, 70)-(305, 59): DRAW AU$
LINE (134, 130)-(144, 138), 14, B: LINE (137, 130)-(139, 115), 14, B
LINE (122, 109)-(100, 109): LINE (100, 109)-(100, 102)
DRAW "TA165 NU5 TA195 NU5": COLOR 15
'********* Put Label *********
```

```
    COLOR 9: LOCATE 8, 4: PRINT "NITROGEN GAS": LOCATE 20, 72
    PRINT "Filtrate": LOCATE 21, 79: PRINT "F1": LOCATE 22, 72
    PRINT "C-Butanol": LOCATE 5, 42: PRINT "F2 (Cell Remove)"
    COLOR 12: LOCATE 19, 39: PRINT "TERBIDI": LOCATE 20, 40: PRINT "METER"
    LOCATE 16, 72: PRINT "G.C.": LOCATE 4, 64: PRINT "COMPUTER"
    LOCATE 18, 53: PRINT "MICRO FILTER"
    LOCATE 14, 17: PRINT "LC": LOCATE 11, 51: PRINT "PH"
    LOCATE 8, 55: PRINT "D/A": LOCATE 8, 77: PRINT "A/D": COLOR 15
    LINE (182, 100)-(202, 110), 10, B: LINE (192, 105)-(192, 150), 10
    LOCATE 4, 1: PRINT "Press SPACEBAR key to STOP"
  '********* Draw Blade 1 ********
    COLOR 10: LINE (192, 150)-(192, 163): LINE (152, 158)-(172, 168), , B
    LINE (212, 158)-(232, 168), , B: LINE (172, 163)-(212, 163)
    GET (152, 150)-(232, 177), RUBBER1%
    VIEW (152, 262)-(232, 297): CLS : VIEW
  '********* Draw Blade 2 ********
    LINE (192, 150)-(192, 163): LINE (162, 168)-(180, 162)
    LINE (180, 162)-(180, 172): LINE (180, 172)-(162, 177)
    LINE (162, 177)-(162, 168): LINE (204, 156)-(222, 150)
    LINE (222, 150)-(222, 160): LINE (222, 160)-(204, 165)
    LINE (204, 165)-(204, 156): LINE (180, 167)-(204, 160)
    GET (152, 150)-(232, 177), RUBBER2%
    VIEW (152, 262)-(232, 309): CLS : VIEW
  '********* Draw Blade 3 ********
    LINE (192, 163)-(192, 177)
    GET (192, 163)-(192, 177), RUBBER3%
    VIEW (191, 287)-(193, 309): CLS : VIEW
  '********* Draw Blade 4 *********
```

```
    LINE (192, 150)-(192, 163): LINE (222, 168)-(204, 162)
    LINE (204, 162)-(204, 172): LINE (204, 172)-(222, 177)
    LINE (222, 177)-(222, 168): LINE (180, 156)-(162, 150)
    LINE (162, 150)-(162, 160): LINE (162, 160)-(180, 165)
    LINE (180, 165)-(180, 156): LINE (180, 160)-(204, 167)
    GET (152, 150)-(232, 177), RUBBER4%
    VIEW (152, 262)-(232, 309): CLS : VIEW: COLOR 15
  '********** Put Blade for Animation *******
  DO
    PUT (152, 150), RUBBER1%, PSET
    FOR SLOW = 1 TO 100: NEXT SLOW
    PUT (152, 150), RUBBER2%, PSET
    FOR SLOW = 1 TO 100: NEXT SLOW
    PUT (192, 163), RUBBER3%, PSET
    FOR SLOW = 1 TO 100: NEXT SLOW
    PUT (152, 150), RUBBER4%, PSET
    FOR SLOW = 1 TO 100: NEXT SLOW
    K$ = INKEY$
  LOOP UNTIL K$ = " "
  WINDOW
END SUB
'#### SUBPROGRAM SIMULATION ####
SUB SIMULATE STATIC
DIM V!(1001, 11), VM!(5, 7), RK!(5, 11), VTIME!(1001)
SHARED PreCb!, PreCet!, PreCaco!, PreCaca!, PreCbl!, PreCbt!, PreCglu!
SHARED Cglu!, Chy!, Cco2!, GCTIMES, PGCCb!()
' ********** INPUT DATA *****
CLS : COLOR 15
```

```
I% = 0
'*** TRANSFER DATA FROM ONLINE PART
IF GCTIMES < 1 THEN
  VI(0, 1) = PreCbI ' BIOMASS CONC. (X)
  VI(0, 2) = 1      ' DIMENSIONLESS RNA (Y)
  VI(0, 3) = PreCgluI ' SUBSTRATE CONC. (S)
  VI(0, 4) = PreCbt   ' BUTYRIC ACID CONC. (BA)
  VI(0, 5) = PreCbII  ' BUTANOL CONC. (B)
  VI(0, 6) = PreCacaI ' ACETIC ACID CONC. (AA)
  VI(0, 7) = PreCacoI ' ACETONE CONC. (A)
  VI(0, 8) = PreCetI  ' ETHANOL CONC. (E)
  VI(0, 9) = 0        ' CARBON DIOXIDE CONC. (CO2)
  VI(0, 10) = 0       ' HYDROGEN CONC. (H2)
ELSE
  VI(0, 1) = PreCbI   ' BIOMASS CONC. (X)
  VI(0, 2) = 1        ' DIMENSIONLESS RNA (Y)
  VI(0, 3) = CgluI    ' SUBSTRATE CONC. (S)
  VI(0, 4) = PreCbt   ' BUTYRIC ACID CONC. (BA)
  VI(0, 5) = PreCbII  ' BUTANOL CONC. (B)
  VI(0, 6) = PreCacaI ' ACETIC ACID CONC. (AA)
  VI(0, 7) = PreCacoI ' ACETONE CONC. (A)
  VI(0, 8) = PreCetI  ' ETHANOL CONC. (E)
  VI(0, 9) = Cco2I    ' CARBON DIOXIDE CONC. (CO2)
  VI(0, 10) = ChyI    ' HYDROGEN CONC. (H2)
END IF
FOR W = 1 TO 6
  VMI(0, W) = VI(0, W)
NEXT W
```

```
'******* PRINT HEADER & CONC.OF MATERIAL AT TIME = 0 Hr *******
CLS
LOCATE 1, 1: PRINT " Time (Hr) "; "   X   "; "   Y   ";
PRINT "   S   "; "  BA   "; "   B   "
LOCATE 2, 26: PRINT "(g/l)"
LOCATE 3, 1: PRINT USING "   ###   "; VTIME(I%);
PRINT USING " ##.### "; V!(I%, 1);
PRINT USING " ##.### "; V!(I%, 2);
PRINT USING " ##.### "; V!(I%, 3);
PRINT USING " ##.### "; V!(I%, 4);
PRINT USING " ##.### "; V!(I%, 5);
LOCATE 12, 1: PRINT " Time (Hr) "; "   AA   "; "   A   ";
PRINT "   E   "; "  CO2   "; "   H2   "
LOCATE 13, 26: PRINT "(g/l)"
LOCATE 14, 1: PRINT USING "   ###   "; VTIME(I%);
PRINT USING " ##.### "; V!(I%, 6);
PRINT USING " ##.### "; V!(I%, 7);
PRINT USING " ##.### "; V!(I%, 8);
PRINT USING " ##.### "; V!(I%, 9);
PRINT USING " ##.### "; V!(I%, 10);
'******** BEGIN CALCULATION *****
DELTATIME! = .1: J% = 100: K% = 100
VTIME!(0) = 0
LTOP = 4: LBOT = 15
DO
 WHILE I% < J%
   VTIME!(I% + 1) = VTIME!(I%) + DELTATIME!
   '****** Use RUNG-KATTA *****
```

```
    FOR Z = 1 TO 4
      RK!(Z, 1) = FNBIOMASS!(VM!(Z - 1, 1), V!(I%, 2), V!(I%, 5))
      RK!(Z, 2) = FNCELLLYSIS!(VM!(Z - 1, 2), V!(I%, 3), V!(I%, 5))
      RK!(Z, 3) = FNSUBSTRATE!(VM!(Z - 1, 3), V!(I%, 1))
      RK!(Z, 4) = FNBUTYRATE!(VM!(Z - 1, 4), V!(I%, 3), V!(I%, 5), V!(I%, 1))
      RK!(Z, 5) = FNBUTANOL!(VM!(Z - 1, 5), V!(I%, 3), V!(I%, 1), V!(I%, 4))
      RK!(Z, 6) = FNACETATE!(VM!(Z - 1, 6), V!(I%, 3), V!(I%, 5), V!(I%, 1))
      'RK!(Z, 7) = FNACETONE!(V!(I%, 6), V!(I%, 3), V!(I%, 5), V!(I%, 1))
      'RK!(Z, 8) = FNETHANOL!(V!(I%, 3), V!(I%, 1))
      'RK!(Z, 9) = FNCARBON!(V!(I%, 3), V!(I%, 1))
      'RK!(Z, 10) = FNHYDROGEN!(V!(I%, 3), V!(I%, 1))
      FOR W = 1 TO 6
        IF Z <= 2 THEN
          VM!(Z, W) = V!(I%, W) + (DELTATIME! / 2 * RK!(Z, W))
        ELSE
          VM!(Z, W) = V!(I%, W) + (DELTATIME! * RK!(Z, W))
        END IF
      NEXT W
    NEXT Z
    FOR W = 1 TO 6
      V!(I% + 1, W) = V!(I%, W) + (1 / 6 * (RK!(1, W) + (2 * RK!(2, W)) + (2 * RK!(3, W)) + RK!(4, W)) *
DELTATIME!)
      'IF W <= 6 THEN
        VM!(0, W) = V!(I% + 1, W)
      'END IF
    NEXT W
    IF GCTIMES < 1 THEN
      PGCCb!(I% + 1) = V!(I% + 1, 1)
```

```
    END IF
    '****** Use EULER *****
    XMID! = (V!(I%, 1) + V!(I% + 1, 1)) / 2
    SMID! = (V!(I%, 3) + V!(I% + 1, 3)) / 2
    BMID! = (V!(I%, 5) + V!(I% + 1, 5)) / 2
    AAMID! = (V!(I%, 6) + V!(I% + 1, 6)) / 2
    RK!(1, 7) = FNACETONE!(AAMID!, SMID!, BMID!, XMID!)
    RK!(1, 8) = FNETHANOL!(SMID!, XMID!)
    RK!(1, 9) = FNCARBON!(SMID!, XMID!)
    RK!(1, 10) = FNHYDROGEN!(SMID!, XMID!)
    FOR W = 7 TO 10
      V!(I% + 1, W) = V!(I%, W) + (RK!(1, W) * DELTATIME!)
        NEXT W
    '******* INNER LOOP *****
    I% = I% + 1
  WEND
  '******* SHOW RESULT ****
  LOCATE LTOP, 1: PRINT USING "    ###    "; VTIME(I%);
  PRINT USING " ##.### "; V!(I%, 1);
  PRINT USING " ##.### "; V!(I%, 2);
  PRINT USING " ##.### "; V!(I%, 3);
  PRINT USING " ##.### "; V!(I%, 4);
  PRINT USING " ##.### "; V!(I%, 5)
  LOCATE LBOT, 1: PRINT USING "    ###    "; VTIME(I%);
  PRINT USING " ##.### "; V!(I%, 6);
  PRINT USING " ##.### "; V!(I%, 7);
  PRINT USING " ##.### "; V!(I%, 8);
  PRINT USING " ##.### "; V!(I%, 9);
```

```
    PRINT USING " ##.### "; V!(I%, 10)
    J% = J% + K%
    LTOP = LTOP + 1: LBOT = LBOT + 1
  LOOP UNTIL VTIME!(I%) >= 60
  COLOR 12: LOCATE 23, 20: PRINT "PRESS ANY KEY TO CONTINUE"
  P$ = INPUT$(1): CLS
  XTMIN = VTIME!(0): XTMAX = VTIME!(I%): YXMIN = 0: YXMAX = V!(280, 1) + .5
  YSMIN = 0: YSMAX = V!(0, 3)
  NRANGE% = XTMAX / 10
  '******** PLOT GRAPH OF PRODUCT **********
  VIEW (85, 20)-(385, 135): CLS
  WINDOW (XTMIN, YXMIN)-(XTMAX, YXMAX)
  LINE (XTMIN, YXMIN)-(XTMAX, YXMAX), 10, B
   '*** DRAW SCALE FOR X-AXIS
  DELTA2 = (XTMAX - XTMIN) / NRANGE%
  FOR E = 1 TO (NRANGE% - 1)
    X2 = E * DELTA2 + XTMIN
    LINE (X2, YXMIN)-(X2, YXMAX), 11, , &HCCCC
  NEXT E
  LOCATE 12, 11: PRINT USING "#"; XTMIN
  LOCATE 12, 26: PRINT "TIME (Hr)"
  LOCATE 12, 48: PRINT USING "##"; XTMAX
   '*** DRAW SCALE FOR Y-AXIS
  DELTA2 = (YXMAX - YXMIN) / 5
  FOR E = 1 TO 4
    Y2 = E * DELTA2 + YXMIN
    LINE (XTMIN, Y2)-(XTMAX, Y2), 11, , &HCCCC
  NEXT E
```

```
LOCATE 2, 7: PRINT USING "##.#"; YXMAX
LOCATE 6, 2: PRINT "CONC(g/l)"
LOCATE 10, 7: PRINT USING "##.#"; YXMIN
'*** DRAW GRAPH
PSET (VTIME!(0), V!(0, 1)), 6
FOR E = 1 TO I%
  LINE -(VTIME!(E), V!(E, 1)), 6
NEXT E
PSET (VTIME!(0), V!(0, 4)), 9
FOR E = 1 TO I%
  LINE -(VTIME!(E), V!(E, 4)), 9
NEXT E
PSET (VTIME!(0), V!(0, 6)), 10
FOR E = 1 TO I%
  LINE -(VTIME!(E), V!(E, 6)), 10
NEXT E
PSET (VTIME!(0), V!(0, 8)), 11
FOR E = 1 TO I%
  LINE -(VTIME!(E), V!(E, 8)), 11
NEXT E
PSET (VTIME!(0), V!(0, 10)), 12
FOR E = 1 TO I%
  LINE -(VTIME!(E), V!(E, 10)), 12
NEXT E
VIEW (85, 175)-(385, 290): CLS
WINDOW (XTMIN, YSMIN)-(XTMAX, YSMAX)
LINE (XTMIN, YSMIN)-(XTMAX, YSMAX), 10, B
'*** DRAW SCALE FOR X-AXIS
```

```
DELTA2 = (XTMAX - XTMIN) / NRANGE%
FOR E = 1 TO (NRANGE% - 1)
 X2 = E * DELTA2 + XTMIN
 LINE (X2, YSMIN)-(X2, YSMAX), 11, , &HCCCC
NEXT E
'*** DRAW SCALE FOR Y-AXIS
DELTA2 = (YSMAX - YSMIN) / 5
FOR E = 1 TO 4
 Y2 = E * DELTA2 + YSMIN
 LINE (XTMIN, Y2)-(XTMAX, Y2), 11, , &HCCCC
NEXT E
LOCATE 13, 7: PRINT USING "##.#"; YSMAX
LOCATE 17, 2: PRINT "CONC(g/l)"
LOCATE 21, 7: PRINT USING "##.#"; YSMIN
'*** DRAW GRAPH
PSET (VTIME!(0), V!(0, 3)), 13
FOR E = 1 TO I%
 LINE -(VTIME!(E), V!(E, 3)), 13
NEXT E
PSET (VTIME!(0), V!(0, 5)), 14
FOR E = 1 TO I%
 LINE -(VTIME!(E), V!(E, 5)), 14
NEXT E
PSET (VTIME!(0), V!(0, 7)), 15
FOR E = 1 TO I%
 LINE -(VTIME!(E), V!(E, 7)), 15
NEXT E
```

```
PSET (VTIME!(0), V!(0, 9)), 4
FOR E = 1 TO I%
  LINE -(VTIME!(E), V!(E, 9)), 4
NEXT E
VIEW (500, 40)-(600, 270)
WINDOW (500, 40)-(600, 270)
LINE (500, 40)-(600, 270), 7, B
LINE (520, 245)-(540, 245), 6
LINE (560, 245)-(580, 245), 9
LINE (520, 200)-(540, 200), 10
LOCATE 6, 66: COLOR 15: PRINT " X "; "  BA "
LOCATE 9, 66: PRINT "AA "; "   E "
LOCATE 12, 66: PRINT "H2 "; "   S "
LINE (560, 200)-(580, 200), 11
LINE (520, 160)-(540, 160), 12
LINE (560, 160)-(580, 160), 13
LOCATE 15, 66: PRINT " B "; "   A "
LOCATE 18, 66: PRINT "CO2 "
LINE (520, 120)-(540, 120), 14
LINE (560, 120)-(580, 120), 15
LINE (520, 80)-(540, 80), 4
COLOR 12: LOCATE 23, 30: PRINT "PRESS ANY KEY TO CONTINUE"
P$ = INPUT$(1)
VIEW: WINDOW: CLS
END SUB
```

คุณลักษณะของอุปกรณ์อินเทอร์เฟซ

คุณลักษณะของการ์ดแปลงผันแอนะล็อกเป็นดิจิตอล PCL-711S (5)

* Analog Input (A/D Converter)

Channels :	8 single-ended.
Resolution :	12 bits.
Input Range :	Bipolar : +/- 5V.
Overvoltage :	Continuous +/- 30V Max.
Conversion type :	Successive Approximation.
Converter :	AD574 or equivalent.
Conversion speed :	25 microsecond max.
Accuracy :	0.015 % of reading +/- 1bit.
Linearity :	+/- 1 bit.
Trigger mode :	Software trigger.
Data transfer :	Program control.

* Analog Output (D/A Converter)

Channels :	1 channel.
Resolution :	12 Bits.
Output Range :	0 to +5V or 0 to +10V.
Reference voltage :	Internal -5V and -10V (+/- 0.005V).
Conversion type :	12 bit monolithic multiplying.
Analog devices :	AD7541AKN or equivalent.
Linearity :	+/- 1/2 bit.
Output drive :	+/- 5mA max.
Settling time :	30 microseconds.

* Digital Input

Channel :	16 bits.
Level :	TTL compatible.
Input voltage :	Low - 0.8V max. High - 2.0V min.
Input load :	Low - 0.4mA max. at 0.5V.
	High - 0.05mA max. at 2.7V.

* Digital Output

Channel :	16 bits.
Level :	TTL compatible.
Output voltage :	Low - Sink 8mA at 0.5V max.
	High - Source -0.4mA at 2.4V min.

* General Specifications

Power consumption :	+5V : typ. 100mA, max. 500mA.
	+12V : typ. 40mA, max. 100mA.
	-12V : typ. 20mA, max. 50mA.
I/O connector :	20-pin post headers for Analog/Digital I/O ports. Adapter available to convert to 37 pin D-type connecter.
I/Obaseaddress :	Requires 16 consecutive address locations Base address definable by the DIP switch for address line A9 - A4. (Factory setting is hex 220).
Operating Temp :	0 to +50 deg.C.
Storage Temp :	-20 to +65 deg.C.
Weight :	4.49 oz (127 gms).

คุณลักษณะของบอร์ด Amplifier/Multiplexer PCLD-789 (6)

Input Channel :	16 differential channels.
Input Range :	+/- 10V Maximum, varies on the gain selection.
Input Conditions:	

Gains	Commom Mode Rejection	Non-Linearity	Settling Time	Maximum Input Voltage
1000	125 dB	0.005	0.05 mS	+/- 10mV
100	125 dB	0.005	0.015 mS	+/- 100mV
10	110 dB	0.007	0.013 mS	+/- 1V
1	90 dB	0.015	0.012 mS	+/- 10V

Overvoltage Protection : +/- 30V continuous.

Common Mode Voltage : +/- 10V max.

Output Range : +/- 10V max.

Output Current : 20 mA max.

Cold Junction Compensation : +24.4mV/Deg. C. (0.0 V at 0.0 Deg. C.)

Power Consumption : +5V (10mA max).

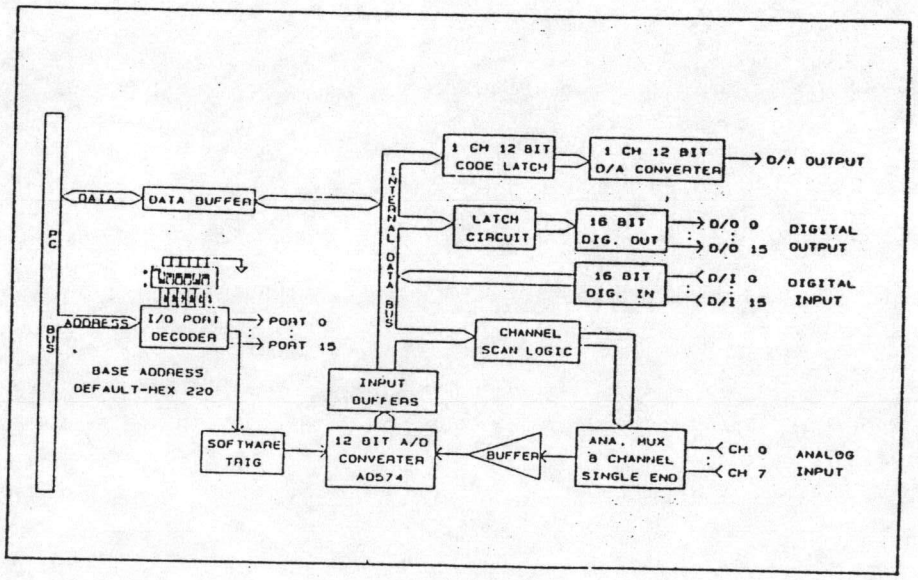
+12V (80mA max).

Connector type : 20 pin flat cable connectors.

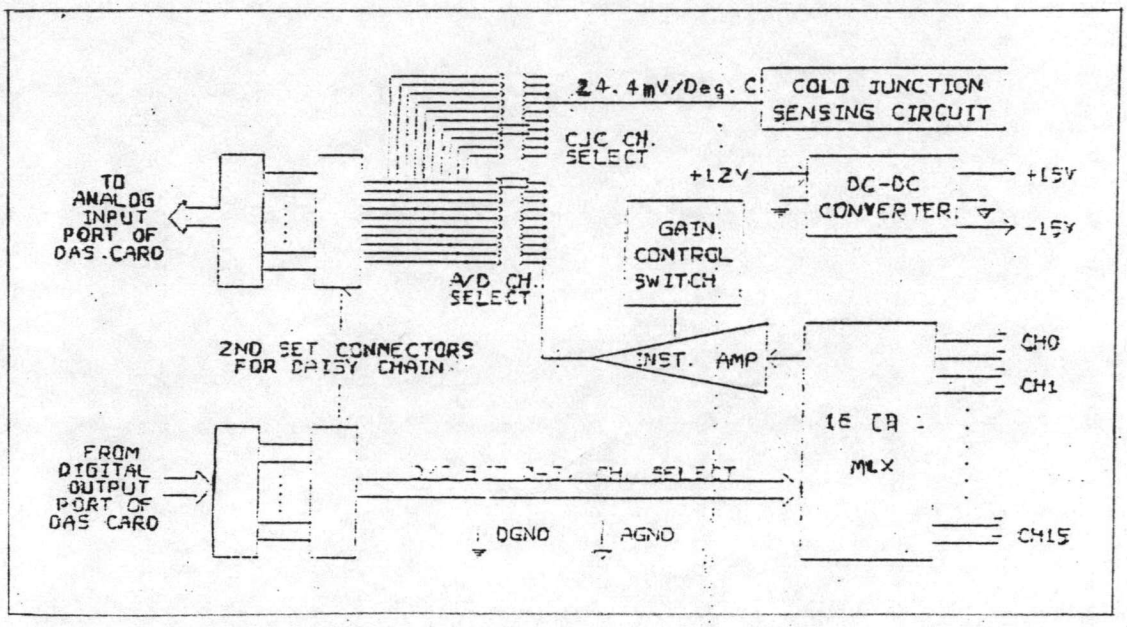
Dimensions : 20.5 cm (L) x 11.43 cm (W) or 8.07 " (L) x 4.5 " (W).

Weight : 0.527 Lbs (239 gms).

PCL-711S BLOCK DIAGRAM

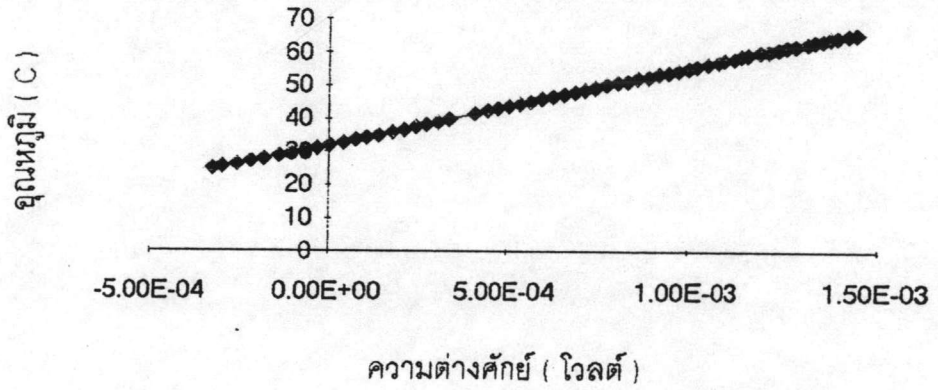


PCLD-789 BLOCK DIAGRAM

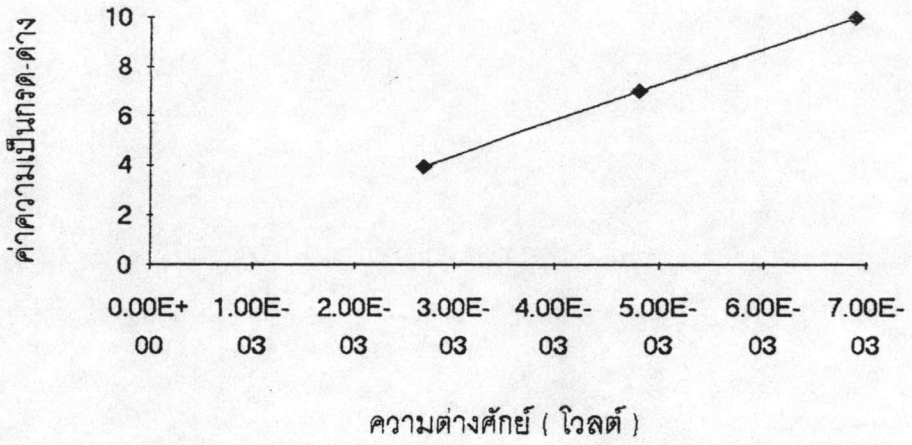


การสอบเทียบความถูกต้องของเทอร์มอคัปเปิลและเครื่องควบคุมความเป็นกรด-ด่าง

ในการสอบเทียบความถูกต้องของเทอร์มอคัปเปิลนั้น จะเก็บข้อมูลค่าความต่างศักย์ที่ส่งออกมาจากเทอร์มอคัปเปิล ณ ค่าอุณหภูมิจุดต่างๆ โดยค่าอุณหภูมิจุดต่างๆนี้ จะอ่านได้จากจอแสดงผลของเครื่องอังน้ำซึ่งใช้เทอร์มอคัปเปิลแบบ Pt 100 ในการวัดค่าอุณหภูมิ สำหรับเครื่องควบคุมความเป็นกรด-ด่างก็เช่นเดียวกัน โดยจะเก็บข้อมูลค่าความต่างศักย์ที่ส่งออกมา ณ ค่าความเป็นกรด-ด่างจุดต่างๆ โดยสารละลายที่นำมาใช้วัดค่าความเป็นกรด-ด่างนั้น จะใช้สารละลายบัฟเฟอร์มาตราฐาน ซึ่งมีค่าความเป็นกรด-ด่างเท่ากับ 4.0 , 7.0 และ 10.0 มาใช้งาน และจะได้ชุดข้อมูลดังแสดงในกราฟรูปที่ 1 และ 2 ตามลำดับ



รูปที่ 1 ค่าความต่างศักย์ที่ออกจากเทอร์มอคัปเปิล ณ อุณหภูมิต่างๆ



รูปที่ 2 ค่าความต่างศักย์ที่ออกจากเครื่องควบคุมความเป็นกรด-ด่าง ณ ค่าความเป็นกรด-ด่างต่างๆ

จากข้อมูลที่ได้จะอาศัยโปรแกรมชื่อ GRAPHER รุ่น 1.8 เพื่อหาสมการที่แสดงถึงความสัมพันธ์ระหว่างค่าอุณหภูมิกับค่าความต่างศักย์ และ ค่าความเป็นกรด-ด่างกับค่าความต่างศักย์ ที่สอดคล้องกับข้อมูลที่ได้มา โดยจะได้สมการสหสัมพันธ์ออกมาดังนี้

สมการสหสัมพันธ์ระหว่างอุณหภูมิกับความต่างศักย์

ค่าอุณหภูมิ = (ค่าความต่างศักย์ * 23298) + 31.98

สมการสหสัมพันธ์ระหว่างค่าความเป็นกรด-ด่างกับความต่างศักย์

ค่าความเป็นกรด-ด่าง = (ค่าความต่างศักย์ * 1436.09)

เวลา	X	S	B	A	E	AA	BA
(ชม.)	(กรัม/ลิตร)						
0	0.143	46	0.3952	0.3025	0.0426	0.1751	0
10	0.143	45.5	0.464	0.1312	0.0082	0.4038	0.4379
16.5	0.259	39	1.7147	0.6412	0.0548	1.0061	0.4777
22	3.231	32	3.2689	1.3074	0.1567	0.9233	0.2329
28	3.76	26	4.5458	1.8749	0.0798	0.4516	0.1
34	2.803	19.5	6.484	2.7127	0.1884	0.7562	0.1763
40.5	3.769	14.5	7.9676	3.1107	0.2644	0.6985	0.1132
46	3.577	11	8.9897	3.5507	0.2859	0.7639	0.1814
54.5	3.054	8.6	10.6163	4.4517	0.2164	0.3498	0.1003
59	4	7	10.4221	4.3235	0.3107	0.6313	0.1817
65	3.615	6	10.852	4.5265	0.3623	0.79	0.2839
70	4.231	5.9	10.9137	4.4259	0.2205	0.2563	0.1404
76	3.515	5.1	11.58	4.4929	0.3387	0.6093	0.1127
82	4.054	4.9	10.9014	4.3751	0.227	0.2414	0.1551
88	3.677	4.7	10.7843	4.0107	0.2197	0.2652	0.2138

ตารางที่ 1 ข้อมูลความเข้มข้นของผลิตภัณฑ์ที่ได้จากการหมักเดคโตส ที่ pH 5.5 (7)

เวลา	X	S	B	A	E	AA	BA
(ชม.)	(กรัม/ลิตร)						
0	0.143	46	0.395	0.303	0.043	0.175	0
10	0.977	38.664	1.624	0.408	0.148	1.273	0.962
20	1.962	17.882	6.602	2.497	0.524	1.652	0.851
30	2.07	2.3	9.359	5.254	0.938	1.121	0.092
40	1.937	0.004	9.548	5.777	1.016	1.016	0.001
50	1.795	0	9.549	5.778	1.017	1.016	0
60	1.663	0	9.549	5.778	1.017	1.016	0
70	1.54	0	9.549	5.778	1.017	1.016	0

ตารางที่ 2 ข้อมูลความเข้มข้นของผลิตภัณฑ์ที่ได้จากการทำนายโดยใช้ค่าคงที่ของ Votruba และคณะ

เวลา	X	S	B	A	E	AA	BA
(ชม.)	(กรัม/ลิตร)						
0	0.143	46	0.395	0.303	0.043	0.175	0
10	0.466	42.142	1.242	0.346	0.057	0.732	0.436
20	0.817	33.383	4.168	1.013	0.094	0.979	0.087
30	0.885	23.806	6.799	2.093	0.141	0.824	0.035
40	0.799	16.176	8.58	3.156	0.186	0.601	0.018
50	0.66	10.807	9.614	4.021	0.223	0.44	0.011
60	0.52	7.223	10.17	4.66	0.251	0.347	0.007
70	0.4	4.89	10.461	5.109	0.27	0.298	0.005

ตารางที่ 3 ข้อมูลความเข้มข้นของผลิตภัณฑ์ที่ได้จากการทำนายโดยใช้ค่าคงที่หาได้

ค่าคงที่	ค่า
k1	0.009
k2	0.0008
k3	0.0255
k4	0.6764
k5	0.0136
k6	0.117
k7	0.0113
k8	0.715
k9	0.135
k10	0.1558
k11	0.0258
k12	0.6139
k13	0.0185
k14	0.00013

ตารางที่ 4 ค่าคงที่ (k1-k14) ที่ได้จากการศึกษาของ Votruba และคณะ (4)

เวลา	[กลูโคส]	[กลูโคส]	[H2]	[CO2]	ATP*tot	ATP*ex	NfF*	Y-atp	Y-s
(ชม.)	(โมล/ลิตร)	(กรัม/ลิตร)			(โมล/ลิตร)			กรัมชีวมวล/ โมล ATP	กรัมชีวมวล/ กรัมกลูโคส
10	0.2353	42.40106	0.06	1.59	0.048	0.033	0.004	9.742	0.129
22	0.1714	30.88628	0.242	7.92	0.198	0.172	0.036	4.13	0.054
28	0.1502	27.06604	0.282	10.16	0.24	0.212	0.057	3.692	0.047
40.5	0.0791	14.25382	0.472	17.42	0.409	0.383	0.106	1.956	0.025
50	0.0534	9.62268	0.488	19.89	0.463	0.442	0.118	1.464	0.019
59	0.0262	4.72124	0.64	23.14	0.539	0.522	0.131	0.966	0.013
70	0.0231	4.16262	0.628	23.54	0.541	0.529	0.145	0.739	0.009

ตารางที่ 5 ตัวแปรที่วัดไม่ได้ที่ได้จากการคำนวณ

เวลา	ความเข้มข้นของกลูโคส		
(ซม.)	ค่าที่รายงานโดยจิรกานต์และคณะ	จากการคำนวณ	
		โดยใช้ค่าชีวมวลที่คำนวณได้	โดยใช้ค่าชีวมวลที่รายงานโดยจิรกานต์
	(กรัม/ลิตร)		
10	45.5	42.4	42.78
16.5	39		36.96
22	32	30.89	28.09
28	26	27.07	23.75
34	19.5		16.7
40.5	14.5	14.25	10.83
46	11		11.41
50		9.62	
54.5	8.6		1.75
59	7	4.72	0.7
65	6		0
70	5.9	4.16	0

ตารางที่ 6 ค่าความเข้มข้นของกลูโคสที่ได้จากการใช้สมการสำหรับการหาค่าของตัวแปรที่วัดไม่ได้มาคำนวณหา

ประวัติผู้เขียน

นาย จำเริญ เลี่ยวศรีสุข เกิดวันที่ 16 กรกฎาคม พ.ศ. 2511 ที่อำเภอเมือง จังหวัดนครปฐม สำเร็จการศึกษาปริญญาตรีวิทยาศาสตรบัณฑิต สาขาวิชาเคมีอุตสาหกรรม ภาควิชาเคมี คณะวิทยาศาสตร์ มหาวิทยาลัยเชียงใหม่ ในปีการศึกษา 2532