latency\_on\_WIO\_20160627.R

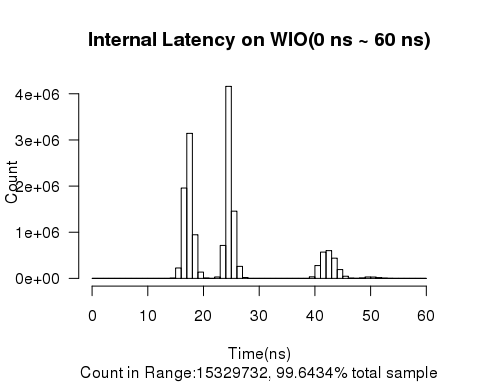
xing

Thu Jun 30 10:07:18 2016

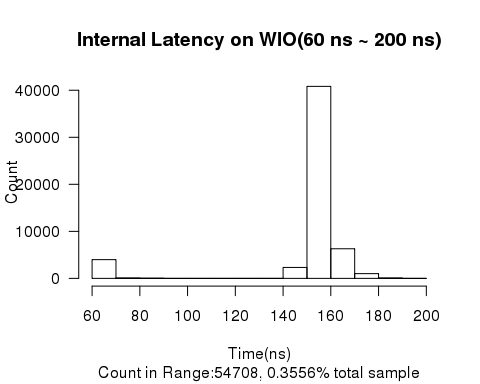
library("knitr")  
filenames = c("20160627\_LL\_WIO/Latency/20160627\_WIO\_SF\_64b.csv",  
 "20160627\_LL\_WIO/Latency/20160627\_WIO\_SF\_128b.csv",  
 "20160627\_LL\_WIO/Latency/20160627\_WIO\_SF\_256b.csv",  
 "20160627\_LL\_WIO/Latency/20160627\_WIO\_SF\_512b.csv"  
 )  
system.name = "WIO"  
sep = "##############################\n"  
for(filename in filenames){  
 data.wio = read.csv(filename, header = TRUE, colClasses=c("NULL", "integer", "integer"))  
 data.wio.length = length(data.wio$inner)  
 data.wio$inner[1] = as.integer(mean(data.wio$inner[2:data.wio.length]))  
 latency.wio.in = data.wio$inner  
 latency.wio.net = data.wio$net  
 latency.wio.net.total = sum(data.wio$net)/1000  
 latency.wio.in.total = sum(data.wio$inner)/1000  
 ranges = rbind(c(0,60,1),  
 c(60,200,10),  
 c(200,500,50),  
 c(500,4000,500)  
 )  
   
 hist.table = data.frame(start=integer(), end=integer(), count=integer())  
   
 for(c in 1:nrow(ranges)){  
 range.start = ranges[c,1]  
 range.end = ranges[c,2]  
 range.step = ranges[c,3]  
 b = seq(range.start, range.end, range.step)  
 range.target = (latency.wio.in > range.start & latency.wio.in <= range.end)  
 range.count = sum(range.target)  
 range.count.ratio = range.count/(data.wio.length-1)  
 range.latency = sum(latency.wio.in[range.target])  
 range.latency.ratio = range.latency/latency.wio.in.total/1000  
 text.capital = sprintf("Internal Latency on %s(%d ns ~ %d ns)", system.name, range.start, range.end)  
 text.comment = sprintf("Count in Range:%d, %.4f%% total sample", range.count, range.count.ratio\*100)  
   
 cat(sprintf(sep))  
 cat(sprintf("DATA:%s on %s\n", filename, system.name))  
 cat(sprintf("RANGE:%d ns - %d ns\n", range.start, range.end))  
 cat(sprintf("Sample Number:%d, %.4f%% of total %d samples\n", range.count, range.count.ratio\*100, data.wio.length))  
   
 h.data = hist(latency.wio.in[range.target], breaks = b,right = TRUE,   
 main = text.capital, sub = text.comment, xlab = "Time(ns)", ylab = "Count",   
 las = 1  
 )  
 hist.table = rbind(hist.table, cbind(h.data$breaks[1:length(h.data$breaks)-1],   
 h.data$breaks[1:length(h.data$breaks)-1]+range.step,  
 h.data$counts))  
 }  
 cat(sprintf(sep))  
 cat(sprintf("DATA:%s on %s\n", filename, system.name))  
 cat(sprintf("Sample number:%d\n", data.wio.length))  
 cat(sprintf("latency avg=%.1f ns, sd=%.1f ns\n", mean(latency.wio.in), sd(latency.wio.in)))  
 print(summary(latency.wio.in))  
 print(kable(hist.table, col.names = c("Start", "End", "Count")))  
}

## Warning in sum(data.wio$net): integer overflow - use sum(as.numeric(.))

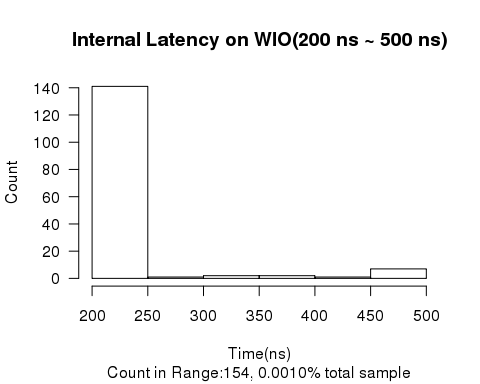
## ##############################  
## DATA:20160627\_LL\_WIO/Latency/20160627\_WIO\_SF\_64b.csv on WIO  
## RANGE:0 ns - 60 ns  
## Sample Number:15329732, 99.6434% of total 15384598 samples



## ##############################  
## DATA:20160627\_LL\_WIO/Latency/20160627\_WIO\_SF\_64b.csv on WIO  
## RANGE:60 ns - 200 ns  
## Sample Number:54708, 0.3556% of total 15384598 samples



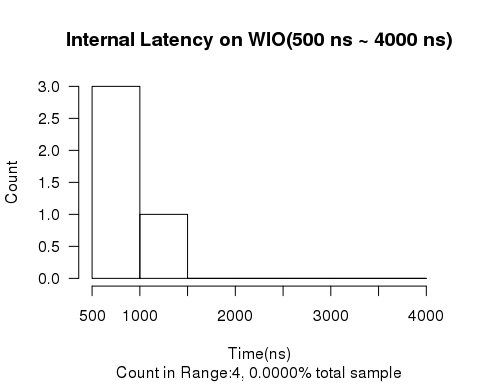
## ##############################  
## DATA:20160627\_LL\_WIO/Latency/20160627\_WIO\_SF\_64b.csv on WIO  
## RANGE:200 ns - 500 ns  
## Sample Number:154, 0.0010% of total 15384598 samples



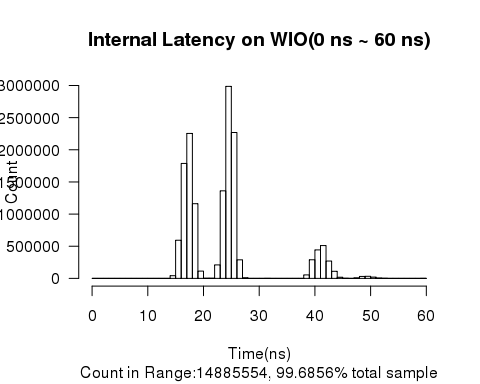
## ##############################  
## DATA:20160627\_LL\_WIO/Latency/20160627\_WIO\_SF\_64b.csv on WIO  
## RANGE:500 ns - 4000 ns  
## Sample Number:4, 0.0000% of total 15384598 samples

## ##############################  
## DATA:20160627\_LL\_WIO/Latency/20160627\_WIO\_SF\_64b.csv on WIO  
## Sample number:15384598  
## latency avg=25.2 ns, sd=11.4 ns  
## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 14.00 18.00 25.00 25.22 26.00 1419.00   
##   
##   
## Start End Count  
## ------ ----- --------  
## 0 1 0  
## 1 2 0  
## 2 3 0  
## 3 4 0  
## 4 5 0  
## 5 6 0  
## 6 7 0  
## 7 8 0  
## 8 9 0  
## 9 10 0  
## 10 11 0  
## 11 12 0  
## 12 13 0  
## 13 14 7  
## 14 15 5877  
## 15 16 225573  
## 16 17 1957976  
## 17 18 3143019  
## 18 19 945169  
## 19 20 135879  
## 20 21 7600  
## 21 22 679  
## 22 23 30350  
## 23 24 714753  
## 24 25 4159985  
## 25 26 1456602  
## 26 27 260799  
## 27 28 19198  
## 28 29 486  
## 29 30 24  
## 30 31 10  
## 31 32 3  
## 32 33 0  
## 33 34 0  
## 34 35 0  
## 35 36 0  
## 36 37 0  
## 37 38 3  
## 38 39 514  
## 39 40 34595  
## 40 41 277917  
## 41 42 569538  
## 42 43 600926  
## 43 44 438678  
## 44 45 190923  
## 45 46 47573  
## 46 47 6296  
## 47 48 1793  
## 48 49 10031  
## 49 50 30915  
## 50 51 29414  
## 51 52 15487  
## 52 53 5973  
## 53 54 1587  
## 54 55 1233  
## 55 56 596  
## 56 57 427  
## 57 58 337  
## 58 59 364  
## 59 60 623  
## 60 70 3990  
## 70 80 97  
## 80 90 49  
## 90 100 10  
## 100 110 6  
## 110 120 7  
## 120 130 1  
## 130 140 0  
## 140 150 2336  
## 150 160 40814  
## 160 170 6305  
## 170 180 1006  
## 180 190 86  
## 190 200 1  
## 200 250 141  
## 250 300 1  
## 300 350 2  
## 350 400 2  
## 400 450 1  
## 450 500 7  
## 500 1000 3  
## 1000 1500 1  
## 1500 2000 0  
## 2000 2500 0  
## 2500 3000 0  
## 3000 3500 0  
## 3500 4000 0

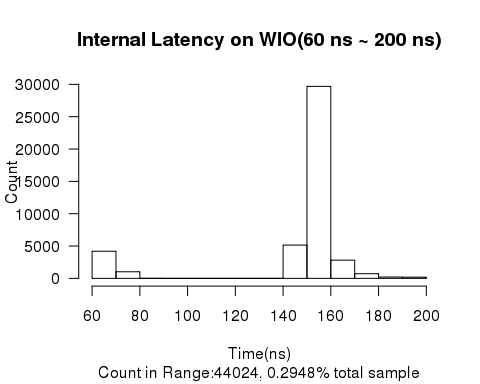
## Warning in sum(data.wio$net): integer overflow - use sum(as.numeric(.))



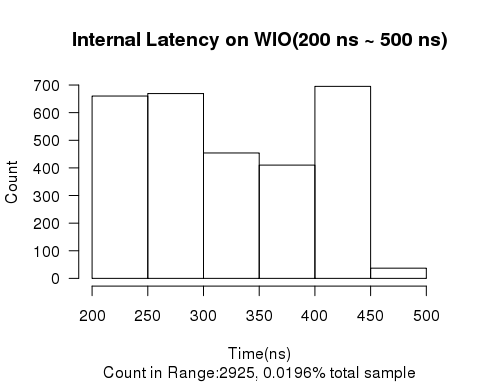
## ##############################  
## DATA:20160627\_LL\_WIO/Latency/20160627\_WIO\_SF\_128b.csv on WIO  
## RANGE:0 ns - 60 ns  
## Sample Number:14885554, 99.6856% of total 14932505 samples



## ##############################  
## DATA:20160627\_LL\_WIO/Latency/20160627\_WIO\_SF\_128b.csv on WIO  
## RANGE:60 ns - 200 ns  
## Sample Number:44024, 0.2948% of total 14932505 samples



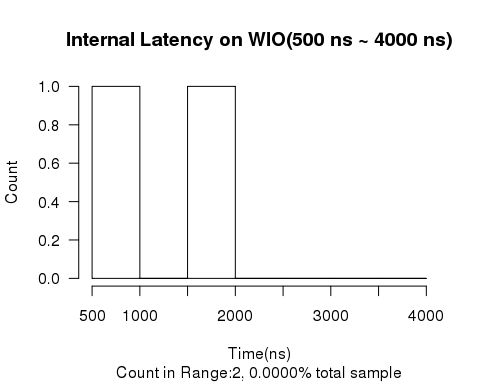
## ##############################  
## DATA:20160627\_LL\_WIO/Latency/20160627\_WIO\_SF\_128b.csv on WIO  
## RANGE:200 ns - 500 ns  
## Sample Number:2925, 0.0196% of total 14932505 samples



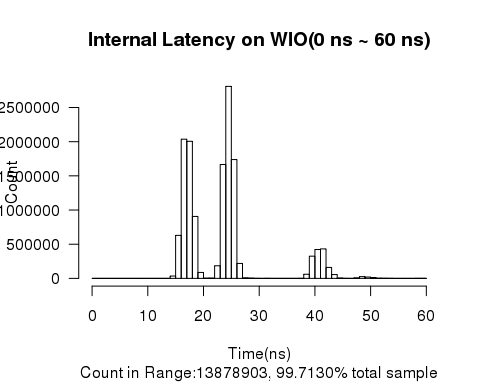
## ##############################  
## DATA:20160627\_LL\_WIO/Latency/20160627\_WIO\_SF\_128b.csv on WIO  
## RANGE:500 ns - 4000 ns  
## Sample Number:2, 0.0000% of total 14932505 samples

## ##############################  
## DATA:20160627\_LL\_WIO/Latency/20160627\_WIO\_SF\_128b.csv on WIO  
## Sample number:14932505  
## latency avg=24.6 ns, sd=11.0 ns  
## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 14.00 18.00 24.00 24.65 26.00 1711.00   
##   
##   
## Start End Count  
## ------ ----- --------  
## 0 1 0  
## 1 2 0  
## 2 3 0  
## 3 4 0  
## 4 5 0  
## 5 6 0  
## 6 7 0  
## 7 8 0  
## 8 9 0  
## 9 10 0  
## 10 11 0  
## 11 12 0  
## 12 13 0  
## 13 14 16  
## 14 15 41837  
## 15 16 593316  
## 16 17 1787778  
## 17 18 2254385  
## 18 19 1161550  
## 19 20 113297  
## 20 21 2620  
## 21 22 2776  
## 22 23 209985  
## 23 24 1361562  
## 24 25 2985682  
## 25 26 2268683  
## 26 27 288499  
## 27 28 8936  
## 28 29 143  
## 29 30 47  
## 30 31 476  
## 31 32 1284  
## 32 33 0  
## 33 34 2  
## 34 35 2  
## 35 36 1  
## 36 37 8  
## 37 38 368  
## 38 39 54228  
## 39 40 290236  
## 40 41 443558  
## 41 42 511050  
## 42 43 270506  
## 43 44 110450  
## 44 45 18219  
## 45 46 2089  
## 46 47 121  
## 47 48 7671  
## 48 49 30521  
## 49 50 32592  
## 50 51 19794  
## 51 52 5403  
## 52 53 2431  
## 53 54 878  
## 54 55 468  
## 55 56 178  
## 56 57 124  
## 57 58 126  
## 58 59 416  
## 59 60 1242  
## 60 70 4204  
## 70 80 1027  
## 80 90 10  
## 90 100 4  
## 100 110 2  
## 110 120 2  
## 120 130 0  
## 130 140 0  
## 140 150 5152  
## 150 160 29683  
## 160 170 2815  
## 170 180 730  
## 180 190 206  
## 190 200 189  
## 200 250 660  
## 250 300 669  
## 300 350 454  
## 350 400 410  
## 400 450 695  
## 450 500 37  
## 500 1000 1  
## 1000 1500 0  
## 1500 2000 1  
## 2000 2500 0  
## 2500 3000 0  
## 3000 3500 0  
## 3500 4000 0

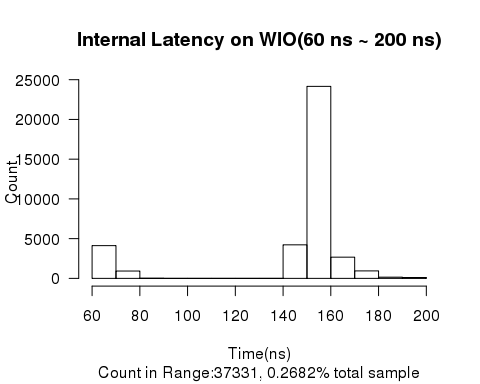
## Warning in sum(data.wio$net): integer overflow - use sum(as.numeric(.))



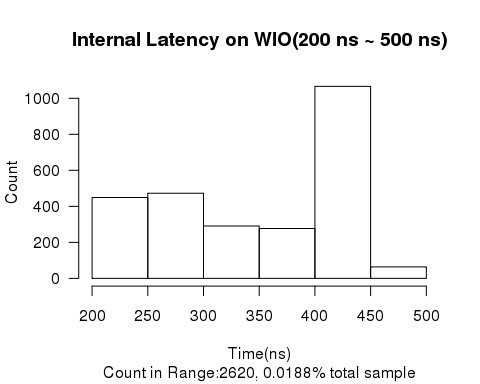
## ##############################  
## DATA:20160627\_LL\_WIO/Latency/20160627\_WIO\_SF\_256b.csv on WIO  
## RANGE:0 ns - 60 ns  
## Sample Number:13878903, 99.7130% of total 13918854 samples



## ##############################  
## DATA:20160627\_LL\_WIO/Latency/20160627\_WIO\_SF\_256b.csv on WIO  
## RANGE:60 ns - 200 ns  
## Sample Number:37331, 0.2682% of total 13918854 samples



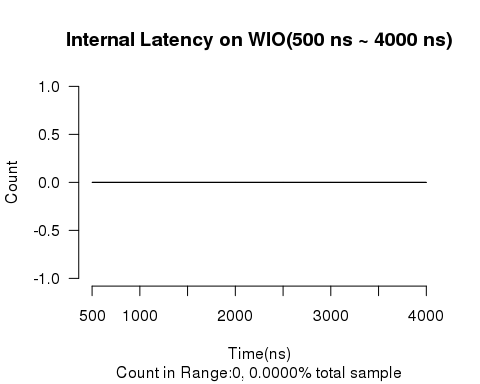
## ##############################  
## DATA:20160627\_LL\_WIO/Latency/20160627\_WIO\_SF\_256b.csv on WIO  
## RANGE:200 ns - 500 ns  
## Sample Number:2620, 0.0188% of total 13918854 samples



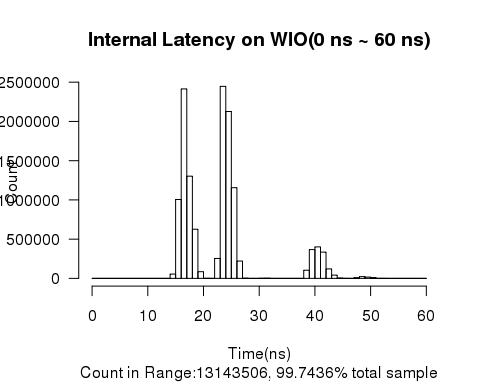
## ##############################  
## DATA:20160627\_LL\_WIO/Latency/20160627\_WIO\_SF\_256b.csv on WIO  
## RANGE:500 ns - 4000 ns  
## Sample Number:0, 0.0000% of total 13918854 samples

## ##############################  
## DATA:20160627\_LL\_WIO/Latency/20160627\_WIO\_SF\_256b.csv on WIO  
## Sample number:13918854  
## latency avg=24.2 ns, sd=10.7 ns  
## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 13.00 18.00 24.00 24.22 26.00 493.00   
##   
##   
## Start End Count  
## ------ ----- --------  
## 0 1 0  
## 1 2 0  
## 2 3 0  
## 3 4 0  
## 4 5 0  
## 5 6 0  
## 6 7 0  
## 7 8 0  
## 8 9 0  
## 9 10 0  
## 10 11 0  
## 11 12 0  
## 12 13 5  
## 13 14 102  
## 14 15 34379  
## 15 16 629172  
## 16 17 2036551  
## 17 18 2004967  
## 18 19 906024  
## 19 20 86807  
## 20 21 3753  
## 21 22 4205  
## 22 23 185122  
## 23 24 1666176  
## 24 25 2808495  
## 25 26 1737778  
## 26 27 218792  
## 27 28 7206  
## 28 29 3980  
## 29 30 1168  
## 30 31 814  
## 31 32 1503  
## 32 33 6  
## 33 34 3  
## 34 35 0  
## 35 36 4  
## 36 37 10  
## 37 38 1422  
## 38 39 61341  
## 39 40 324766  
## 40 41 422884  
## 41 42 431259  
## 42 43 160394  
## 43 44 56222  
## 44 45 5574  
## 45 46 1059  
## 46 47 570  
## 47 48 7447  
## 48 49 25694  
## 49 50 20143  
## 50 51 10753  
## 51 52 3139  
## 52 53 2425  
## 53 54 1463  
## 54 55 621  
## 55 56 559  
## 56 57 754  
## 57 58 902  
## 58 59 949  
## 59 60 1541  
## 60 70 4123  
## 70 80 930  
## 80 90 19  
## 90 100 7  
## 100 110 6  
## 110 120 3  
## 120 130 0  
## 130 140 0  
## 140 150 4221  
## 150 160 24156  
## 160 170 2667  
## 170 180 941  
## 180 190 152  
## 190 200 106  
## 200 250 449  
## 250 300 473  
## 300 350 291  
## 350 400 277  
## 400 450 1066  
## 450 500 64  
## 500 1000 0  
## 1000 1500 0  
## 1500 2000 0  
## 2000 2500 0  
## 2500 3000 0  
## 3000 3500 0  
## 3500 4000 0

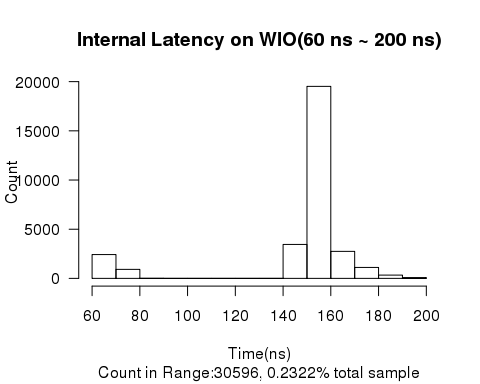
## Warning in sum(data.wio$net): integer overflow - use sum(as.numeric(.))



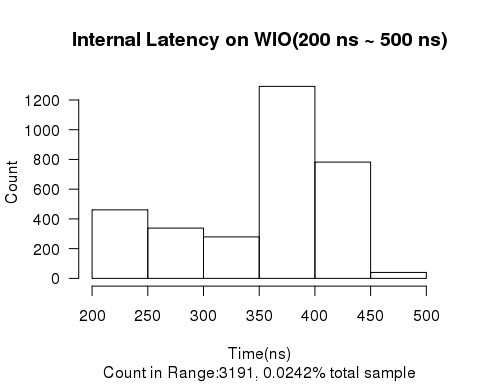
## ##############################  
## DATA:20160627\_LL\_WIO/Latency/20160627\_WIO\_SF\_512b.csv on WIO  
## RANGE:0 ns - 60 ns  
## Sample Number:13143506, 99.7436% of total 13177296 samples



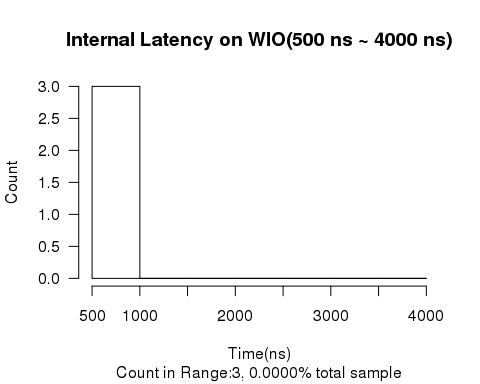
## ##############################  
## DATA:20160627\_LL\_WIO/Latency/20160627\_WIO\_SF\_512b.csv on WIO  
## RANGE:60 ns - 200 ns  
## Sample Number:30596, 0.2322% of total 13177296 samples



## ##############################  
## DATA:20160627\_LL\_WIO/Latency/20160627\_WIO\_SF\_512b.csv on WIO  
## RANGE:200 ns - 500 ns  
## Sample Number:3191, 0.0242% of total 13177296 samples



## ##############################  
## DATA:20160627\_LL\_WIO/Latency/20160627\_WIO\_SF\_512b.csv on WIO  
## RANGE:500 ns - 4000 ns  
## Sample Number:3, 0.0000% of total 13177296 samples



## ##############################  
## DATA:20160627\_LL\_WIO/Latency/20160627\_WIO\_SF\_512b.csv on WIO  
## Sample number:13177296  
## latency avg=23.9 ns, sd=10.8 ns  
## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 14.00 17.00 24.00 23.86 25.00 618.00   
##   
##   
## Start End Count  
## ------ ----- --------  
## 0 1 0  
## 1 2 0  
## 2 3 0  
## 3 4 0  
## 4 5 0  
## 5 6 0  
## 6 7 0  
## 7 8 0  
## 8 9 0  
## 9 10 0  
## 10 11 0  
## 11 12 0  
## 12 13 0  
## 13 14 9  
## 14 15 55078  
## 15 16 1005439  
## 16 17 2414267  
## 17 18 1302459  
## 18 19 626202  
## 19 20 85310  
## 20 21 1066  
## 21 22 789  
## 22 23 254532  
## 23 24 2446273  
## 24 25 2126626  
## 25 26 1155239  
## 26 27 221281  
## 27 28 3341  
## 28 29 524  
## 29 30 805  
## 30 31 1704  
## 31 32 2611  
## 32 33 30  
## 33 34 5  
## 34 35 1  
## 35 36 1  
## 36 37 1  
## 37 38 725  
## 38 39 104045  
## 39 40 367143  
## 40 41 401528  
## 41 42 335370  
## 42 43 120970  
## 43 44 41751  
## 44 45 4582  
## 45 46 1425  
## 46 47 786  
## 47 48 8386  
## 48 49 22165  
## 49 50 16855  
## 50 51 8606  
## 51 52 1560  
## 52 53 915  
## 53 54 510  
## 54 55 309  
## 55 56 261  
## 56 57 647  
## 57 58 631  
## 58 59 399  
## 59 60 344  
## 60 70 2423  
## 70 80 919  
## 80 90 11  
## 90 100 4  
## 100 110 4  
## 110 120 3  
## 120 130 0  
## 130 140 0  
## 140 150 3450  
## 150 160 19517  
## 160 170 2747  
## 170 180 1112  
## 180 190 334  
## 190 200 72  
## 200 250 461  
## 250 300 338  
## 300 350 279  
## 350 400 1291  
## 400 450 782  
## 450 500 40  
## 500 1000 3  
## 1000 1500 0  
## 1500 2000 0  
## 2000 2500 0  
## 2500 3000 0  
## 3000 3500 0  
## 3500 4000 0