## 2015. 11. 21. P. u. 26. vasárnap, utóünnep, 1. hang

I. antifona

<?xml version="1.0" encoding="utf-16"?>
<item>
  <title>November 21., I. antifona</title>
  <LilyPond>\version "2.18.2"

\include "gregorian.ly"

\paper {
  paper-width = 123.0136\mm
  paper-height = 184.6262\mm
  top-margin = 0\mm
  bottom-margin = 0\mm
  left-margin = 0\mm
  right-margin = 2\mm
  ragged-bottom = ##t
  print-page-number = ##f
  page-breaking = #ly:minimal-breaking
  system-system-spacing =
    #'((basic-distance . 12)
       (minimum-distance . 12)
       (padding . 0)
       (stretchability . 0))
%  page-breaking-system-system-spacing = #'((padding . 2))
  markup-system-spacing = #'((padding . 11.62))
  top-markup-spacing = #'((basic-distance . 0) (minimum-distance . 0) (padding . 0))
}

#(set-global-staff-size 14)

#(define currBSign 0)
#(define hideStems #t)

#(define (set-breathe bsign)
 (begin
  (set! currBSign bsign)
 (let* ((columnOfBSign (ly:grob-parent (ly:grob-parent (ly:grob-parent currBSign X) X) X))
   (system (ly:grob-parent columnOfBSign X))   
   (columns (ly:grob-object system 'columns))
   (lastColumn (ly:grob-array-ref columns (- (ly:grob-array-length columns) 1)))
  )
  (if (eqv? columnOfBSign lastColumn)
   (ly:grob-set-property! currBSign 'extra-offset '(0.89 . 0))
  )
 )
 )
)

#(define (staff-print staffgrob)
 (let* ((stencil (ly:staff-symbol::print staffgrob))
   (Xextent (ly:stencil-extent stencil X))
   (Yextent (ly:stencil-extent stencil Y))
            (X0 (car Xextent))
   (X1 (cdr Xextent))
   (X1sc (+ X1 0.89))
   )
   (ly:make-stencil (ly:stencil-expr (ly:stencil-scale stencil (/ (- X1sc X0) (- X1 X0)) 1.0))
                         (cons X0 X1sc)
                         Yextent
   )
 )
)

#(define (staff-break-callback staffgrob)
    (let*
  (
    (system (ly:grob-parent (ly:grob-parent staffgrob X) X))
    (columns (ly:grob-object system 'columns))
    (columns-count (ly:grob-array-length columns))
    (goI 1)
    (goJ 1)
  )
  (if (ly:grob? currBSign)
   (let* ((columnOfBSign (ly:grob-parent (ly:grob-parent (ly:grob-parent currBSign X) X) X))
      (lastColumn (ly:grob-array-ref columns (- columns-count 1)))
     )
     (if (eqv? columnOfBSign lastColumn)
      (ly:grob-set-property! currBSign 'extra-offset '(0.89 . 0))
      (set! currBSign 0)
     )
     (set! currBSign 0)
   )
  )
  (let loopI  ((i 0)
     )
       (if (&lt; i columns-count)
      (loopI (let* ((column (ly:grob-array-ref columns i))
          (columnname (grob-name column))
         )
         (if (and (eqv? columnname 'PaperColumn) (ly:grob-array? (ly:grob-object column 'elements)))
          (let*
           (
           (elems (ly:grob-object column 'elements))
           (elems-count (ly:grob-array-length elems))
           )
           (let loopJ ((j 0)
                )
                (if (&lt; j elems-count) 
               (loopJ (let* ((grob (ly:grob-array-ref elems j))
                   (grobname (grob-name grob))
                  )
                  (if (eqv? grobname 'NoteColumn)
                   (begin
                    (show-notes grob)
                    (set! j elems-count)
                   )
                  )
                  (1+ j)
                 )
               )
              )
           )
           (set! i columns-count)
          )
         )
         (1+ i)
        )
      )
     )
  )
 )
)

#(define (grob-name grob)
   (if (ly:grob? grob)
       (assoc-ref (ly:grob-property grob 'meta) 'name)
       #f))

#(define (show-flag flag)
 (ly:grob-set-property! flag 'transparent hideStems)
)
   
#(define (show-stem stem)
 (let*  ((beam (ly:grob-object stem 'beam))
   )
   (begin
    (ly:grob-set-property! stem 'transparent hideStems)
    (if (ly:grob? beam)(ly:grob-set-property! beam 'transparent hideStems))
   )
 )
)

#(define (show-note notehead)
 (let* ((stem (ly:grob-object (ly:grob-parent notehead X) 'stem))
        (beam (ly:grob-object stem 'beam))
     (flag (ly:grob-object notehead 'flag))
        (acci (ly:grob-object notehead 'accidental-grob))
        (dots (ly:grob-object notehead 'dot)))
 (begin
  (ly:grob-set-property! notehead 'transparent #f)
  (if (ly:grob? dots)(ly:grob-set-property! dots 'transparent #f))
  (if (ly:grob? acci)(ly:grob-set-property! acci 'transparent #f))
  (ly:grob-set-property! stem 'transparent hideStems)
  (if (ly:grob? beam)(ly:grob-set-property! beam 'transparent hideStems))
  (if (ly:grob? flag)(ly:grob-set-property! flag 'transparent hideStems))
 ))
)

#(define (show-notes note-column) 
 (begin 
  (let* ((elems (ly:grob-object note-column 'elements))
    (elems-count (ly:grob-array-length elems))
     ) 
       (let loop ((staff-pos-sum 0)
      (i 0)
        )
        (if (&lt; i elems-count)
       (loop (let* ((grob (ly:grob-array-ref elems i))
           (grobname (grob-name grob))
          )
          (if (eqv? grobname 'NoteHead)
           (show-note grob)
           (if (eqv? grobname 'Stem)
            (show-stem grob)
            (if (eqv? grobname 'Flag)
             (show-flag grob)
            )
           )
          )
         )
        (1+ i)
       )
      )
    )
  )
 )
)




leftAlign = {  \once \override LyricText #'self-alignment-X = #LEFT }
centerAlign = {   \once \override LyricText #'self-alignment-X = #CENTER }
rightAlign = {   \once \override LyricText #'self-alignment-X = #1 }
hairpinToBarline = { \override Hairpin #'to-barline = ##t  }
hairpinToNote = { \override Hairpin #'to-barline = ##f  }

alignAt =
#(define-music-function
     (parser location xOffset)
     (number?)
  (display xOffset)
   #{
      \once \override LyricText #'self-alignment-X = $xOffset
   #})

stemOn = { \revert Staff.Stem #'transparent  \revert Staff.Beam #'transparent \revert Staff.Flag #'transparent \revert Voice.Stem #'length }
stemOff = { \override Staff.Stem #'transparent = ##t \override Staff.Beam #'transparent = ##t \override Staff.Flag #'transparent = ##t \override Stem #'length = #0 }

bigNotes = { \override NoteHead #'font-size = #3 }
smallNotes = { \override NoteHead #'font-size = #-3 }
normalNotes = { \override NoteHead #'font-size = #0 }

bN = { \once \override NoteHead #'font-size = #3 }
sN = { \once \override NoteHead #'font-size = #-3 }
nN = { \once \override NoteHead #'font-size = #0 }

shOn = { \override NoteColumn #'force-hshift = #0.18 }
shOff = { \revert NoteColumn #'force-hshift }

midRest = { \override Voice.Rest #'staff-position = #0 }
shRest = { \revert Voice.Rest #'staff-position }

modernAccidentals = #(set-accidental-style 'modern)
modernCautionaryAccidentals = #(set-accidental-style 'modern-cautionary)
defaultAccidentals = #(set-accidental-style 'default)
pianoAccidentals = #(set-accidental-style 'piano)
voiceAccidentals = #(set-accidental-style 'voice)
modernVoiceAccidentals = #(set-accidental-style 'modern-voice)
modernVoiceCautionaryAccidentals = #(set-accidental-style 'modern-voice-cautionary)
pianoCautionaryAccidentals = #(set-accidental-style 'piano-cautionary)
neoModernAccidentals = #(set-accidental-style 'neo-modern)
neoModernCautionaryAccidentals = #(set-accidental-style 'neo-modern-cautionary)
dodecapohnicAccidentals = #(set-accidental-style 'dodecaphonic)
teachingAccidentals = #(set-accidental-style 'teaching)
noResetAccidentals = #(set-accidental-style 'no-reset)
forgetAccidentals = #(set-accidental-style 'forget)

go = {
\grace {s16 } 
}



chant = {
\key f \major
f'8 g' a'\breve \hideNotes a' a' a' \unHideNotes g'4( a') bes'2 a' \divisioMaxima
f'4 g' a'\breve \hideNotes a' a' a' \unHideNotes g'4( a') \break bes'2 a' \divisioMaxima \divisioMaxima
a'4( bes') c'' bes' a' g'4. a'4 g' f' e' f'2 g'\breve \unHideNotes a'4 g' f'2 e' d' \finalis \break
f'4 g' a'\breve \hideNotes a' a' \unHideNotes g'4( a') bes'2 a' \divisioMaxima
a'4 a' g'( a') bes'2 a' \finalis \divisioMaxima
a'4( bes') c'' bes' a' g'4. \break a'4 g' f' e' f'2 g'\breve \unHideNotes a'4 g' f'2 e' d' \finalis
f'4 g' a'\breve \hideNotes a' a' a' \unHideNotes g'4( a') bes'2 a' \break f'4 g' a' g'( a') bes'2 a' \divisioMaxima \divisioMaxima
a'4( bes') c'' bes' a' g'4. a'4 g' f' e' f'2 g'\breve \break \unHideNotes a'4 g' f'2 e' d' \finalis
f'4 g' a'\breve \hideNotes a' a' a' a' a' a' \unHideNotes g'4( a') bes'2 a' \break \divisioMaxima
f'4 g' a'\breve \hideNotes a' a' \unHideNotes a'4 a' g'( a') bes'2 a' \divisioMaxima \divisioMaxima
a'4( bes') c'' bes' a' g'4. a'4 g' f' e' f'2 \break g'\breve \unHideNotes a'4 g' f'2 e' d' 
\hideNotes \grace {s16 } \finalis
}

verba = \lyricmode {
"Nagy" "az" \alignAt #-1.149 "Úr" \alignAt #0 "és" \alignAt #-1.345 "igen" \alignAt #0 "dí" -- "csé" -- "re" -- "tes"
"a" "mi" \alignAt #-1.161 "Istenünk" \alignAt #-0.872 "városában," \alignAt #-1.482 "az" \alignAt #0 "ő" "szent" "he" -- "gyén."
"Az" "Is" -- "ten" -- "szü" -- "lő" "i" -- "má" -- "i" "ál" -- "tal," \alignAt #-1.039 "Üdvözítő," "üd" -- "vö" -- "zíts" "min" -- "ket!"
"Di" -- "cső" -- \alignAt #-0.841 "séges" \alignAt #-0.774 "dolgok" \alignAt #-0.72 "mondattak" "fe" -- "lő" -- "led,"
"Is" -- "ten" "vá" -- "ro" -- "sa."
"Az" "Is" -- "ten" -- "szü" -- "lő" "i" -- "má" -- "i" "ál" -- "tal," \alignAt #-1.039 "Üdvözítő," "üd" -- "vö" -- "zíts" "min" -- "ket!"
"Há" -- "za" -- \alignAt #-1.343 "iban" \alignAt #-1.482 "az" \alignAt #-1.285 "Isten" \alignAt #0 "is" -- "me" -- "re" -- "tes," "mint" "az" "ő" "ol" -- "tal" -- "muk."
"Az" "Is" -- "ten" -- "szü" -- "lő" "i" -- "má" -- "i" "ál" -- "tal," \alignAt #-1.039 "Üdvözítő," "üd" -- "vö" -- "zíts" "min" -- "ket!"
"Di" -- "cső" -- \alignAt #-0.74 "ség" \alignAt #-1.482 "az" \alignAt #-1.046 "Atyának" \alignAt #0 "és" \alignAt #-0.643 "Fiúnak" \alignAt #0 "és" \alignAt #-0.3 "Szent" -- "lé" -- "lek" -- "nek,"
"most" "és" \alignAt #-0.777 "mindenkor" \alignAt #0 "és" \alignAt #-1.112 "örökkön" -- "ö" -- "rök" -- "ké." "A" -- "men."
"Az" "Is" -- "ten" -- "szü" -- "lő" "i" -- "má" -- "i" "ál" -- "tal," \alignAt #-1.039 "Üdvözítő," "üd" -- "vö" -- "zíts" "min" -- "ket!"
}


\score {
  \new Staff &lt;&lt;
    #(set-accidental-style 'modern-cautionary)
    \new Voice = "melody" \chant
    \new Lyrics = "one" \lyricsto melody \verba
  &gt;&gt;
  \layout {
    line-thickness = #(/ (ly:output-def-lookup $defaultpaper 'staff-space) 4)
    \context {
      \Staff
      \override BarLine  #'stencil = ##f
      \remove Time_signature_engraver
      \override Stem #'transparent = #hideStems
      \override Beam #'transparent = #hideStems
      \override Flag #'transparent = #hideStems
      \override StaffSymbol #'after-line-breaking = #staff-break-callback
      \override StaffSymbol #'thickness = #(/ (ly:output-def-lookup $defaultpaper 'staff-space) 3.0)
      \override StaffSymbol #'stencil = #staff-print
    }
    \context {
      \Voice
      \override Stem #'length = #0
   \override BreathingSign #'after-line-breaking = #set-breathe
    }
    \context {
      \Score
      \override  RehearsalMark #'self-alignment-X = #LEFT
      \override RehearsalMark #'break-align-symbols = #'(key-signature)
      \override  NonMusicalPaperColumn #'padding = #0.5
      skipBars= ##t
      barAlways = ##t
      lyricMelismaAlignment = #CENTER
    }
    \context {
      \Lyrics
      \override LyricSpace #'minimum-distance = #0.5
      \override LyricText #'font-size = #1.065
      \override LyricText #'font-name = #"TimesNewRoman"
      \override LyricText #'color = #(rgb-color 0 0 0)
      \override LyricHyphen #'color = #(rgb-color 0 0 0)
      \override LyricExtender #'color = #(rgb-color 0 0 0)
    }
  }
}

\header {
tagline = "" 
title = ""
}
</LilyPond>
  <EasyChant>Nagy az Úr és igen dí- csé- re- tes
d8   r  m0             r4 m f2  m  

a  mi Istenünk városában, az ő szent he- gyén.
d4 r  m0                       r m / f2  m    

Az   Is- ten- szü- lő  i- má- i ál- tal, Üdvözítő, üd- vö- zíts min- ket! 
m4 f s   f    m    r4. m4 r   d t,  d2   r0        m4  r   d2   t,   l, ||
/
Di- cső- séges dolgok mondattak fe- lő- led,
d4  r    m0                     r m f2  m   

Is- ten vá- ro- sa.  
m4  m   r m f2  m || 

Az   Is- ten- szü- lő    i- má- i ál- tal, Üdvözítő, üd- vö- zíts min- ket! 
m4 f s   f    m    r4. / m4 r   d t,  d2   r0        m4  r   d2   t,   l, ||

Há- za- iban az Isten is- me- re- tes, mint az ő ol- tal- muk.
d4  r   m0                r m f2  m  / d4   r  m r m f2   m   

Az   Is- ten- szü- lő  i- má- i ál- tal, Üdvözítő,   üd- vö- zíts min- ket! 
m4 f s   f    m    r4. m4 r   d t,  d2   r0        / m4  r   d2   t,   l, ||

Di- cső- ség az Atyának és Fiúnak és Szent- lé- lek- nek,
d4  r    m0                                 r m f2   m   
/
most és mindenkor és örökkön - ö- rök- ké. A- men.
d4   r  m0                     m4 m    r m f2 m   

Az   Is- ten- szü- lő  i- má- i ál- tal,   Üdvözítő, üd- vö- zíts min- ket!
m4 f s   f    m    r4. m4 r   d t,  d2   / r0        m4  r   d2   t,   l,  

</EasyChant>
  <Clef>0</Clef>
  <Key>8</Key>
  <LyricsSpacing>0.5</LyricsSpacing>
  <TimeOption>0</TimeOption>
  <TimeText>N/A (cadenza)</TimeText>
  <StaffSizeOption>2</StaffSizeOption>
  <StaffSizeText>14</StaffSizeText>
  <LyricsSize>1.1</LyricsSize>
  <LyricsFontName>Times New Roman</LyricsFontName>
  <LyricsFontBold>False</LyricsFontBold>
  <LyricsFontItalic>False</LyricsFontItalic>
  <LyricsFontColor>Color [A=255, R=0, G=0, B=0]</LyricsFontColor>
  <LineSpacing>12</LineSpacing>
  <LineThickness>4</LineThickness>
  <Options>0,1,2,4,6,7,8</Options>
  <pages>1</pages>
  <page>1</page>
</item>

II. antifona

<?xml version="1.0" encoding="utf-16"?>
<item>
  <title>II. antifona2</title>
  <LilyPond>\version "2.18.2"

\include "gregorian.ly"

\paper {
  paper-width = 123.0136\mm
  paper-height = 184.6262\mm
  top-margin = 0\mm
  bottom-margin = 0\mm
  left-margin = 0\mm
  right-margin = 2\mm
  ragged-bottom = ##t
  print-page-number = ##f
  page-breaking = #ly:minimal-breaking
  system-system-spacing =
    #'((basic-distance . 12)
       (minimum-distance . 12)
       (padding . 0)
       (stretchability . 0))
%  page-breaking-system-system-spacing = #'((padding . 2))
  markup-system-spacing = #'((padding . 0))
  top-markup-spacing = #'((basic-distance . 0) (minimum-distance . 0) (padding . 0))
}

#(set-global-staff-size 14)

#(define currBSign 0)
#(define hideStems #t)

#(define (set-breathe bsign)
 (begin
  (set! currBSign bsign)
 (let* ((columnOfBSign (ly:grob-parent (ly:grob-parent (ly:grob-parent currBSign X) X) X))
   (system (ly:grob-parent columnOfBSign X))   
   (columns (ly:grob-object system 'columns))
   (lastColumn (ly:grob-array-ref columns (- (ly:grob-array-length columns) 1)))
  )
  (if (eqv? columnOfBSign lastColumn)
   (ly:grob-set-property! currBSign 'extra-offset '(0.89 . 0))
  )
 )
 )
)

#(define (staff-print staffgrob)
 (let* ((stencil (ly:staff-symbol::print staffgrob))
   (Xextent (ly:stencil-extent stencil X))
   (Yextent (ly:stencil-extent stencil Y))
            (X0 (car Xextent))
   (X1 (cdr Xextent))
   (X1sc (+ X1 0.89))
   )
   (ly:make-stencil (ly:stencil-expr (ly:stencil-scale stencil (/ (- X1sc X0) (- X1 X0)) 1.0))
                         (cons X0 X1sc)
                         Yextent
   )
 )
)

#(define (staff-break-callback staffgrob)
    (let*
  (
    (system (ly:grob-parent (ly:grob-parent staffgrob X) X))
    (columns (ly:grob-object system 'columns))
    (columns-count (ly:grob-array-length columns))
    (goI 1)
    (goJ 1)
  )
  (if (ly:grob? currBSign)
   (let* ((columnOfBSign (ly:grob-parent (ly:grob-parent (ly:grob-parent currBSign X) X) X))
      (lastColumn (ly:grob-array-ref columns (- columns-count 1)))
     )
     (if (eqv? columnOfBSign lastColumn)
      (ly:grob-set-property! currBSign 'extra-offset '(0.89 . 0))
      (set! currBSign 0)
     )
     (set! currBSign 0)
   )
  )
  (let loopI  ((i 0)
     )
       (if (&lt; i columns-count)
      (loopI (let* ((column (ly:grob-array-ref columns i))
          (columnname (grob-name column))
         )
         (if (and (eqv? columnname 'PaperColumn) (ly:grob-array? (ly:grob-object column 'elements)))
          (let*
           (
           (elems (ly:grob-object column 'elements))
           (elems-count (ly:grob-array-length elems))
           )
           (let loopJ ((j 0)
                )
                (if (&lt; j elems-count) 
               (loopJ (let* ((grob (ly:grob-array-ref elems j))
                   (grobname (grob-name grob))
                  )
                  (if (eqv? grobname 'NoteColumn)
                   (begin
                    (show-notes grob)
                    (set! j elems-count)
                   )
                  )
                  (1+ j)
                 )
               )
              )
           )
           (set! i columns-count)
          )
         )
         (1+ i)
        )
      )
     )
  )
 )
)

#(define (grob-name grob)
   (if (ly:grob? grob)
       (assoc-ref (ly:grob-property grob 'meta) 'name)
       #f))

#(define (show-flag flag)
 (ly:grob-set-property! flag 'transparent hideStems)
)
   
#(define (show-stem stem)
 (let*  ((beam (ly:grob-object stem 'beam))
   )
   (begin
    (ly:grob-set-property! stem 'transparent hideStems)
    (if (ly:grob? beam)(ly:grob-set-property! beam 'transparent hideStems))
   )
 )
)

#(define (show-note notehead)
 (let* ((stem (ly:grob-object (ly:grob-parent notehead X) 'stem))
        (beam (ly:grob-object stem 'beam))
     (flag (ly:grob-object notehead 'flag))
        (acci (ly:grob-object notehead 'accidental-grob))
        (dots (ly:grob-object notehead 'dot)))
 (begin
  (ly:grob-set-property! notehead 'transparent #f)
  (if (ly:grob? dots)(ly:grob-set-property! dots 'transparent #f))
  (if (ly:grob? acci)(ly:grob-set-property! acci 'transparent #f))
  (ly:grob-set-property! stem 'transparent hideStems)
  (if (ly:grob? beam)(ly:grob-set-property! beam 'transparent hideStems))
  (if (ly:grob? flag)(ly:grob-set-property! flag 'transparent hideStems))
 ))
)

#(define (show-notes note-column) 
 (begin 
  (let* ((elems (ly:grob-object note-column 'elements))
    (elems-count (ly:grob-array-length elems))
     ) 
       (let loop ((staff-pos-sum 0)
      (i 0)
        )
        (if (&lt; i elems-count)
       (loop (let* ((grob (ly:grob-array-ref elems i))
           (grobname (grob-name grob))
          )
          (if (eqv? grobname 'NoteHead)
           (show-note grob)
           (if (eqv? grobname 'Stem)
            (show-stem grob)
            (if (eqv? grobname 'Flag)
             (show-flag grob)
            )
           )
          )
         )
        (1+ i)
       )
      )
    )
  )
 )
)




leftAlign = {  \once \override LyricText #'self-alignment-X = #LEFT }
centerAlign = {   \once \override LyricText #'self-alignment-X = #CENTER }
rightAlign = {   \once \override LyricText #'self-alignment-X = #1 }
hairpinToBarline = { \override Hairpin #'to-barline = ##t  }
hairpinToNote = { \override Hairpin #'to-barline = ##f  }

alignAt =
#(define-music-function
     (parser location xOffset)
     (number?)
  (display xOffset)
   #{
      \once \override LyricText #'self-alignment-X = $xOffset
   #})

stemOn = { \revert Staff.Stem #'transparent  \revert Staff.Beam #'transparent \revert Staff.Flag #'transparent \revert Voice.Stem #'length }
stemOff = { \override Staff.Stem #'transparent = ##t \override Staff.Beam #'transparent = ##t \override Staff.Flag #'transparent = ##t \override Stem #'length = #0 }

bigNotes = { \override NoteHead #'font-size = #3 }
smallNotes = { \override NoteHead #'font-size = #-3 }
normalNotes = { \override NoteHead #'font-size = #0 }

bN = { \once \override NoteHead #'font-size = #3 }
sN = { \once \override NoteHead #'font-size = #-3 }
nN = { \once \override NoteHead #'font-size = #0 }

shOn = { \override NoteColumn #'force-hshift = #0.18 }
shOff = { \revert NoteColumn #'force-hshift }

midRest = { \override Voice.Rest #'staff-position = #0 }
shRest = { \revert Voice.Rest #'staff-position }

modernAccidentals = #(set-accidental-style 'modern)
modernCautionaryAccidentals = #(set-accidental-style 'modern-cautionary)
defaultAccidentals = #(set-accidental-style 'default)
pianoAccidentals = #(set-accidental-style 'piano)
voiceAccidentals = #(set-accidental-style 'voice)
modernVoiceAccidentals = #(set-accidental-style 'modern-voice)
modernVoiceCautionaryAccidentals = #(set-accidental-style 'modern-voice-cautionary)
pianoCautionaryAccidentals = #(set-accidental-style 'piano-cautionary)
neoModernAccidentals = #(set-accidental-style 'neo-modern)
neoModernCautionaryAccidentals = #(set-accidental-style 'neo-modern-cautionary)
dodecapohnicAccidentals = #(set-accidental-style 'dodecaphonic)
teachingAccidentals = #(set-accidental-style 'teaching)
noResetAccidentals = #(set-accidental-style 'no-reset)
forgetAccidentals = #(set-accidental-style 'forget)

go = {
\grace {s16 } 
}



chant = {
\key f \major
&lt;&lt;
{
\voiceOne
f'4 g' a'\breve \hideNotes a' a' \unHideNotes g'4( a') bes'2 a' \divisioMaxima
a'4 bes' c''2 bes'4 a' g' f' e' f'2 \break \divisioMaxima
f'4 g' a'\breve \hideNotes a' a' \unHideNotes g'4( a') bes'2 a' \break \divisioMaxima
a'2 bes'4 c'' bes' a' g' f'( e') f'4. g'2 a'4( g') f'( e') d'2 
\hideNotes \grace {s16 } \finalis
}
\new Voice {
\voiceTwo
\smallNotes
s\longa s\longa
f'4 g' a'2 bes'4 a' g' f' e' f'2
s\longa s\longa
s\longa s2 s8
}
&gt;&gt;
\oneVoice

}

verba = \lyricmode {
"Meg" -- "szen" -- \alignAt #-0.895 "teli" \alignAt #-0.82 "hajlékát" \alignAt #0 "a" "Föl" -- "sé" -- "ges."
"Üd" -- "vö" -- "zíts" "min" -- "ket," "Is" -- "ten" "Fi" -- "a,"
"a" -- "ki" \alignAt #-0.881 "halottaidból" \alignAt #-0.944 "föltámadtál," \alignAt #-1.182 "éneke" -- "lünk" "né" -- "ked:"
"Al" -- "le" -- "lu" -- "ja," "al" -- "le" -- "lu" -- "ja," "al" -- "le" -- "lu" -- "ja."
}


\score {
  \new Staff &lt;&lt;
    #(set-accidental-style 'modern-cautionary)
    \new Voice = "melody" \chant
    \new Lyrics = "one" \lyricsto melody \verba
  &gt;&gt;
  \layout {
    ragged-last = ##f
    line-thickness = #(/ (ly:output-def-lookup $defaultpaper 'staff-space) 4)
    \context {
      \Staff
      \override BarLine  #'stencil = ##f
      \remove Time_signature_engraver
      \override Stem #'transparent = #hideStems
      \override Beam #'transparent = #hideStems
      \override Flag #'transparent = #hideStems
      \override StaffSymbol #'after-line-breaking = #staff-break-callback
      \override StaffSymbol #'thickness = #(/ (ly:output-def-lookup $defaultpaper 'staff-space) 3.0)
      \override StaffSymbol #'stencil = #staff-print
    }
    \context {
      \Voice
      \override Stem #'length = #0
   \override BreathingSign #'after-line-breaking = #set-breathe
    }
    \context {
      \Score
      \override  RehearsalMark #'self-alignment-X = #LEFT
      \override RehearsalMark #'break-align-symbols = #'(key-signature)
      \override  NonMusicalPaperColumn #'padding = #0.8
      skipBars= ##t
      barAlways = ##t
      lyricMelismaAlignment = #CENTER
    }
    \context {
      \Lyrics
      \override LyricSpace #'minimum-distance = #0.8
      \override LyricText #'font-size = #1.065
      \override LyricText #'font-name = #"TimesNewRoman"
      \override LyricText #'color = #(rgb-color 0 0 0)
      \override LyricHyphen #'color = #(rgb-color 0 0 0)
      \override LyricExtender #'color = #(rgb-color 0 0 0)
    }
  }
}

\header {
tagline = "" 
title = ""
}
</LilyPond>
  <EasyChant>Meg- szen- teli hajlékát a Föl- sé- ges.
d4   r     m0              r m  f2  m   

Üd- vö- zíts min- ket, Is- ten Fi- a,
m4  f   s2   f4   m    r   d   t,  d2
d4  r   m2   f4   m    r   d   t,  d2
/
a- ki halottaidból föltámadtál, éneke- lünk né- ked:
d4 r  m0                               r m  f2  m   
/
Al- le- lu- ja, al- le- lu-  ja,  al- le-  lu-  ja. 
m2  f4  s   f   m   r   d t, d4.  r2  m4 r d t, l,2 
</EasyChant>
  <Clef>0</Clef>
  <Key>8</Key>
  <LyricsSpacing>0.8</LyricsSpacing>
  <TimeOption>0</TimeOption>
  <TimeText>N/A (cadenza)</TimeText>
  <StaffSizeOption>2</StaffSizeOption>
  <StaffSizeText>14</StaffSizeText>
  <LyricsSize>1.1</LyricsSize>
  <LyricsFontName>Times New Roman</LyricsFontName>
  <LyricsFontBold>False</LyricsFontBold>
  <LyricsFontItalic>False</LyricsFontItalic>
  <LyricsFontColor>Color [A=255, R=0, G=0, B=0]</LyricsFontColor>
  <LineSpacing>12</LineSpacing>
  <LineThickness>4</LineThickness>
  <Options>0,1,2,6,7,8</Options>
  <pages>1</pages>
  <page>1</page>
</item>

<?xml version="1.0" encoding="utf-16"?>
<item>
  <title>II. antifona</title>
  <LilyPond>\version "2.18.2"

\include "gregorian.ly"

\paper {
  paper-width = 123.0136\mm
  paper-height = 184.6262\mm
  top-margin = 0\mm
  bottom-margin = 0\mm
  left-margin = 0\mm
  right-margin = 2\mm
  ragged-bottom = ##t
  print-page-number = ##f
  page-breaking = #ly:minimal-breaking
  system-system-spacing =
    #'((basic-distance . 12)
       (minimum-distance . 12)
       (padding . 0)
       (stretchability . 0))
%  page-breaking-system-system-spacing = #'((padding . 2))
  markup-system-spacing = #'((padding . 4.31))
  top-markup-spacing = #'((basic-distance . 0) (minimum-distance . 0) (padding . 0))
}

#(set-global-staff-size 14)

#(define currBSign 0)
#(define hideStems #t)

#(define (set-breathe bsign)
 (begin
  (set! currBSign bsign)
 (let* ((columnOfBSign (ly:grob-parent (ly:grob-parent (ly:grob-parent currBSign X) X) X))
   (system (ly:grob-parent columnOfBSign X))   
   (columns (ly:grob-object system 'columns))
   (lastColumn (ly:grob-array-ref columns (- (ly:grob-array-length columns) 1)))
  )
  (if (eqv? columnOfBSign lastColumn)
   (ly:grob-set-property! currBSign 'extra-offset '(0.89 . 0))
  )
 )
 )
)

#(define (staff-print staffgrob)
 (let* ((stencil (ly:staff-symbol::print staffgrob))
   (Xextent (ly:stencil-extent stencil X))
   (Yextent (ly:stencil-extent stencil Y))
            (X0 (car Xextent))
   (X1 (cdr Xextent))
   (X1sc (+ X1 0.89))
   )
   (ly:make-stencil (ly:stencil-expr (ly:stencil-scale stencil (/ (- X1sc X0) (- X1 X0)) 1.0))
                         (cons X0 X1sc)
                         Yextent
   )
 )
)

#(define (staff-break-callback staffgrob)
    (let*
  (
    (system (ly:grob-parent (ly:grob-parent staffgrob X) X))
    (columns (ly:grob-object system 'columns))
    (columns-count (ly:grob-array-length columns))
    (goI 1)
    (goJ 1)
  )
  (if (ly:grob? currBSign)
   (let* ((columnOfBSign (ly:grob-parent (ly:grob-parent (ly:grob-parent currBSign X) X) X))
      (lastColumn (ly:grob-array-ref columns (- columns-count 1)))
     )
     (if (eqv? columnOfBSign lastColumn)
      (ly:grob-set-property! currBSign 'extra-offset '(0.89 . 0))
      (set! currBSign 0)
     )
     (set! currBSign 0)
   )
  )
  (let loopI  ((i 0)
     )
       (if (&lt; i columns-count)
      (loopI (let* ((column (ly:grob-array-ref columns i))
          (columnname (grob-name column))
         )
         (if (and (eqv? columnname 'PaperColumn) (ly:grob-array? (ly:grob-object column 'elements)))
          (let*
           (
           (elems (ly:grob-object column 'elements))
           (elems-count (ly:grob-array-length elems))
           )
           (let loopJ ((j 0)
                )
                (if (&lt; j elems-count) 
               (loopJ (let* ((grob (ly:grob-array-ref elems j))
                   (grobname (grob-name grob))
                  )
                  (if (eqv? grobname 'NoteColumn)
                   (begin
                    (show-notes grob)
                    (set! j elems-count)
                   )
                  )
                  (1+ j)
                 )
               )
              )
           )
           (set! i columns-count)
          )
         )
         (1+ i)
        )
      )
     )
  )
 )
)

#(define (grob-name grob)
   (if (ly:grob? grob)
       (assoc-ref (ly:grob-property grob 'meta) 'name)
       #f))

#(define (show-flag flag)
 (ly:grob-set-property! flag 'transparent hideStems)
)
   
#(define (show-stem stem)
 (let*  ((beam (ly:grob-object stem 'beam))
   )
   (begin
    (ly:grob-set-property! stem 'transparent hideStems)
    (if (ly:grob? beam)(ly:grob-set-property! beam 'transparent hideStems))
   )
 )
)

#(define (show-note notehead)
 (let* ((stem (ly:grob-object (ly:grob-parent notehead X) 'stem))
        (beam (ly:grob-object stem 'beam))
     (flag (ly:grob-object notehead 'flag))
        (acci (ly:grob-object notehead 'accidental-grob))
        (dots (ly:grob-object notehead 'dot)))
 (begin
  (ly:grob-set-property! notehead 'transparent #f)
  (if (ly:grob? dots)(ly:grob-set-property! dots 'transparent #f))
  (if (ly:grob? acci)(ly:grob-set-property! acci 'transparent #f))
  (ly:grob-set-property! stem 'transparent hideStems)
  (if (ly:grob? beam)(ly:grob-set-property! beam 'transparent hideStems))
  (if (ly:grob? flag)(ly:grob-set-property! flag 'transparent hideStems))
 ))
)

#(define (show-notes note-column) 
 (begin 
  (let* ((elems (ly:grob-object note-column 'elements))
    (elems-count (ly:grob-array-length elems))
     ) 
       (let loop ((staff-pos-sum 0)
      (i 0)
        )
        (if (&lt; i elems-count)
       (loop (let* ((grob (ly:grob-array-ref elems i))
           (grobname (grob-name grob))
          )
          (if (eqv? grobname 'NoteHead)
           (show-note grob)
           (if (eqv? grobname 'Stem)
            (show-stem grob)
            (if (eqv? grobname 'Flag)
             (show-flag grob)
            )
           )
          )
         )
        (1+ i)
       )
      )
    )
  )
 )
)




leftAlign = {  \once \override LyricText #'self-alignment-X = #LEFT }
centerAlign = {   \once \override LyricText #'self-alignment-X = #CENTER }
rightAlign = {   \once \override LyricText #'self-alignment-X = #1 }
hairpinToBarline = { \override Hairpin #'to-barline = ##t  }
hairpinToNote = { \override Hairpin #'to-barline = ##f  }

alignAt =
#(define-music-function
     (parser location xOffset)
     (number?)
  (display xOffset)
   #{
      \once \override LyricText #'self-alignment-X = $xOffset
   #})

stemOn = { \revert Staff.Stem #'transparent  \revert Staff.Beam #'transparent \revert Staff.Flag #'transparent \revert Voice.Stem #'length }
stemOff = { \override Staff.Stem #'transparent = ##t \override Staff.Beam #'transparent = ##t \override Staff.Flag #'transparent = ##t \override Stem #'length = #0 }

bigNotes = { \override NoteHead #'font-size = #3 }
smallNotes = { \override NoteHead #'font-size = #-3 }
normalNotes = { \override NoteHead #'font-size = #0 }

bN = { \once \override NoteHead #'font-size = #3 }
sN = { \once \override NoteHead #'font-size = #-3 }
nN = { \once \override NoteHead #'font-size = #0 }

shOn = { \override NoteColumn #'force-hshift = #0.18 }
shOff = { \revert NoteColumn #'force-hshift }

midRest = { \override Voice.Rest #'staff-position = #0 }
shRest = { \revert Voice.Rest #'staff-position }

modernAccidentals = #(set-accidental-style 'modern)
modernCautionaryAccidentals = #(set-accidental-style 'modern-cautionary)
defaultAccidentals = #(set-accidental-style 'default)
pianoAccidentals = #(set-accidental-style 'piano)
voiceAccidentals = #(set-accidental-style 'voice)
modernVoiceAccidentals = #(set-accidental-style 'modern-voice)
modernVoiceCautionaryAccidentals = #(set-accidental-style 'modern-voice-cautionary)
pianoCautionaryAccidentals = #(set-accidental-style 'piano-cautionary)
neoModernAccidentals = #(set-accidental-style 'neo-modern)
neoModernCautionaryAccidentals = #(set-accidental-style 'neo-modern-cautionary)
dodecapohnicAccidentals = #(set-accidental-style 'dodecaphonic)
teachingAccidentals = #(set-accidental-style 'teaching)
noResetAccidentals = #(set-accidental-style 'no-reset)
forgetAccidentals = #(set-accidental-style 'forget)

go = {
\grace {s16 } 
}



chant = {
\key f \major
&lt;&lt;
{
\voiceOne
f'8 g'4 a'8 g'([ a')] bes'2 a' f'8 g' a'\breve \hideNotes a' \unHideNotes g'4( a') bes'2 a' \break \divisioMaxima
a'4 bes' c''2 bes'4 a' g' f' e' f'2 \divisioMaxima
f'4 g' a'\breve \hideNotes a' a' \unHideNotes g'4( a') bes'2 a' \break \divisioMaxima
a'2 bes'4 c'' bes' a' g' f'( e') f'4. g'2 a'4( g') f'( e') d'2 \finalis
f'4 g' a' g'( a') bes'2 a' f'4 g' a' a' \break a' a' g'( a') bes'2 a' \divisioMaxima
a'4 bes' c''2 bes'4 a' g' f' e' f'2 \divisioMaxima
f'4 g' a'\breve \hideNotes a' a' \unHideNotes g'4( a') bes'2 a' \divisioMaxima
a'2 bes'4 c'' bes' a' g' f'( e') f'4. g'2 a'4( g') f'( e') d'2 
\hideNotes \grace {s16 } \finalis
}
\new Voice {
\voiceTwo
\smallNotes
s\longa s\breve s1 s2
f'4 g' a'2 bes'4 a' g' f' e' f'2
s\longa s\longa
s\longa s2 s8
s\longa s1 s4
f'4 g' a'2 bes'4 a' g' f' e' f'2
s\longa s\longa
s\longa s2 s8
}
&gt;&gt;
\oneVoice

}

verba = \lyricmode {
"Ő" "az" "Úr" "aj" -- "ta" -- "ja:" "i" -- "ga" -- \alignAt #-0.677 "zak" \alignAt #-0.643 "mennek" "át" "raj" -- "ta."
"Üd" -- "vö" -- "zíts" "min" -- "ket," "Is" -- "ten" "Fi" -- "a,"
"a" -- "ki" \alignAt #-0.881 "halottaidból" \alignAt #-0.944 "föltámadtál," \alignAt #-1.182 "éneke" -- "lünk" "né" -- "ked:"
"Al" -- "le" -- "lu" -- "ja," "al" -- "le" -- "lu" -- "ja," "al" -- "le" -- "lu" -- "ja."
"Szent" "a" "te" "temp" -- "lo" -- "mod," "cso" -- "dá" -- "la" -- "tos" "az" "i" -- "gaz" -- "ság" -- "ban."
"Üd" -- "vö" -- "zíts" "min" -- "ket," "Is" -- "ten" "Fi" -- "a,"
"a" -- "ki" \alignAt #-0.881 "halottaidból" \alignAt #-0.944 "föltámadtál," \alignAt #-1.182 "éneke" -- "lünk" "né" -- "ked:"
"Al" -- "le" -- "lu" -- "ja," "al" -- "le" -- "lu" -- "ja," "al" -- "le" -- "lu" -- "ja."
}


\score {
  \new Staff &lt;&lt;
    #(set-accidental-style 'modern-cautionary)
    \new Voice = "melody" \chant
    \new Lyrics = "one" \lyricsto melody \verba
  &gt;&gt;
  \layout {
    ragged-last = ##f
    line-thickness = #(/ (ly:output-def-lookup $defaultpaper 'staff-space) 4)
    \context {
      \Staff
      \override BarLine  #'stencil = ##f
      \remove Time_signature_engraver
      \override Stem #'transparent = #hideStems
      \override Beam #'transparent = #hideStems
      \override Flag #'transparent = #hideStems
      \override StaffSymbol #'after-line-breaking = #staff-break-callback
      \override StaffSymbol #'thickness = #(/ (ly:output-def-lookup $defaultpaper 'staff-space) 3.0)
      \override StaffSymbol #'stencil = #staff-print
    }
    \context {
      \Voice
      \override Stem #'length = #0
   \override BreathingSign #'after-line-breaking = #set-breathe
    }
    \context {
      \Score
      \override  RehearsalMark #'self-alignment-X = #LEFT
      \override RehearsalMark #'break-align-symbols = #'(key-signature)
      \override  NonMusicalPaperColumn #'padding = #0.8
      skipBars= ##t
      barAlways = ##t
      lyricMelismaAlignment = #CENTER
    }
    \context {
      \Lyrics
      \override LyricSpace #'minimum-distance = #0.8
      \override LyricText #'font-size = #1.065
      \override LyricText #'font-name = #"TimesNewRoman"
      \override LyricText #'color = #(rgb-color 0 0 0)
      \override LyricHyphen #'color = #(rgb-color 0 0 0)
      \override LyricExtender #'color = #(rgb-color 0 0 0)
    }
  }
}

\header {
tagline = "" 
title = ""
}
</LilyPond>
  <EasyChant>Ő  az Úr aj- ta- ja: i- ga- zak mennek át   raj- ta.
d8 r4 m8 r m f2  m   d8 r   m0         r4 m f2   m  
/
Üd- vö- zíts min- ket, Is- ten Fi- a,
m4  f   s2   f4   m    r   d   t,  d2
d4  r   m2   f4   m    r   d   t,  d2

a- ki halottaidból föltámadtál, éneke- lünk né- ked:
d4 r  m0                               r m  f2  m   
/
Al- le- lu- ja, al- le- lu-  ja, al- le-  lu-  ja.   
m2  f4  s   f   m   r   d t, d4. r2  m4 r d t, l,2 ||

Szent a te temp- lo- mod, cso- dá- la- tos az   i- gaz- ság- ban.
d4    r m  r m   f2  m    d4   r   m   m / m    m  r m  f2   m   

Üd- vö- zíts min- ket, Is- ten Fi- a,
m4  f   s2   f4   m    r   d   t,  d2
d4  r   m2   f4   m    r   d   t,  d2

a- ki halottaidból föltámadtál, éneke- lünk né- ked:
d4 r  m0                               r m  f2  m   

Al- le- lu- ja, al- le- lu-  ja, al- le-  lu-  ja.
m2  f4  s   f   m   r   d t, d4. r2  m4 r d t, l,2

</EasyChant>
  <Clef>0</Clef>
  <Key>8</Key>
  <LyricsSpacing>0.8</LyricsSpacing>
  <TimeOption>0</TimeOption>
  <TimeText>N/A (cadenza)</TimeText>
  <StaffSizeOption>2</StaffSizeOption>
  <StaffSizeText>14</StaffSizeText>
  <LyricsSize>1.1</LyricsSize>
  <LyricsFontName>Times New Roman</LyricsFontName>
  <LyricsFontBold>False</LyricsFontBold>
  <LyricsFontItalic>False</LyricsFontItalic>
  <LyricsFontColor>Color [A=255, R=0, G=0, B=0]</LyricsFontColor>
  <LineSpacing>12</LineSpacing>
  <LineThickness>4</LineThickness>
  <Options>0,1,2,6,7,8</Options>
  <pages>1</pages>
  <page>1</page>
</item>

III. antifona: Boldogságok,

Jertek, imádjuk... aki halottaidból...

Ft. tropár, 1. hang

<?xml version="1.0" encoding="utf-16"?>
<item>
  <title>Tropár, 1. hang1</title>
  <LilyPond>\version "2.18.2"

\include "gregorian.ly"

\paper {
  paper-width = 123.0136\mm
  paper-height = 184.6262\mm
  top-margin = 0\mm
  bottom-margin = 0\mm
  left-margin = 0\mm
  right-margin = 2\mm
  ragged-bottom = ##t
  print-page-number = ##f
  page-breaking = #ly:minimal-breaking
  system-system-spacing =
    #'((basic-distance . 12)
       (minimum-distance . 12)
       (padding . 0)
       (stretchability . 0))
%  page-breaking-system-system-spacing = #'((padding . 2))
  markup-system-spacing = #'((padding . 9.15))
  top-markup-spacing = #'((basic-distance . 0) (minimum-distance . 0) (padding . 0))
}

#(set-global-staff-size 14)

#(define currBSign 0)
#(define hideStems #t)

#(define (set-breathe bsign)
 (begin
  (set! currBSign bsign)
 (let* ((columnOfBSign (ly:grob-parent (ly:grob-parent (ly:grob-parent currBSign X) X) X))
   (system (ly:grob-parent columnOfBSign X))   
   (columns (ly:grob-object system 'columns))
   (lastColumn (ly:grob-array-ref columns (- (ly:grob-array-length columns) 1)))
  )
  (if (eqv? columnOfBSign lastColumn)
   (ly:grob-set-property! currBSign 'extra-offset '(0.89 . 0))
  )
 )
 )
)

#(define (staff-print staffgrob)
 (let* ((stencil (ly:staff-symbol::print staffgrob))
   (Xextent (ly:stencil-extent stencil X))
   (Yextent (ly:stencil-extent stencil Y))
            (X0 (car Xextent))
   (X1 (cdr Xextent))
   (X1sc (+ X1 0.89))
   )
   (ly:make-stencil (ly:stencil-expr (ly:stencil-scale stencil (/ (- X1sc X0) (- X1 X0)) 1.0))
                         (cons X0 X1sc)
                         Yextent
   )
 )
)

#(define (staff-break-callback staffgrob)
    (let*
  (
    (system (ly:grob-parent (ly:grob-parent staffgrob X) X))
    (columns (ly:grob-object system 'columns))
    (columns-count (ly:grob-array-length columns))
    (goI 1)
    (goJ 1)
  )
  (if (ly:grob? currBSign)
   (let* ((columnOfBSign (ly:grob-parent (ly:grob-parent (ly:grob-parent currBSign X) X) X))
      (lastColumn (ly:grob-array-ref columns (- columns-count 1)))
     )
     (if (eqv? columnOfBSign lastColumn)
      (ly:grob-set-property! currBSign 'extra-offset '(0.89 . 0))
      (set! currBSign 0)
     )
     (set! currBSign 0)
   )
  )
  (let loopI  ((i 0)
     )
       (if (&lt; i columns-count)
      (loopI (let* ((column (ly:grob-array-ref columns i))
          (columnname (grob-name column))
         )
         (if (and (eqv? columnname 'PaperColumn) (ly:grob-array? (ly:grob-object column 'elements)))
          (let*
           (
           (elems (ly:grob-object column 'elements))
           (elems-count (ly:grob-array-length elems))
           )
           (let loopJ ((j 0)
                )
                (if (&lt; j elems-count) 
               (loopJ (let* ((grob (ly:grob-array-ref elems j))
                   (grobname (grob-name grob))
                  )
                  (if (eqv? grobname 'NoteColumn)
                   (begin
                    (show-notes grob)
                    (set! j elems-count)
                   )
                  )
                  (1+ j)
                 )
               )
              )
           )
           (set! i columns-count)
          )
         )
         (1+ i)
        )
      )
     )
  )
 )
)

#(define (grob-name grob)
   (if (ly:grob? grob)
       (assoc-ref (ly:grob-property grob 'meta) 'name)
       #f))

#(define (show-flag flag)
 (ly:grob-set-property! flag 'transparent hideStems)
)
   
#(define (show-stem stem)
 (let*  ((beam (ly:grob-object stem 'beam))
   )
   (begin
    (ly:grob-set-property! stem 'transparent hideStems)
    (if (ly:grob? beam)(ly:grob-set-property! beam 'transparent hideStems))
   )
 )
)

#(define (show-note notehead)
 (let* ((stem (ly:grob-object (ly:grob-parent notehead X) 'stem))
        (beam (ly:grob-object stem 'beam))
     (flag (ly:grob-object notehead 'flag))
        (acci (ly:grob-object notehead 'accidental-grob))
        (dots (ly:grob-object notehead 'dot)))
 (begin
  (ly:grob-set-property! notehead 'transparent #f)
  (if (ly:grob? dots)(ly:grob-set-property! dots 'transparent #f))
  (if (ly:grob? acci)(ly:grob-set-property! acci 'transparent #f))
  (ly:grob-set-property! stem 'transparent hideStems)
  (if (ly:grob? beam)(ly:grob-set-property! beam 'transparent hideStems))
  (if (ly:grob? flag)(ly:grob-set-property! flag 'transparent hideStems))
 ))
)

#(define (show-notes note-column) 
 (begin 
  (let* ((elems (ly:grob-object note-column 'elements))
    (elems-count (ly:grob-array-length elems))
     ) 
       (let loop ((staff-pos-sum 0)
      (i 0)
        )
        (if (&lt; i elems-count)
       (loop (let* ((grob (ly:grob-array-ref elems i))
           (grobname (grob-name grob))
          )
          (if (eqv? grobname 'NoteHead)
           (show-note grob)
           (if (eqv? grobname 'Stem)
            (show-stem grob)
            (if (eqv? grobname 'Flag)
             (show-flag grob)
            )
           )
          )
         )
        (1+ i)
       )
      )
    )
  )
 )
)




leftAlign = {  \once \override LyricText #'self-alignment-X = #LEFT }
centerAlign = {   \once \override LyricText #'self-alignment-X = #CENTER }
rightAlign = {   \once \override LyricText #'self-alignment-X = #1 }
hairpinToBarline = { \override Hairpin #'to-barline = ##t  }
hairpinToNote = { \override Hairpin #'to-barline = ##f  }

alignAt =
#(define-music-function
     (parser location xOffset)
     (number?)
  (display xOffset)
   #{
      \once \override LyricText #'self-alignment-X = $xOffset
   #})

stemOn = { \revert Staff.Stem #'transparent  \revert Staff.Beam #'transparent \revert Staff.Flag #'transparent \revert Voice.Stem #'length }
stemOff = { \override Staff.Stem #'transparent = ##t \override Staff.Beam #'transparent = ##t \override Staff.Flag #'transparent = ##t \override Stem #'length = #0 }

bigNotes = { \override NoteHead #'font-size = #3 }
smallNotes = { \override NoteHead #'font-size = #-3 }
normalNotes = { \override NoteHead #'font-size = #0 }

bN = { \once \override NoteHead #'font-size = #3 }
sN = { \once \override NoteHead #'font-size = #-3 }
nN = { \once \override NoteHead #'font-size = #0 }

shOn = { \override NoteColumn #'force-hshift = #0.18 }
shOff = { \revert NoteColumn #'force-hshift }

midRest = { \override Voice.Rest #'staff-position = #0 }
shRest = { \revert Voice.Rest #'staff-position }

modernAccidentals = #(set-accidental-style 'modern)
modernCautionaryAccidentals = #(set-accidental-style 'modern-cautionary)
defaultAccidentals = #(set-accidental-style 'default)
pianoAccidentals = #(set-accidental-style 'piano)
voiceAccidentals = #(set-accidental-style 'voice)
modernVoiceAccidentals = #(set-accidental-style 'modern-voice)
modernVoiceCautionaryAccidentals = #(set-accidental-style 'modern-voice-cautionary)
pianoCautionaryAccidentals = #(set-accidental-style 'piano-cautionary)
neoModernAccidentals = #(set-accidental-style 'neo-modern)
neoModernCautionaryAccidentals = #(set-accidental-style 'neo-modern-cautionary)
dodecapohnicAccidentals = #(set-accidental-style 'dodecaphonic)
teachingAccidentals = #(set-accidental-style 'teaching)
noResetAccidentals = #(set-accidental-style 'no-reset)
forgetAccidentals = #(set-accidental-style 'forget)

go = {
\grace {s16 } 
}



chant = {
\key f \major
d'4 e'2 f'4 e' f' g' a'\breve \hideNotes a' a' a' \unHideNotes g'4( fis') g'2 \divisioMaxima
a'4 g'2 f'4 e' f' g' a'\breve \hideNotes a' a' \unHideNotes f'4 e'2 e' \divisioMaxima
f'4 g'2 f'4 e' f' g' a'\breve \hideNotes a' \unHideNotes g'4( fis') g'2 \break \divisioMaxima
a'4 g'2 f'?4 e' f' g' a'\breve \hideNotes a' \unHideNotes f'4 e'2 e' \divisioMaxima
f'4 g'2 f'4 e' f' g' a'\breve \hideNotes a' a' a' a' \unHideNotes g'4( fis') g'2 \divisioMaxima
a'4 g'2 f'?4 e' f' g' a'\breve \unHideNotes f'4 e'2 e' \break \divisioMaxima
f'4 g'2 f'4 e' f' g' a' a' f'8 e'2 d' \divisioMaxima
g'\breve \hideNotes g' g' \unHideNotes a'\breve \hideNotes a' \unHideNotes g'8 f'4 f'8 e'2( d'8[ cis')] d'2 
\hideNotes \grace {s16 } \finalis
}

verba = \lyricmode {
"Ha" -- "bár" "sí" -- "rod" "kö" -- "ve" \alignAt #0 "a" \alignAt #-0.596 "zsidók" \alignAt #-1.246 "által" \alignAt #-0.771 "megpecsétel" -- "te" -- "tett,"
"és" "leg" -- "szen" -- "tebb" "tes" -- "ted" \alignAt #-0.82 "katonák" \alignAt #-1.246 "által" \alignAt #0 "ő" -- "riz" -- "te" -- "tett,"
"még" -- "is" "föl" -- "tá" -- "mad" -- "tál" \alignAt #-0.9 "harmadnapon," \alignAt #-1.071 "Üdvö" -- "zí" -- "tőnk,"
"a" -- "ján" -- "dé" -- "koz" -- "ván" "é" -- \alignAt #-0.922 "letet" \alignAt #0 "a" "vi" -- "lág" -- "nak."
"A" -- "zért," "ó" "É" -- "let" -- "a" -- \alignAt #-0.5 "dó," \alignAt #0 "a" \alignAt #-0.67 "mennyei" \alignAt #-1.134 "Erők" \alignAt #-0.717 "kiáltják" "né" -- "ked:"
"Di" -- "cső" -- "ség" "a" "te" "fel" -- \alignAt #-0.968 "támadásod" -- "nak," "Krisz" -- "tus!"
"Di" -- "cső" -- "ség" "a" "te" "u" -- "ral" -- "ko" -- "dá" "sod" -- "nak!"
\alignAt #-0.769 "Dicsőség" \alignAt #0 "a" \alignAt #0 "te" \alignAt #-0.918 "gondviselésednek," \alignAt #-1.127 "egyetlen" "Em" -- "ber" -- "sze" -- "re" -- "tő!"
}


\score {
  \new Staff &lt;&lt;
    #(set-accidental-style 'modern-cautionary)
    \new Voice = "melody" \chant
    \new Lyrics = "one" \lyricsto melody \verba
  &gt;&gt;
  \layout {
    line-thickness = #(/ (ly:output-def-lookup $defaultpaper 'staff-space) 4)
    \context {
      \Staff
      \override BarLine  #'stencil = ##f
      \remove Time_signature_engraver
      \override Stem #'transparent = #hideStems
      \override Beam #'transparent = #hideStems
      \override Flag #'transparent = #hideStems
      \override StaffSymbol #'after-line-breaking = #staff-break-callback
      \override StaffSymbol #'thickness = #(/ (ly:output-def-lookup $defaultpaper 'staff-space) 3.0)
      \override StaffSymbol #'stencil = #staff-print
    }
    \context {
      \Voice
      \override Stem #'length = #0
   \override BreathingSign #'after-line-breaking = #set-breathe
    }
    \context {
      \Score
      \override  RehearsalMark #'self-alignment-X = #LEFT
      \override RehearsalMark #'break-align-symbols = #'(key-signature)
      \override  NonMusicalPaperColumn #'padding = #0.6
      skipBars= ##t
      barAlways = ##t
      lyricMelismaAlignment = #CENTER
    }
    \context {
      \Lyrics
      \override LyricSpace #'minimum-distance = #0.6
      \override LyricText #'font-size = #1.065
      \override LyricText #'font-name = #"TimesNewRoman"
      \override LyricText #'color = #(rgb-color 0 0 0)
      \override LyricHyphen #'color = #(rgb-color 0 0 0)
      \override LyricExtender #'color = #(rgb-color 0 0 0)
    }
  }
}

\header {
tagline = "" 
title = ""
}
</LilyPond>
  <EasyChant>Ha- bár sí- rod kö- ve a zsidók által megpecsétel- te-  tett,
l,4 t,2 d4  t,  d   r  m0                          r di r2   

és leg- szen- tebb tes- ted katonák által ő- riz- te- tett,
m4 r2   d4    t,   d    r   m0               d    t,2 t,   

még- is   föl- tá- mad- tál harmadnapon, Üdvö- zí-  tőnk,
d4   r2   d4   t,  d    r   m0                 r di r2   
/
a- ján- dé-  koz- ván   é- letet a vi- lág- nak.
m4 r2   d?4  t,   d     r  m0      d   t,2  t,  

A- zért, ó  É- let- a- dó, a mennyei Erők kiáltják né-  ked:
d4 r2    d4 t, d    r  m0                          r di r2  

Di- cső- ség  a  te fel- támadásod- nak, Krisz- tus!
m4  r2   d?4  t, d  r    m0         d    t,2    t,  
/
Di- cső- ség  a  te u- ral- ko- dá sod- nak!
d4  r2   d4   t, d  r  m    m   d8 t,2  l,  

Dicsőség a te gondviselésednek, egyetlen Em- ber- sze- re-         tő!
r0            m0                         r8  d4   d8   t,2 l,8 si, l,2

</EasyChant>
  <Clef>0</Clef>
  <Key>8</Key>
  <LyricsSpacing>0.6</LyricsSpacing>
  <TimeOption>0</TimeOption>
  <TimeText>N/A (cadenza)</TimeText>
  <StaffSizeOption>2</StaffSizeOption>
  <StaffSizeText>14</StaffSizeText>
  <LyricsSize>1.1</LyricsSize>
  <LyricsFontName>Times New Roman</LyricsFontName>
  <LyricsFontBold>False</LyricsFontBold>
  <LyricsFontItalic>False</LyricsFontItalic>
  <LyricsFontColor>Color [A=255, R=0, G=0, B=0]</LyricsFontColor>
  <LineSpacing>12</LineSpacing>
  <LineThickness>4</LineThickness>
  <Options>0,1,2,4,6,7,8</Options>
  <pages>1</pages>
  <page>1</page>
</item>

Ün. tropár, 4. hang

<?xml version="1.0" encoding="utf-16"?>
<item>
  <title>Tropár, 4. hang</title>
  <LilyPond>\version "2.18.2"

\include "gregorian.ly"

\paper {
  paper-width = 123.0136\mm
  paper-height = 184.6262\mm
  top-margin = 0\mm
  bottom-margin = 0\mm
  left-margin = 0\mm
  right-margin = 2\mm
  ragged-bottom = ##t
  print-page-number = ##f
  page-breaking = #ly:minimal-breaking
  system-system-spacing =
    #'((basic-distance . 12)
       (minimum-distance . 12)
       (padding . 0)
       (stretchability . 0))
%  page-breaking-system-system-spacing = #'((padding . 2))
  markup-system-spacing = #'((padding . 112.83))
  top-markup-spacing = #'((basic-distance . 0) (minimum-distance . 0) (padding . 0))
}

#(set-global-staff-size 14)

#(define currBSign 0)
#(define hideStems #t)

#(define (set-breathe bsign)
 (begin
  (set! currBSign bsign)
 (let* ((columnOfBSign (ly:grob-parent (ly:grob-parent (ly:grob-parent currBSign X) X) X))
   (system (ly:grob-parent columnOfBSign X))   
   (columns (ly:grob-object system 'columns))
   (lastColumn (ly:grob-array-ref columns (- (ly:grob-array-length columns) 1)))
  )
  (if (eqv? columnOfBSign lastColumn)
   (ly:grob-set-property! currBSign 'extra-offset '(0.89 . 0))
  )
 )
 )
)

#(define (staff-print staffgrob)
 (let* ((stencil (ly:staff-symbol::print staffgrob))
   (Xextent (ly:stencil-extent stencil X))
   (Yextent (ly:stencil-extent stencil Y))
            (X0 (car Xextent))
   (X1 (cdr Xextent))
   (X1sc (+ X1 0.89))
   )
   (ly:make-stencil (ly:stencil-expr (ly:stencil-scale stencil (/ (- X1sc X0) (- X1 X0)) 1.0))
                         (cons X0 X1sc)
                         Yextent
   )
 )
)

#(define (staff-break-callback staffgrob)
    (let*
  (
    (system (ly:grob-parent (ly:grob-parent staffgrob X) X))
    (columns (ly:grob-object system 'columns))
    (columns-count (ly:grob-array-length columns))
    (goI 1)
    (goJ 1)
  )
  (if (ly:grob? currBSign)
   (let* ((columnOfBSign (ly:grob-parent (ly:grob-parent (ly:grob-parent currBSign X) X) X))
      (lastColumn (ly:grob-array-ref columns (- columns-count 1)))
     )
     (if (eqv? columnOfBSign lastColumn)
      (ly:grob-set-property! currBSign 'extra-offset '(0.89 . 0))
      (set! currBSign 0)
     )
     (set! currBSign 0)
   )
  )
  (let loopI  ((i 0)
     )
       (if (&lt; i columns-count)
      (loopI (let* ((column (ly:grob-array-ref columns i))
          (columnname (grob-name column))
         )
         (if (and (eqv? columnname 'PaperColumn) (ly:grob-array? (ly:grob-object column 'elements)))
          (let*
           (
           (elems (ly:grob-object column 'elements))
           (elems-count (ly:grob-array-length elems))
           )
           (let loopJ ((j 0)
                )
                (if (&lt; j elems-count) 
               (loopJ (let* ((grob (ly:grob-array-ref elems j))
                   (grobname (grob-name grob))
                  )
                  (if (eqv? grobname 'NoteColumn)
                   (begin
                    (show-notes grob)
                    (set! j elems-count)
                   )
                  )
                  (1+ j)
                 )
               )
              )
           )
           (set! i columns-count)
          )
         )
         (1+ i)
        )
      )
     )
  )
 )
)

#(define (grob-name grob)
   (if (ly:grob? grob)
       (assoc-ref (ly:grob-property grob 'meta) 'name)
       #f))

#(define (show-flag flag)
 (ly:grob-set-property! flag 'transparent hideStems)
)
   
#(define (show-stem stem)
 (let*  ((beam (ly:grob-object stem 'beam))
   )
   (begin
    (ly:grob-set-property! stem 'transparent hideStems)
    (if (ly:grob? beam)(ly:grob-set-property! beam 'transparent hideStems))
   )
 )
)

#(define (show-note notehead)
 (let* ((stem (ly:grob-object (ly:grob-parent notehead X) 'stem))
        (beam (ly:grob-object stem 'beam))
     (flag (ly:grob-object notehead 'flag))
        (acci (ly:grob-object notehead 'accidental-grob))
        (dots (ly:grob-object notehead 'dot)))
 (begin
  (ly:grob-set-property! notehead 'transparent #f)
  (if (ly:grob? dots)(ly:grob-set-property! dots 'transparent #f))
  (if (ly:grob? acci)(ly:grob-set-property! acci 'transparent #f))
  (ly:grob-set-property! stem 'transparent hideStems)
  (if (ly:grob? beam)(ly:grob-set-property! beam 'transparent hideStems))
  (if (ly:grob? flag)(ly:grob-set-property! flag 'transparent hideStems))
 ))
)

#(define (show-notes note-column) 
 (begin 
  (let* ((elems (ly:grob-object note-column 'elements))
    (elems-count (ly:grob-array-length elems))
     ) 
       (let loop ((staff-pos-sum 0)
      (i 0)
        )
        (if (&lt; i elems-count)
       (loop (let* ((grob (ly:grob-array-ref elems i))
           (grobname (grob-name grob))
          )
          (if (eqv? grobname 'NoteHead)
           (show-note grob)
           (if (eqv? grobname 'Stem)
            (show-stem grob)
            (if (eqv? grobname 'Flag)
             (show-flag grob)
            )
           )
          )
         )
        (1+ i)
       )
      )
    )
  )
 )
)




leftAlign = {  \once \override LyricText #'self-alignment-X = #LEFT }
centerAlign = {   \once \override LyricText #'self-alignment-X = #CENTER }
rightAlign = {   \once \override LyricText #'self-alignment-X = #1 }
hairpinToBarline = { \override Hairpin #'to-barline = ##t  }
hairpinToNote = { \override Hairpin #'to-barline = ##f  }

alignAt =
#(define-music-function
     (parser location xOffset)
     (number?)
  (display xOffset)
   #{
      \once \override LyricText #'self-alignment-X = $xOffset
   #})

stemOn = { \revert Staff.Stem #'transparent  \revert Staff.Beam #'transparent \revert Staff.Flag #'transparent \revert Voice.Stem #'length }
stemOff = { \override Staff.Stem #'transparent = ##t \override Staff.Beam #'transparent = ##t \override Staff.Flag #'transparent = ##t \override Stem #'length = #0 }

bigNotes = { \override NoteHead #'font-size = #3 }
smallNotes = { \override NoteHead #'font-size = #-3 }
normalNotes = { \override NoteHead #'font-size = #0 }

bN = { \once \override NoteHead #'font-size = #3 }
sN = { \once \override NoteHead #'font-size = #-3 }
nN = { \once \override NoteHead #'font-size = #0 }

shOn = { \override NoteColumn #'force-hshift = #0.18 }
shOff = { \revert NoteColumn #'force-hshift }

midRest = { \override Voice.Rest #'staff-position = #0 }
shRest = { \revert Voice.Rest #'staff-position }

modernAccidentals = #(set-accidental-style 'modern)
modernCautionaryAccidentals = #(set-accidental-style 'modern-cautionary)
defaultAccidentals = #(set-accidental-style 'default)
pianoAccidentals = #(set-accidental-style 'piano)
voiceAccidentals = #(set-accidental-style 'voice)
modernVoiceAccidentals = #(set-accidental-style 'modern-voice)
modernVoiceCautionaryAccidentals = #(set-accidental-style 'modern-voice-cautionary)
pianoCautionaryAccidentals = #(set-accidental-style 'piano-cautionary)
neoModernAccidentals = #(set-accidental-style 'neo-modern)
neoModernCautionaryAccidentals = #(set-accidental-style 'neo-modern-cautionary)
dodecapohnicAccidentals = #(set-accidental-style 'dodecaphonic)
teachingAccidentals = #(set-accidental-style 'teaching)
noResetAccidentals = #(set-accidental-style 'no-reset)
forgetAccidentals = #(set-accidental-style 'forget)

go = {
\grace {s16 } 
}



chant = {
\key f \major
f'2 g' a'4 bes' c''\breve \hideNotes c'' c'' c'' \unHideNotes d''4 c''2 bes' a' \divisioMaxima
bes'2 g'4 bes'\breve \hideNotes bes' bes' \unHideNotes c''4 d''\breve \unHideNotes c''4.( bes'4) a'2 \divisioMaxima
c''\breve \hideNotes c'' c'' c'' c'' \unHideNotes d''4 c''2 bes' a' \divisioMaxima
bes'2 g'4 bes'\breve \hideNotes bes' bes' bes' \unHideNotes c''4 d'' d'' c''4.( bes'4) a'2 \divisioMaxima
c''\breve \hideNotes c'' c'' c'' c'' c'' \unHideNotes d''4 c''2 bes' a' \divisioMaxima
c''\breve \hideNotes c'' c'' c'' \break \unHideNotes a'4 bes' a'2 g' f' 
\hideNotes \grace {s16 } \finalis
}

verba = \lyricmode {
"Ma" "van" "el" -- "ső" \alignAt #-0.964 "jelensége" \alignAt #-1.482 "az" \alignAt #-1.285 "Isten" \alignAt #-0.837 "boldogító" "ke" -- "gyel" -- "mé" -- "nek,"
"és" "hir" -- \alignAt #-0.772 "detése" \alignAt #-1.482 "az" \alignAt #-1.196 "embe" -- "rek" \alignAt #-1.117 "üdvössé" -- "gé" -- "nek:"
\alignAt #-1.482 "az" \alignAt #-1.285 "Isten" \alignAt #-0.868 "hajlékában" \alignAt #-0.542 "nyilván" \alignAt #-0.339 "meg" -- "je" -- "len" "a" "Szűz,"
"és" "Krisz" -- \alignAt #-0.839 "tus" \alignAt #-0.783 "megjelenését" \alignAt #-0.815 "mindenkinek" \alignAt #0 "e" -- "lő" -- "re" "hir" -- "de" -- "ti."
\alignAt #-0.271 "Miért" \alignAt #0 "is" \alignAt #0 "a" \alignAt #-0.3 "Szent" \alignAt #-0.525 "Szűznek" \alignAt #-1.177 "öröm" -- "mel" "ki" -- "ált" -- "juk:"
\alignAt #-1.034 "Üdvözlégy," \alignAt #0 "a" \alignAt #-0.765 "Teremtő" \alignAt #-1.052 "üdvgondozásának" "tel" -- "je" -- "sü" -- "lé" -- "se!"
}


\score {
  \new Staff &lt;&lt;
    #(set-accidental-style 'modern-cautionary)
    \new Voice = "melody" \chant
    \new Lyrics = "one" \lyricsto melody \verba
  &gt;&gt;
  \layout {
    line-thickness = #(/ (ly:output-def-lookup $defaultpaper 'staff-space) 4)
    \context {
      \Staff
      \override BarLine  #'stencil = ##f
      \remove Time_signature_engraver
      \override Stem #'transparent = #hideStems
      \override Beam #'transparent = #hideStems
      \override Flag #'transparent = #hideStems
      \override StaffSymbol #'after-line-breaking = #staff-break-callback
      \override StaffSymbol #'thickness = #(/ (ly:output-def-lookup $defaultpaper 'staff-space) 3.0)
      \override StaffSymbol #'stencil = #staff-print
    }
    \context {
      \Voice
      \override Stem #'length = #0
   \override BreathingSign #'after-line-breaking = #set-breathe
    }
    \context {
      \Score
      \override  RehearsalMark #'self-alignment-X = #LEFT
      \override RehearsalMark #'break-align-symbols = #'(key-signature)
      \override  NonMusicalPaperColumn #'padding = #0.8
      skipBars= ##t
      barAlways = ##t
      lyricMelismaAlignment = #CENTER
    }
    \context {
      \Lyrics
      \override LyricSpace #'minimum-distance = #0.8
      \override LyricText #'font-size = #1.065
      \override LyricText #'font-name = #"TimesNewRoman"
      \override LyricText #'color = #(rgb-color 0 0 0)
      \override LyricHyphen #'color = #(rgb-color 0 0 0)
      \override LyricExtender #'color = #(rgb-color 0 0 0)
    }
  }
}

\header {
tagline = "" 
title = ""
}
</LilyPond>
  <EasyChant>Ma van el- ső jelensége az Isten boldogító ke- gyel- mé- nek,
d2 r   m4  f  s0                           l   s2    f   m   

és hir- detése az embe- rek üdvössé- gé-    nek:
f2 r4   f0              s   l0       s4. f4 m2  

az Isten hajlékában nyilván meg- je- len a Szűz,
s0                               l4  s2  f m    

és Krisz- tus megjelenését mindenkinek e- lő- re hir- de-    ti.
f2 r4     f0                              s   l  l    s4. f4 m2 

Miért is a Szent Szűznek öröm- mel ki- ált- juk:
s0                             l4  s2  f    m   

Üdvözlégy, a Teremtő üdvgondozásának tel- je- sü- lé- se!
s0                                 / m4   f   m2  r   d  

</EasyChant>
  <Clef>0</Clef>
  <Key>8</Key>
  <LyricsSpacing>0.8</LyricsSpacing>
  <TimeOption>0</TimeOption>
  <TimeText>N/A (cadenza)</TimeText>
  <StaffSizeOption>2</StaffSizeOption>
  <StaffSizeText>14</StaffSizeText>
  <LyricsSize>1.1</LyricsSize>
  <LyricsFontName>Times New Roman</LyricsFontName>
  <LyricsFontBold>False</LyricsFontBold>
  <LyricsFontItalic>False</LyricsFontItalic>
  <LyricsFontColor>Color [A=255, R=0, G=0, B=0]</LyricsFontColor>
  <LineSpacing>12</LineSpacing>
  <LineThickness>4</LineThickness>
  <Options>0,1,2,4,6,7,8</Options>
  <pages>2</pages>
  <page>1</page>
</item> <?xml version="1.0" encoding="utf-16"?>
<item>
  <title>Tropár, 4. hang</title>
  <LilyPond>\version "2.18.2"

\include "gregorian.ly"

\paper {
  paper-width = 123.0136\mm
  paper-height = 184.6262\mm
  top-margin = 0\mm
  bottom-margin = 0\mm
  left-margin = 0\mm
  right-margin = 2\mm
  ragged-bottom = ##t
  print-page-number = ##f
  page-breaking = #ly:minimal-breaking
  system-system-spacing =
    #'((basic-distance . 12)
       (minimum-distance . 12)
       (padding . 0)
       (stretchability . 0))
%  page-breaking-system-system-spacing = #'((padding . 2))
  markup-system-spacing = #'((padding . 112.83))
  top-markup-spacing = #'((basic-distance . 0) (minimum-distance . 0) (padding . 0))
}

#(set-global-staff-size 14)

#(define currBSign 0)
#(define hideStems #t)

#(define (set-breathe bsign)
 (begin
  (set! currBSign bsign)
 (let* ((columnOfBSign (ly:grob-parent (ly:grob-parent (ly:grob-parent currBSign X) X) X))
   (system (ly:grob-parent columnOfBSign X))   
   (columns (ly:grob-object system 'columns))
   (lastColumn (ly:grob-array-ref columns (- (ly:grob-array-length columns) 1)))
  )
  (if (eqv? columnOfBSign lastColumn)
   (ly:grob-set-property! currBSign 'extra-offset '(0.89 . 0))
  )
 )
 )
)

#(define (staff-print staffgrob)
 (let* ((stencil (ly:staff-symbol::print staffgrob))
   (Xextent (ly:stencil-extent stencil X))
   (Yextent (ly:stencil-extent stencil Y))
            (X0 (car Xextent))
   (X1 (cdr Xextent))
   (X1sc (+ X1 0.89))
   )
   (ly:make-stencil (ly:stencil-expr (ly:stencil-scale stencil (/ (- X1sc X0) (- X1 X0)) 1.0))
                         (cons X0 X1sc)
                         Yextent
   )
 )
)

#(define (staff-break-callback staffgrob)
    (let*
  (
    (system (ly:grob-parent (ly:grob-parent staffgrob X) X))
    (columns (ly:grob-object system 'columns))
    (columns-count (ly:grob-array-length columns))
    (goI 1)
    (goJ 1)
  )
  (if (ly:grob? currBSign)
   (let* ((columnOfBSign (ly:grob-parent (ly:grob-parent (ly:grob-parent currBSign X) X) X))
      (lastColumn (ly:grob-array-ref columns (- columns-count 1)))
     )
     (if (eqv? columnOfBSign lastColumn)
      (ly:grob-set-property! currBSign 'extra-offset '(0.89 . 0))
      (set! currBSign 0)
     )
     (set! currBSign 0)
   )
  )
  (let loopI  ((i 0)
     )
       (if (&lt; i columns-count)
      (loopI (let* ((column (ly:grob-array-ref columns i))
          (columnname (grob-name column))
         )
         (if (and (eqv? columnname 'PaperColumn) (ly:grob-array? (ly:grob-object column 'elements)))
          (let*
           (
           (elems (ly:grob-object column 'elements))
           (elems-count (ly:grob-array-length elems))
           )
           (let loopJ ((j 0)
                )
                (if (&lt; j elems-count) 
               (loopJ (let* ((grob (ly:grob-array-ref elems j))
                   (grobname (grob-name grob))
                  )
                  (if (eqv? grobname 'NoteColumn)
                   (begin
                    (show-notes grob)
                    (set! j elems-count)
                   )
                  )
                  (1+ j)
                 )
               )
              )
           )
           (set! i columns-count)
          )
         )
         (1+ i)
        )
      )
     )
  )
 )
)

#(define (grob-name grob)
   (if (ly:grob? grob)
       (assoc-ref (ly:grob-property grob 'meta) 'name)
       #f))

#(define (show-flag flag)
 (ly:grob-set-property! flag 'transparent hideStems)
)
   
#(define (show-stem stem)
 (let*  ((beam (ly:grob-object stem 'beam))
   )
   (begin
    (ly:grob-set-property! stem 'transparent hideStems)
    (if (ly:grob? beam)(ly:grob-set-property! beam 'transparent hideStems))
   )
 )
)

#(define (show-note notehead)
 (let* ((stem (ly:grob-object (ly:grob-parent notehead X) 'stem))
        (beam (ly:grob-object stem 'beam))
     (flag (ly:grob-object notehead 'flag))
        (acci (ly:grob-object notehead 'accidental-grob))
        (dots (ly:grob-object notehead 'dot)))
 (begin
  (ly:grob-set-property! notehead 'transparent #f)
  (if (ly:grob? dots)(ly:grob-set-property! dots 'transparent #f))
  (if (ly:grob? acci)(ly:grob-set-property! acci 'transparent #f))
  (ly:grob-set-property! stem 'transparent hideStems)
  (if (ly:grob? beam)(ly:grob-set-property! beam 'transparent hideStems))
  (if (ly:grob? flag)(ly:grob-set-property! flag 'transparent hideStems))
 ))
)

#(define (show-notes note-column) 
 (begin 
  (let* ((elems (ly:grob-object note-column 'elements))
    (elems-count (ly:grob-array-length elems))
     ) 
       (let loop ((staff-pos-sum 0)
      (i 0)
        )
        (if (&lt; i elems-count)
       (loop (let* ((grob (ly:grob-array-ref elems i))
           (grobname (grob-name grob))
          )
          (if (eqv? grobname 'NoteHead)
           (show-note grob)
           (if (eqv? grobname 'Stem)
            (show-stem grob)
            (if (eqv? grobname 'Flag)
             (show-flag grob)
            )
           )
          )
         )
        (1+ i)
       )
      )
    )
  )
 )
)




leftAlign = {  \once \override LyricText #'self-alignment-X = #LEFT }
centerAlign = {   \once \override LyricText #'self-alignment-X = #CENTER }
rightAlign = {   \once \override LyricText #'self-alignment-X = #1 }
hairpinToBarline = { \override Hairpin #'to-barline = ##t  }
hairpinToNote = { \override Hairpin #'to-barline = ##f  }

alignAt =
#(define-music-function
     (parser location xOffset)
     (number?)
  (display xOffset)
   #{
      \once \override LyricText #'self-alignment-X = $xOffset
   #})

stemOn = { \revert Staff.Stem #'transparent  \revert Staff.Beam #'transparent \revert Staff.Flag #'transparent \revert Voice.Stem #'length }
stemOff = { \override Staff.Stem #'transparent = ##t \override Staff.Beam #'transparent = ##t \override Staff.Flag #'transparent = ##t \override Stem #'length = #0 }

bigNotes = { \override NoteHead #'font-size = #3 }
smallNotes = { \override NoteHead #'font-size = #-3 }
normalNotes = { \override NoteHead #'font-size = #0 }

bN = { \once \override NoteHead #'font-size = #3 }
sN = { \once \override NoteHead #'font-size = #-3 }
nN = { \once \override NoteHead #'font-size = #0 }

shOn = { \override NoteColumn #'force-hshift = #0.18 }
shOff = { \revert NoteColumn #'force-hshift }

midRest = { \override Voice.Rest #'staff-position = #0 }
shRest = { \revert Voice.Rest #'staff-position }

modernAccidentals = #(set-accidental-style 'modern)
modernCautionaryAccidentals = #(set-accidental-style 'modern-cautionary)
defaultAccidentals = #(set-accidental-style 'default)
pianoAccidentals = #(set-accidental-style 'piano)
voiceAccidentals = #(set-accidental-style 'voice)
modernVoiceAccidentals = #(set-accidental-style 'modern-voice)
modernVoiceCautionaryAccidentals = #(set-accidental-style 'modern-voice-cautionary)
pianoCautionaryAccidentals = #(set-accidental-style 'piano-cautionary)
neoModernAccidentals = #(set-accidental-style 'neo-modern)
neoModernCautionaryAccidentals = #(set-accidental-style 'neo-modern-cautionary)
dodecapohnicAccidentals = #(set-accidental-style 'dodecaphonic)
teachingAccidentals = #(set-accidental-style 'teaching)
noResetAccidentals = #(set-accidental-style 'no-reset)
forgetAccidentals = #(set-accidental-style 'forget)

go = {
\grace {s16 } 
}



chant = {
\key f \major
f'2 g' a'4 bes' c''\breve \hideNotes c'' c'' c'' \unHideNotes d''4 c''2 bes' a' \divisioMaxima
bes'2 g'4 bes'\breve \hideNotes bes' bes' \unHideNotes c''4 d''\breve \unHideNotes c''4.( bes'4) a'2 \divisioMaxima
c''\breve \hideNotes c'' c'' c'' c'' \unHideNotes d''4 c''2 bes' a' \divisioMaxima
bes'2 g'4 bes'\breve \hideNotes bes' bes' bes' \unHideNotes c''4 d'' d'' c''4.( bes'4) a'2 \divisioMaxima
c''\breve \hideNotes c'' c'' c'' c'' c'' \unHideNotes d''4 c''2 bes' a' \divisioMaxima
c''\breve \hideNotes c'' c'' c'' \break \unHideNotes a'4 bes' a'2 g' f' 
\hideNotes \grace {s16 } \finalis
}

verba = \lyricmode {
"Ma" "van" "el" -- "ső" \alignAt #-0.964 "jelensége" \alignAt #-1.482 "az" \alignAt #-1.285 "Isten" \alignAt #-0.837 "boldogító" "ke" -- "gyel" -- "mé" -- "nek,"
"és" "hir" -- \alignAt #-0.772 "detése" \alignAt #-1.482 "az" \alignAt #-1.196 "embe" -- "rek" \alignAt #-1.117 "üdvössé" -- "gé" -- "nek:"
\alignAt #-1.482 "az" \alignAt #-1.285 "Isten" \alignAt #-0.868 "hajlékában" \alignAt #-0.542 "nyilván" \alignAt #-0.339 "meg" -- "je" -- "len" "a" "Szűz,"
"és" "Krisz" -- \alignAt #-0.839 "tus" \alignAt #-0.783 "megjelenését" \alignAt #-0.815 "mindenkinek" \alignAt #0 "e" -- "lő" -- "re" "hir" -- "de" -- "ti."
\alignAt #-0.271 "Miért" \alignAt #0 "is" \alignAt #0 "a" \alignAt #-0.3 "Szent" \alignAt #-0.525 "Szűznek" \alignAt #-1.177 "öröm" -- "mel" "ki" -- "ált" -- "juk:"
\alignAt #-1.034 "Üdvözlégy," \alignAt #0 "a" \alignAt #-0.765 "Teremtő" \alignAt #-1.052 "üdvgondozásának" "tel" -- "je" -- "sü" -- "lé" -- "se!"
}


\score {
  \new Staff &lt;&lt;
    #(set-accidental-style 'modern-cautionary)
    \new Voice = "melody" \chant
    \new Lyrics = "one" \lyricsto melody \verba
  &gt;&gt;
  \layout {
    line-thickness = #(/ (ly:output-def-lookup $defaultpaper 'staff-space) 4)
    \context {
      \Staff
      \override BarLine  #'stencil = ##f
      \remove Time_signature_engraver
      \override Stem #'transparent = #hideStems
      \override Beam #'transparent = #hideStems
      \override Flag #'transparent = #hideStems
      \override StaffSymbol #'after-line-breaking = #staff-break-callback
      \override StaffSymbol #'thickness = #(/ (ly:output-def-lookup $defaultpaper 'staff-space) 3.0)
      \override StaffSymbol #'stencil = #staff-print
    }
    \context {
      \Voice
      \override Stem #'length = #0
   \override BreathingSign #'after-line-breaking = #set-breathe
    }
    \context {
      \Score
      \override  RehearsalMark #'self-alignment-X = #LEFT
      \override RehearsalMark #'break-align-symbols = #'(key-signature)
      \override  NonMusicalPaperColumn #'padding = #0.8
      skipBars= ##t
      barAlways = ##t
      lyricMelismaAlignment = #CENTER
    }
    \context {
      \Lyrics
      \override LyricSpace #'minimum-distance = #0.8
      \override LyricText #'font-size = #1.065
      \override LyricText #'font-name = #"TimesNewRoman"
      \override LyricText #'color = #(rgb-color 0 0 0)
      \override LyricHyphen #'color = #(rgb-color 0 0 0)
      \override LyricExtender #'color = #(rgb-color 0 0 0)
    }
  }
}

\header {
tagline = "" 
title = ""
}
</LilyPond>
  <EasyChant>Ma van el- ső jelensége az Isten boldogító ke- gyel- mé- nek,
d2 r   m4  f  s0                           l   s2    f   m   

és hir- detése az embe- rek üdvössé- gé-    nek:
f2 r4   f0              s   l0       s4. f4 m2  

az Isten hajlékában nyilván meg- je- len a Szűz,
s0                               l4  s2  f m    

és Krisz- tus megjelenését mindenkinek e- lő- re hir- de-    ti.
f2 r4     f0                              s   l  l    s4. f4 m2 

Miért is a Szent Szűznek öröm- mel ki- ált- juk:
s0                             l4  s2  f    m   

Üdvözlégy, a Teremtő üdvgondozásának tel- je- sü- lé- se!
s0                                 / m4   f   m2  r   d  

</EasyChant>
  <Clef>0</Clef>
  <Key>8</Key>
  <LyricsSpacing>0.8</LyricsSpacing>
  <TimeOption>0</TimeOption>
  <TimeText>N/A (cadenza)</TimeText>
  <StaffSizeOption>2</StaffSizeOption>
  <StaffSizeText>14</StaffSizeText>
  <LyricsSize>1.1</LyricsSize>
  <LyricsFontName>Times New Roman</LyricsFontName>
  <LyricsFontBold>False</LyricsFontBold>
  <LyricsFontItalic>False</LyricsFontItalic>
  <LyricsFontColor>Color [A=255, R=0, G=0, B=0]</LyricsFontColor>
  <LineSpacing>12</LineSpacing>
  <LineThickness>4</LineThickness>
  <Options>0,1,2,4,6,7,8</Options>
  <pages>2</pages>
  <page>2</page>
</item>

Dicsőség... 1. hang, tropár-dallam

<?xml version="1.0" encoding="utf-16"?>
<item>
  <title>Dicsőség, 1. hang</title>
  <LilyPond>\version "2.18.2"

\include "gregorian.ly"

\paper {
  paper-width = 123.0136\mm
  paper-height = 184.6262\mm
  top-margin = 0\mm
  bottom-margin = 0\mm
  left-margin = 0\mm
  right-margin = 2\mm
  ragged-bottom = ##t
  print-page-number = ##f
  page-breaking = #ly:minimal-breaking
  system-system-spacing =
    #'((basic-distance . 12)
       (minimum-distance . 12)
       (padding . 0)
       (stretchability . 0))
%  page-breaking-system-system-spacing = #'((padding . 2))
  markup-system-spacing = #'((padding . 85.4))
  top-markup-spacing = #'((basic-distance . 0) (minimum-distance . 0) (padding . 0))
}

#(set-global-staff-size 14)

#(define currBSign 0)
#(define hideStems #t)

#(define (set-breathe bsign)
 (begin
  (set! currBSign bsign)
 (let* ((columnOfBSign (ly:grob-parent (ly:grob-parent (ly:grob-parent currBSign X) X) X))
   (system (ly:grob-parent columnOfBSign X))   
   (columns (ly:grob-object system 'columns))
   (lastColumn (ly:grob-array-ref columns (- (ly:grob-array-length columns) 1)))
  )
  (if (eqv? columnOfBSign lastColumn)
   (ly:grob-set-property! currBSign 'extra-offset '(0.89 . 0))
  )
 )
 )
)

#(define (staff-print staffgrob)
 (let* ((stencil (ly:staff-symbol::print staffgrob))
   (Xextent (ly:stencil-extent stencil X))
   (Yextent (ly:stencil-extent stencil Y))
            (X0 (car Xextent))
   (X1 (cdr Xextent))
   (X1sc (+ X1 0.89))
   )
   (ly:make-stencil (ly:stencil-expr (ly:stencil-scale stencil (/ (- X1sc X0) (- X1 X0)) 1.0))
                         (cons X0 X1sc)
                         Yextent
   )
 )
)

#(define (staff-break-callback staffgrob)
    (let*
  (
    (system (ly:grob-parent (ly:grob-parent staffgrob X) X))
    (columns (ly:grob-object system 'columns))
    (columns-count (ly:grob-array-length columns))
    (goI 1)
    (goJ 1)
  )
  (if (ly:grob? currBSign)
   (let* ((columnOfBSign (ly:grob-parent (ly:grob-parent (ly:grob-parent currBSign X) X) X))
      (lastColumn (ly:grob-array-ref columns (- columns-count 1)))
     )
     (if (eqv? columnOfBSign lastColumn)
      (ly:grob-set-property! currBSign 'extra-offset '(0.89 . 0))
      (set! currBSign 0)
     )
     (set! currBSign 0)
   )
  )
  (let loopI  ((i 0)
     )
       (if (&lt; i columns-count)
      (loopI (let* ((column (ly:grob-array-ref columns i))
          (columnname (grob-name column))
         )
         (if (and (eqv? columnname 'PaperColumn) (ly:grob-array? (ly:grob-object column 'elements)))
          (let*
           (
           (elems (ly:grob-object column 'elements))
           (elems-count (ly:grob-array-length elems))
           )
           (let loopJ ((j 0)
                )
                (if (&lt; j elems-count) 
               (loopJ (let* ((grob (ly:grob-array-ref elems j))
                   (grobname (grob-name grob))
                  )
                  (if (eqv? grobname 'NoteColumn)
                   (begin
                    (show-notes grob)
                    (set! j elems-count)
                   )
                  )
                  (1+ j)
                 )
               )
              )
           )
           (set! i columns-count)
          )
         )
         (1+ i)
        )
      )
     )
  )
 )
)

#(define (grob-name grob)
   (if (ly:grob? grob)
       (assoc-ref (ly:grob-property grob 'meta) 'name)
       #f))

#(define (show-flag flag)
 (ly:grob-set-property! flag 'transparent hideStems)
)
   
#(define (show-stem stem)
 (let*  ((beam (ly:grob-object stem 'beam))
   )
   (begin
    (ly:grob-set-property! stem 'transparent hideStems)
    (if (ly:grob? beam)(ly:grob-set-property! beam 'transparent hideStems))
   )
 )
)

#(define (show-note notehead)
 (let* ((stem (ly:grob-object (ly:grob-parent notehead X) 'stem))
        (beam (ly:grob-object stem 'beam))
     (flag (ly:grob-object notehead 'flag))
        (acci (ly:grob-object notehead 'accidental-grob))
        (dots (ly:grob-object notehead 'dot)))
 (begin
  (ly:grob-set-property! notehead 'transparent #f)
  (if (ly:grob? dots)(ly:grob-set-property! dots 'transparent #f))
  (if (ly:grob? acci)(ly:grob-set-property! acci 'transparent #f))
  (ly:grob-set-property! stem 'transparent hideStems)
  (if (ly:grob? beam)(ly:grob-set-property! beam 'transparent hideStems))
  (if (ly:grob? flag)(ly:grob-set-property! flag 'transparent hideStems))
 ))
)

#(define (show-notes note-column) 
 (begin 
  (let* ((elems (ly:grob-object note-column 'elements))
    (elems-count (ly:grob-array-length elems))
     ) 
       (let loop ((staff-pos-sum 0)
      (i 0)
        )
        (if (&lt; i elems-count)
       (loop (let* ((grob (ly:grob-array-ref elems i))
           (grobname (grob-name grob))
          )
          (if (eqv? grobname 'NoteHead)
           (show-note grob)
           (if (eqv? grobname 'Stem)
            (show-stem grob)
            (if (eqv? grobname 'Flag)
             (show-flag grob)
            )
           )
          )
         )
        (1+ i)
       )
      )
    )
  )
 )
)




leftAlign = {  \once \override LyricText #'self-alignment-X = #LEFT }
centerAlign = {   \once \override LyricText #'self-alignment-X = #CENTER }
rightAlign = {   \once \override LyricText #'self-alignment-X = #1 }
hairpinToBarline = { \override Hairpin #'to-barline = ##t  }
hairpinToNote = { \override Hairpin #'to-barline = ##f  }

alignAt =
#(define-music-function
     (parser location xOffset)
     (number?)
  (display xOffset)
   #{
      \once \override LyricText #'self-alignment-X = $xOffset
   #})

stemOn = { \revert Staff.Stem #'transparent  \revert Staff.Beam #'transparent \revert Staff.Flag #'transparent \revert Voice.Stem #'length }
stemOff = { \override Staff.Stem #'transparent = ##t \override Staff.Beam #'transparent = ##t \override Staff.Flag #'transparent = ##t \override Stem #'length = #0 }

bigNotes = { \override NoteHead #'font-size = #3 }
smallNotes = { \override NoteHead #'font-size = #-3 }
normalNotes = { \override NoteHead #'font-size = #0 }

bN = { \once \override NoteHead #'font-size = #3 }
sN = { \once \override NoteHead #'font-size = #-3 }
nN = { \once \override NoteHead #'font-size = #0 }

shOn = { \override NoteColumn #'force-hshift = #0.18 }
shOff = { \revert NoteColumn #'force-hshift }

midRest = { \override Voice.Rest #'staff-position = #0 }
shRest = { \revert Voice.Rest #'staff-position }

modernAccidentals = #(set-accidental-style 'modern)
modernCautionaryAccidentals = #(set-accidental-style 'modern-cautionary)
defaultAccidentals = #(set-accidental-style 'default)
pianoAccidentals = #(set-accidental-style 'piano)
voiceAccidentals = #(set-accidental-style 'voice)
modernVoiceAccidentals = #(set-accidental-style 'modern-voice)
modernVoiceCautionaryAccidentals = #(set-accidental-style 'modern-voice-cautionary)
pianoCautionaryAccidentals = #(set-accidental-style 'piano-cautionary)
neoModernAccidentals = #(set-accidental-style 'neo-modern)
neoModernCautionaryAccidentals = #(set-accidental-style 'neo-modern-cautionary)
dodecapohnicAccidentals = #(set-accidental-style 'dodecaphonic)
teachingAccidentals = #(set-accidental-style 'teaching)
noResetAccidentals = #(set-accidental-style 'no-reset)
forgetAccidentals = #(set-accidental-style 'forget)

go = {
\grace {s16 } 
}



chant = {
\key f \major
d'4 e'2 f'4 e' f' g' a'\breve \hideNotes a' a' \unHideNotes g'4 f' f' e'2( d'4 cis') d'2 
\hideNotes \grace {s16 } \finalis
}

verba = \lyricmode {
"Di" -- "cső" -- "ség" "az" "A" -- "tyá" -- \alignAt #-0.604 "nak" \alignAt #0 "és" \alignAt #-0.643 "Fiúnak" "és" "Szent" -- "lé" -- "lek" -- "nek."
}


\score {
  \new Staff &lt;&lt;
    #(set-accidental-style 'modern-cautionary)
    \new Voice = "melody" \chant
    \new Lyrics = "one" \lyricsto melody \verba
  &gt;&gt;
  \layout {
    line-thickness = #(/ (ly:output-def-lookup $defaultpaper 'staff-space) 4)
    \context {
      \Staff
      \override BarLine  #'stencil = ##f
      \remove Time_signature_engraver
      \override Stem #'transparent = #hideStems
      \override Beam #'transparent = #hideStems
      \override Flag #'transparent = #hideStems
      \override StaffSymbol #'after-line-breaking = #staff-break-callback
      \override StaffSymbol #'thickness = #(/ (ly:output-def-lookup $defaultpaper 'staff-space) 3.0)
      \override StaffSymbol #'stencil = #staff-print
    }
    \context {
      \Voice
      \override Stem #'length = #0
   \override BreathingSign #'after-line-breaking = #set-breathe
    }
    \context {
      \Score
      \override  RehearsalMark #'self-alignment-X = #LEFT
      \override RehearsalMark #'break-align-symbols = #'(key-signature)
      \override  NonMusicalPaperColumn #'padding = #0.6
      skipBars= ##t
      barAlways = ##t
      lyricMelismaAlignment = #CENTER
    }
    \context {
      \Lyrics
      \override LyricSpace #'minimum-distance = #0.6
      \override LyricText #'font-size = #1.065
      \override LyricText #'font-name = #"TimesNewRoman"
      \override LyricText #'color = #(rgb-color 0 0 0)
      \override LyricHyphen #'color = #(rgb-color 0 0 0)
      \override LyricExtender #'color = #(rgb-color 0 0 0)
    }
  }
}

\header {
tagline = "" 
title = ""
}
</LilyPond>
  <EasyChant>Di- cső- ség az A- tyá- nak és Fiúnak és Szent- lé- lek-        nek.
l,4 t,2  d4  t, d  r    m0            r  d      d   t,2 l,4 si, l,2 

</EasyChant>
  <Clef>0</Clef>
  <Key>8</Key>
  <LyricsSpacing>0.6</LyricsSpacing>
  <TimeOption>0</TimeOption>
  <TimeText>N/A (cadenza)</TimeText>
  <StaffSizeOption>2</StaffSizeOption>
  <StaffSizeText>14</StaffSizeText>
  <LyricsSize>1.1</LyricsSize>
  <LyricsFontName>Times New Roman</LyricsFontName>
  <LyricsFontBold>False</LyricsFontBold>
  <LyricsFontItalic>False</LyricsFontItalic>
  <LyricsFontColor>Color [A=255, R=0, G=0, B=0]</LyricsFontColor>
  <LineSpacing>12</LineSpacing>
  <LineThickness>4</LineThickness>
  <Options>0,1,2,4,6,7,8</Options>
  <pages>1</pages>
  <page>1</page>
</item>

Ft. konták, 1. hang

<?xml version="1.0" encoding="utf-16"?>
<item>
  <title>Konták, 1. hang</title>
  <LilyPond>\version "2.18.2"

\include "gregorian.ly"

\paper {
  paper-width = 123.0136\mm
  paper-height = 184.6262\mm
  top-margin = 0\mm
  bottom-margin = 0\mm
  left-margin = 0\mm
  right-margin = 2\mm
  ragged-bottom = ##t
  print-page-number = ##f
  page-breaking = #ly:minimal-breaking
  system-system-spacing =
    #'((basic-distance . 12)
       (minimum-distance . 12)
       (padding . 0)
       (stretchability . 0))
%  page-breaking-system-system-spacing = #'((padding . 2))
  markup-system-spacing = #'((padding . 1.73))
  top-markup-spacing = #'((basic-distance . 0) (minimum-distance . 0) (padding . 0))
}

#(set-global-staff-size 14)

#(define currBSign 0)
#(define hideStems #t)

#(define (set-breathe bsign)
 (begin
  (set! currBSign bsign)
 (let* ((columnOfBSign (ly:grob-parent (ly:grob-parent (ly:grob-parent currBSign X) X) X))
   (system (ly:grob-parent columnOfBSign X))   
   (columns (ly:grob-object system 'columns))
   (lastColumn (ly:grob-array-ref columns (- (ly:grob-array-length columns) 1)))
  )
  (if (eqv? columnOfBSign lastColumn)
   (ly:grob-set-property! currBSign 'extra-offset '(0.89 . 0))
  )
 )
 )
)

#(define (staff-print staffgrob)
 (let* ((stencil (ly:staff-symbol::print staffgrob))
   (Xextent (ly:stencil-extent stencil X))
   (Yextent (ly:stencil-extent stencil Y))
            (X0 (car Xextent))
   (X1 (cdr Xextent))
   (X1sc (+ X1 0.89))
   )
   (ly:make-stencil (ly:stencil-expr (ly:stencil-scale stencil (/ (- X1sc X0) (- X1 X0)) 1.0))
                         (cons X0 X1sc)
                         Yextent
   )
 )
)

#(define (staff-break-callback staffgrob)
    (let*
  (
    (system (ly:grob-parent (ly:grob-parent staffgrob X) X))
    (columns (ly:grob-object system 'columns))
    (columns-count (ly:grob-array-length columns))
    (goI 1)
    (goJ 1)
  )
  (if (ly:grob? currBSign)
   (let* ((columnOfBSign (ly:grob-parent (ly:grob-parent (ly:grob-parent currBSign X) X) X))
      (lastColumn (ly:grob-array-ref columns (- columns-count 1)))
     )
     (if (eqv? columnOfBSign lastColumn)
      (ly:grob-set-property! currBSign 'extra-offset '(0.89 . 0))
      (set! currBSign 0)
     )
     (set! currBSign 0)
   )
  )
  (let loopI  ((i 0)
     )
       (if (&lt; i columns-count)
      (loopI (let* ((column (ly:grob-array-ref columns i))
          (columnname (grob-name column))
         )
         (if (and (eqv? columnname 'PaperColumn) (ly:grob-array? (ly:grob-object column 'elements)))
          (let*
           (
           (elems (ly:grob-object column 'elements))
           (elems-count (ly:grob-array-length elems))
           )
           (let loopJ ((j 0)
                )
                (if (&lt; j elems-count) 
               (loopJ (let* ((grob (ly:grob-array-ref elems j))
                   (grobname (grob-name grob))
                  )
                  (if (eqv? grobname 'NoteColumn)
                   (begin
                    (show-notes grob)
                    (set! j elems-count)
                   )
                  )
                  (1+ j)
                 )
               )
              )
           )
           (set! i columns-count)
          )
         )
         (1+ i)
        )
      )
     )
  )
 )
)

#(define (grob-name grob)
   (if (ly:grob? grob)
       (assoc-ref (ly:grob-property grob 'meta) 'name)
       #f))

#(define (show-flag flag)
 (ly:grob-set-property! flag 'transparent hideStems)
)
   
#(define (show-stem stem)
 (let*  ((beam (ly:grob-object stem 'beam))
   )
   (begin
    (ly:grob-set-property! stem 'transparent hideStems)
    (if (ly:grob? beam)(ly:grob-set-property! beam 'transparent hideStems))
   )
 )
)

#(define (show-note notehead)
 (let* ((stem (ly:grob-object (ly:grob-parent notehead X) 'stem))
        (beam (ly:grob-object stem 'beam))
     (flag (ly:grob-object notehead 'flag))
        (acci (ly:grob-object notehead 'accidental-grob))
        (dots (ly:grob-object notehead 'dot)))
 (begin
  (ly:grob-set-property! notehead 'transparent #f)
  (if (ly:grob? dots)(ly:grob-set-property! dots 'transparent #f))
  (if (ly:grob? acci)(ly:grob-set-property! acci 'transparent #f))
  (ly:grob-set-property! stem 'transparent hideStems)
  (if (ly:grob? beam)(ly:grob-set-property! beam 'transparent hideStems))
  (if (ly:grob? flag)(ly:grob-set-property! flag 'transparent hideStems))
 ))
)

#(define (show-notes note-column) 
 (begin 
  (let* ((elems (ly:grob-object note-column 'elements))
    (elems-count (ly:grob-array-length elems))
     ) 
       (let loop ((staff-pos-sum 0)
      (i 0)
        )
        (if (&lt; i elems-count)
       (loop (let* ((grob (ly:grob-array-ref elems i))
           (grobname (grob-name grob))
          )
          (if (eqv? grobname 'NoteHead)
           (show-note grob)
           (if (eqv? grobname 'Stem)
            (show-stem grob)
            (if (eqv? grobname 'Flag)
             (show-flag grob)
            )
           )
          )
         )
        (1+ i)
       )
      )
    )
  )
 )
)




leftAlign = {  \once \override LyricText #'self-alignment-X = #LEFT }
centerAlign = {   \once \override LyricText #'self-alignment-X = #CENTER }
rightAlign = {   \once \override LyricText #'self-alignment-X = #1 }
hairpinToBarline = { \override Hairpin #'to-barline = ##t  }
hairpinToNote = { \override Hairpin #'to-barline = ##f  }

alignAt =
#(define-music-function
     (parser location xOffset)
     (number?)
  (display xOffset)
   #{
      \once \override LyricText #'self-alignment-X = $xOffset
   #})

stemOn = { \revert Staff.Stem #'transparent  \revert Staff.Beam #'transparent \revert Staff.Flag #'transparent \revert Voice.Stem #'length }
stemOff = { \override Staff.Stem #'transparent = ##t \override Staff.Beam #'transparent = ##t \override Staff.Flag #'transparent = ##t \override Stem #'length = #0 }

bigNotes = { \override NoteHead #'font-size = #3 }
smallNotes = { \override NoteHead #'font-size = #-3 }
normalNotes = { \override NoteHead #'font-size = #0 }

bN = { \once \override NoteHead #'font-size = #3 }
sN = { \once \override NoteHead #'font-size = #-3 }
nN = { \once \override NoteHead #'font-size = #0 }

shOn = { \override NoteColumn #'force-hshift = #0.18 }
shOff = { \revert NoteColumn #'force-hshift }

midRest = { \override Voice.Rest #'staff-position = #0 }
shRest = { \revert Voice.Rest #'staff-position }

modernAccidentals = #(set-accidental-style 'modern)
modernCautionaryAccidentals = #(set-accidental-style 'modern-cautionary)
defaultAccidentals = #(set-accidental-style 'default)
pianoAccidentals = #(set-accidental-style 'piano)
voiceAccidentals = #(set-accidental-style 'voice)
modernVoiceAccidentals = #(set-accidental-style 'modern-voice)
modernVoiceCautionaryAccidentals = #(set-accidental-style 'modern-voice-cautionary)
pianoCautionaryAccidentals = #(set-accidental-style 'piano-cautionary)
neoModernAccidentals = #(set-accidental-style 'neo-modern)
neoModernCautionaryAccidentals = #(set-accidental-style 'neo-modern-cautionary)
dodecapohnicAccidentals = #(set-accidental-style 'dodecaphonic)
teachingAccidentals = #(set-accidental-style 'teaching)
noResetAccidentals = #(set-accidental-style 'no-reset)
forgetAccidentals = #(set-accidental-style 'forget)

go = {
\grace {s16 } 
}



chant = {
\key f \major
d'4 e'2 f'4 e' f' g' a'\breve \hideNotes a' \unHideNotes g'4( fis') g'2 \divisioMaxima
a'2 g' f'4 e' f' g' a' a' f' e'2 e' \break \divisioMaxima
f'4 g'2 f'4 e' f' g' a'\breve \hideNotes a' a' a'  \unHideNotes \divisioMaxima
a'\breve \unHideNotes g'4( fis') g'2 \divisioMaxima
a'4 g'2 f'4 e' \break f' g' a' a' f' e'2 e' \divisioMaxima
f'4 g'2 f'4 e' f' g' a'\breve \hideNotes a' a' \break \unHideNotes a'4 g'( fis') g'2 \divisioMaxima
a'4 g'2 f'?\breve \unHideNotes e'2 d' \divisioMaxima
g'4 g' a'\breve \hideNotes a' a' a' \unHideNotes g'4 f' f' e'2( d'4 cis') d'2 
\hideNotes \grace {s16 } \finalis
}

verba = \lyricmode {
"Föl" -- "tá" -- "mad" -- "tál" "di" -- "cső" -- \alignAt #-0.865 "séggel" \alignAt #0 "a" "sír" -- "ból,"
"és" "a" "vi" -- "lá" -- "got" "is" "föl" -- "tá" -- "masz" -- "tot" -- "tad,"
"mi" -- "ért" "a" "mi" "em" -- "be" -- \alignAt #0 "ri" \alignAt #-0.976 "természetünk," \alignAt #-0.475 "mint" \alignAt #-1.224 "Istent,"
\alignAt #-0.716 "magasztal" "té" -- "ged,"
"mert" "a" "ha" -- "lál" "ál" -- "ta" -- "lad" "el" -- "e" -- "nyé" -- "szett."
"Á" -- "dám" "vi" -- "ga" -- "doz," "a" \alignAt #-0.913 "bilincseiből" \alignAt #-0.794 "megszabadult" \alignAt #-1.168 "Éva" "ör" -- "ven" -- "dez,"
"szív" -- "ből" \alignAt #-0.735 "kiáltván" "hoz" -- "zád:"
"te" "vagy" \alignAt #-0.577 "Krisztus," \alignAt #0 "ki" \alignAt #-0.856 "mindnyájunknak" \alignAt #-0.942 "föltámadást" "a" -- "ján" "dé" -- "koz" -- "tál."
}


\score {
  \new Staff &lt;&lt;
    #(set-accidental-style 'modern-cautionary)
    \new Voice = "melody" \chant
    \new Lyrics = "one" \lyricsto melody \verba
  &gt;&gt;
  \layout {
    line-thickness = #(/ (ly:output-def-lookup $defaultpaper 'staff-space) 4)
    \context {
      \Staff
      \override BarLine  #'stencil = ##f
      \remove Time_signature_engraver
      \override Stem #'transparent = #hideStems
      \override Beam #'transparent = #hideStems
      \override Flag #'transparent = #hideStems
      \override StaffSymbol #'after-line-breaking = #staff-break-callback
      \override StaffSymbol #'thickness = #(/ (ly:output-def-lookup $defaultpaper 'staff-space) 3.0)
      \override StaffSymbol #'stencil = #staff-print
    }
    \context {
      \Voice
      \override Stem #'length = #0
   \override BreathingSign #'after-line-breaking = #set-breathe
    }
    \context {
      \Score
      \override  RehearsalMark #'self-alignment-X = #LEFT
      \override RehearsalMark #'break-align-symbols = #'(key-signature)
      \override  NonMusicalPaperColumn #'padding = #0.6
      skipBars= ##t
      barAlways = ##t
      lyricMelismaAlignment = #CENTER
    }
    \context {
      \Lyrics
      \override LyricSpace #'minimum-distance = #0.6
      \override LyricText #'font-size = #1.065
      \override LyricText #'font-name = #"TimesNewRoman"
      \override LyricText #'color = #(rgb-color 0 0 0)
      \override LyricHyphen #'color = #(rgb-color 0 0 0)
      \override LyricExtender #'color = #(rgb-color 0 0 0)
    }
  }
}

\header {
tagline = "" 
title = ""
}
</LilyPond>
  <EasyChant>Föl- tá- mad- tál di- cső- séggel a sír- ból,
l,4  t,2 d4   t,  d   r    m0       r di r2  

és a vi- lá- got is   föl- tá- masz- tot- tad,
m2 r d4  t,  d   r    m    m   d     t,2  t,  
/
mi- ért a  mi em- be- ri természetünk, mint Istent,  
d4  r2  d4 t, d   r   m0                             

magasztal té-  ged, 
m0        r di r2   

mert a   ha- lál   ál- ta- lad el- e- nyé- szett.
m4   r2  d4  t,  / d   r   m   m   d  t,2  t,    

Á- dám vi- ga- doz, a bilincseiből megszabadult Éva   ör- ven- dez,
d4 r2  d4  t,  d    r m0                            / m   r di r2  

szív- ből kiáltván  hoz- zád:
m4    r2  d?0       t,   l,  

te vagy Krisztus, ki mindnyájunknak föltámadást  a- ján dé- koz-        tál.
r4 r    m0                                       r  d   d   t,2 l,4 si, l,2 </EasyChant>
  <Clef>0</Clef>
  <Key>8</Key>
  <LyricsSpacing>0.6</LyricsSpacing>
  <TimeOption>0</TimeOption>
  <TimeText>N/A (cadenza)</TimeText>
  <StaffSizeOption>2</StaffSizeOption>
  <StaffSizeText>14</StaffSizeText>
  <LyricsSize>1.1</LyricsSize>
  <LyricsFontName>Times New Roman</LyricsFontName>
  <LyricsFontBold>False</LyricsFontBold>
  <LyricsFontItalic>False</LyricsFontItalic>
  <LyricsFontColor>Color [A=255, R=0, G=0, B=0]</LyricsFontColor>
  <LineSpacing>12</LineSpacing>
  <LineThickness>4</LineThickness>
  <Options>0,1,2,4,6,7,8</Options>
  <pages>1</pages>
  <page>1</page>
</item>

Most és... 4. hang, konták-dallam

<?xml version="1.0" encoding="utf-16"?>
<item>
  <title>Most és, konták, 4. hang</title>
  <LilyPond>\version "2.18.2"

\include "gregorian.ly"

\paper {
  paper-width = 123.0136\mm
  paper-height = 184.6262\mm
  top-margin = 0\mm
  bottom-margin = 0\mm
  left-margin = 0\mm
  right-margin = 2\mm
  ragged-bottom = ##t
  print-page-number = ##f
  page-breaking = #ly:minimal-breaking
  system-system-spacing =
    #'((basic-distance . 12)
       (minimum-distance . 12)
       (padding . 0)
       (stretchability . 0))
%  page-breaking-system-system-spacing = #'((padding . 2))
  markup-system-spacing = #'((padding . 33.56))
  top-markup-spacing = #'((basic-distance . 0) (minimum-distance . 0) (padding . 0))
}

#(set-global-staff-size 14)

#(define currBSign 0)
#(define hideStems #t)

#(define (set-breathe bsign)
 (begin
  (set! currBSign bsign)
 (let* ((columnOfBSign (ly:grob-parent (ly:grob-parent (ly:grob-parent currBSign X) X) X))
   (system (ly:grob-parent columnOfBSign X))   
   (columns (ly:grob-object system 'columns))
   (lastColumn (ly:grob-array-ref columns (- (ly:grob-array-length columns) 1)))
  )
  (if (eqv? columnOfBSign lastColumn)
   (ly:grob-set-property! currBSign 'extra-offset '(0.89 . 0))
  )
 )
 )
)

#(define (staff-print staffgrob)
 (let* ((stencil (ly:staff-symbol::print staffgrob))
   (Xextent (ly:stencil-extent stencil X))
   (Yextent (ly:stencil-extent stencil Y))
            (X0 (car Xextent))
   (X1 (cdr Xextent))
   (X1sc (+ X1 0.89))
   )
   (ly:make-stencil (ly:stencil-expr (ly:stencil-scale stencil (/ (- X1sc X0) (- X1 X0)) 1.0))
                         (cons X0 X1sc)
                         Yextent
   )
 )
)

#(define (staff-break-callback staffgrob)
    (let*
  (
    (system (ly:grob-parent (ly:grob-parent staffgrob X) X))
    (columns (ly:grob-object system 'columns))
    (columns-count (ly:grob-array-length columns))
    (goI 1)
    (goJ 1)
  )
  (if (ly:grob? currBSign)
   (let* ((columnOfBSign (ly:grob-parent (ly:grob-parent (ly:grob-parent currBSign X) X) X))
      (lastColumn (ly:grob-array-ref columns (- columns-count 1)))
     )
     (if (eqv? columnOfBSign lastColumn)
      (ly:grob-set-property! currBSign 'extra-offset '(0.89 . 0))
      (set! currBSign 0)
     )
     (set! currBSign 0)
   )
  )
  (let loopI  ((i 0)
     )
       (if (&lt; i columns-count)
      (loopI (let* ((column (ly:grob-array-ref columns i))
          (columnname (grob-name column))
         )
         (if (and (eqv? columnname 'PaperColumn) (ly:grob-array? (ly:grob-object column 'elements)))
          (let*
           (
           (elems (ly:grob-object column 'elements))
           (elems-count (ly:grob-array-length elems))
           )
           (let loopJ ((j 0)
                )
                (if (&lt; j elems-count) 
               (loopJ (let* ((grob (ly:grob-array-ref elems j))
                   (grobname (grob-name grob))
                  )
                  (if (eqv? grobname 'NoteColumn)
                   (begin
                    (show-notes grob)
                    (set! j elems-count)
                   )
                  )
                  (1+ j)
                 )
               )
              )
           )
           (set! i columns-count)
          )
         )
         (1+ i)
        )
      )
     )
  )
 )
)

#(define (grob-name grob)
   (if (ly:grob? grob)
       (assoc-ref (ly:grob-property grob 'meta) 'name)
       #f))

#(define (show-flag flag)
 (ly:grob-set-property! flag 'transparent hideStems)
)
   
#(define (show-stem stem)
 (let*  ((beam (ly:grob-object stem 'beam))
   )
   (begin
    (ly:grob-set-property! stem 'transparent hideStems)
    (if (ly:grob? beam)(ly:grob-set-property! beam 'transparent hideStems))
   )
 )
)

#(define (show-note notehead)
 (let* ((stem (ly:grob-object (ly:grob-parent notehead X) 'stem))
        (beam (ly:grob-object stem 'beam))
     (flag (ly:grob-object notehead 'flag))
        (acci (ly:grob-object notehead 'accidental-grob))
        (dots (ly:grob-object notehead 'dot)))
 (begin
  (ly:grob-set-property! notehead 'transparent #f)
  (if (ly:grob? dots)(ly:grob-set-property! dots 'transparent #f))
  (if (ly:grob? acci)(ly:grob-set-property! acci 'transparent #f))
  (ly:grob-set-property! stem 'transparent hideStems)
  (if (ly:grob? beam)(ly:grob-set-property! beam 'transparent hideStems))
  (if (ly:grob? flag)(ly:grob-set-property! flag 'transparent hideStems))
 ))
)

#(define (show-notes note-column) 
 (begin 
  (let* ((elems (ly:grob-object note-column 'elements))
    (elems-count (ly:grob-array-length elems))
     ) 
       (let loop ((staff-pos-sum 0)
      (i 0)
        )
        (if (&lt; i elems-count)
       (loop (let* ((grob (ly:grob-array-ref elems i))
           (grobname (grob-name grob))
          )
          (if (eqv? grobname 'NoteHead)
           (show-note grob)
           (if (eqv? grobname 'Stem)
            (show-stem grob)
            (if (eqv? grobname 'Flag)
             (show-flag grob)
            )
           )
          )
         )
        (1+ i)
       )
      )
    )
  )
 )
)




leftAlign = {  \once \override LyricText #'self-alignment-X = #LEFT }
centerAlign = {   \once \override LyricText #'self-alignment-X = #CENTER }
rightAlign = {   \once \override LyricText #'self-alignment-X = #1 }
hairpinToBarline = { \override Hairpin #'to-barline = ##t  }
hairpinToNote = { \override Hairpin #'to-barline = ##f  }

alignAt =
#(define-music-function
     (parser location xOffset)
     (number?)
  (display xOffset)
   #{
      \once \override LyricText #'self-alignment-X = $xOffset
   #})

stemOn = { \revert Staff.Stem #'transparent  \revert Staff.Beam #'transparent \revert Staff.Flag #'transparent \revert Voice.Stem #'length }
stemOff = { \override Staff.Stem #'transparent = ##t \override Staff.Beam #'transparent = ##t \override Staff.Flag #'transparent = ##t \override Stem #'length = #0 }

bigNotes = { \override NoteHead #'font-size = #3 }
smallNotes = { \override NoteHead #'font-size = #-3 }
normalNotes = { \override NoteHead #'font-size = #0 }

bN = { \once \override NoteHead #'font-size = #3 }
sN = { \once \override NoteHead #'font-size = #-3 }
nN = { \once \override NoteHead #'font-size = #0 }

shOn = { \override NoteColumn #'force-hshift = #0.18 }
shOff = { \revert NoteColumn #'force-hshift }

midRest = { \override Voice.Rest #'staff-position = #0 }
shRest = { \revert Voice.Rest #'staff-position }

modernAccidentals = #(set-accidental-style 'modern)
modernCautionaryAccidentals = #(set-accidental-style 'modern-cautionary)
defaultAccidentals = #(set-accidental-style 'default)
pianoAccidentals = #(set-accidental-style 'piano)
voiceAccidentals = #(set-accidental-style 'voice)
modernVoiceAccidentals = #(set-accidental-style 'modern-voice)
modernVoiceCautionaryAccidentals = #(set-accidental-style 'modern-voice-cautionary)
pianoCautionaryAccidentals = #(set-accidental-style 'piano-cautionary)
neoModernAccidentals = #(set-accidental-style 'neo-modern)
neoModernCautionaryAccidentals = #(set-accidental-style 'neo-modern-cautionary)
dodecapohnicAccidentals = #(set-accidental-style 'dodecaphonic)
teachingAccidentals = #(set-accidental-style 'teaching)
noResetAccidentals = #(set-accidental-style 'no-reset)
forgetAccidentals = #(set-accidental-style 'forget)

go = {
\grace {s16 } 
}



chant = {
\key f \major
c''\breve \hideNotes c'' c'' c'' c'' \unHideNotes a'4 d'' c''2 bes' a'2 
\hideNotes \grace {s16 } \finalis
}

verba = \lyricmode {
\alignAt #-0.32 "Most" \alignAt #0 "és" \alignAt #-0.777 "mindenkor" \alignAt #0 "és" \alignAt #-1.112 "örökkön" -- "ö" -- "rök" -- "ké." "A" -- "men."
}


\score {
  \new Staff &lt;&lt;
    #(set-accidental-style 'modern-cautionary)
    \new Voice = "melody" \chant
    \new Lyrics = "one" \lyricsto melody \verba
  &gt;&gt;
  \layout {
    line-thickness = #(/ (ly:output-def-lookup $defaultpaper 'staff-space) 4)
    \context {
      \Staff
      \override BarLine  #'stencil = ##f
      \remove Time_signature_engraver
      \override Stem #'transparent = #hideStems
      \override Beam #'transparent = #hideStems
      \override Flag #'transparent = #hideStems
      \override StaffSymbol #'after-line-breaking = #staff-break-callback
      \override StaffSymbol #'thickness = #(/ (ly:output-def-lookup $defaultpaper 'staff-space) 3.0)
      \override StaffSymbol #'stencil = #staff-print
    }
    \context {
      \Voice
      \override Stem #'length = #0
   \override BreathingSign #'after-line-breaking = #set-breathe
    }
    \context {
      \Score
      \override  RehearsalMark #'self-alignment-X = #LEFT
      \override RehearsalMark #'break-align-symbols = #'(key-signature)
      \override  NonMusicalPaperColumn #'padding = #0.6
      skipBars= ##t
      barAlways = ##t
      lyricMelismaAlignment = #CENTER
    }
    \context {
      \Lyrics
      \override LyricSpace #'minimum-distance = #0.6
      \override LyricText #'font-size = #1.065
      \override LyricText #'font-name = #"TimesNewRoman"
      \override LyricText #'color = #(rgb-color 0 0 0)
      \override LyricHyphen #'color = #(rgb-color 0 0 0)
      \override LyricExtender #'color = #(rgb-color 0 0 0)
    }
  }
}

\header {
tagline = "" 
title = ""
}
</LilyPond>
  <EasyChant>Most és mindenkor és örökkön- ö- rök- ké. A- men.
s0                            m4 l    s2  f  m2  

</EasyChant>
  <Clef>0</Clef>
  <Key>8</Key>
  <LyricsSpacing>0.6</LyricsSpacing>
  <TimeOption>0</TimeOption>
  <TimeText>N/A (cadenza)</TimeText>
  <StaffSizeOption>2</StaffSizeOption>
  <StaffSizeText>14</StaffSizeText>
  <LyricsSize>1.1</LyricsSize>
  <LyricsFontName>Times New Roman</LyricsFontName>
  <LyricsFontBold>False</LyricsFontBold>
  <LyricsFontItalic>False</LyricsFontItalic>
  <LyricsFontColor>Color [A=255, R=0, G=0, B=0]</LyricsFontColor>
  <LineSpacing>12</LineSpacing>
  <LineThickness>4</LineThickness>
  <Options>0,1,2,4,6,7,8</Options>
  <pages>1</pages>
  <page>1</page>
</item>

Ün. konták, 4. hang

<?xml version="1.0" encoding="utf-16"?>
<item>
  <title>Konták, 4. hang</title>
  <LilyPond>\version "2.18.2"

\include "gregorian.ly"

\paper {
  paper-width = 123.0136\mm
  paper-height = 184.6262\mm
  top-margin = 0\mm
  bottom-margin = 0\mm
  left-margin = 0\mm
  right-margin = 2\mm
  ragged-bottom = ##t
  print-page-number = ##f
  page-breaking = #ly:minimal-breaking
  system-system-spacing =
    #'((basic-distance . 12)
       (minimum-distance . 12)
       (padding . 0)
       (stretchability . 0))
%  page-breaking-system-system-spacing = #'((padding . 2))
  markup-system-spacing = #'((padding . 0))
  top-markup-spacing = #'((basic-distance . 0) (minimum-distance . 0) (padding . 0))
}

#(set-global-staff-size 14)

#(define currBSign 0)
#(define hideStems #t)

#(define (set-breathe bsign)
 (begin
  (set! currBSign bsign)
 (let* ((columnOfBSign (ly:grob-parent (ly:grob-parent (ly:grob-parent currBSign X) X) X))
   (system (ly:grob-parent columnOfBSign X))   
   (columns (ly:grob-object system 'columns))
   (lastColumn (ly:grob-array-ref columns (- (ly:grob-array-length columns) 1)))
  )
  (if (eqv? columnOfBSign lastColumn)
   (ly:grob-set-property! currBSign 'extra-offset '(0.89 . 0))
  )
 )
 )
)

#(define (staff-print staffgrob)
 (let* ((stencil (ly:staff-symbol::print staffgrob))
   (Xextent (ly:stencil-extent stencil X))
   (Yextent (ly:stencil-extent stencil Y))
            (X0 (car Xextent))
   (X1 (cdr Xextent))
   (X1sc (+ X1 0.89))
   )
   (ly:make-stencil (ly:stencil-expr (ly:stencil-scale stencil (/ (- X1sc X0) (- X1 X0)) 1.0))
                         (cons X0 X1sc)
                         Yextent
   )
 )
)

#(define (staff-break-callback staffgrob)
    (let*
  (
    (system (ly:grob-parent (ly:grob-parent staffgrob X) X))
    (columns (ly:grob-object system 'columns))
    (columns-count (ly:grob-array-length columns))
    (goI 1)
    (goJ 1)
  )
  (if (ly:grob? currBSign)
   (let* ((columnOfBSign (ly:grob-parent (ly:grob-parent (ly:grob-parent currBSign X) X) X))
      (lastColumn (ly:grob-array-ref columns (- columns-count 1)))
     )
     (if (eqv? columnOfBSign lastColumn)
      (ly:grob-set-property! currBSign 'extra-offset '(0.89 . 0))
      (set! currBSign 0)
     )
     (set! currBSign 0)
   )
  )
  (let loopI  ((i 0)
     )
       (if (&lt; i columns-count)
      (loopI (let* ((column (ly:grob-array-ref columns i))
          (columnname (grob-name column))
         )
         (if (and (eqv? columnname 'PaperColumn) (ly:grob-array? (ly:grob-object column 'elements)))
          (let*
           (
           (elems (ly:grob-object column 'elements))
           (elems-count (ly:grob-array-length elems))
           )
           (let loopJ ((j 0)
                )
                (if (&lt; j elems-count) 
               (loopJ (let* ((grob (ly:grob-array-ref elems j))
                   (grobname (grob-name grob))
                  )
                  (if (eqv? grobname 'NoteColumn)
                   (begin
                    (show-notes grob)
                    (set! j elems-count)
                   )
                  )
                  (1+ j)
                 )
               )
              )
           )
           (set! i columns-count)
          )
         )
         (1+ i)
        )
      )
     )
  )
 )
)

#(define (grob-name grob)
   (if (ly:grob? grob)
       (assoc-ref (ly:grob-property grob 'meta) 'name)
       #f))

#(define (show-flag flag)
 (ly:grob-set-property! flag 'transparent hideStems)
)
   
#(define (show-stem stem)
 (let*  ((beam (ly:grob-object stem 'beam))
   )
   (begin
    (ly:grob-set-property! stem 'transparent hideStems)
    (if (ly:grob? beam)(ly:grob-set-property! beam 'transparent hideStems))
   )
 )
)

#(define (show-note notehead)
 (let* ((stem (ly:grob-object (ly:grob-parent notehead X) 'stem))
        (beam (ly:grob-object stem 'beam))
     (flag (ly:grob-object notehead 'flag))
        (acci (ly:grob-object notehead 'accidental-grob))
        (dots (ly:grob-object notehead 'dot)))
 (begin
  (ly:grob-set-property! notehead 'transparent #f)
  (if (ly:grob? dots)(ly:grob-set-property! dots 'transparent #f))
  (if (ly:grob? acci)(ly:grob-set-property! acci 'transparent #f))
  (ly:grob-set-property! stem 'transparent hideStems)
  (if (ly:grob? beam)(ly:grob-set-property! beam 'transparent hideStems))
  (if (ly:grob? flag)(ly:grob-set-property! flag 'transparent hideStems))
 ))
)

#(define (show-notes note-column) 
 (begin 
  (let* ((elems (ly:grob-object note-column 'elements))
    (elems-count (ly:grob-array-length elems))
     ) 
       (let loop ((staff-pos-sum 0)
      (i 0)
        )
        (if (&lt; i elems-count)
       (loop (let* ((grob (ly:grob-array-ref elems i))
           (grobname (grob-name grob))
          )
          (if (eqv? grobname 'NoteHead)
           (show-note grob)
           (if (eqv? grobname 'Stem)
            (show-stem grob)
            (if (eqv? grobname 'Flag)
             (show-flag grob)
            )
           )
          )
         )
        (1+ i)
       )
      )
    )
  )
 )
)




leftAlign = {  \once \override LyricText #'self-alignment-X = #LEFT }
centerAlign = {   \once \override LyricText #'self-alignment-X = #CENTER }
rightAlign = {   \once \override LyricText #'self-alignment-X = #1 }
hairpinToBarline = { \override Hairpin #'to-barline = ##t  }
hairpinToNote = { \override Hairpin #'to-barline = ##f  }

alignAt =
#(define-music-function
     (parser location xOffset)
     (number?)
  (display xOffset)
   #{
      \once \override LyricText #'self-alignment-X = $xOffset
   #})

stemOn = { \revert Staff.Stem #'transparent  \revert Staff.Beam #'transparent \revert Staff.Flag #'transparent \revert Voice.Stem #'length }
stemOff = { \override Staff.Stem #'transparent = ##t \override Staff.Beam #'transparent = ##t \override Staff.Flag #'transparent = ##t \override Stem #'length = #0 }

bigNotes = { \override NoteHead #'font-size = #3 }
smallNotes = { \override NoteHead #'font-size = #-3 }
normalNotes = { \override NoteHead #'font-size = #0 }

bN = { \once \override NoteHead #'font-size = #3 }
sN = { \once \override NoteHead #'font-size = #-3 }
nN = { \once \override NoteHead #'font-size = #0 }

shOn = { \override NoteColumn #'force-hshift = #0.18 }
shOff = { \revert NoteColumn #'force-hshift }

midRest = { \override Voice.Rest #'staff-position = #0 }
shRest = { \revert Voice.Rest #'staff-position }

modernAccidentals = #(set-accidental-style 'modern)
modernCautionaryAccidentals = #(set-accidental-style 'modern-cautionary)
defaultAccidentals = #(set-accidental-style 'default)
pianoAccidentals = #(set-accidental-style 'piano)
voiceAccidentals = #(set-accidental-style 'voice)
modernVoiceAccidentals = #(set-accidental-style 'modern-voice)
modernVoiceCautionaryAccidentals = #(set-accidental-style 'modern-voice-cautionary)
pianoCautionaryAccidentals = #(set-accidental-style 'piano-cautionary)
neoModernAccidentals = #(set-accidental-style 'neo-modern)
neoModernCautionaryAccidentals = #(set-accidental-style 'neo-modern-cautionary)
dodecapohnicAccidentals = #(set-accidental-style 'dodecaphonic)
teachingAccidentals = #(set-accidental-style 'teaching)
noResetAccidentals = #(set-accidental-style 'no-reset)
forgetAccidentals = #(set-accidental-style 'forget)

go = {
\grace {s16 } 
}



chant = {
\key f \major
c''4 bes' a'\breve \hideNotes a' \unHideNotes bes'4 c''2 bes' a' \divisioMaxima
c''\breve \hideNotes c'' c'' c'' c'' \unHideNotes a'4 d''2 \break c'' bes' a' \divisioMaxima
c''4 bes' a'\breve \hideNotes a' a' \unHideNotes bes'4 c''2 bes' a' \divisioMaxima
c''\breve \hideNotes c'' \unHideNotes a'4 d'' c''2 bes' a'2 \divisioMaxima
c''4 bes' a'\breve \unHideNotes bes'4 c''2 bes' a' \divisioMaxima
c''\breve \hideNotes c'' c'' \unHideNotes a'4 d'' c''4 bes' a'2 
\hideNotes \grace {s16 } \finalis
}

verba = \lyricmode {
"Üd" -- "vö" -- \alignAt #-0.888 "zítőnk" \alignAt #-0.948 "legtisz" -- "tább" "haj" -- "lé" -- "ka,"
\alignAt #-0.441 "drága" \alignAt #-0.788 "palota," \alignAt #-1.482 "az" \alignAt #-1.273 "isteni" \alignAt #-0.808 "dicső" -- "ség" "szent" "kincs" -- "tá" -- "ra,"
"ma" "mu" -- \alignAt #-0.952 "tattatik" \alignAt #-0.396 "be" \alignAt #-1.482 "az" "Úr" "há" -- "zá" -- "ban,"
\alignAt #0 "a" \alignAt #-0.628 "Szentlélek" "ma" -- "laszt" -- "ját" "hoz" -- "ván,"
"kit" "így" \alignAt #-1.105 "énekelnek" "az" "an" -- "gya" -- "lok:"
\alignAt #-1.248 "Ez" \alignAt #0 "a" \alignAt #-0.486 "meny" -- "nye" -- "i" "pa" -- "lo" -- "ta!"
}


\score {
  \new Staff &lt;&lt;
    #(set-accidental-style 'modern-cautionary)
    \new Voice = "melody" \chant
    \new Lyrics = "one" \lyricsto melody \verba
  &gt;&gt;
  \layout {
    ragged-last = ##f
    line-thickness = #(/ (ly:output-def-lookup $defaultpaper 'staff-space) 4)
    \context {
      \Staff
      \override BarLine  #'stencil = ##f
      \remove Time_signature_engraver
      \override Stem #'transparent = #hideStems
      \override Beam #'transparent = #hideStems
      \override Flag #'transparent = #hideStems
      \override StaffSymbol #'after-line-breaking = #staff-break-callback
      \override StaffSymbol #'thickness = #(/ (ly:output-def-lookup $defaultpaper 'staff-space) 3.0)
      \override StaffSymbol #'stencil = #staff-print
    }
    \context {
      \Voice
      \override Stem #'length = #0
   \override BreathingSign #'after-line-breaking = #set-breathe
    }
    \context {
      \Score
      \override  RehearsalMark #'self-alignment-X = #LEFT
      \override RehearsalMark #'break-align-symbols = #'(key-signature)
      \override  NonMusicalPaperColumn #'padding = #0.8
      skipBars= ##t
      barAlways = ##t
      lyricMelismaAlignment = #CENTER
    }
    \context {
      \Lyrics
      \override LyricSpace #'minimum-distance = #0.8
      \override LyricText #'font-size = #1.065
      \override LyricText #'font-name = #"TimesNewRoman"
      \override LyricText #'color = #(rgb-color 0 0 0)
      \override LyricHyphen #'color = #(rgb-color 0 0 0)
      \override LyricExtender #'color = #(rgb-color 0 0 0)
    }
  }
}

\header {
tagline = "" 
title = ""
}
</LilyPond>
  <EasyChant>Üd- vö- zítőnk legtisz- tább haj- lé- ka,
s4  f   m0              f    s2   f   m  

drága palota, az isteni dicső- ség szent   kincs- tá- ra,
s0                             m4  l2    / s      f   m  

ma mu- tattatik be az Úr há- zá- ban,
s4 f   m0             f  s2  f   m   

a Szentlélek ma- laszt- ját hoz- ván,
s0           m4  l      s2  f    m2  

kit így énekelnek az an- gya- lok:
s4  f   m0        f  s2  f    m   

Ez a meny- nye- i  pa- lo- ta!
s0         m4   l  s4  f   m2 

</EasyChant>
  <Clef>0</Clef>
  <Key>8</Key>
  <LyricsSpacing>0.8</LyricsSpacing>
  <TimeOption>0</TimeOption>
  <TimeText>N/A (cadenza)</TimeText>
  <StaffSizeOption>2</StaffSizeOption>
  <StaffSizeText>14</StaffSizeText>
  <LyricsSize>1.1</LyricsSize>
  <LyricsFontName>Times New Roman</LyricsFontName>
  <LyricsFontBold>False</LyricsFontBold>
  <LyricsFontItalic>False</LyricsFontItalic>
  <LyricsFontColor>Color [A=255, R=0, G=0, B=0]</LyricsFontColor>
  <LineSpacing>12</LineSpacing>
  <LineThickness>4</LineThickness>
  <Options>0,1,2,6,7,8</Options>
  <pages>1</pages>
  <page>1</page>
</item>

Ft. prokimen, 1. hang

<?xml version="1.0" encoding="utf-16"?>
<item>
  <title>Prokimen, 1. hang</title>
  <LilyPond>\version "2.18.2"

\include "gregorian.ly"

\paper {
  paper-width = 123.0136\mm
  paper-height = 184.6262\mm
  top-margin = 0\mm
  bottom-margin = 0\mm
  left-margin = 0\mm
  right-margin = 2\mm
  ragged-bottom = ##t
  print-page-number = ##f
  page-breaking = #ly:minimal-breaking
  system-system-spacing =
    #'((basic-distance . 12)
       (minimum-distance . 12)
       (padding . 0)
       (stretchability . 0))
%  page-breaking-system-system-spacing = #'((padding . 2))
  markup-system-spacing = #'((padding . 1.73))
  top-markup-spacing = #'((basic-distance . 0) (minimum-distance . 0) (padding . 0))
}

#(set-global-staff-size 14)

#(define currBSign 0)
#(define hideStems #t)

#(define (set-breathe bsign)
 (begin
  (set! currBSign bsign)
 (let* ((columnOfBSign (ly:grob-parent (ly:grob-parent (ly:grob-parent currBSign X) X) X))
   (system (ly:grob-parent columnOfBSign X))   
   (columns (ly:grob-object system 'columns))
   (lastColumn (ly:grob-array-ref columns (- (ly:grob-array-length columns) 1)))
  )
  (if (eqv? columnOfBSign lastColumn)
   (ly:grob-set-property! currBSign 'extra-offset '(0.89 . 0))
  )
 )
 )
)

#(define (staff-print staffgrob)
 (let* ((stencil (ly:staff-symbol::print staffgrob))
   (Xextent (ly:stencil-extent stencil X))
   (Yextent (ly:stencil-extent stencil Y))
            (X0 (car Xextent))
   (X1 (cdr Xextent))
   (X1sc (+ X1 0.89))
   )
   (ly:make-stencil (ly:stencil-expr (ly:stencil-scale stencil (/ (- X1sc X0) (- X1 X0)) 1.0))
                         (cons X0 X1sc)
                         Yextent
   )
 )
)

#(define (staff-break-callback staffgrob)
    (let*
  (
    (system (ly:grob-parent (ly:grob-parent staffgrob X) X))
    (columns (ly:grob-object system 'columns))
    (columns-count (ly:grob-array-length columns))
    (goI 1)
    (goJ 1)
  )
  (if (ly:grob? currBSign)
   (let* ((columnOfBSign (ly:grob-parent (ly:grob-parent (ly:grob-parent currBSign X) X) X))
      (lastColumn (ly:grob-array-ref columns (- columns-count 1)))
     )
     (if (eqv? columnOfBSign lastColumn)
      (ly:grob-set-property! currBSign 'extra-offset '(0.89 . 0))
      (set! currBSign 0)
     )
     (set! currBSign 0)
   )
  )
  (let loopI  ((i 0)
     )
       (if (&lt; i columns-count)
      (loopI (let* ((column (ly:grob-array-ref columns i))
          (columnname (grob-name column))
         )
         (if (and (eqv? columnname 'PaperColumn) (ly:grob-array? (ly:grob-object column 'elements)))
          (let*
           (
           (elems (ly:grob-object column 'elements))
           (elems-count (ly:grob-array-length elems))
           )
           (let loopJ ((j 0)
                )
                (if (&lt; j elems-count) 
               (loopJ (let* ((grob (ly:grob-array-ref elems j))
                   (grobname (grob-name grob))
                  )
                  (if (eqv? grobname 'NoteColumn)
                   (begin
                    (show-notes grob)
                    (set! j elems-count)
                   )
                  )
                  (1+ j)
                 )
               )
              )
           )
           (set! i columns-count)
          )
         )
         (1+ i)
        )
      )
     )
  )
 )
)

#(define (grob-name grob)
   (if (ly:grob? grob)
       (assoc-ref (ly:grob-property grob 'meta) 'name)
       #f))

#(define (show-flag flag)
 (ly:grob-set-property! flag 'transparent hideStems)
)
   
#(define (show-stem stem)
 (let*  ((beam (ly:grob-object stem 'beam))
   )
   (begin
    (ly:grob-set-property! stem 'transparent hideStems)
    (if (ly:grob? beam)(ly:grob-set-property! beam 'transparent hideStems))
   )
 )
)

#(define (show-note notehead)
 (let* ((stem (ly:grob-object (ly:grob-parent notehead X) 'stem))
        (beam (ly:grob-object stem 'beam))
     (flag (ly:grob-object notehead 'flag))
        (acci (ly:grob-object notehead 'accidental-grob))
        (dots (ly:grob-object notehead 'dot)))
 (begin
  (ly:grob-set-property! notehead 'transparent #f)
  (if (ly:grob? dots)(ly:grob-set-property! dots 'transparent #f))
  (if (ly:grob? acci)(ly:grob-set-property! acci 'transparent #f))
  (ly:grob-set-property! stem 'transparent hideStems)
  (if (ly:grob? beam)(ly:grob-set-property! beam 'transparent hideStems))
  (if (ly:grob? flag)(ly:grob-set-property! flag 'transparent hideStems))
 ))
)

#(define (show-notes note-column) 
 (begin 
  (let* ((elems (ly:grob-object note-column 'elements))
    (elems-count (ly:grob-array-length elems))
     ) 
       (let loop ((staff-pos-sum 0)
      (i 0)
        )
        (if (&lt; i elems-count)
       (loop (let* ((grob (ly:grob-array-ref elems i))
           (grobname (grob-name grob))
          )
          (if (eqv? grobname 'NoteHead)
           (show-note grob)
           (if (eqv? grobname 'Stem)
            (show-stem grob)
            (if (eqv? grobname 'Flag)
             (show-flag grob)
            )
           )
          )
         )
        (1+ i)
       )
      )
    )
  )
 )
)




leftAlign = {  \once \override LyricText #'self-alignment-X = #LEFT }
centerAlign = {   \once \override LyricText #'self-alignment-X = #CENTER }
rightAlign = {   \once \override LyricText #'self-alignment-X = #1 }
hairpinToBarline = { \override Hairpin #'to-barline = ##t  }
hairpinToNote = { \override Hairpin #'to-barline = ##f  }

alignAt =
#(define-music-function
     (parser location xOffset)
     (number?)
  (display xOffset)
   #{
      \once \override LyricText #'self-alignment-X = $xOffset
   #})

stemOn = { \revert Staff.Stem #'transparent  \revert Staff.Beam #'transparent \revert Staff.Flag #'transparent \revert Voice.Stem #'length }
stemOff = { \override Staff.Stem #'transparent = ##t \override Staff.Beam #'transparent = ##t \override Staff.Flag #'transparent = ##t \override Stem #'length = #0 }

bigNotes = { \override NoteHead #'font-size = #3 }
smallNotes = { \override NoteHead #'font-size = #-3 }
normalNotes = { \override NoteHead #'font-size = #0 }

bN = { \once \override NoteHead #'font-size = #3 }
sN = { \once \override NoteHead #'font-size = #-3 }
nN = { \once \override NoteHead #'font-size = #0 }

shOn = { \override NoteColumn #'force-hshift = #0.18 }
shOff = { \revert NoteColumn #'force-hshift }

midRest = { \override Voice.Rest #'staff-position = #0 }
shRest = { \revert Voice.Rest #'staff-position }

modernAccidentals = #(set-accidental-style 'modern)
modernCautionaryAccidentals = #(set-accidental-style 'modern-cautionary)
defaultAccidentals = #(set-accidental-style 'default)
pianoAccidentals = #(set-accidental-style 'piano)
voiceAccidentals = #(set-accidental-style 'voice)
modernVoiceAccidentals = #(set-accidental-style 'modern-voice)
modernVoiceCautionaryAccidentals = #(set-accidental-style 'modern-voice-cautionary)
pianoCautionaryAccidentals = #(set-accidental-style 'piano-cautionary)
neoModernAccidentals = #(set-accidental-style 'neo-modern)
neoModernCautionaryAccidentals = #(set-accidental-style 'neo-modern-cautionary)
dodecapohnicAccidentals = #(set-accidental-style 'dodecaphonic)
teachingAccidentals = #(set-accidental-style 'teaching)
noResetAccidentals = #(set-accidental-style 'no-reset)
forgetAccidentals = #(set-accidental-style 'forget)

go = {
\grace {s16 } 
}



chant = {
\key f \major
a'4 bes' c''\breve \hideNotes c'' \unHideNotes a'4 bes'2 a'4( c'' bes' a') g'2 \divisioMaxima \divisioMaxima \divisioMaxima \divisioMaxima
g'4( a' g' fis') g'( a' bes') c''( bes') a'( g') c''2( bes'4 a') g'2 \break \divisioMaxima
e'4 f'? g'2 a' g'4( f' e' d') c'2 \finalis
\override Lyrics.LyricText.transparent = ##t f'\breve \hideNotes  \unHideNotes 
\hideNotes \grace {s16 } \finalis
}

verba = \lyricmode {
"U" -- "runk," \alignAt #-0.949 "legyen" \alignAt #-0.775 "raj" -- "tunk" "ir" -- "gal" -- "mad,"
"a" -- "mint" "ben" -- "ned" "bíz" -- "tunk,"
"a" -- "mint" "ben" -- "ned" "bíz" -- "tunk!"
\alignAt #-0.574 "Recit"
}


\score {
  \new Staff &lt;&lt;
    #(set-accidental-style 'modern-cautionary)
    \new Voice = "melody" \chant
    \new Lyrics = "one" \lyricsto melody \verba
  &gt;&gt;
  \layout {
    line-thickness = #(/ (ly:output-def-lookup $defaultpaper 'staff-space) 4)
    \context {
      \Staff
      \override BarLine  #'stencil = ##f
      \remove Time_signature_engraver
      \override Stem #'transparent = #hideStems
      \override Beam #'transparent = #hideStems
      \override Flag #'transparent = #hideStems
      \override StaffSymbol #'after-line-breaking = #staff-break-callback
      \override StaffSymbol #'thickness = #(/ (ly:output-def-lookup $defaultpaper 'staff-space) 3.0)
      \override StaffSymbol #'stencil = #staff-print
    }
    \context {
      \Voice
      \override Stem #'length = #0
   \override BreathingSign #'after-line-breaking = #set-breathe
    }
    \context {
      \Score
      \override  RehearsalMark #'self-alignment-X = #LEFT
      \override RehearsalMark #'break-align-symbols = #'(key-signature)
      \override  NonMusicalPaperColumn #'padding = #0.6
      skipBars= ##t
      barAlways = ##t
      lyricMelismaAlignment = #CENTER
    }
    \context {
      \Lyrics
      \override LyricSpace #'minimum-distance = #0.6
      \override LyricText #'font-size = #1.065
      \override LyricText #'font-name = #"TimesNewRoman"
      \override LyricText #'color = #(rgb-color 0 0 0)
      \override LyricHyphen #'color = #(rgb-color 0 0 0)
      \override LyricExtender #'color = #(rgb-color 0 0 0)
    }
  }
}

\header {
tagline = "" 
title = ""
}
</LilyPond>
  <EasyChant>U- runk, le-gyen raj- tunk ir- gal-     mad,
m4  f    s0              m f2  m4 s f m r2  

a-        mint  ben- ned   bíz-    tunk,
r4 m r di r m f s f  m r   s2 f4 m r2   
/
a-  mint ben - ned bíz-       tunk!
t,4 d?    r2   m   r4 d t, l, s,2   ||

Recit
\"override Lyrics.LyricText.transparent = ##t" d0</EasyChant>
  <Clef>0</Clef>
  <Key>8</Key>
  <LyricsSpacing>0.6</LyricsSpacing>
  <TimeOption>0</TimeOption>
  <TimeText>N/A (cadenza)</TimeText>
  <StaffSizeOption>2</StaffSizeOption>
  <StaffSizeText>14</StaffSizeText>
  <LyricsSize>1.1</LyricsSize>
  <LyricsFontName>Times New Roman</LyricsFontName>
  <LyricsFontBold>False</LyricsFontBold>
  <LyricsFontItalic>False</LyricsFontItalic>
  <LyricsFontColor>Color [A=255, R=0, G=0, B=0]</LyricsFontColor>
  <LineSpacing>12</LineSpacing>
  <LineThickness>4</LineThickness>
  <Options>0,1,2,4,6,7,8</Options>
  <pages>1</pages>
  <page>1</page>
</item>

Vers: Örvendjetek, igazak az Úrban, az igazakhoz illik a dicséret!

Urunk, legyen rajtunk irgalmad, amint benned bíztunk!

(32. zsoltár)

Ün. prokimen, 3. hang

<?xml version="1.0" encoding="utf-16"?>
<item>
  <title>Prokimen, 3. hang2</title>
  <LilyPond>\version "2.18.2"

\include "gregorian.ly"

\paper {
  paper-width = 123.0136\mm
  paper-height = 184.6262\mm
  top-margin = 0\mm
  bottom-margin = 0\mm
  left-margin = 0\mm
  right-margin = 2\mm
  ragged-bottom = ##t
  print-page-number = ##f
  page-breaking = #ly:minimal-breaking
  system-system-spacing =
    #'((basic-distance . 12)
       (minimum-distance . 12)
       (padding . 0)
       (stretchability . 0))
%  page-breaking-system-system-spacing = #'((padding . 2))
  markup-system-spacing = #'((padding . 112.83))
  top-markup-spacing = #'((basic-distance . 0) (minimum-distance . 0) (padding . 0))
}

#(set-global-staff-size 14)

#(define currBSign 0)
#(define hideStems #t)

#(define (set-breathe bsign)
 (begin
  (set! currBSign bsign)
 (let* ((columnOfBSign (ly:grob-parent (ly:grob-parent (ly:grob-parent currBSign X) X) X))
   (system (ly:grob-parent columnOfBSign X))   
   (columns (ly:grob-object system 'columns))
   (lastColumn (ly:grob-array-ref columns (- (ly:grob-array-length columns) 1)))
  )
  (if (eqv? columnOfBSign lastColumn)
   (ly:grob-set-property! currBSign 'extra-offset '(0.89 . 0))
  )
 )
 )
)

#(define (staff-print staffgrob)
 (let* ((stencil (ly:staff-symbol::print staffgrob))
   (Xextent (ly:stencil-extent stencil X))
   (Yextent (ly:stencil-extent stencil Y))
            (X0 (car Xextent))
   (X1 (cdr Xextent))
   (X1sc (+ X1 0.89))
   )
   (ly:make-stencil (ly:stencil-expr (ly:stencil-scale stencil (/ (- X1sc X0) (- X1 X0)) 1.0))
                         (cons X0 X1sc)
                         Yextent
   )
 )
)

#(define (staff-break-callback staffgrob)
    (let*
  (
    (system (ly:grob-parent (ly:grob-parent staffgrob X) X))
    (columns (ly:grob-object system 'columns))
    (columns-count (ly:grob-array-length columns))
    (goI 1)
    (goJ 1)
  )
  (if (ly:grob? currBSign)
   (let* ((columnOfBSign (ly:grob-parent (ly:grob-parent (ly:grob-parent currBSign X) X) X))
      (lastColumn (ly:grob-array-ref columns (- columns-count 1)))
     )
     (if (eqv? columnOfBSign lastColumn)
      (ly:grob-set-property! currBSign 'extra-offset '(0.89 . 0))
      (set! currBSign 0)
     )
     (set! currBSign 0)
   )
  )
  (let loopI  ((i 0)
     )
       (if (&lt; i columns-count)
      (loopI (let* ((column (ly:grob-array-ref columns i))
          (columnname (grob-name column))
         )
         (if (and (eqv? columnname 'PaperColumn) (ly:grob-array? (ly:grob-object column 'elements)))
          (let*
           (
           (elems (ly:grob-object column 'elements))
           (elems-count (ly:grob-array-length elems))
           )
           (let loopJ ((j 0)
                )
                (if (&lt; j elems-count) 
               (loopJ (let* ((grob (ly:grob-array-ref elems j))
                   (grobname (grob-name grob))
                  )
                  (if (eqv? grobname 'NoteColumn)
                   (begin
                    (show-notes grob)
                    (set! j elems-count)
                   )
                  )
                  (1+ j)
                 )
               )
              )
           )
           (set! i columns-count)
          )
         )
         (1+ i)
        )
      )
     )
  )
 )
)

#(define (grob-name grob)
   (if (ly:grob? grob)
       (assoc-ref (ly:grob-property grob 'meta) 'name)
       #f))

#(define (show-flag flag)
 (ly:grob-set-property! flag 'transparent hideStems)
)
   
#(define (show-stem stem)
 (let*  ((beam (ly:grob-object stem 'beam))
   )
   (begin
    (ly:grob-set-property! stem 'transparent hideStems)
    (if (ly:grob? beam)(ly:grob-set-property! beam 'transparent hideStems))
   )
 )
)

#(define (show-note notehead)
 (let* ((stem (ly:grob-object (ly:grob-parent notehead X) 'stem))
        (beam (ly:grob-object stem 'beam))
     (flag (ly:grob-object notehead 'flag))
        (acci (ly:grob-object notehead 'accidental-grob))
        (dots (ly:grob-object notehead 'dot)))
 (begin
  (ly:grob-set-property! notehead 'transparent #f)
  (if (ly:grob? dots)(ly:grob-set-property! dots 'transparent #f))
  (if (ly:grob? acci)(ly:grob-set-property! acci 'transparent #f))
  (ly:grob-set-property! stem 'transparent hideStems)
  (if (ly:grob? beam)(ly:grob-set-property! beam 'transparent hideStems))
  (if (ly:grob? flag)(ly:grob-set-property! flag 'transparent hideStems))
 ))
)

#(define (show-notes note-column) 
 (begin 
  (let* ((elems (ly:grob-object note-column 'elements))
    (elems-count (ly:grob-array-length elems))
     ) 
       (let loop ((staff-pos-sum 0)
      (i 0)
        )
        (if (&lt; i elems-count)
       (loop (let* ((grob (ly:grob-array-ref elems i))
           (grobname (grob-name grob))
          )
          (if (eqv? grobname 'NoteHead)
           (show-note grob)
           (if (eqv? grobname 'Stem)
            (show-stem grob)
            (if (eqv? grobname 'Flag)
             (show-flag grob)
            )
           )
          )
         )
        (1+ i)
       )
      )
    )
  )
 )
)




leftAlign = {  \once \override LyricText #'self-alignment-X = #LEFT }
centerAlign = {   \once \override LyricText #'self-alignment-X = #CENTER }
rightAlign = {   \once \override LyricText #'self-alignment-X = #1 }
hairpinToBarline = { \override Hairpin #'to-barline = ##t  }
hairpinToNote = { \override Hairpin #'to-barline = ##f  }

alignAt =
#(define-music-function
     (parser location xOffset)
     (number?)
  (display xOffset)
   #{
      \once \override LyricText #'self-alignment-X = $xOffset
   #})

stemOn = { \revert Staff.Stem #'transparent  \revert Staff.Beam #'transparent \revert Staff.Flag #'transparent \revert Voice.Stem #'length }
stemOff = { \override Staff.Stem #'transparent = ##t \override Staff.Beam #'transparent = ##t \override Staff.Flag #'transparent = ##t \override Stem #'length = #0 }

bigNotes = { \override NoteHead #'font-size = #3 }
smallNotes = { \override NoteHead #'font-size = #-3 }
normalNotes = { \override NoteHead #'font-size = #0 }

bN = { \once \override NoteHead #'font-size = #3 }
sN = { \once \override NoteHead #'font-size = #-3 }
nN = { \once \override NoteHead #'font-size = #0 }

shOn = { \override NoteColumn #'force-hshift = #0.18 }
shOff = { \revert NoteColumn #'force-hshift }

midRest = { \override Voice.Rest #'staff-position = #0 }
shRest = { \revert Voice.Rest #'staff-position }

modernAccidentals = #(set-accidental-style 'modern)
modernCautionaryAccidentals = #(set-accidental-style 'modern-cautionary)
defaultAccidentals = #(set-accidental-style 'default)
pianoAccidentals = #(set-accidental-style 'piano)
voiceAccidentals = #(set-accidental-style 'voice)
modernVoiceAccidentals = #(set-accidental-style 'modern-voice)
modernVoiceCautionaryAccidentals = #(set-accidental-style 'modern-voice-cautionary)
pianoCautionaryAccidentals = #(set-accidental-style 'piano-cautionary)
neoModernAccidentals = #(set-accidental-style 'neo-modern)
neoModernCautionaryAccidentals = #(set-accidental-style 'neo-modern-cautionary)
dodecapohnicAccidentals = #(set-accidental-style 'dodecaphonic)
teachingAccidentals = #(set-accidental-style 'teaching)
noResetAccidentals = #(set-accidental-style 'no-reset)
forgetAccidentals = #(set-accidental-style 'forget)

go = {
\grace {s16 } 
}



chant = {
\key f \major
a'\breve \hideNotes a' a' a' \unHideNotes g'4 a' g'2 \divisioMaxima \divisioMaxima
f'4( g') a'\breve \hideNotes a' a' a' a' \unHideNotes a'4 a' g'( a') bes'2 \break \divisioMaxima
a'4 bes' c''\breve \unHideNotes bes'4 a' a' g'( fis' g' a') e'2 \finalis
\override Lyrics.LyricText.transparent = ##t f'\breve \hideNotes  \unHideNotes 
\hideNotes \grace {s16 } \finalis
}

verba = \lyricmode {
\alignAt #-0.722 "Magasztalja" \alignAt #-1.482 "az" \alignAt #-1.451 "én" \alignAt #-0.951 "lelkem" "az" "U" -- "rat,"
"és" \alignAt #-1.103 "örvendez" \alignAt #-0.951 "lelkem" \alignAt #-1.482 "az" \alignAt #-1.451 "én" \alignAt #-1.107 "üdvözítő" "Is" -- "te" -- "nem" -- "ben,"
"az" "én" \alignAt #-1.138 "üdvözí" -- "tő" "Is" -- "te" -- "nem" -- "ben!"
\alignAt #-0.574 "Recit"
}


\score {
  \new Staff &lt;&lt;
    #(set-accidental-style 'modern-cautionary)
    \new Voice = "melody" \chant
    \new Lyrics = "one" \lyricsto melody \verba
  &gt;&gt;
  \layout {
    ragged-last = ##f
    line-thickness = #(/ (ly:output-def-lookup $defaultpaper 'staff-space) 4)
    \context {
      \Staff
      \override BarLine  #'stencil = ##f
      \remove Time_signature_engraver
      \override Stem #'transparent = #hideStems
      \override Beam #'transparent = #hideStems
      \override Flag #'transparent = #hideStems
      \override StaffSymbol #'after-line-breaking = #staff-break-callback
      \override StaffSymbol #'thickness = #(/ (ly:output-def-lookup $defaultpaper 'staff-space) 3.0)
      \override StaffSymbol #'stencil = #staff-print
    }
    \context {
      \Voice
      \override Stem #'length = #0
   \override BreathingSign #'after-line-breaking = #set-breathe
    }
    \context {
      \Score
      \override  RehearsalMark #'self-alignment-X = #LEFT
      \override RehearsalMark #'break-align-symbols = #'(key-signature)
      \override  NonMusicalPaperColumn #'padding = #1
      skipBars= ##t
      barAlways = ##t
      lyricMelismaAlignment = #CENTER
    }
    \context {
      \Lyrics
      \override LyricSpace #'minimum-distance = #1
      \override LyricText #'font-size = #1.065
      \override LyricText #'font-name = #"TimesNewRoman"
      \override LyricText #'color = #(rgb-color 0 0 0)
      \override LyricHyphen #'color = #(rgb-color 0 0 0)
      \override LyricExtender #'color = #(rgb-color 0 0 0)
    }
  }
}

\header {
tagline = "" 
title = ""
}
</LilyPond>
  <EasyChant>Magasztalja az én lelkem az U- rat,
m0                       r4 m  r2  

és   örvendez lelkem az én üdvözítő Is- te- nem- ben,
d4 r m0                             m   m   r m  f2  
/
az én üdvözí- tő Is- te- nem-     ben!  
m4 f  s0      f  m   m   r di r m t,2 ||

Recit
\"override Lyrics.LyricText.transparent = ##t" d0

</EasyChant>
  <Clef>0</Clef>
  <Key>8</Key>
  <LyricsSpacing>1</LyricsSpacing>
  <TimeOption>0</TimeOption>
  <TimeText>N/A (cadenza)</TimeText>
  <StaffSizeOption>2</StaffSizeOption>
  <StaffSizeText>14</StaffSizeText>
  <LyricsSize>1.1</LyricsSize>
  <LyricsFontName>Times New Roman</LyricsFontName>
  <LyricsFontBold>False</LyricsFontBold>
  <LyricsFontItalic>False</LyricsFontItalic>
  <LyricsFontColor>Color [A=255, R=0, G=0, B=0]</LyricsFontColor>
  <LineSpacing>12</LineSpacing>
  <LineThickness>4</LineThickness>
  <Options>0,1,2,6,7,8</Options>
  <pages>1</pages>
  <page>1</page>
</item>

Vers: Mert megtekintette szolgálójának alázatosságát; íme, mostantól boldognak hirdet engem minden nemzedék. (Lk 1,46-48)

Ft. alleluja, 1. hang

<?xml version="1.0" encoding="utf-16"?>
<item>
  <title>Alleluja, 1. hang</title>
  <LilyPond>\version "2.18.2"

\include "gregorian.ly"

\paper {
  paper-width = 123.1194\mm
  paper-height = 184.6262\mm
  top-margin = 0\mm
  bottom-margin = 0\mm
  left-margin = 0\mm
  right-margin = 2\mm
  ragged-bottom = ##t
  print-page-number = ##f
  page-breaking = #ly:minimal-breaking
  system-system-spacing =
    #'((basic-distance . 12)
       (minimum-distance . 12)
       (padding . 0)
       (stretchability . 0))
%  page-breaking-system-system-spacing = #'((padding . 2))
  markup-system-spacing = #'((padding . 92.49))
  top-markup-spacing = #'((basic-distance . 0) (minimum-distance . 0) (padding . 0))
}

#(set-global-staff-size 14)

#(define currBSign 0)
#(define hideStems #t)

#(define (set-breathe bsign)
 (begin
  (set! currBSign bsign)
 (let* ((columnOfBSign (ly:grob-parent (ly:grob-parent (ly:grob-parent currBSign X) X) X))
   (system (ly:grob-parent columnOfBSign X))   
   (columns (ly:grob-object system 'columns))
   (lastColumn (ly:grob-array-ref columns (- (ly:grob-array-length columns) 1)))
  )
  (if (eqv? columnOfBSign lastColumn)
   (ly:grob-set-property! currBSign 'extra-offset '(0.89 . 0))
  )
 )
 )
)

#(define (staff-print staffgrob)
 (let* ((stencil (ly:staff-symbol::print staffgrob))
   (Xextent (ly:stencil-extent stencil X))
   (Yextent (ly:stencil-extent stencil Y))
            (X0 (car Xextent))
   (X1 (cdr Xextent))
   (X1sc (+ X1 0.89))
   )
   (ly:make-stencil (ly:stencil-expr (ly:stencil-scale stencil (/ (- X1sc X0) (- X1 X0)) 1.0))
                         (cons X0 X1sc)
                         Yextent
   )
 )
)

#(define (staff-break-callback staffgrob)
    (let*
  (
    (system (ly:grob-parent (ly:grob-parent staffgrob X) X))
    (columns (ly:grob-object system 'columns))
    (columns-count (ly:grob-array-length columns))
    (goI 1)
    (goJ 1)
  )
  (if (ly:grob? currBSign)
   (let* ((columnOfBSign (ly:grob-parent (ly:grob-parent (ly:grob-parent currBSign X) X) X))
      (lastColumn (ly:grob-array-ref columns (- columns-count 1)))
     )
     (if (eqv? columnOfBSign lastColumn)
      (ly:grob-set-property! currBSign 'extra-offset '(0.89 . 0))
      (set! currBSign 0)
     )
     (set! currBSign 0)
   )
  )
  (let loopI  ((i 0)
     )
       (if (&lt; i columns-count)
      (loopI (let* ((column (ly:grob-array-ref columns i))
          (columnname (grob-name column))
         )
         (if (and (eqv? columnname 'PaperColumn) (ly:grob-array? (ly:grob-object column 'elements)))
          (let*
           (
           (elems (ly:grob-object column 'elements))
           (elems-count (ly:grob-array-length elems))
           )
           (let loopJ ((j 0)
                )
                (if (&lt; j elems-count) 
               (loopJ (let* ((grob (ly:grob-array-ref elems j))
                   (grobname (grob-name grob))
                  )
                  (if (eqv? grobname 'NoteColumn)
                   (begin
                    (show-notes grob)
                    (set! j elems-count)
                   )
                  )
                  (1+ j)
                 )
               )
              )
           )
           (set! i columns-count)
          )
         )
         (1+ i)
        )
      )
     )
  )
 )
)

#(define (grob-name grob)
   (if (ly:grob? grob)
       (assoc-ref (ly:grob-property grob 'meta) 'name)
       #f))

#(define (show-flag flag)
 (ly:grob-set-property! flag 'transparent hideStems)
)
   
#(define (show-stem stem)
 (let*  ((beam (ly:grob-object stem 'beam))
   )
   (begin
    (ly:grob-set-property! stem 'transparent hideStems)
    (if (ly:grob? beam)(ly:grob-set-property! beam 'transparent hideStems))
   )
 )
)

#(define (show-note notehead)
 (let* ((stem (ly:grob-object (ly:grob-parent notehead X) 'stem))
        (beam (ly:grob-object stem 'beam))
     (flag (ly:grob-object notehead 'flag))
        (acci (ly:grob-object notehead 'accidental-grob))
        (dots (ly:grob-object notehead 'dot)))
 (begin
  (ly:grob-set-property! notehead 'transparent #f)
  (if (ly:grob? dots)(ly:grob-set-property! dots 'transparent #f))
  (if (ly:grob? acci)(ly:grob-set-property! acci 'transparent #f))
  (ly:grob-set-property! stem 'transparent hideStems)
  (if (ly:grob? beam)(ly:grob-set-property! beam 'transparent hideStems))
  (if (ly:grob? flag)(ly:grob-set-property! flag 'transparent hideStems))
 ))
)

#(define (show-notes note-column) 
 (begin 
  (let* ((elems (ly:grob-object note-column 'elements))
    (elems-count (ly:grob-array-length elems))
     ) 
       (let loop ((staff-pos-sum 0)
      (i 0)
        )
        (if (&lt; i elems-count)
       (loop (let* ((grob (ly:grob-array-ref elems i))
           (grobname (grob-name grob))
          )
          (if (eqv? grobname 'NoteHead)
           (show-note grob)
           (if (eqv? grobname 'Stem)
            (show-stem grob)
            (if (eqv? grobname 'Flag)
             (show-flag grob)
            )
           )
          )
         )
        (1+ i)
       )
      )
    )
  )
 )
)


% GREGORIAN REPEAT BARS BEGIN
finalisLeftRepeat = {
  \once \override BreathingSign.stencil = 
    #(lambda (grob)
       (ly:stencil-combine-at-edge
        (bar-line::compound-bar-line grob ":" '(-2.0 . 2.0))
        X RIGHT
        (ly:breathing-sign::finalis grob)
        0.3))
  \once \override BreathingSign.Y-offset = #0
  % Workaround: add padding.  Correct fix would be spacing engine handle this.
  \once \override BreathingSign.minimum-X-extent = #'(-1.0 . 0.0)
  \once \override BreathingSign.minimum-Y-extent = #'(-2.5 . 2.5)

  \breathe
}

finalisRightRepeat = {
  \once \override BreathingSign.stencil = 
    #(lambda (grob)
       (ly:stencil-combine-at-edge
        (ly:breathing-sign::finalis grob)
        X RIGHT
        (bar-line::compound-bar-line grob ":" '(-2.0 . 2.0))
        0.3))
  \once \override BreathingSign.Y-offset = #0
  % Workaround: add padding.  Correct fix would be spacing engine handle this.
  \once \override BreathingSign.minimum-X-extent = #'(-1.0 . 0.0)
  \once \override BreathingSign.minimum-Y-extent = #'(-2.5 . 2.5)

  \breathe
}
% GREGORIAN REPEAT BARS END


leftAlign = {  \once \override LyricText #'self-alignment-X = #LEFT }
centerAlign = {   \once \override LyricText #'self-alignment-X = #CENTER }
rightAlign = {   \once \override LyricText #'self-alignment-X = #1 }
hairpinToBarline = { \override Hairpin #'to-barline = ##t  }
hairpinToNote = { \override Hairpin #'to-barline = ##f  }

alignAt =
#(define-music-function
     (parser location xOffset)
     (number?)
  (display xOffset)
   #{
      \once \override LyricText #'self-alignment-X = $xOffset
   #})

stemOn = { \revert Staff.Stem #'transparent  \revert Staff.Beam #'transparent \revert Staff.Flag #'transparent \revert Voice.Stem #'length }
stemOff = { \override Staff.Stem #'transparent = ##t \override Staff.Beam #'transparent = ##t \override Staff.Flag #'transparent = ##t \override Stem #'length = #0 }

bigNotes = { \override NoteHead #'font-size = #3 }
smallNotes = { \override NoteHead #'font-size = #-3 }
normalNotes = { \override NoteHead #'font-size = #0 }

bN = { \once \override NoteHead #'font-size = #3 }
sN = { \once \override NoteHead #'font-size = #-3 }
nN = { \once \override NoteHead #'font-size = #0 }

shOn = { \override NoteColumn #'force-hshift = #0.18 }
shOff = { \revert NoteColumn #'force-hshift }

midRest = { \override Voice.Rest #'staff-position = #0 }
shRest = { \revert Voice.Rest #'staff-position }

modernAccidentals = #(set-accidental-style 'modern)
modernCautionaryAccidentals = #(set-accidental-style 'modern-cautionary)
defaultAccidentals = #(set-accidental-style 'default)
pianoAccidentals = #(set-accidental-style 'piano)
voiceAccidentals = #(set-accidental-style 'voice)
modernVoiceAccidentals = #(set-accidental-style 'modern-voice)
modernVoiceCautionaryAccidentals = #(set-accidental-style 'modern-voice-cautionary)
pianoCautionaryAccidentals = #(set-accidental-style 'piano-cautionary)
neoModernAccidentals = #(set-accidental-style 'neo-modern)
neoModernCautionaryAccidentals = #(set-accidental-style 'neo-modern-cautionary)
dodecapohnicAccidentals = #(set-accidental-style 'dodecaphonic)
teachingAccidentals = #(set-accidental-style 'teaching)
noResetAccidentals = #(set-accidental-style 'no-reset)
forgetAccidentals = #(set-accidental-style 'forget)

go = {
\grace {s16 } 
}



chant = {
\key f \major
a'4 bes' c''( a') bes'2( a'4 c'' bes' a' g'2) \divisioMaxima \divisioMaxima \divisioMaxima \divisioMaxima \divisioMaxima \divisioMaxima \divisioMaxima \divisioMaxima \divisioMaxima
g'4( a' g' fis' g' a' bes'2 c''4 bes') a' g' c''2( bes'4 a' g'2) \divisioMaxima
g'2 a' g'4( f' e' d') c'2 \finalisLeftRepeat
\override Lyrics.LyricText.transparent = ##t f'\breve \hideNotes  \unHideNotes 
\hideNotes \grace {s16 } \finalis
}

verba = \lyricmode {
"Al" -- "le" -- "lu" -- "ja,"
"al" -- "le" -- "lu" -- "ja,"
"al" -- "le" -- "lu" -- "ja."
\alignAt #-0.574 "Recit"
}


\score {
  \new Staff &lt;&lt;
    #(set-accidental-style 'modern-cautionary)
    \new Voice = "melody" \chant
    \new Lyrics = "one" \lyricsto melody \verba
  &gt;&gt;
  \layout {
    ragged-last = ##f
    line-thickness = #(/ (ly:output-def-lookup $defaultpaper 'staff-space) 4)
    \context {
      \Staff
      \override BarLine  #'stencil = ##f
      \remove Time_signature_engraver
      \override Stem #'transparent = #hideStems
      \override Beam #'transparent = #hideStems
      \override Flag #'transparent = #hideStems
      \override StaffSymbol #'after-line-breaking = #staff-break-callback
      \override StaffSymbol #'thickness = #(/ (ly:output-def-lookup $defaultpaper 'staff-space) 3.0)
      \override StaffSymbol #'stencil = #staff-print
    }
    \context {
      \Voice
      \override Stem #'length = #0
   \override BreathingSign #'after-line-breaking = #set-breathe
    }
    \context {
      \Score
      \override  RehearsalMark #'self-alignment-X = #LEFT
      \override RehearsalMark #'break-align-symbols = #'(key-signature)
      \override  NonMusicalPaperColumn #'padding = #0.6
      skipBars= ##t
      barAlways = ##t
      lyricMelismaAlignment = #CENTER
    }
    \context {
      \Lyrics
      \override LyricSpace #'minimum-distance = #0.6
      \override LyricText #'font-size = #1.065
      \override LyricText #'font-name = #"TimesNewRoman"
      \override LyricText #'color = #(rgb-color 0 0 0)
      \override LyricHyphen #'color = #(rgb-color 0 0 0)
      \override LyricExtender #'color = #(rgb-color 0 0 0)
    }
  }
}

\header {
tagline = "" 
title = ""
}
</LilyPond>
  <EasyChant>Al- le- lu- ja,           
m4  f   s m f2 m4 s f m r2

al-                   le- lu- ja,       
r4 m r di r m f2 s4 f m   r   s2 f4 m r2

al- le- lu-        ja.
r2  m   r4 d t, l, s,2


:||
Recit
\"override Lyrics.LyricText.transparent = ##t" d0</EasyChant>
  <Clef>0</Clef>
  <Key>8</Key>
  <LyricsSpacing>0.6</LyricsSpacing>
  <TimeOption>0</TimeOption>
  <TimeText>N/A (cadenza)</TimeText>
  <StaffSizeOption>2</StaffSizeOption>
  <StaffSizeText>14</StaffSizeText>
  <LyricsSize>1.1</LyricsSize>
  <LyricsFontName>Times New Roman</LyricsFontName>
  <LyricsFontBold>False</LyricsFontBold>
  <LyricsFontItalic>False</LyricsFontItalic>
  <LyricsFontColor>Color [A=255, R=0, G=0, B=0]</LyricsFontColor>
  <LineSpacing>12</LineSpacing>
  <LineThickness>4</LineThickness>
  <Options>0,1,2,6,7,8</Options>
  <pages>1</pages>
  <page>1</page>
</item>

Vers: Isten, aki megtorlást adsz nekem, a népeket alám vetetted, én szabadítóm haragos ellenségeimtől.

Vers: Aki nagy szabadulást adtál királyodnak, és irgalmasságot cselekszel fölkenteddel, Dáviddal, és az Ő ivadékával mindörökké. (17. zsoltár)

Ün. alleluja, 8. hang

<?xml version="1.0" encoding="utf-16"?>
<item>
  <title>Alleluja, 8. hang2</title>
  <LilyPond>\version "2.18.2"

\include "gregorian.ly"

\paper {
  paper-width = 123.0136\mm
  paper-height = 184.6262\mm
  top-margin = 0\mm
  bottom-margin = 0\mm
  left-margin = 0\mm
  right-margin = 2\mm
  ragged-bottom = ##t
  print-page-number = ##f
  page-breaking = #ly:minimal-breaking
  system-system-spacing =
    #'((basic-distance . 12)
       (minimum-distance . 12)
       (padding . 0)
       (stretchability . 0))
%  page-breaking-system-system-spacing = #'((padding . 2))
  markup-system-spacing = #'((padding . 112.83))
  top-markup-spacing = #'((basic-distance . 0) (minimum-distance . 0) (padding . 0))
}

#(set-global-staff-size 14)

#(define currBSign 0)
#(define hideStems #t)

#(define (set-breathe bsign)
 (begin
  (set! currBSign bsign)
 (let* ((columnOfBSign (ly:grob-parent (ly:grob-parent (ly:grob-parent currBSign X) X) X))
   (system (ly:grob-parent columnOfBSign X))   
   (columns (ly:grob-object system 'columns))
   (lastColumn (ly:grob-array-ref columns (- (ly:grob-array-length columns) 1)))
  )
  (if (eqv? columnOfBSign lastColumn)
   (ly:grob-set-property! currBSign 'extra-offset '(0.89 . 0))
  )
 )
 )
)

#(define (staff-print staffgrob)
 (let* ((stencil (ly:staff-symbol::print staffgrob))
   (Xextent (ly:stencil-extent stencil X))
   (Yextent (ly:stencil-extent stencil Y))
            (X0 (car Xextent))
   (X1 (cdr Xextent))
   (X1sc (+ X1 0.89))
   )
   (ly:make-stencil (ly:stencil-expr (ly:stencil-scale stencil (/ (- X1sc X0) (- X1 X0)) 1.0))
                         (cons X0 X1sc)
                         Yextent
   )
 )
)

#(define (staff-break-callback staffgrob)
    (let*
  (
    (system (ly:grob-parent (ly:grob-parent staffgrob X) X))
    (columns (ly:grob-object system 'columns))
    (columns-count (ly:grob-array-length columns))
    (goI 1)
    (goJ 1)
  )
  (if (ly:grob? currBSign)
   (let* ((columnOfBSign (ly:grob-parent (ly:grob-parent (ly:grob-parent currBSign X) X) X))
      (lastColumn (ly:grob-array-ref columns (- columns-count 1)))
     )
     (if (eqv? columnOfBSign lastColumn)
      (ly:grob-set-property! currBSign 'extra-offset '(0.89 . 0))
      (set! currBSign 0)
     )
     (set! currBSign 0)
   )
  )
  (let loopI  ((i 0)
     )
       (if (&lt; i columns-count)
      (loopI (let* ((column (ly:grob-array-ref columns i))
          (columnname (grob-name column))
         )
         (if (and (eqv? columnname 'PaperColumn) (ly:grob-array? (ly:grob-object column 'elements)))
          (let*
           (
           (elems (ly:grob-object column 'elements))
           (elems-count (ly:grob-array-length elems))
           )
           (let loopJ ((j 0)
                )
                (if (&lt; j elems-count) 
               (loopJ (let* ((grob (ly:grob-array-ref elems j))
                   (grobname (grob-name grob))
                  )
                  (if (eqv? grobname 'NoteColumn)
                   (begin
                    (show-notes grob)
                    (set! j elems-count)
                   )
                  )
                  (1+ j)
                 )
               )
              )
           )
           (set! i columns-count)
          )
         )
         (1+ i)
        )
      )
     )
  )
 )
)

#(define (grob-name grob)
   (if (ly:grob? grob)
       (assoc-ref (ly:grob-property grob 'meta) 'name)
       #f))

#(define (show-flag flag)
 (ly:grob-set-property! flag 'transparent hideStems)
)
   
#(define (show-stem stem)
 (let*  ((beam (ly:grob-object stem 'beam))
   )
   (begin
    (ly:grob-set-property! stem 'transparent hideStems)
    (if (ly:grob? beam)(ly:grob-set-property! beam 'transparent hideStems))
   )
 )
)

#(define (show-note notehead)
 (let* ((stem (ly:grob-object (ly:grob-parent notehead X) 'stem))
        (beam (ly:grob-object stem 'beam))
     (flag (ly:grob-object notehead 'flag))
        (acci (ly:grob-object notehead 'accidental-grob))
        (dots (ly:grob-object notehead 'dot)))
 (begin
  (ly:grob-set-property! notehead 'transparent #f)
  (if (ly:grob? dots)(ly:grob-set-property! dots 'transparent #f))
  (if (ly:grob? acci)(ly:grob-set-property! acci 'transparent #f))
  (ly:grob-set-property! stem 'transparent hideStems)
  (if (ly:grob? beam)(ly:grob-set-property! beam 'transparent hideStems))
  (if (ly:grob? flag)(ly:grob-set-property! flag 'transparent hideStems))
 ))
)

#(define (show-notes note-column) 
 (begin 
  (let* ((elems (ly:grob-object note-column 'elements))
    (elems-count (ly:grob-array-length elems))
     ) 
       (let loop ((staff-pos-sum 0)
      (i 0)
        )
        (if (&lt; i elems-count)
       (loop (let* ((grob (ly:grob-array-ref elems i))
           (grobname (grob-name grob))
          )
          (if (eqv? grobname 'NoteHead)
           (show-note grob)
           (if (eqv? grobname 'Stem)
            (show-stem grob)
            (if (eqv? grobname 'Flag)
             (show-flag grob)
            )
           )
          )
         )
        (1+ i)
       )
      )
    )
  )
 )
)




leftAlign = {  \once \override LyricText #'self-alignment-X = #LEFT }
centerAlign = {   \once \override LyricText #'self-alignment-X = #CENTER }
rightAlign = {   \once \override LyricText #'self-alignment-X = #1 }
hairpinToBarline = { \override Hairpin #'to-barline = ##t  }
hairpinToNote = { \override Hairpin #'to-barline = ##f  }

alignAt =
#(define-music-function
     (parser location xOffset)
     (number?)
  (display xOffset)
   #{
      \once \override LyricText #'self-alignment-X = $xOffset
   #})

stemOn = { \revert Staff.Stem #'transparent  \revert Staff.Beam #'transparent \revert Staff.Flag #'transparent \revert Voice.Stem #'length }
stemOff = { \override Staff.Stem #'transparent = ##t \override Staff.Beam #'transparent = ##t \override Staff.Flag #'transparent = ##t \override Stem #'length = #0 }

bigNotes = { \override NoteHead #'font-size = #3 }
smallNotes = { \override NoteHead #'font-size = #-3 }
normalNotes = { \override NoteHead #'font-size = #0 }

bN = { \once \override NoteHead #'font-size = #3 }
sN = { \once \override NoteHead #'font-size = #-3 }
nN = { \once \override NoteHead #'font-size = #0 }

shOn = { \override NoteColumn #'force-hshift = #0.18 }
shOff = { \revert NoteColumn #'force-hshift }

midRest = { \override Voice.Rest #'staff-position = #0 }
shRest = { \revert Voice.Rest #'staff-position }

modernAccidentals = #(set-accidental-style 'modern)
modernCautionaryAccidentals = #(set-accidental-style 'modern-cautionary)
defaultAccidentals = #(set-accidental-style 'default)
pianoAccidentals = #(set-accidental-style 'piano)
voiceAccidentals = #(set-accidental-style 'voice)
modernVoiceAccidentals = #(set-accidental-style 'modern-voice)
modernVoiceCautionaryAccidentals = #(set-accidental-style 'modern-voice-cautionary)
pianoCautionaryAccidentals = #(set-accidental-style 'piano-cautionary)
neoModernAccidentals = #(set-accidental-style 'neo-modern)
neoModernCautionaryAccidentals = #(set-accidental-style 'neo-modern-cautionary)
dodecapohnicAccidentals = #(set-accidental-style 'dodecaphonic)
teachingAccidentals = #(set-accidental-style 'teaching)
noResetAccidentals = #(set-accidental-style 'no-reset)
forgetAccidentals = #(set-accidental-style 'forget)

go = {
\grace {s16 } 
}



chant = {
\key f \major
f'4 f' e'( f') g'2 f'4( g' f' e' f' g' a') a' e'2( f'4 e') d'2 \divisioMaxima \divisioMaxima \divisioMaxima
a'4( g' a'2) e'4( f' g'2) f'4( e') d'2 \finalis
\override Lyrics.LyricText.transparent = ##t d'\breve \hideNotes  \unHideNotes 
\hideNotes \grace {s16 } \finalis
}

verba = \lyricmode {
"Al" -- "le" -- "lu" -- "ja," "al" -- "le" -- "lu" -- "ja,"
"al" -- "le" -- "lu" -- "ja."
\alignAt #-0.574 "Recit"
}


\score {
  \new Staff &lt;&lt;
    #(set-accidental-style 'modern-cautionary)
    \new Voice = "melody" \chant
    \new Lyrics = "one" \lyricsto melody \verba
  &gt;&gt;
  \layout {
    line-thickness = #(/ (ly:output-def-lookup $defaultpaper 'staff-space) 4)
    \context {
      \Staff
      \override BarLine  #'stencil = ##f
      \remove Time_signature_engraver
      \override Stem #'transparent = #hideStems
      \override Beam #'transparent = #hideStems
      \override Flag #'transparent = #hideStems
      \override StaffSymbol #'after-line-breaking = #staff-break-callback
      \override StaffSymbol #'thickness = #(/ (ly:output-def-lookup $defaultpaper 'staff-space) 3.0)
      \override StaffSymbol #'stencil = #staff-print
    }
    \context {
      \Voice
      \override Stem #'length = #0
   \override BreathingSign #'after-line-breaking = #set-breathe
    }
    \context {
      \Score
      \override  RehearsalMark #'self-alignment-X = #LEFT
      \override RehearsalMark #'break-align-symbols = #'(key-signature)
      \override  NonMusicalPaperColumn #'padding = #1
      skipBars= ##t
      barAlways = ##t
      lyricMelismaAlignment = #CENTER
    }
    \context {
      \Lyrics
      \override LyricSpace #'minimum-distance = #1
      \override LyricText #'font-size = #1.065
      \override LyricText #'font-name = #"TimesNewRoman"
      \override LyricText #'color = #(rgb-color 0 0 0)
      \override LyricHyphen #'color = #(rgb-color 0 0 0)
      \override LyricExtender #'color = #(rgb-color 0 0 0)
    }
  }
}

\header {
tagline = "" 
title = ""
}
</LilyPond>
  <EasyChant>Al- le- lu-  ja, al-             le- lu-       ja,
d4  d   t, d r2  d4 r d t, d r m m   t,2 d4 t, l,2

al-     le-       lu-   ja.   
m4 r m2 t,4 d r2  d4 t, l,2 ||

Recit
\"override Lyrics.LyricText.transparent = ##t" l,0

</EasyChant>
  <Clef>0</Clef>
  <Key>8</Key>
  <LyricsSpacing>1</LyricsSpacing>
  <TimeOption>0</TimeOption>
  <TimeText>N/A (cadenza)</TimeText>
  <StaffSizeOption>2</StaffSizeOption>
  <StaffSizeText>14</StaffSizeText>
  <LyricsSize>1.1</LyricsSize>
  <LyricsFontName>Times New Roman</LyricsFontName>
  <LyricsFontBold>False</LyricsFontBold>
  <LyricsFontItalic>False</LyricsFontItalic>
  <LyricsFontColor>Color [A=255, R=0, G=0, B=0]</LyricsFontColor>
  <LineSpacing>12</LineSpacing>
  <LineThickness>4</LineThickness>
  <Options>0,1,2,4,6,7,8</Options>
  <pages>1</pages>
  <page>1</page>
</item>

Vers: Halljad, leányom és lásd, és hajtsd ide füledet!

Vers: A nép gazdagjai könyörögni fognak színed előtt. (44. zsoltár)

Magasztalás

<?xml version="1.0" encoding="utf-16"?>
<item>
  <title>Magasztalás4</title>
  <LilyPond>\version "2.18.2"

\include "gregorian.ly"

\paper {
  paper-width = 123.0136\mm
  paper-height = 184.6262\mm
  top-margin = 0\mm
  bottom-margin = 0\mm
  left-margin = 0\mm
  right-margin = 2\mm
  ragged-bottom = ##t
  print-page-number = ##f
  page-breaking = #ly:minimal-breaking
  system-system-spacing =
    #'((basic-distance . 12)
       (minimum-distance . 12)
       (padding . 0)
       (stretchability . 0))
%  page-breaking-system-system-spacing = #'((padding . 2))
  markup-system-spacing = #'((padding . 112.83))
  top-markup-spacing = #'((basic-distance . 0) (minimum-distance . 0) (padding . 0))
}

#(set-global-staff-size 14)

#(define currBSign 0)
#(define hideStems #t)

#(define (set-breathe bsign)
 (begin
  (set! currBSign bsign)
 (let* ((columnOfBSign (ly:grob-parent (ly:grob-parent (ly:grob-parent currBSign X) X) X))
   (system (ly:grob-parent columnOfBSign X))   
   (columns (ly:grob-object system 'columns))
   (lastColumn (ly:grob-array-ref columns (- (ly:grob-array-length columns) 1)))
  )
  (if (eqv? columnOfBSign lastColumn)
   (ly:grob-set-property! currBSign 'extra-offset '(0.89 . 0))
  )
 )
 )
)

#(define (staff-print staffgrob)
 (let* ((stencil (ly:staff-symbol::print staffgrob))
   (Xextent (ly:stencil-extent stencil X))
   (Yextent (ly:stencil-extent stencil Y))
            (X0 (car Xextent))
   (X1 (cdr Xextent))
   (X1sc (+ X1 0.89))
   )
   (ly:make-stencil (ly:stencil-expr (ly:stencil-scale stencil (/ (- X1sc X0) (- X1 X0)) 1.0))
                         (cons X0 X1sc)
                         Yextent
   )
 )
)

#(define (staff-break-callback staffgrob)
    (let*
  (
    (system (ly:grob-parent (ly:grob-parent staffgrob X) X))
    (columns (ly:grob-object system 'columns))
    (columns-count (ly:grob-array-length columns))
    (goI 1)
    (goJ 1)
  )
  (if (ly:grob? currBSign)
   (let* ((columnOfBSign (ly:grob-parent (ly:grob-parent (ly:grob-parent currBSign X) X) X))
      (lastColumn (ly:grob-array-ref columns (- columns-count 1)))
     )
     (if (eqv? columnOfBSign lastColumn)
      (ly:grob-set-property! currBSign 'extra-offset '(0.89 . 0))
      (set! currBSign 0)
     )
     (set! currBSign 0)
   )
  )
  (let loopI  ((i 0)
     )
       (if (&lt; i columns-count)
      (loopI (let* ((column (ly:grob-array-ref columns i))
          (columnname (grob-name column))
         )
         (if (and (eqv? columnname 'PaperColumn) (ly:grob-array? (ly:grob-object column 'elements)))
          (let*
           (
           (elems (ly:grob-object column 'elements))
           (elems-count (ly:grob-array-length elems))
           )
           (let loopJ ((j 0)
                )
                (if (&lt; j elems-count) 
               (loopJ (let* ((grob (ly:grob-array-ref elems j))
                   (grobname (grob-name grob))
                  )
                  (if (eqv? grobname 'NoteColumn)
                   (begin
                    (show-notes grob)
                    (set! j elems-count)
                   )
                  )
                  (1+ j)
                 )
               )
              )
           )
           (set! i columns-count)
          )
         )
         (1+ i)
        )
      )
     )
  )
 )
)

#(define (grob-name grob)
   (if (ly:grob? grob)
       (assoc-ref (ly:grob-property grob 'meta) 'name)
       #f))

#(define (show-flag flag)
 (ly:grob-set-property! flag 'transparent hideStems)
)
   
#(define (show-stem stem)
 (let*  ((beam (ly:grob-object stem 'beam))
   )
   (begin
    (ly:grob-set-property! stem 'transparent hideStems)
    (if (ly:grob? beam)(ly:grob-set-property! beam 'transparent hideStems))
   )
 )
)

#(define (show-note notehead)
 (let* ((stem (ly:grob-object (ly:grob-parent notehead X) 'stem))
        (beam (ly:grob-object stem 'beam))
     (flag (ly:grob-object notehead 'flag))
        (acci (ly:grob-object notehead 'accidental-grob))
        (dots (ly:grob-object notehead 'dot)))
 (begin
  (ly:grob-set-property! notehead 'transparent #f)
  (if (ly:grob? dots)(ly:grob-set-property! dots 'transparent #f))
  (if (ly:grob? acci)(ly:grob-set-property! acci 'transparent #f))
  (ly:grob-set-property! stem 'transparent hideStems)
  (if (ly:grob? beam)(ly:grob-set-property! beam 'transparent hideStems))
  (if (ly:grob? flag)(ly:grob-set-property! flag 'transparent hideStems))
 ))
)

#(define (show-notes note-column) 
 (begin 
  (let* ((elems (ly:grob-object note-column 'elements))
    (elems-count (ly:grob-array-length elems))
     ) 
       (let loop ((staff-pos-sum 0)
      (i 0)
        )
        (if (&lt; i elems-count)
       (loop (let* ((grob (ly:grob-array-ref elems i))
           (grobname (grob-name grob))
          )
          (if (eqv? grobname 'NoteHead)
           (show-note grob)
           (if (eqv? grobname 'Stem)
            (show-stem grob)
            (if (eqv? grobname 'Flag)
             (show-flag grob)
            )
           )
          )
         )
        (1+ i)
       )
      )
    )
  )
 )
)




leftAlign = {  \once \override LyricText #'self-alignment-X = #LEFT }
centerAlign = {   \once \override LyricText #'self-alignment-X = #CENTER }
rightAlign = {   \once \override LyricText #'self-alignment-X = #1 }
hairpinToBarline = { \override Hairpin #'to-barline = ##t  }
hairpinToNote = { \override Hairpin #'to-barline = ##f  }

alignAt =
#(define-music-function
     (parser location xOffset)
     (number?)
  (display xOffset)
   #{
      \once \override LyricText #'self-alignment-X = $xOffset
   #})

stemOn = { \revert Staff.Stem #'transparent  \revert Staff.Beam #'transparent \revert Staff.Flag #'transparent \revert Voice.Stem #'length }
stemOff = { \override Staff.Stem #'transparent = ##t \override Staff.Beam #'transparent = ##t \override Staff.Flag #'transparent = ##t \override Stem #'length = #0 }

bigNotes = { \override NoteHead #'font-size = #3 }
smallNotes = { \override NoteHead #'font-size = #-3 }
normalNotes = { \override NoteHead #'font-size = #0 }

bN = { \once \override NoteHead #'font-size = #3 }
sN = { \once \override NoteHead #'font-size = #-3 }
nN = { \once \override NoteHead #'font-size = #0 }

shOn = { \override NoteColumn #'force-hshift = #0.18 }
shOff = { \revert NoteColumn #'force-hshift }

midRest = { \override Voice.Rest #'staff-position = #0 }
shRest = { \revert Voice.Rest #'staff-position }

modernAccidentals = #(set-accidental-style 'modern)
modernCautionaryAccidentals = #(set-accidental-style 'modern-cautionary)
defaultAccidentals = #(set-accidental-style 'default)
pianoAccidentals = #(set-accidental-style 'piano)
voiceAccidentals = #(set-accidental-style 'voice)
modernVoiceAccidentals = #(set-accidental-style 'modern-voice)
modernVoiceCautionaryAccidentals = #(set-accidental-style 'modern-voice-cautionary)
pianoCautionaryAccidentals = #(set-accidental-style 'piano-cautionary)
neoModernAccidentals = #(set-accidental-style 'neo-modern)
neoModernCautionaryAccidentals = #(set-accidental-style 'neo-modern-cautionary)
dodecapohnicAccidentals = #(set-accidental-style 'dodecaphonic)
teachingAccidentals = #(set-accidental-style 'teaching)
noResetAccidentals = #(set-accidental-style 'no-reset)
forgetAccidentals = #(set-accidental-style 'forget)

go = {
\grace {s16 } 
}



chant = {
\key f \major
d'4( e' f'2) e' d' f'4( g' a'2) g' f' \divisioMaxima
a'2 bes' a'( g') f' e' d' \divisioMaxima
g'4 g' \break g'\breve \unHideNotes a'4 bes' a'2( g'4 a') e'2 \divisioMaxima
f'4 g' a'\breve \hideNotes a' a' \unHideNotes g'4 a'2 g'4( f' e' f') d'2 
\hideNotes \grace {s16 } \finalis
}

verba = \lyricmode {
"Ma" -- "gasz" -- "tald," "ma" -- "gasz" -- "tald,"
"én" "lel" -- "kem," "én" "lel" -- "kem,"
"az" "Úr" \alignAt #-1.102 "egyházába" "be" -- "ve" -- "ze" -- "tett"
"és" "fő" -- \alignAt #-0.604 "pap" \alignAt #-0.828 "kezeivel" \alignAt #-0.339 "meg" -- "ál" -- "dott" "Szű" -- "zet!"
}


\score {
  \new Staff &lt;&lt;
    #(set-accidental-style 'modern-cautionary)
    \new Voice = "melody" \chant
    \new Lyrics = "one" \lyricsto melody \verba
  &gt;&gt;
  \layout {
    ragged-last = ##f
    line-thickness = #(/ (ly:output-def-lookup $defaultpaper 'staff-space) 4)
    \context {
      \Staff
      \override BarLine  #'stencil = ##f
      \remove Time_signature_engraver
      \override Stem #'transparent = #hideStems
      \override Beam #'transparent = #hideStems
      \override Flag #'transparent = #hideStems
      \override StaffSymbol #'after-line-breaking = #staff-break-callback
      \override StaffSymbol #'thickness = #(/ (ly:output-def-lookup $defaultpaper 'staff-space) 3.0)
      \override StaffSymbol #'stencil = #staff-print
    }
    \context {
      \Voice
      \override Stem #'length = #0
   \override BreathingSign #'after-line-breaking = #set-breathe
    }
    \context {
      \Score
      \override  RehearsalMark #'self-alignment-X = #LEFT
      \override RehearsalMark #'break-align-symbols = #'(key-signature)
      \override  NonMusicalPaperColumn #'padding = #0.8
      skipBars= ##t
      barAlways = ##t
      lyricMelismaAlignment = #CENTER
    }
    \context {
      \Lyrics
      \override LyricSpace #'minimum-distance = #0.8
      \override LyricText #'font-size = #1.065
      \override LyricText #'font-name = #"TimesNewRoman"
      \override LyricText #'color = #(rgb-color 0 0 0)
      \override LyricHyphen #'color = #(rgb-color 0 0 0)
      \override LyricExtender #'color = #(rgb-color 0 0 0)
    }
  }
}

\header {
tagline = "" 
title = ""
}
</LilyPond>
  <EasyChant>Ma-       gasz- tald, ma-     gasz- tald,
l,4 t, d2 t,    l,    d4 r m2 r     d    

én lel- kem, én lel- kem,
m2 f    m r  d  t,   l,  

az Úr   egyházába be- ve- ze-     tett
r4 r  / r0        m4  f   m2 r4 m t,2 

és fő- pap kezeivel meg- ál- dott Szű-      zet!
d4 r   m0                r   m2   r4 d t, d l,2 

</EasyChant>
  <Clef>0</Clef>
  <Key>8</Key>
  <LyricsSpacing>0.8</LyricsSpacing>
  <TimeOption>0</TimeOption>
  <TimeText>N/A (cadenza)</TimeText>
  <StaffSizeOption>2</StaffSizeOption>
  <StaffSizeText>14</StaffSizeText>
  <LyricsSize>1.1</LyricsSize>
  <LyricsFontName>Times New Roman</LyricsFontName>
  <LyricsFontBold>False</LyricsFontBold>
  <LyricsFontItalic>False</LyricsFontItalic>
  <LyricsFontColor>Color [A=255, R=0, G=0, B=0]</LyricsFontColor>
  <LineSpacing>12</LineSpacing>
  <LineThickness>4</LineThickness>
  <Options>0,1,2,6,7,8</Options>
  <pages>1</pages>
  <page>1</page>
</item>

9. irmosz, 1. hang



Áldozási vers

Dicsérjétek... Az üdvösség kelyhét...