Figure 7B upper panel, Figure 8B upper panel, Figure S7, and Figure S8

Toshihiro Arae

General directory setting

```r
wd <- here::here()
shared <- fs::path(fs::path_dir(wd), "shared")
```

Loading packages

```r
library(magrittr)
library(ggplot2)
```

Load common R scripts

```r
#source(fs::path(wd, "script_r", "MISC.R"))
#source(fs::path(here::here(), "script_r", "MISC_PALETTE.R"))
```

Load script

```r
source(fs::path(wd, "script_r", "MISC_FIG.R"))
readLines(fs::path(wd, "script_r", "MISC_FIG.R")) %>% cat(sep = "\n")
```

```
library(magrittr)
library(ggplot2)

COL_PALETTE <-
  viridis::inferno(6, begin = .1, end = .9) %>%
  rev() %>%
  setNames(nm = c("ZT0", "ZT3", "ZT6", "ZT12", "ZT18", "ZT21"))

LABEL_PALETTE <-
  COL_PALETTE %>%
  prismatic::clr_darken(shift = .15) %>%
  setNames(names(COL_PALETTE))

label_number_si <-
  purrr::partial(scales::label_number, scale_cut = scales::cut_short_scale())

ggsave_single <- function(..., width = 86, height = 230, dpi = 300) {
  f <- purrr::partial(ggsave, width = width, height = height, dpi = dpi, units = "mm")
  f(...)
}

ggsave_double <- function(..., width = 178, height = 230, dpi = 300) {
  f <- purrr::partial(ggsave, width = width, height = height, dpi = dpi, units = "mm")
  f(...)
}

#' Utility functions for making secondary y-axis
#' @param y1 numeric vector
#' @param y2 numeric vector
#' @name util_2nd_axis
#' @examples
#' make_scale_y1_to_y2(1:5, 6:10)(1:10)
#' make_scale_y2_to_y1(1:5, 6:10)(1:10)
#'
#' iris_ <- dplyr::select(iris, x = Sepal.Length, y1 = Petal.Length, y2 = Petal.Width)
#' gp1 <-
#'   iris_ %>%
#'   ggplot() +
#'   geom_point(aes(x, y1), color = "#CD3700") +
#'   geom_point(aes(x, y2), color = "#473C8B")
#'
#' to_y1 <- with(iris_, {make_scale_y2_to_y1(y1, y2)})
#' to_y2 <- with(iris_, {make_scale_y1_to_y2(y1, y2)})
#' gp2 <-
#'   iris_ %>%
#'   ggplot() +
#'   geom_point(aes(x, y1), color = "#CD3700") +
#'   geom_point(aes(x, y = to_y1(y2)), color = "#473C8B") +
#'   scale_y_continuous(sec.axis = sec_axis(trans = to_y2, name = "y2"))
#' patchwork::wrap_plots(gp1, gp2)
#'
NULL

#' Create transformation function of range(y1) to range(y2)
#' @rdname util_2nd_axis
#' @export
#'
make_scale_y1_to_y2 <- function(y1, y2) {
  function(n) {
    scales:::rescale.numeric(
      n,
      to = range(y2, na.rm = TRUE, finite = TRUE),
      from = range(y1, na.rm = TRUE, finite = TRUE)
    )
  }
}

#' Create transformation function of range(y2) to range(y1)
#' @rdname util_2nd_axis
#' @export
#'
make_scale_y2_to_y1 <- function(y1, y2) {
  function(n) {
    scales:::rescale.numeric(
      n,
      to = range(y1, na.rm = TRUE, finite = TRUE),
      from = range(y2, na.rm = TRUE, finite = TRUE)
    )
  }
}

#' Create transformation function of range(y2) to range(y1)
#' @rdname util_2nd_axis
#' @export
#'
make_scale_y2_to_y1_se <- function(y1, y2) {
  to <- range(y1, na.rm = TRUE, finite = TRUE)
  from <- range(y2, na.rm = TRUE, finite = TRUE)
  function(n) n / (diff(from) / diff(to))
}
```

Directory setting

```
path_out_fig7Bup <- function(...) fs::path(wd, "analysis", "fig", "fig07Bup", ...)
path_out_fig8Bup <- function(...) fs::path(wd, "analysis", "fig", "fig08Bup", ...)
path_out_figS7 <- function(...) fs::path(wd, "analysis", "fig", "figS07", ...)
path_out_figS8up <- function(...) fs::path(wd, "analysis", "fig", "figS08up", ...)
fs::dir_create(path_out_fig7Bup())
fs::dir_create(path_out_fig8Bup())
fs::dir_create(path_out_figS7())
fs::dir_create(path_out_figS8up())
```

Define some functions

Functions to create tibbles

```
# function to change pattern scaling
# choose either one
rel_to_ave <- function(x) x / mean(x)
rel_to_max <- function(x) x / max(x)

#' Extract normalized count of gene of interest and return a tibble for plot
#' @param GOI a character vector. AGI code of gene of interest
make_tbl_plot <- function(GOI) {
  tbl_nc %>%
    dplyr::filter(AGI == GOI) %>%
    tidyr::pivot_longer(!AGI) %>%
    dplyr::mutate(
      ZT =
        stringr::str_extract(name, "zt\\d+") %>%
        stringr::str_to_upper() %>%
        forcats::fct_relevel(names(COL_PALETTE)),
      type =
        stringr::str_extract(name, "rna|morf|uorf") %>%
        {c(rna = "RNA", morf = "mORF", uorf = "uORF")[.]} %>%
        forcats::fct_relevel(c("RNA", "mORF", "uORF"))
    ) %>%
    dplyr::with_groups(c(type), dplyr::mutate, rel = rel_to_ave(value))
}

#' Extract effect size of gene of interest and return a tibble for plot
#' @param GOI a character vector. AGI code of gene of interest
make_tbl_plot_coef <- function(tbl, GOI) {
  if(nrow(tbl) == 0) {
    return(
      tibble::tibble(
        AGI = GOI,
        ZT = paste0("ZT", c(0,3,6,12,18,21)) %>%
          forcats::fct_relevel(names(COL_PALETTE)),
        coef = NA, se = NA
      )
    )
  }
  tbl %>%
    dplyr::select(AGI, coef_zt3:coef_zt21, se_zt3:se_zt21) %>%
    dplyr::mutate(coef_zt0 = 0, .before = coef_zt3) %>%
    dplyr::mutate(se_zt0 = 0, .before = se_zt3) %>%
    tidyr::pivot_longer(!AGI) %>%
    dplyr::mutate(
      ZT =
        stringr::str_extract(name, "zt\\d+") %>%
        stringr::str_to_upper() %>%
        forcats::fct_relevel(names(COL_PALETTE)),
      type = stringr::str_extract(name, "coef|se")
    ) %>%
    dplyr::select(!name) %>%
    tidyr::pivot_wider(names_from = type, values_from = value)
}
```

Functions to plot

```
# Aesthetic parameters
SIZE_POINT <- 2.5
WIDTH_DODGE <- 2/3
PAL_LINECOLOR <-
  c("RNA" = "red", "mORF" = "blue", "uORF" = "green") %>%
  scales::muted(l = 50, c = 80) %>%
  c(., "TE mORF" = "black", "TE uORF" = "black")
SHAPE_MANUAL <-
  c("RNA" = 16, "mORF" = 17, "uORF" = 15, "TE mORF" = 15, "TE uORF" = 15)
LINETYPE_MANUAL <-
  c("RNA" = "solid", "mORF" = "solid", "uORF" = "solid",
    "TE mORF" = "solid", "TE uORF" = "dashed")

#' Function to label numbers in the specific format
label_num <- function(x) {
  signs::signs(x, accuracy = .1, add_plusses = TRUE, label_at_zero = "symbol") %>%
    chartr("+±−", "  -", .)
}

#' Partialized `ggsave()`
ggsave_ <- purrr::partial(ggsave, width = 140, height = 80, dpi = 300, units = "mm")

#' Add some plot layers of normalized count
#' @param tbl
#' @param x
plot_nc <- function(tbl, x) {
  force(tbl)

  list(
    geom_line(
      data =
        tbl %>%
        dplyr::group_by(type, {{x}}, facet) %>%
        dplyr::summarise(ave = mean(rel), .groups = "drop") %>%
        dplyr::ungroup(),
      mapping = aes(x = {{x}}, y = ave, color = type)
    ),
    geom_point(
      data = tbl,
      mapping = aes(
        x = {{x}}, y = rel,
        color = stage(type, after_scale = alpha(color, .5)),
        shape = type),
      size = SIZE_POINT
    ),
    scale_color_manual(values = PAL_LINECOLOR)
  )
}

plot_te <- function(tbl, x, scale_to_x, scale_to_y) {
  force(tbl)
  force(scale_to_x)
  force(scale_to_y)

  list(
    geom_line(
      data = tbl,
      aes(x = {{x}}, y = scale_to_x(coef),
          group = type, color = type, linetype = type)
    ),
    geom_point(
      data = tbl,
      aes(x = {{x}}, y = scale_to_x(coef),
          color = type,
          shape = type),
      size = SIZE_POINT
    ),
    geom_linerange(
      data = dplyr::filter(tbl, type %in% c("TE mORF", "TE uORF")),
      aes(x = {{x}}, y = scale_to_x(coef),
          ymin = scale_to_x(coef - se),
          ymax = scale_to_x(coef + se),
          group = type, color = type),
      linetype = "solid",
      position = position_dodge(width = WIDTH_DODGE),
      linewidth = 1 / .pt,
      show.legend = FALSE
    ),
    scale_y_continuous(
      expand = expansion(mult = c(0, .1)),
      labels = label_num,
      sec.axis = sec_axis(
        name = "Log<sub>2</sub>(FC) (vs. TE at ZT0)",
        trans = scale_to_y,
        labels = label_num
      )
    )
  )
}
```

Plot themes

```
theme_line <- function(base_size = 10, base_line_size = 1/22) {
  list(
    theme_linedraw(
      base_size = base_size,
      base_line_size = base_line_size
    ),
    labs(x = "ZT", y = "Normalized count (relative to average)"),
    theme(
      axis.text.x = element_text(face = "bold"),
      axis.title.y.left = ggtext::element_markdown(),
      axis.title.y.right = ggtext::element_markdown(),
      axis.text.y.left = element_text(hjust = 1),
      axis.text.y.right = element_text(hjust = 1),
      panel.border = element_blank(),
      axis.line = element_line(color = "black", linewidth = .5),
      panel.grid = element_blank(),
      axis.ticks = element_line(linewidth = .5),
      legend.position = "top"
    ),
    guides(
      alpha = guide_legend("", order = 1),
      linetype = guide_legend("", order = 2),
      fill = guide_none()
    )
  )
}

theme_bar <- function(base_size = 10, base_line_size = 1/22) {
  list(
    theme_linedraw(
      base_size = base_size,
      base_line_size = base_line_size
    ),
    labs(x = "ZT", y = "Relative normalized count\nto average"),
    theme(
      axis.text.x = element_text(face = "bold"),
      axis.title.y.left = ggtext::element_markdown(),
      axis.title.y.right = ggtext::element_markdown(),
      axis.text.y.left = element_text(hjust = 1),
      axis.text.y.right = element_text(hjust = 1),
      panel.border = element_blank(),
      axis.line = element_line(color = "black", linewidth = .5),
      axis.line.x.top = element_blank(),
      panel.grid = element_blank(),
      axis.ticks = element_line(linewidth = .5),
      legend.position = "top"
    ),
    guides(
      alpha = guide_legend("", order = 1),
      linetype = guide_legend("", order = 2),
      fill = guide_none()
    )
  )
}
```

Load input data

```
# Load normalized counts
dir_readcount <- fs::path("analysis", "list_summary")

tbl_nc <-
  fs::dir_ls(fs::path(wd, dir_readcount), regexp = "_nc.csv") %>%
  purrr::map(readr::read_csv) %>%
  purrr::reduce(dplyr::left_join, by = "AGI")

# Load DESeq2 results
tbl_out <- readr::read_csv(fs::path(wd, "analysis", "deseq2_te", "deg_all.csv"))
tbl_out_uorf <- readr::read_csv(fs::path(wd, "analysis", "deseq2_te_uorf", "deg_all.csv"))
```

Plot Figure 7B upper panel, Figure 8B upper panel, and Figure S7

```
tbl_goi <-
  tibble::tribble(
    ~AGI,        ~gene_name,
    "AT5G57360", "ZTL",
    "AT5G14060", "AK2",
    "AT3G02020", "AK3",
    "AT5G64170", "LNK1",
    "AT2G46830", "CCA1",
    "AT5G02810", "PRR7",
    "AT5G60100", "PRR3",
    "AT5G61380", "TOC1",
    "AT1G22770", "GI",
  )
```

```
li_gp1 <- list()
for(i in seq_along(tbl_goi$AGI)) {
  temp_GOI <- tbl_goi$AGI[i]
  temp_name <- tbl_goi$gene_name[i]
  label <- stringr::str_glue("{temp_GOI}{temp_name}")

  tbl_plot <- make_tbl_plot(temp_GOI)

  tbl_plot_coef <-
    dplyr::bind_rows(
      tbl_out %>%
        dplyr::filter(AGI == temp_GOI) %>%
        make_tbl_plot_coef(temp_GOI) %>%
        dplyr::mutate(type = "TE mORF"),
      tbl_out_uorf %>%
        dplyr::filter(AGI == temp_GOI) %>%
        make_tbl_plot_coef(temp_GOI) %>%
        dplyr::mutate(type = "TE uORF")
    )

  if(all(is.na(tbl_plot$value) | tbl_plot$value == 0))
    next

  scale_to_y <- make_scale_y1_to_y2(tbl_plot$rel, tbl_plot_coef$coef)
  scale_to_x <- make_scale_y2_to_y1(tbl_plot$rel, tbl_plot_coef$coef)
  scale_to_x_se <- make_scale_y2_to_y1_se(tbl_plot$rel, tbl_plot_coef$coef)

  # Slightly modify the looks of plot when plotting PRR3 data
  if(temp_GOI == "AT5G60100") {
    range_x <- c(0, 6)
    range_y <- range(c(tbl_plot_coef$coef + tbl_plot_coef$se,
                       tbl_plot_coef$coef - tbl_plot_coef$se))
    scale_to_y <- make_scale_y1_to_y2(range_x, range_y)
    scale_to_x <- make_scale_y2_to_y1(range_x, range_y)
    scale_to_x_se <- make_scale_y2_to_y1_se(range_x, range_y)
  }

  gp1 <-
    ggplot() +
    tbl_plot_coef %>%
    dplyr::mutate(zt = as.integer(stringr::str_extract(ZT, "\\d+"))) %>%
    dplyr::mutate(facet = "TE") %>%
    plot_te(x = zt, scale_to_x, scale_to_y) +
    tbl_plot %>%
    dplyr::mutate(zt = as.integer(stringr::str_extract(ZT, "\\d+"))) %>%
    dplyr::mutate(facet = "count") %>%
    plot_nc(zt) +
    scale_x_continuous("ZT", breaks = c(0, 3, 6, 12, 18, 21),
                       expand = expansion(add = .5)) +
    facet_grid(cols = vars(facet), scales = "free_y") +
    theme_line() +
    theme(strip.background = element_blank()) +
    scale_shape_manual(values = SHAPE_MANUAL) +
    scale_linetype_manual(values = LINETYPE_MANUAL)

  # Slightly modify the looks of plot when plotting PRR3 data
  if(temp_GOI == "AT5G60100") {
    gp1 <-
      gp1 +
      scale_y_continuous(
        expand = expansion(mult = c(.05, .1)),
        labels = label_num,
        sec.axis = sec_axis(
          name = "Log<sub>2</sub>(FC) (vs. TE at ZT0)",
          trans = scale_to_y,
          labels = label_num,
          breaks = c(-2, 0, 2, 4)
        ),
      limits = c(0, 6)
      )
  }
  li_gp1[[label]] <- gp1

  force(scale_to_x(1:10))
  force(scale_to_y(1:10))
}
```

```
Scale for y is already present.
Adding another scale for y, which will replace the existing scale.
```

```
max_dims1 <- patchwork::get_max_dim(li_gp1)

for(i in seq_along(tbl_goi$AGI)) {
  temp_GOI <- tbl_goi$AGI[i]
  temp_name <- tbl_goi$gene_name[i] %>% stringr::str_replace_all("/", "_")
  temp_name <- ifelse(is.na(temp_name), "", paste0("_", temp_name))
  label <- stringr::str_glue("{temp_GOI}{temp_name}")

  gp1 <- patchwork::set_dim(li_gp1[[i]], max_dims1)
  if(i == 1) {
    ggsave_(path_out_fig7Bup(stringr::str_glue("{label}.png")), gp1)
    # ggsave_(path_out_fig7Bup(stringr::str_glue("{abel}.svg")), gp1)
    # ggsave_(path_out_fig7Bup(stringr::str_glue("{abel}.pdf")), gp1)
  } else if(i == 2 | i == 3) {
    label <- paste0(sprintf('%02d', i-1), "_", label)
    ggsave_(path_out_fig8Bup(stringr::str_glue("{label}.png")), gp1)
    # ggsave_(path_out_fig8Bup(stringr::str_glue("{label}.svg")), gp1)
    # ggsave_(path_out_fig8Bup(stringr::str_glue("{label}.pdf")), gp1)
  } else {
    label <- paste0(sprintf('%02d', i-3), "_", label)
    ggsave_(path_out_figS7(stringr::str_glue("{label}.png")), gp1)
    # ggsave_(path_out_figS7(stringr::str_glue("{label}.svg")), gp1)
    # ggsave_(path_out_figS7(stringr::str_glue("{label}.pdf")), gp1)
  }

}
```

Figure 7B upper panel

AT5G57360_ZTL.png

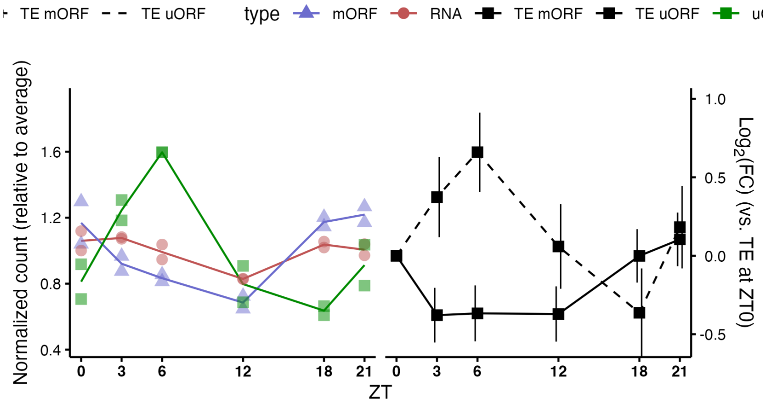
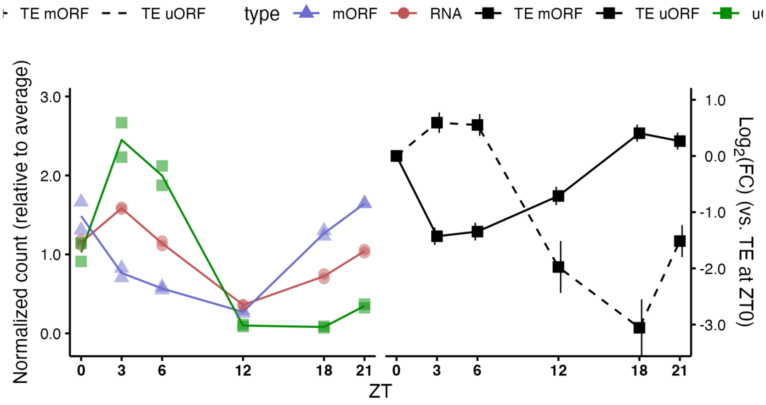


Figure 8B upper panel

01_AT5G14060_AK2.png



02_AT3G02020_AK3.png

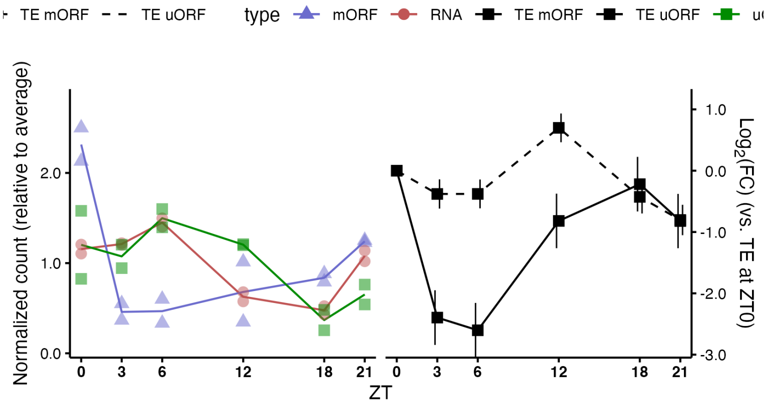
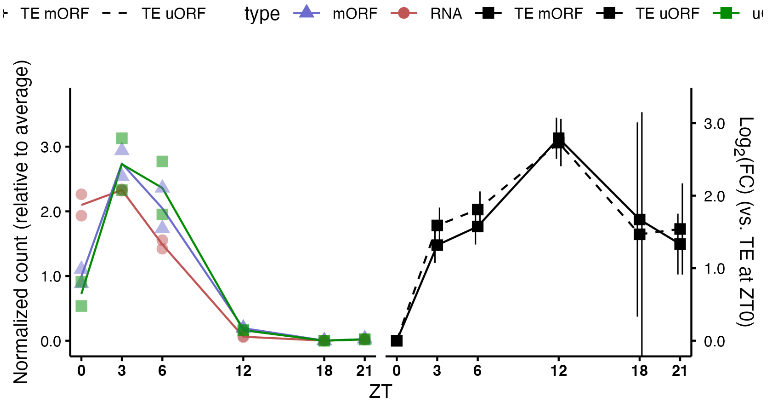
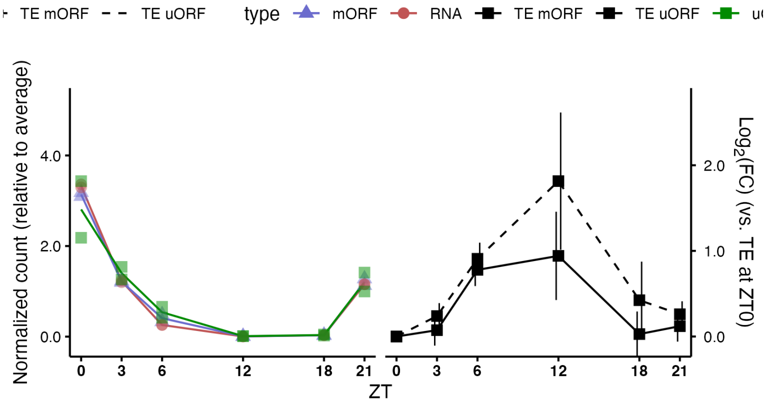


Figure S7

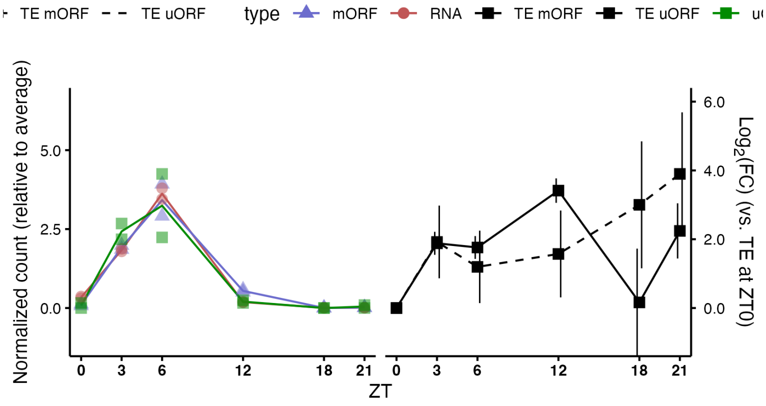
01_AT5G64170_LNK1.png



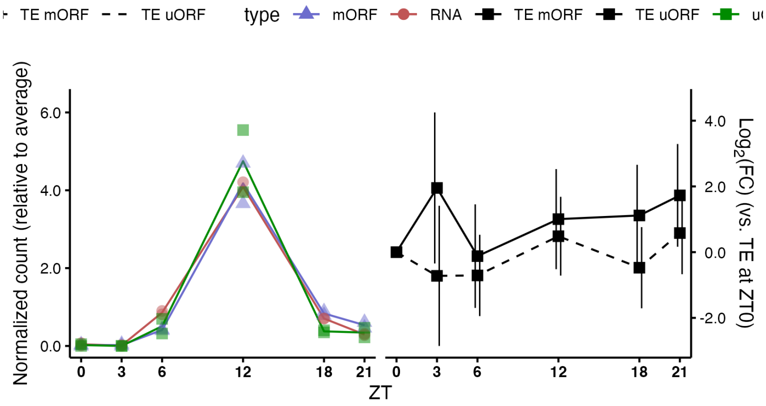
02_AT2G46830_CCA1.png



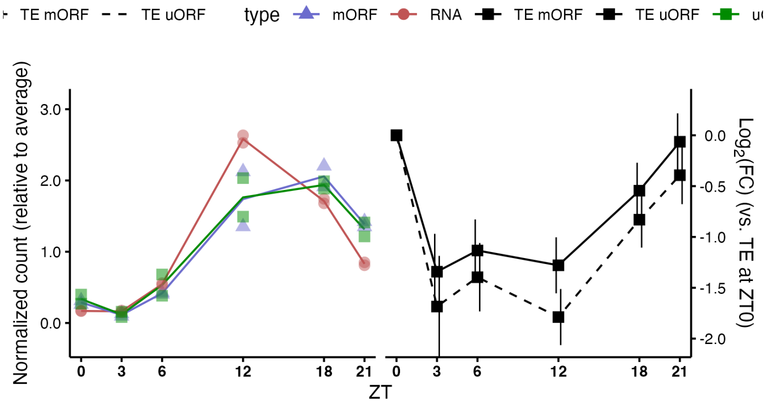
03_AT5G02810_PRR7.png



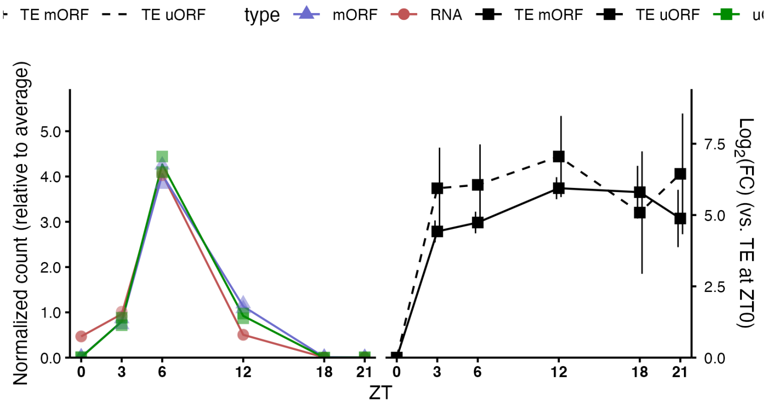
04_AT5G60100_PRR3.png



05_AT5G61380_TOC1.png



06_AT1G22770_GI.png



Plot Figure S8 upper panel

Line plot

```
temp_GOI <- "AT1G01060"
temp_name <- "LHY"
label <- stringr::str_glue("{temp_GOI}_{temp_name}")

tbl_plot <-
  tbl_nc %>%
  dplyr::filter(AGI == temp_GOI) %>%
  tidyr::pivot_longer(!AGI) %>%
  dplyr::mutate(
    ZT =
      stringr::str_extract(name, "zt\\d+") %>%
      stringr::str_to_upper() %>%
      forcats::fct_relevel(names(COL_PALETTE)),
    type =
      stringr::str_extract(name, "rna|morf|uorf") %>%
      {c(rna = "RNA", morf = "mORF", uorf = "uORF")[.]} %>%
      forcats::fct_relevel(c("RNA", "mORF", "uORF"))
  ) %>%
  dplyr::filter(ZT %in% paste0("ZT", c(0, 3, 18, 21))) %>%
  dplyr::with_groups(c(type), dplyr::mutate, rel = rel_to_ave(value))

tbl_plot_coef <-
  dplyr::bind_rows(
    tbl_out %>%
      dplyr::filter(AGI == temp_GOI) %>%
      make_tbl_plot_coef(temp_GOI) %>%
      dplyr::mutate(type = "TE mORF"),
    tbl_out_uorf %>%
      dplyr::filter(AGI == temp_GOI) %>%
      make_tbl_plot_coef(temp_GOI) %>%
      dplyr::mutate(type = "TE uORF")
  ) %>%
  dplyr::filter(ZT %in% paste0("ZT", c(0, 3, 18, 21)))

scale_to_y <- make_scale_y1_to_y2(tbl_plot$rel, tbl_plot_coef$coef)
scale_to_x <- make_scale_y2_to_y1(tbl_plot$rel, tbl_plot_coef$coef)
scale_to_x_se <- make_scale_y2_to_y1_se(tbl_plot$rel, tbl_plot_coef$coef)

gp1 <-
  ggplot() +
  tbl_plot_coef %>%
  dplyr::mutate(zt = as.integer(stringr::str_extract(ZT, "\\d+")) - 18) %>%
  dplyr::mutate(zt = ifelse(zt < 0, zt + 24, zt)) %>%
  dplyr::mutate(facet = "TE") %>%
  plot_te(x = zt, scale_to_x, scale_to_y) +
  tbl_plot %>%
  dplyr::mutate(zt = as.integer(stringr::str_extract(ZT, "\\d+")) - 18) %>%
  dplyr::mutate(zt = ifelse(zt < 0, zt + 24, zt)) %>%
  dplyr::mutate(facet = "count") %>%
  plot_nc(zt) +
  scale_x_continuous(
    "ZT",
    breaks = (c(0, 3, 18, 21) - 18) %>% {ifelse(. < 0, . + 24, .)},
    labels = function(x) (x + 18) %>% {ifelse(. >= 24, . - 24, .)},
    expand = expansion(add = .5)) +
  facet_grid(cols = vars(facet), scales = "free_y") +
  theme_line() +
  theme(strip.background = element_blank()) +
  scale_shape_manual(values = SHAPE_MANUAL)
force(scale_to_x(1:10))
```

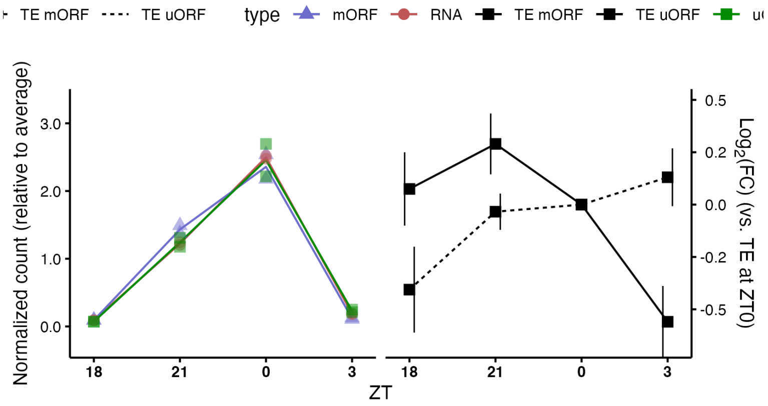
```
 [1]  4.891971  7.984568 11.077165 14.169762 17.262359 20.354956 23.447553
 [8] 26.540150 29.632747 32.725344
```

```
force(scale_to_y(1:10))
```

```
 [1] -0.25848000  0.06487284  0.38822568  0.71157852  1.03493136  1.35828420
 [7]  1.68163704  2.00498988  2.32834272  2.65169556
```

```
gp1 <- patchwork::set_dim(gp1, max_dims1)
ggsave_(path_out_figS8up(stringr::str_glue("{label}.png")), gp1)
# ggsave_(path_out_figS8up(stringr::str_glue("{label}.svg")), gp1)
# ggsave_(path_out_figS8up(stringr::str_glue("{label}.pdf")), gp1)
```

AT1G01060_LHY.png

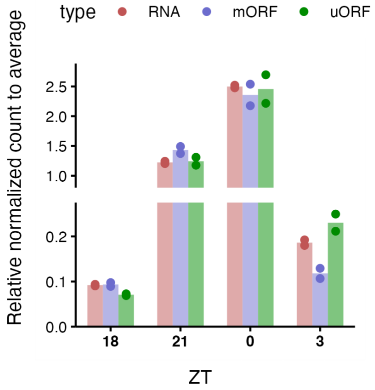


Bar plot

```
tbl_plot <-
  tbl_plot %>%
  dplyr::mutate(zt = as.integer(stringr::str_extract(ZT, "\\d+")) - 18) %>%
  dplyr::mutate(zt = ifelse(zt < 0, zt + 24, zt)) %>%
  dplyr::with_groups(c(type), dplyr::mutate, rel = value / mean(value))

gp1 <-
  tbl_plot %>%
  ggplot(aes(zt, rel, group = type)) +
  geom_col(
    data =
      tbl_plot %>%
      dplyr::with_groups(c(zt, type), dplyr::summarise, rel = mean(rel)),
    mapping = aes(zt, rel, fill = type),
    position = position_dodge(),
    width = 2,
    alpha = .5
  ) +
  geom_point(aes(color = type), position = position_dodge(width = 2)) +
  scale_x_continuous(
    "ZT",
    breaks = (c(0, 3, 18, 21) - 18) %>% {ifelse(. < 0, . + 24, .)},
    labels = function(x) (x + 18) %>% {ifelse(. >= 24, . - 24, .)},
    expand = expansion(add = .5)
  ) +
  ggbreak::scale_y_break(breaks = c(.25, .8), scales = c(1, 1),
                         expand = expansion(c(0, .1))) +
  scale_fill_manual(values = PAL_LINECOLOR) +
  scale_color_manual(values = PAL_LINECOLOR) +
  theme_bar() +
  theme(
    axis.ticks.x.top = element_blank(),
    axis.text.x.top = element_blank(),
    legend.position = "top"
  )
ggsave_ <- purrr::partial(ggsave, width = 70, height = 80, dpi = 300, units = "mm")
ggsave_(path_out_figS8up(stringr::str_glue("{label}_bar.png")), gp1)
# ggsave_(path_out_figS8up(stringr::str_glue("{label}_bar.svg")), gp1)
# ggsave_(path_out_figS8up(stringr::str_glue("{label}_bar.pdf")), gp1)
```

AT1G01060_LHY_bar.png



Sessioninfo

```
sessionInfo()
```

```
R version 4.2.1 (2022-06-23)
Platform: aarch64-apple-darwin20 (64-bit)
Running under: macOS Ventura 13.1

Matrix products: default
BLAS:   /Library/Frameworks/R.framework/Versions/4.2-arm64/Resources/lib/libRblas.0.dylib
LAPACK: /Library/Frameworks/R.framework/Versions/4.2-arm64/Resources/lib/libRlapack.dylib

locale:
[1] en_US.UTF-8/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8

attached base packages:
[1] stats     graphics  grDevices datasets  utils     methods   base

other attached packages:
[1] ggplot2_3.4.2  magrittr_2.0.3
```

```
loaded via a namespace (and not attached):
 [1] Rcpp_1.0.11         here_1.0.1          tidyr_1.3.0
 [4] rprojroot_2.0.3     digest_0.6.31       utf8_1.2.2
 [7] R6_2.5.1            evaluate_0.20       pillar_1.9.0
[10] ggfun_0.1.1         yulab.utils_0.0.5   rlang_1.1.0
[13] rstudioapi_0.14     magick_2.7.3        rmarkdown_2.24
[16] textshaping_0.3.6   labeling_0.4.2      readr_2.1.4
[19] stringr_1.5.0       bit_4.0.5           munsell_0.5.0
[22] gridtext_0.1.5      compiler_4.2.1      xfun_0.40
[25] gridGraphics_0.5-1  pkgconfig_2.0.3     systemfonts_1.0.4
[28] htmltools_0.5.3     ggtext_0.1.2        tidyselect_1.2.0
[31] tibble_3.2.1        gridExtra_2.3       fansi_1.0.3
[34] viridisLite_0.4.1   crayon_1.5.2        dplyr_1.1.1
[37] tzdb_0.3.0          withr_2.5.0         prismatic_1.1.1
[40] commonmark_1.9.0    ggbreak_0.1.2       grid_4.2.1
[43] jsonlite_1.8.4      gtable_0.3.1        lifecycle_1.0.3
[46] scales_1.2.1        cli_3.6.0           stringi_1.7.12
[49] vroom_1.6.0         farver_2.1.1        renv_1.0.3
[52] viridis_0.6.2       fs_1.5.2            xml2_1.3.3
[55] ragg_1.2.5          generics_0.1.3      vctrs_0.6.1
[58] tools_4.2.1         forcats_1.0.0       bit64_4.0.5
[61] ggplotify_0.1.0     glue_1.6.2          markdown_1.9
[64] purrr_1.0.1         hms_1.1.3           parallel_4.2.1
[67] fastmap_1.1.0       yaml_2.3.6          colorspace_2.0-3
[70] BiocManager_1.30.18 aplot_0.1.8         knitr_1.42
[73] patchwork_1.1.2     signs_0.1.2
```