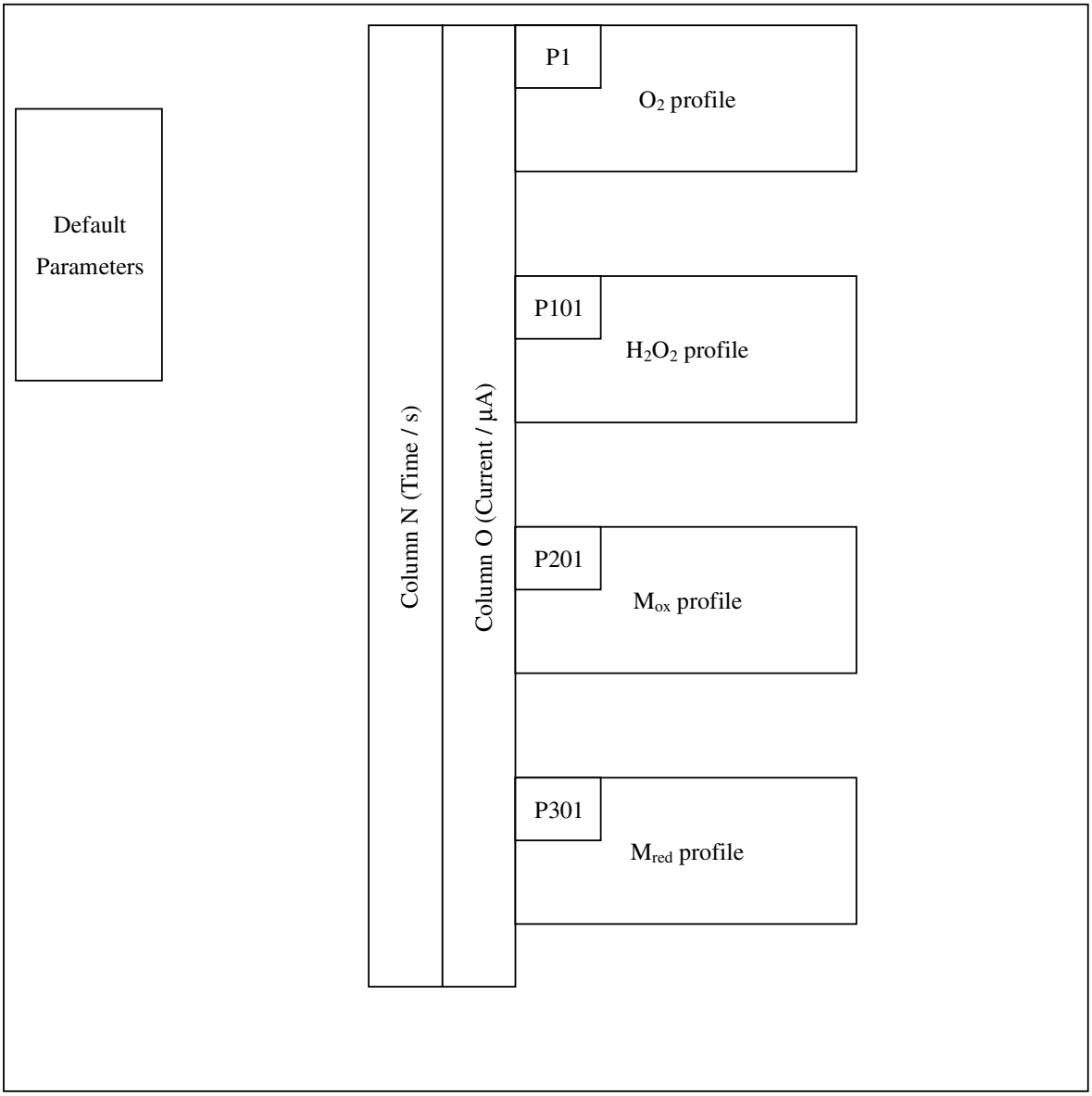
Supplemental Information
on Digital Simulation of Mediated Bienzyme Sensor

Excel® Worksheet Profile



Cell Location of Parameters on Worksheet

B37 : $[M_{red}]$ / mM

B38 : $[M_{ox}]$ / mM

B39 : [Glc] / mM

B40 : $[O_2]$ / mM

B41 : K_s^{OX} / mM

B42 : K_m^{OX} / mM

B43 : V_{max}^{OX} / mM s^{-1}

B44 : K_s^{DH} / mM

B45 : K_m^{DH} / mM

B46 : V_{max}^{DH} / mM s^{-1}

B47 : K_s^{POD} / mM1

B48 : K_m^{POD} / mM

B49 : V_{max}^{POD} / mM s^{-1}

B50 : Δt / s

B51 : Total time / s

B52 : Layer thickness / cm

D35 : Diffusion coefficient of mediator

D36 : Diffusion coefficient of glucose

D37 : Diffusion coefficient of hydrogen peroxide

Excel® Macro Program

```
Sub program()

    D = 0.00001          'Diffusion coefficient of oxygen
    Eo = 0               'Redox potential
    Ei = -800            'Electrode potential
    n = 1                'Mole
    A = 0.01             'Electrode surface area
    F = 96500            'Faraday constant
    R = 8.31             'Gas constant
    T = 298              'Temperature
```

```
initialred = Sheet1.Cells(37, 2)
initialox = Sheet1.Cells(38, 2)
initialglc = Sheet1.Cells(39, 2)
initialH2O2 = 0
initialO2 = Sheet1.Cells(40, 2)
KsO2 = Sheet1.Cells(41, 2)
KmO2 = Sheet1.Cells(42, 2)
kcatEO2 = Sheet1.Cells(43, 2)
KsFc = Sheet1.Cells(44, 2)
KmFc = Sheet1.Cells(45, 2)
kcatEFc = Sheet1.Cells(46, 2)
KsPOD = Sheet1.Cells(47, 2)
kcatEPOD = Sheet1.Cells(49, 2)
dt = Sheet1.Cells(50, 2)
Ttotal = Sheet1.Cells(51, 2)
L = Sheet1.Cells(52, 2)

nFRT = n * F / (R * T * 1000)
dx = Sqr(D * dt / 0.45)
imax = Int(Ttotal / dt)
jmax = Int(L / dx) + 1
ratio = Exp(nFRT * (Ei - Eo))
Dm = 0.45 * Sheet1.Cells(35, 5) / D
Dglc = 0.45 * Sheet1.Cells(36, 5) / D
DH2O2 = 0.45 * Sheet1.Cells(37, 5) / D

ReDim red(imax + 1, jmax)
ReDim ox(imax + 1, jmax)
ReDim glc(imax + 1, jmax)
ReDim O2(imax + 1, jmax)
ReDim H2O2(imax + 1, jmax)
ReDim cred(imax, jmax)
ReDim cox(imax, jmax)
ReDim cglc(imax, jmax)
ReDim cH2O2(imax, jmax)
```

```
    ReDim cO2(imax, jmax)
    ReDim kineticsO2(imax, jmax)
    ReDim kineticsFc(imax, jmax)
    ReDim kineticsPOD(imax, jmax)
    ReDim Time(imax)
    ReDim distance(jmax)

'time axis
Time(0) = 0
For i = 0 To imax - 1
    Time(i + 1) = Time(i) + dt
Next i

'initial concentration
For j = 1 To jmax
    red(0, j) = initialred
    ox(0, j) = initialox
    glc(0, j) = initialglc
    H2O2(0, j) = initialH2O2
    O2(0, j) = initialO2
Next j

    red(0, 0) = (red(0, 1) + ox(0, 1)) / (1 + ratio)
    ox(0, 0) = red(0, 0) * ratio
    O2(0, 0) = initialO2
    glc(0, 0) = initialglc
    H2O2(0, 0) = initialH2O2

'concentration profiles
For i = 0 To imax
    For j = 0 To jmax
        kineticsO2(i, j) = O2(i, j) * glc(i, j) * KmFc * kcatEO2 / (glc(i, j) * KmFc * KmO2 + O2(i,
j) * glc(i, j) * KmFc + ox(i, j) * glc(i, j) * KmO2 + O2(i, j) * KsO2 * KmFc + ox(i, j) * KsFc *
KmO2)
        kineticsFc(i, j) = ox(i, j) * glc(i, j) * KmO2 * kcatEFc / (glc(i, j) * KmFc * KmO2 + O2(i, j)
* glc(i, j) * KmFc + ox(i, j) * glc(i, j) * KmO2 + O2(i, j) * KsO2 * KmFc + ox(i, j) * KsFc * KmO2)
```

```
    kineticsPOD(i, j) = kcatEPOD * H2O2(i, j) / (H2O2(i, j) + KsPOD)
Next j
For j = 1 To jmax - 1
    cglc(i, j) = glc(i, j) - kineticsO2(i, j) * dt - kineticsFc(i, j) * dt
    cO2(i, j) = O2(i, j) - kineticsO2(i, j) * dt
    cH2O2(i, j) = H2O2(i, j) + kineticsO2(i, j) * dt - kineticsPOD(i, j) * dt
    cred(i, j) = red(i, j) + 2 * kineticsFc(i, j) * dt - 2 * kineticsPOD(i, j) * dt
    cox(i, j) = ox(i, j) - 2 * kineticsFc(i, j) * dt + 2 * kineticsPOD(i, j) * dt
Next j

cred(i, 0) = (cred(i, 1) + cox(i, 1)) / (1 + ratio)
cox(i, 0) = cred(i, 0) * ratio
cO2(i, 0) = cO2(i, 1)
cH2O2(i, 0) = cH2O2(i, 1)
cglc(i, 0) = cglc(i, 1)

cglc(i, jmax) = initialglc
cO2(i, jmax) = initialO2
cH2O2(i, jmax) = initialH2O2
cred(i, jmax) = initialred
cox(i, jmax) = initialox

ox(i + 1, 1) = cox(i, 1) + Dm * (cox(i, 2) - 3 * cox(i, 1) + 2 * cox(i, 0))
red(i + 1, 1) = cred(i, 1) + Dm * (cred(i, 2) - 3 * cred(i, 1) + 2 * cred(i, 0))
glc(i + 1, 1) = cglc(i, 1) + Dglc * (cglc(i, 2) - cglc(i, 1))
H2O2(i + 1, 1) = cH2O2(i, 1) + DH2O2 * (cH2O2(i, 2) - cH2O2(i, 1))
O2(i + 1, 1) = cO2(i, 1) + 0.45 * (cO2(i, 2) - cO2(i, 1))

red(i + 1, 0) = (red(i + 1, 1) + ox(i + 1, 1)) / (1 + ratio)
ox(i + 1, 0) = red(i + 1, 0) * ratio
glc(i + 1, 0) = glc(i + 1, 1)
O2(i + 1, 0) = O2(i + 1, 1)
H2O2(i + 1, 0) = H2O2(i + 1, 1)

For j = 1 To jmax - 2
    red(i + 1, j + 1) = cred(i, j + 1) + Dm * (cred(i, j + 2) - 2 * cred(i, j + 1) + cred(i, j))
```

```
        ox(i + 1, j + 1) = cox(i, j + 1) + Dm * (cox(i, j + 2) - 2 * cox(i, j + 1) + cox(i, j))
        glc(i + 1, j + 1) = cglc(i, j + 1) + Dglc * (cglc(i, j + 2) - 2 * cglc(i, j + 1) + cglc(i, j))
        O2(i + 1, j + 1) = cO2(i, j + 1) + 0.45 * (cO2(i, j + 2) - 2 * cO2(i, j + 1) + cO2(i, j))
        H2O2(i + 1, j + 1) = cH2O2(i, j + 1) + DH2O2 * (cH2O2(i, j + 2) - 2 * cH2O2(i, j + 1) +
cH2O2(i, j))
    Next j

    red(i + 1, jmax) = initialred
    ox(i + 1, jmax) = initialox
    glc(i + 1, jmax) = initialglc
    O2(i + 1, jmax) = initialO2
    H2O2(i + 1, jmax) = initialH2O2

Next i

'distance
    distance(0) = 0
    distance(1) = dx / 2
    For j = 1 To jmax - 2
        distance(j + 1) = distance(j) + dx
    Next j
    distance(jmax) = 0.01

'output

    For i = 0 To imax
        Sheet1.Cells(i + 1, 14) = Time(i)
        Sheet1.Cells(i + 1, 15) = 2 * n * F * A * Sheet1.Cells(35, 5) * (cred(i, 1) - cred(i, 0)) / dx
    Next i

    For j = 0 To jmax
        Sheet1.Cells(j + 1, 16) = distance(j)
        Sheet1.Cells(j + 101, 16) = distance(j)
        Sheet1.Cells(j + 201, 16) = distance(j)
        Sheet1.Cells(j + 301, 16) = distance(j)
    Next j
```

```
    For i = 0 To imax Step 50
        For j = 0 To jmax
            Sheet1.Cells(j + 1, i / 50 + 17) = cO2(i, j)
            Sheet1.Cells(j + 101, i / 50 + 17) = cH2O2(i, j)
            Sheet1.Cells(j + 201, i / 50 + 17) = cox(i, j)
            Sheet1.Cells(j + 301, i / 50 + 17) = cred(i, j)
        Next j
    Next i

End Sub
```