## SI Appendix

### CellulaRhythm Code for Tecan Luminometer

```
#########################################################################
#######
# CellulaRhythm 6.T     for R 2.3.1/2.4.0/2.5.0
# read, fit, and plot circadian data
# Tecan format 384
# Warren G. Lewis         February  2008
#########################################################################
#######
#linear detrended exponential damped cosine
#with linear parameter estimates
#########################################################################
#######
#For further info, check out R project online at http://www.r-
project.org/
#########################################################################
#######
# delete all text above Cycle Number   1  2  3  4 ...
# then save the Tecan output file as Text (tab delimited) from
Microsoft Excel
# change the red text to match your file and folder names
# change the red numbers to set values for interval and end of run
# be careful not to change the quotes, or other parts of the program
#########################################################################
#######
method<-"lincosdamp"

# name of tab-delimited text data file
filename<-"filename.txt"

# heatmap limit in # of points
STOP<-54

# last point included in curve fitting (nls)
END<-54

# wells in plate (384 or 96)
plate.wells<-384

# interval in hours  (Tecan gives time in seconds--divide by 3600 to
get hours)
interval=1.66666
totaltime<-interval*END
totaltime

# make a new folder for each analysis
# set the working directory to match the data folder
setwd("D:/folder")
getwd()

# screen date, along with a short description of the experiment
screendate<-"description"

image.size<-450
pdf.long<-11
pdf.short<-8.5

# first row of luminescence data in text file
row.first<-4
row.last<-(row.first+plate.wells-1)
```

```
# number of points you consider to be the acute induction part of the
data
drop.points<-12
startpoint<-(drop.points+1)
clippoint<-(2+startpoint)
halfpoint<-trunc((END-drop.points)/2)

period.est<-24
damp.est<-0.005
large.size<-1200
databins<-100
damp.limit<-0.1
amp.limit<-10
squares.limit<-30000000

ifelse(plate.wells==384, Wells<-
c("P24","P23","P22","P21","P20","P19","P18","P17","P16","P15","P14","P1
3","P12","P11","P10","P09","P08","P07","P06","P05","P04","P03","P02","P
01",
"O24","O23","O22","O21","O20","O19","O18","O17","O16","O15","O14","O13"
,"O12","O11","O10","O09","O08","O07","O06","O05","O04","O03","O02","O01
",
"N24","N23","N22","N21","N20","N19","N18","N17","N16","N15","N14","N13"
,"N12","N11","N10","N09","N08","N07","N06","N05","N04","N03","N02","N01
",
"M24","M23","M22","M21","M20","M19","M18","M17","M16","M15","M14","M13"
,"M12","M11","M10","M09","M08","M07","M06","M05","M04","M03","M02","M01
",
"L24","L23","L22","L21","L20","L19","L18","L17","L16","L15","L14","L13"
,"L12","L11","L10","L09","L08","L07","L06","L05","L04","L03","L02","L01
",
"K24","K23","K22","K21","K20","K19","K18","K17","K16","K15","K14","K13"
,"K12","K11","K10","K09","K08","K07","K06","K05","K04","K03","K02","K01
",
"J24","J23","J22","J21","J20","J19","J18","J17","J16","J15","J14","J13"
,"J12","J11","J10","J09","J08","J07","J06","J05","J04","J03","J02","J01
",
"I24","I23","I22","I21","I20","I19","I18","I17","I16","I15","I14","I13"
,"I12","I11","I10","I09","I08","I07","I06","I05","I04","I03","I02","I01
",
"H24","H23","H22","H21","H20","H19","H18","H17","H16","H15","H14","H13"
,"H12","H11","H10","H09","H08","H07","H06","H05","H04","H03","H02","H01
",
"G24","G23","G22","G21","G20","G19","G18","G17","G16","G15","G14","G13"
,"G12","G11","G10","G09","G08","G07","G06","G05","G04","G03","G02","G01
",
"F24","F23","F22","F21","F20","F19","F18","F17","F16","F15","F14","F13"
,"F12","F11","F10","F09","F08","F07","F06","F05","F04","F03","F02","F01
",
"E24","E23","E22","E21","E20","E19","E18","E17","E16","E15","E14","E13"
,"E12","E11","E10","E09","E08","E07","E06","E05","E04","E03","E02","E01
",
"D24","D23","D22","D21","D20","D19","D18","D17","D16","D15","D14","D13"
,"D12","D11","D10","D09","D08","D07","D06","D05","D04","D03","D02","D01
",
"C24","C23","C22","C21","C20","C19","C18","C17","C16","C15","C14","C13"
,"C12","C11","C10","C09","C08","C07","C06","C05","C04","C03","C02","C01
",
"B24","B23","B22","B21","B20","B19","B18","B17","B16","B15","B14","B13"
,"B12","B11","B10","B09","B08","B07","B06","B05","B04","B03","B02","B01
",
"A24","A23","A22","A21","A20","A19","A18","A17","A16","A15","A14","A13"
,"A12","A11","A10","A09","A08","A07","A06","A05","A04","A03","A02","A01
```

```
"),
Wells<-
c("H12","H11","H10","H09","H08","H07","H06","H05","H04","H03","H02","H0
1",
"G12","G11","G10","G09","G08","G07","G06","G05","G04","G03","G02","G01"
,
"F12","F11","F10","F09","F08","F07","F06","F05","F04","F03","F02","F01"
,
"E12","E11","E10","E09","E08","E07","E06","E05","E04","E03","E02","E01"
,
"D12","D11","D10","D09","D08","D07","D06","D05","D04","D03","D02","D01"
,
"C12","C11","C10","C09","C08","C07","C06","C05","C04","C03","C02","C01"
,
"B12","B11","B10","B09","B08","B07","B06","B05","B04","B03","B02","B01"
,
"A12","A11","A10","A09","A08","A07","A06","A05","A04","A03","A02","A01"
))

WellsA1<-c(sort(Wells), "time(hours)")

Tecandata<-read.table(filename, header=FALSE, skip=(row.first-1),
fill=TRUE, row.names=1)

newdrop<-1
PLATE<-as.matrix((Tecandata[1:(plate.wells),newdrop:STOP]))

#####################################################
# check the data import, if there are errors, using the fix command
# >fix(PLATE)

integertimes<-(newdrop:STOP)

measurementtimes<-integertimes*interval
# if you want to define your own measurement times
# (e.g. there are uneven intervals)
# insert your own measurementtimes, using
# measurementsec<-c(0,3600,7200,10000)
#  you can make that list of times by saving your excel file as a csv
file
#  and pasting it between the parentheses above
# measurementtimes<-measurementsec/3600

PLATEwtime<-rbind(PLATE[1:plate.wells,],measurementtimes)
write.table(PLATEwtime,paste("screendata",filename), sep=",",
quote=FALSE, col.names=FALSE, row.names=FALSE)
READY<-read.table(paste("screendata",filename), sep=",", header=FALSE,
skip=0, row.names=WellsA1, fill=FALSE)
WellsA1<-row.names(READY)
attach(READY)
x<-unlist(READY[(plate.wells+1),startpoint:END])
clipx<- x[clippoint:(END-startpoint)]

grayscale<-colorRampPalette(c(gray(0), gray(1)))
normcolor<-grayscale(64)
hotset<-colorRampPalette(c("black","blue", "green", "yellow", "orange",
"red"))
nonnormcolor<-hotset(128)

beginwell<-(1)
endwell<-(plate.wells)
png(paste(screendate,"normal.png", sep=""), width=large.size,
height=large.size)
heatmap(as.matrix(READY), Colv = NA, Rowv=NA, scale="row",main =
```

```
paste(screendate,"norm","      time ->"),labRow = NULL, labCol = NULL,
xlab = NULL , ylab = NULL, col = normcolor, margin=c(1,0))
dev.off()
png(paste(screendate,"raw.png", sep=""), width=large.size,
height=large.size)
heatmap(as.matrix(READY), Colv = NA, Rowv=NA, scale="none",main =
paste(screendate,"raw","      time ->"),labRow = NULL, labCol = NULL,
xlab = NULL , ylab = NULL, col = nonnormcolor, margin=c(1,0))
dev.off()

pdf(paste("RAW", screendate,".pdf"), width=11, height=8.5)
RAW<-heatmap(as.matrix(READY[1:plate.wells,1:END]), Colv = NA, Rowv=NA,
scale="none",main =
paste("RAW", screendate,"time ->"),labRow = NULL, labCol = NULL, xlab =
NULL , ylab =
NULL, col = nonnormcolor, margin=c(3,3))
dev.off()

pdf(paste("RAW1", screendate,".pdf"), width=11, height=8.5)
RAW<-heatmap(as.matrix(READY[1:96,1:END]), Colv = NA, Rowv=NA,
scale="none",main =
paste("RAW1", screendate,"time ->"),labRow = NULL, labCol = NULL, xlab
= NULL , ylab =
NULL, col = nonnormcolor, margin=c(3,3))
dev.off()
pdf(paste("RAW2", screendate,".pdf"), width=11, height=8.5)
RAW<-heatmap(as.matrix(READY[97:192,1:END]), Colv = NA, Rowv=NA,
scale="none",main =
paste("RAW2", screendate,"time ->"),labRow = NULL, labCol = NULL, xlab
= NULL , ylab =
NULL, col = nonnormcolor, margin=c(3,3))
dev.off()
pdf(paste("RAW3", screendate,".pdf"), width=11, height=8.5)
RAW<-heatmap(as.matrix(READY[193:288,1:END]), Colv = NA, Rowv=NA,
scale="none",main =
paste("RAW3", screendate,"time ->"),labRow = NULL, labCol = NULL, xlab
= NULL , ylab =
NULL, col = nonnormcolor, margin=c(3,3))
dev.off()
pdf(paste("RAW4", screendate,".pdf"), width=11, height=8.5)
RAW<-heatmap(as.matrix(READY[289:384,1:END]), Colv = NA, Rowv=NA,
scale="none",main =
paste("RAW4", screendate,"time ->"),labRow = NULL, labCol = NULL, xlab
= NULL , ylab =
NULL, col = nonnormcolor, margin=c(3,3))
dev.off()

pdf(paste("RAWclust", screendate,".pdf"), width=11, height=8.5)
RAW<-heatmap(as.matrix(READY[1:384,1:END]), Colv = NA, Rowv=TRUE,
scale="none",main =
paste("RAW", screendate,"time ->"),labRow = NULL, labCol = NULL,
cexRow=0.2, xlab = "timepoints" , ylab =
NULL, col = nonnormcolor, margin=c(3,3))
dev.off()

pdf(paste("NORMAL", screendate,".pdf"), width=11, height=8.5)
NORM<-heatmap(as.matrix(READY), Colv = NA, Rowv=NA, scale="row",main =
paste("time ->"),labRow = NULL, labCol = NULL, xlab = NULL , ylab =
NULL, col = normcolor, margin=c(2,2))
dev.off()

pdf(paste("NORM1", screendate,".pdf"), width=11, height=8.5)
RAW<-heatmap(as.matrix(READY[1:96,1:END]), Colv = NA, Rowv=NA,
scale="row",main =
```

```
paste("RAW1", screendate,"time ->"),labRow = NULL, labCol = NULL, xlab
= NULL , ylab =
NULL, col = normcolor, margin=c(3,3))
dev.off()
pdf(paste("NORM2", screendate,".pdf"), width=11, height=8.5)
RAW<-heatmap(as.matrix(READY[97:192,1:END]), Colv = NA, Rowv=NA,
scale="row",main =
paste("RAW2", screendate,"time ->"),labRow = NULL, labCol = NULL, xlab
= NULL , ylab =
NULL, col = normcolor, margin=c(3,3))
dev.off()
pdf(paste("NORM3", screendate,".pdf"), width=11, height=8.5)
RAW<-heatmap(as.matrix(READY[193:288,1:END]), Colv = NA, Rowv=NA,
scale="row",main =
paste("RAW3", screendate,"time ->"),labRow = NULL, labCol = NULL, xlab
= NULL , ylab =
NULL, col = normcolor, margin=c(3,3))
dev.off()
pdf(paste("NORM4", screendate,".pdf"), width=11, height=8.5)
RAW<-heatmap(as.matrix(READY[289:384,1:END]), Colv = NA, Rowv=NA,
scale="row",main =
paste("RAW4", screendate,"time ->"),labRow = NULL, labCol = NULL, xlab
= NULL , ylab =
NULL, col = normcolor, margin=c(3,3))
dev.off()

png(paste("RAW1", screendate,".png"), width=large.size,
height=large.size)
RAW<-heatmap(as.matrix(READY[1:96,1:END]), Colv = NA, Rowv=NA,
scale="none",main =
paste("RAW1", screendate,"time ->"),labRow = NULL, labCol = NULL, xlab
= NULL , ylab =
NULL, col = nonnormcolor, margin=c(3,3))
dev.off()
png(paste("RAW2", screendate,".png"), width=large.size,
height=large.size)
RAW<-heatmap(as.matrix(READY[97:192,1:END]), Colv = NA, Rowv=NA,
scale="none",main =
paste("RAW2", screendate,"time ->"),labRow = NULL, labCol = NULL, xlab
= NULL , ylab =
NULL, col = nonnormcolor, margin=c(3,3))
dev.off()
png(paste("RAW3", screendate,".png"), width=large.size,
height=large.size)
RAW<-heatmap(as.matrix(READY[193:288,1:END]), Colv = NA, Rowv=NA,
scale="none",main =
paste("RAW3", screendate,"time ->"),labRow = NULL, labCol = NULL, xlab
= NULL , ylab =
NULL, col = nonnormcolor, margin=c(3,3))
dev.off()
png(paste("RAW4", screendate,".png"), width=large.size,
height=large.size)
RAW<-heatmap(as.matrix(READY[289:384,1:END]), Colv = NA, Rowv=NA,
scale="none",main =
paste("RAW4", screendate,"time ->"),labRow = NULL, labCol = NULL, xlab
= NULL , ylab =
NULL, col = nonnormcolor, margin=c(3,3))
dev.off()

png(paste("RAWclust", screendate,".png"), width=large.size,
height=large.size)
RAW<-heatmap(as.matrix(READY[1:384,1:END]), Colv = NA, Rowv=TRUE,
scale="none",main =
paste("RAW", screendate,"time ->"),labRow = NULL, labCol = NULL,
```

```
cexRow=0.2, xlab = "timepoints" , ylab =
NULL, col = nonnormcolor, margin=c(3,3))
dev.off()

png(paste("NORMAL", screendate,".png"), width=large.size,
height=large.size)
NORM<-heatmap(as.matrix(READY), Colv = NA, Rowv=NA, scale="row",main =
paste("time ->"),labRow = NULL, labCol = NULL, xlab = NULL , ylab =
NULL, col = normcolor, margin=c(2,2))
dev.off()

png(paste("NORM1", screendate,".png"), width=large.size,
height=large.size)
RAW<-heatmap(as.matrix(READY[1:96,1:END]), Colv = NA, Rowv=NA,
scale="row",main =
paste("NORM1", screendate,"time ->"),labRow = NULL, labCol = NULL, xlab
= NULL , ylab =
NULL, col = normcolor, margin=c(3,3))
dev.off()
png(paste("NORM2", screendate,".png"), width=large.size,
height=large.size)
RAW<-heatmap(as.matrix(READY[97:192,1:END]), Colv = NA, Rowv=NA,
scale="row",main =
paste("NORM2", screendate,"time ->"),labRow = NULL, labCol = NULL, xlab
= NULL , ylab =
NULL, col = normcolor, margin=c(3,3))
dev.off()
png(paste("NORM3", screendate,".png"), width=large.size,
height=large.size)
RAW<-heatmap(as.matrix(READY[193:288,1:END]), Colv = NA, Rowv=NA,
scale="row",main =
paste("NORM3", screendate,"time ->"),labRow = NULL, labCol = NULL, xlab
= NULL , ylab =
NULL, col = normcolor, margin=c(3,3))
dev.off()
png(paste("NORM4", screendate,".png"), width=large.size,
height=large.size)
RAW<-heatmap(as.matrix(READY[289:384,1:END]), Colv = NA, Rowv=NA,
scale="row",main =
paste("NORM4", screendate,"time ->"),labRow = NULL, labCol = NULL, xlab
= NULL , ylab =
NULL, col = normcolor, margin=c(3,3))
dev.off()

pdf(paste("ALT", screendate,".pdf"), width=11, height=8.5)
heatmap(as.matrix(READY), Colv = NA, Rowv=NA, scale="none",main =
paste("time ->"),labRow = NULL, labCol = NULL, xlab = NULL , ylab =
NULL, col = nonnormcolor, margin=c(2,2))
dev.off()

############################
#nonlinear least squares curvefitting
lincosdampdata<-matrix(nrow=plate.wells,ncol=10)
colnames(lincosdampdata)<-
c("well","baseline","amplitude","dampingrate","phase","period","phase.a
ngle","sumsquares","slope", "Relamp")
ss.residuals<-numeric()
for (i in 1:plate.wells)
{png(paste(screendate,WellsA1[i],".png"), width=image.size,
height=image.size)
roe<-unlist(READY[i,startpoint:END])
cliproe<-unlist(roe[clippoint:(END-startpoint)])
trend.est<-((median(roe[startpoint:halfpoint])-
median(roe[halfpoint:(END-startpoint)]))/(x[halfpoint]-x[END-
```

```
startpoint]))
phase.est<-clipx[which.max(cliproe)]
baseline.est<-median(cliproe)
amp.est<-((max(cliproe)-min(cliproe))/2)
plot(roe~x, xlab="Time (h)", ylab="Luminescence", pch = 1, cex=2)
lincosdamp<- try(nls(roe~ baseline + amplitude*(exp(-
damp*x))*cos(2*pi*(x-phase)/period) + trend*x,
start=list(baseline=baseline.est, amplitude=amp.est, damp= damp.est,
phase=phase.est, period=period.est, trend=trend.est),
trace=TRUE, algorithm="port"),silent=TRUE
)
lines(x, fitted.values(lincosdamp), lwd=3)
title(WellsA1[i])
try(ss.residuals<-sum((residuals(lincosdamp))^2))
try(baseline<-summary(lincosdamp)$coef[1])
try(amplitude<-(summary(lincosdamp)$coef[2]))
try(dampingrate<-summary(lincosdamp)$coef[3])
try(phase<-summary(lincosdamp)$coef[4])
try(period<-(summary(lincosdamp)$coef[5]))
try(phase.angle<-(-2*pi*((period-phase)/period-trunc((period-
phase)/period))))
try(phase<-format(phase, digits=3))
try(period<-format(period, digits=4))
try(text(55, max(roe), paste("period=",period,"h"), cex=0.8))
try(trend<-(summary(lincosdamp)$coef[6]))
thewell<-WellsA1[i]
Relamp<-NA
try(lincosdampdata[i,]<-
cbind(thewell,baseline,amplitude,dampingrate,phase,period,phase.angle,s
s.residuals,trend,Relamp))
dev.off()
}

write.table(lincosdampdata,paste("Parameters",screendate,".txt",
sep=""), sep=",", quote=FALSE, row.names=FALSE)

detach(READY)

save.image(paste(getwd(),"/run.RData", sep=""))

###STOP HERE
```

## CellulaRhythm Code for VieLux System

```
################################################
# CellulaRhythm 4.V384     for R 2.4.0
# read, fit, and plot circadian data
# 384 Viewlux format
# Warren G. Lewis         October,  2007
################################################
#linear detrended exponential damped cosine
#with linear parameter estimates
method<-"lincosdamp"

screendate<-"description"

#number of traces in a file—e.g. 384 or 1536
plate.wells<-384

large.size<-2000
databins<-100
damp.limit<-0.1
amp.limit<-10
squares.limit<-30000000
image.size<-2000

row.first<-103
row.last<-(row.first+plate.wells-1)
drop.points<-10
startpoint<-(drop.points+1)
clippoint<-(3+startpoint)

period.est<-24
damp.est<-0.005

#name of folder the data is in
setwd("D:/folder")

Wells<-as.matrix(read.table(paste("./well",plate.wells,".txt",
sep="")))

myfilelist<-list.files(pattern="G")
myfileinfo<-file.info(myfilelist)
myfilenames<-row.names(myfileinfo)

##Determines number of files in folder
##the only text files should be your data
fileno<-length(myfilenames)
filenamesplit<-unlist(strsplit(myfilenames[1], " "))
plateID<-vector()
for (i in 1:fileno){
filenamesplit<-unlist(strsplit(myfilenames[i], " "))
plateID<-c(plateID, filenamesplit[1])
}

platelist<-unique(plateID)
plateno<-length(platelist)
alltimepoints<-fileno/plateno
timepoints<-trunc(alltimepoints)
halfpoint<-trunc(timepoints/2)

for (i in 1:plateno){
dataPLATE<-data.frame(nrow=5000)
platefilelist<-list.files(pattern=platelist[i])
platefileinfo<-file.info(platefilelist)
myplatenames<-row.names(platefileinfo)
```

```
platetimepoints<-length(myplatenames)
for (j in 1:timepoints){
dataPLATE <-cbind(dataPLATE, read.delim(myplatenames[j], header=FALSE))
}
dataPLATE<-dataPLATE[,-1]
platedatafile<- paste(platelist[i],"raw.txt", sep="")
write.table (dataPLATE[row.first:row.last,1:timepoints],platedatafile)
datatrim<-read.table(platedatafile)
row.names(datatrim)<-paste(platelist[i],Wells,sep="_")
platedatafile<- paste(platelist[i],"data.txt", sep="")
write.table (datatrim,platedatafile)
}

datafilelist<-list.files(pattern="data.txt")
datafileinfo<-file.info(datafilelist)
datafilenames<-row.names(datafileinfo)
datano<-length(datafilenames)

screendata<-data.frame()
for (i in 1:datano){
activeplate<-datafilenames[i]
screendata<-rbind(screendata,read.table(activeplate))
}

filetimes<-numeric()
for (i in 1:timepoints){rel.read.time<-difftime(myfileinfo$mtime[i],
myfileinfo$mtime[1], units="hours")
filetimes<-c(filetimes,rel.read.time)}

screendata<-rbind(screendata,filetimes)
write.table(screendata,("screendata.txt"), sep=",", quote=FALSE,
col.names=FALSE, row.names=TRUE)

screen.wells<-(datano*plate.wells)
plate.well<-row.names(screendata)
attach(screendata)
x<-unlist(screendata[(screen.wells+1),startpoint:timepoints])
clipx<- x[clippoint:(timepoints-startpoint)]

grayscale<-colorRampPalette(c(gray(0), gray(1)))
normcolor<-grayscale(64)
hotset<-colorRampPalette(c("black","blue", "green", "yellow", "orange",
"red"))
nonnormcolor<-hotset(32)

for (i in 1:plateno){
beginwell<-((plate.wells*(i-1))+1)
endwell<-(plate.wells*i)
png(paste(platelist[i],"normal.png"), width=large.size,
height=large.size)
heatmap(as.matrix(screendata[beginwell:endwell,startpoint:timepoints]),
Colv = NA, Rowv=NA, scale="row",main = paste(platelist[i],"norm","
time ->"),labRow = NULL, labCol = NULL, xlab = NULL , ylab = NULL, col
= normcolor, margin=c(1,0))
dev.off()
png(paste(platelist[i],"raw.png"), width=large.size, height=large.size)
heatmap(as.matrix(screendata[beginwell:endwell,startpoint:timepoints]),
Colv = NA, Rowv=NA, scale="none",main = paste(platelist[i],"raw","
time ->"),labRow = NULL, labCol = NULL, xlab = NULL , ylab = NULL, col
= nonnormcolor, margin=c(1,0))
dev.off()
}

######################
```

```
lincosdampdata<-matrix(nrow=screen.wells,ncol=11)
colnames(lincosdampdata)<-
c("plate_well","plate","well","baseline","amplitude","dampingrate","pha
se","period","phase.angle","sumsquares","slope")
ss.residuals<-numeric()
for (i in 1:screen.wells)
{roe<-unlist(screendata[i,startpoint:timepoints])
cliproe<-unlist(roe[clippoint:(timepoints-startpoint)])
trend.est<-((median(roe[startpoint:halfpoint])-
median(roe[halfpoint:(timepoints-startpoint)]))/(x[halfpoint]-
x[timepoints-startpoint]))
phase.est<-clipx[which.max(cliproe)]
baseline.est<-median(cliproe)
amp.est<-((max(cliproe)-min(cliproe))/2)
lincosdamp<- try(nls(roe~ baseline + amplitude*(exp(-
damp*x))*cos(2*pi*(x-phase)/period) + trend*x,
start=list(baseline=baseline.est, amplitude=amp.est, damp= damp.est,
phase=phase.est, period=period.est, trend=trend.est),
trace=TRUE, algorithm="port"),silent=TRUE
)
try(ss.residuals<-sum((residuals(lincosdamp))^2))
try(baseline<-summary(lincosdamp)$coef[1])
try(amplitude<-(summary(lincosdamp)$coef[2]))
try(dampingrate<-summary(lincosdamp)$coef[3])
try(phase<-summary(lincosdamp)$coef[4])
try(period<-(summary(lincosdamp)$coef[5]))
try(phase.angle<-(-2*pi*((period-phase)/period-trunc((period-
phase)/period))))
try(phase<-format(phase, digits=3))
try(period<-format(period, digits=4))
try(trend<-(summary(lincosdamp)$coef[6]))
thewell<-Wells[(i-(trunc(i/plate.wells)*plate.wells))]
theplate<-platelist[trunc((i-1)/plate.wells)+1]
try(lincosdampdata[i,]<-
cbind(plate.well[i],theplate,thewell,baseline,amplitude,dampingrate,pha
se,period,phase.angle,ss.residuals,trend))
}

write.table(lincosdampdata,paste(method,".txt"), sep=",", quote=FALSE,
row.names=FALSE)

detach(screendata)
```