## 2015. 12. 06. P. u. 28. vasárnap, csodatevő Mikós főpap, 3. hang

Ft. tropár, 3. hang

<?xml version="1.0" encoding="utf-16"?>
<item>
  <title>Tropár, 3. hang4</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 . 13.46))
  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 bes \major
g'4 fis'( g') d' e' fis' g' a' bes' a' bes' c'' bes' a' g' g' a'2 bes'4 bes' a' \divisioMaxima
g'4 fis'( g') \break d' e' fis' g' a' bes' a' bes' c'' bes' a' g' g' a'2 bes'4 bes' a' \divisioMaxima
g'4 fis'( g') d' e' fis' g' a' bes' a' g' a'2 bes'4 bes' a'2 \divisioMaxima
g'4 fis'( g') d' e' fis' g' a' bes' a' g' g' a'2 bes'4 \break bes' a'2 \divisioMaxima
bes'4 c'' bes' a' bes' a' g' g'\breve \unHideNotes g'4 a'2 g'4 g' fis'2 
\hideNotes \grace {s16 } \finalis
}

verba = \lyricmode {
"Vi" -- "gad" -- "ja" -- "nak" "a" "meny" -- "nye" -- "i" -- "ek" "és" "ör" -- "ven" -- "dez" -- "ze" -- "nek" "a" "föl" -- "di" -- "ek,"
"mert" "az" "Úr" "a" "ma" -- "ga" "jobb" -- "já" -- "val" "új" "bi" -- "ro" -- "dal" -- "mat" "te" -- "rem" -- "tett" "ne" -- "künk,"
"és" "le" -- "győz" -- "vén" "a" "ha" -- "lált," "a" "hol" -- "tak" "el" -- "se" -- "je" "lőn,"
"ki" "a" "po" -- "kol" "mély" -- "sé" -- "gé" -- "ből" "ki" -- "sza" -- "ba" -- "dí" -- "tott" "min" -- "ket,"
"és" "a" "vi" -- "lág" -- "nak" "gaz" -- "dag" \alignAt #-0.862 "kegyelmet" "a" -- "ján" -- "dé" -- "ko" -- "zott."
}


\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>Vi- gad-   ja- nak a   meny- nye- i- ek és ör- ven- dez- ze- nek a     föl- di- ek,
l,4 si, l, m,  fi, si, l,    t,   d  t, d  r   d    t,   l,  l,  t,2   d4   d   t, 

mert az       Úr a   ma- ga jobb- já- val új bi- ro- dal- mat   te- rem- tett ne- künk,
l,4  si, l, / m, fi, si, l, t,    d   t,  d  r   d   t,   l,    l,  t,2  d4   d   t,   

és  le-    győz- vén a   ha- lált, a hol- tak   el- se- je lőn,
l,4 si, l, m,    fi, si, l,  t,    d t,   l,    t,2 d4  d  t,2 

ki  a      po- kol mély- sé- gé- ből   ki- sza- ba- dí- tott min- ket,
l,4 si, l, m,  fi, si,   l,  t,  d     t,  l,   l,  t,2 d4 / d    t,2 

és a vi- lág- nak gaz- dag kegyelmet   a-  ján- dé- ko- zott.
d4 r d   t,   d   t,   l,  l,0         l,4 t,2  l,4 l,  si,2 </EasyChant>
  <Clef>0</Clef>
  <Key>9</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>

Tropár, főpap, 4. hang

<?xml version="1.0" encoding="utf-16"?>
<item>
  <title>2. 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 . 63.89))
  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'' \unHideNotes d''4 c''2 bes' a'2 \divisioMaxima
bes'2 g'4 bes'\breve \break \unHideNotes bes'4 c'' d'' d'' c''4.( bes'4) a'2 \divisioMaxima
c''\breve \hideNotes c'' c'' c'' c'' c'' c'' \unHideNotes d''4 c''2 bes' a' \break \divisioMaxima
bes'4 g' bes'\breve \hideNotes bes' \unHideNotes c''4 d'' d'' c''4.( bes'4) a'2 \divisioMaxima
c''\breve \hideNotes c'' \break \unHideNotes d''4 c'' bes' a'2 \divisioMaxima
bes'4 g' d''4 d'' c''4.( bes'4) a'2 \divisioMaxima
c''4 c'' d'' c''2 bes' a' \break \divisioMaxima
c''\breve \hideNotes c'' \unHideNotes a'4 bes' a'2 g' f' 
\hideNotes \grace {s16 } \finalis
}

verba = \lyricmode {
"A" "hit" -- "nek" "sza" -- \alignAt #-0.762 "bálya," \alignAt #0 "a" \alignAt #-0.892 "béketűrésnek" "pél" -- "dány" -- "ké" -- "pe"
"és" "az" \alignAt #-1.052 "önmegtagadásnak" "ta" -- "ní" -- "tó" -- "ja" "gya" -- "nánt"
\alignAt #-1.212 "adott" \alignAt #-0.938 "téged" \alignAt #0 "a" \alignAt #-0.476 "maga" \alignAt #-0.575 "nyájának" \alignAt #-1.482 "az" \alignAt #0 "ö" -- "rök" "i" -- "gaz" -- "ság,"
"a" -- "zért" \alignAt #-1.063 "alázatosságoddal" \alignAt #-1.115 "elnyerted" "a" "ma" -- "gas" -- "sá" -- "got,"
\alignAt #-0.818 "szegénységeddel" \alignAt #0 "a" "gaz" -- "dag" -- "sá" -- "got,"
"Mik" -- "lós" "püs" -- "pök" "a" -- "tyánk,"
"i" -- "mádd" "Krisz" -- "tus" "Is" -- "tent,"
\alignAt #-0.683 "hogy" \alignAt #-1.098 "üdvözítse" "a" "mi" "lel" -- "kün" -- "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 = #0.6
      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>A  hit- nek sza- bálya, a béketűrésnek pél- dány- ké- pe
d2 r    m4  f    s0                    l    s2    f   m2

és az önmegtagadásnak   ta- ní- tó- ja gya-   nánt
f2 r4 f0              / f   s   l   l  s4. f4 m2  

adott téged a maga nyájának az ö- rök i- gaz- ság,
s0                                l4  s2 f    m   
/
a- zért alázatosságoddal elnyerted a ma- gas- sá-    got,
f4 r    f0                         s l   l    s4. f4 m2  

szegénységeddel a  gaz- dag- sá- got,
s0               / l4   s    f   m2  

Mik- lós püs- pök a-     tyánk,
f4   r   l4   l   s4. f4 m2    

i- mádd Krisz- tus Is- tent,
s4 s    l      s2  f   m    
/
hogy üdvözítse a  mi lel- kün- ket!
s0             m4 f  m2   r    d   

</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, 3. hang

<?xml version="1.0" encoding="utf-16"?>
<item>
  <title>Konták, 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 . 99.16))
  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
g'4( a') bes'2 a'4 g' f' a' c''\breve \hideNotes c'' c'' c'' \unHideNotes g'4 c''2 bes'4( a') g'2 \divisioMaxima \divisioMaxima
g'4( a') bes'2 a'4 g' \break f' a' c''\breve \hideNotes c'' c'' c'' \unHideNotes g'4 c''2 bes'4( a') g'2 \divisioMaxima \divisioMaxima
g'4( a') bes'2 a'4( g') f' a' c'' c'' \break c'' g' c''2 bes'4( a') g'2 \divisioMaxima \divisioMaxima
g'4( a') bes'2 a'4 g' f' a' c''\breve \hideNotes c'' c'' c'' \break \unHideNotes g'4 c''2 bes'4( a') g'2 \divisioMaxima \divisioMaxima
g'4( a') bes'2 a'4 g' f' a' c''\breve \hideNotes c'' \unHideNotes g'4 c''2 bes'4( a' bes' c'') g'2 
\hideNotes \grace {s16 } \finalis
}

verba = \lyricmode {
"Föl" -- "tá" -- "mad" -- "tál" "a" "ma" -- \alignAt #0 "i" \alignAt #-0.766 "napon" \alignAt #0 "a" \alignAt #-0.93 "sírból," "ó" "Ir" -- "gal" -- "mas,"
"és" "min" -- "ket" "is" "ki" -- "ve" -- \alignAt #-0.794 "zettél" \alignAt #0 "a" \alignAt #-0.707 "halál" \alignAt #0 "bi" -- "ro" -- "dal" -- "má" -- "ból."
"Ma" "Á" -- "dám" "vi" -- "ga" -- "doz" "és" "É" -- "va" "ör" -- "ven" -- "dez,"
"és" "a" "pró" -- "fé" -- "ták" -- "kal" \alignAt #0 "és" \alignAt #-0.892 "pátriárkákkal" \alignAt #-1.172 "együtt" \alignAt #-0.664 "szüntelen" "di" -- "cső" -- "í" -- "tik"
"a" "te" "is" -- "te" -- "ni" "ha" -- \alignAt #-0.951 "talmad" \alignAt #0 "u" -- "ral" -- "ko" -- "dá" -- "sát."
}


\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 a ma- i napon a sírból, ó Ir- gal- mas,
r4 m f2  m4   r   d m   s0                r s2  f4 m r2  

és   min- ket is  ki- ve- zettél a halál bi- ro- dal- má-  ból.
r4 m f2   m4  r / d   m   s0                 r   s2   f4 m r2  

Ma   Á- dám  vi- ga- doz és  É- va ör- ven- dez,
r4 m f2 m4 r d   m   s   s / s  r  s2  f4 m r2  

és   a  pró- fé- ták- kal és pátriárkákkal együtt szüntelen di- cső- í-   tik
r4 m f2 m4   r   d    m   s0                              / r   s2   f4 m r2 

a    te is- te- ni ha- talmad u- ral- ko- dá-      sát.
r4 m f2 m4  r   d  m   s0        r    s2  f4 m f s r2  </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>

Dicsőség... 3, hang, konták-dallam

<?xml version="1.0" encoding="utf-16"?>
<item>
  <title>Dicsőség, konták, 3. 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 . 53.89))
  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
g'8 a' bes'2 a'4( g') f' a' c'' c'' c''\breve \hideNotes c'' \unHideNotes g'4 c''2 bes'8([ a' bes' c'')] g'2 
\hideNotes \grace {s16 } \finalis
}

verba = \lyricmode {
"Di" -- "cső" -- "ség" "az" "A" -- "tyá" -- "nak" "és" \alignAt #-0.643 "Fiúnak" \alignAt #0 "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.
r8  m    f2  m4 r d  m    s   s   s0          r4     s2  f8 m f s r2 </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>

Konták, főpap, 3. hang

<?xml version="1.0" encoding="utf-16"?>
<item>
  <title>2. konták, 3. 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 . 21.2))
  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
g'4( a') bes'2 a'4 g' f' a' c''\breve \hideNotes c'' c'' c'' c'' \unHideNotes g'4 c''2 bes'4( a') g'2 \break \divisioMaxima \divisioMaxima
g'4( a') bes'2 a'4( g') f' a' c''\breve \unHideNotes g'4 c''2 bes'4( a') g'2 \divisioMaxima
g'4 a' bes'2 c''\breve \hideNotes c'' \unHideNotes g'4 c''2 bes'4( a') g'2 \break \divisioMaxima \divisioMaxima
g'4( a') bes'2 a'4 g' f' a' c''\breve \unHideNotes g'4 c''2 bes'4( a') g'2 \divisioMaxima \divisioMaxima
g'4( a') bes'2 c''4 c'' \break c'' g'4 c''2 bes'4( a') g'2 \divisioMaxima
g'4 a' bes'2 a'4 g' f' a' c''\breve \hideNotes c'' c'' \unHideNotes g'4 c''2 bes'4( a' bes' c'') g'2 
\hideNotes \grace {s16 } \finalis
}

verba = \lyricmode {
"Mű" -- "ra" "vá" -- "ro" -- "sá" -- "ban," \alignAt #-0.3 "Szent" \alignAt #-0.585 "Miklós" \alignAt #-1.224 "atya," \alignAt #-1.306 "igazi" \alignAt #0 "fő" -- "pász" -- "tor" "vol" -- "tál,"
"mert" "Krisz" -- "tus" "e" -- "van" -- \alignAt #-0.724 "géliu" -- "mát" "be" -- "tölt" -- "vén,"
"lel" -- "ke" -- "det" \alignAt #-1.196 "adtad" \alignAt #0 "a" "te" "né" -- "pe" -- "dért"
"és" "meg" -- "men" -- "te" -- "ted" "az" \alignAt #-1.09 "ártatlanokat" "a" "ha" -- "lál" -- "tól."
"A" -- "zért" "föl" "is" "ma" -- "gasz" -- "tal" -- "tat" -- "tál,"
"mint" "az" "is" -- "te" -- "ni" "ma" -- "laszt" -- \alignAt #-0.604 "nak" \alignAt #-0.829 "hathatós" \alignAt #0 "ki" -- "szol" -- "gá" -- "ló" -- "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 = #0.6
      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>Mű-  ra vá- ro- sá- ban, Szent Miklós atya, igazi fő- pász- tor   vol- tál,
r4 m f2 m4  r   d   m    s0                           r     s2    f4 m r2  
/
mert Krisz- tus  e- van- géliu- mát be- tölt- vén,
r4 m f2     m4 r d  m    s0     r   s2  f4 m  r2  

lel- ke- det adtad a te né- pe-  dért
r4   m   f2  s0      r4 s2  f4 m r2  
/
és   meg- men- te- ted az ártatlanokat a ha- lál- tól.
r4 m f2   m4   r   d   m  s0           r s2  f4 m r2  

A-   zért föl is  ma- gasz- tal- tat- tál,
r4 m f2   s4  s / s   r4    s2   f4 m r2  

mint az is- te- ni ma- laszt- nak hathatós ki- szol- gá- ló-      ja.
r4   m  f2  m4  r  d   m      s0               r     s2  f4 m f s r2 </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>

Most és... 3. hang, konták-dallam

<?xml version="1.0" encoding="utf-16"?>
<item>
  <title>Most és, konták, 3. 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 . 53.89))
  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
g'4( a') bes'2 a'4 g' f' a' c''\breve \unHideNotes c''4 g' c''2 bes'4( a' bes' c'') g'2 
\hideNotes \grace {s16 } \finalis
}

verba = \lyricmode {
"Most" "és" "min" -- "den" -- "kor" "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 min- den- kor és örökkön- ö- rök- ké. A-       men.
r4 m f2 m4   r    d   m  s0       s4 r    s2  f4 m f s r2

</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>

Istenszülői, 3. hang

<?xml version="1.0" encoding="utf-16"?>
<item>
  <title>Konták, 3. 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
g'4( a') bes'2 a'4 g' f' a' c'' c'' g' c''2 bes'4( a') g'2 \divisioMaxima \divisioMaxima
g'4( a') bes'2 a'4 g' f' a' c'' c'' \break c''\breve \hideNotes c'' c'' \unHideNotes g'4 c''2 bes'4( a') g'2 \divisioMaxima \divisioMaxima
g'4( a') bes'2 a'4 g' f' \break a' c''\breve \unHideNotes g'4 c''2 bes'4( a') g'2 \divisioMaxima \divisioMaxima
g'4( a') bes'2 a'4 g' f' a' c''\breve \hideNotes c'' c'' \unHideNotes g'4 \break c''2 bes'4( a') g'2 \divisioMaxima \divisioMaxima
g'4( a') bes'2 a'4 g' f' a' c''\breve \unHideNotes g'4 c''2 bes'4( a') g'2 \divisioMaxima \divisioMaxima \divisioMaxima
g'4( a' bes'2) c''\breve \unHideNotes g'4 \break c''2 bes'4( a' bes' c'') g'2 
\hideNotes \grace {s16 } \finalis
}

verba = \lyricmode {
"A" "Szűz" "ma" "je" -- "len" "van" "az" "Úr" "haj" -- "lé" -- "ká" -- "ban,"
"és" "a" "meny" -- "nye" -- "i" "ka" -- "rok" -- "kal" \alignAt #-0.973 "láthatatlanul" \alignAt #-1.14 "imádkozik" \alignAt #-1.283 "éret" -- "tünk" "Is" -- "ten" -- "hez."
"Az" "an" -- "gya" -- "lok" "a" "fő" -- \alignAt #-0.844 "papokkal" "le" -- "bo" -- "rul" -- "nak,"
"az" "a" -- "pos" -- "to" -- "lok" "a" \alignAt #-0.717 "prófétákkal" \alignAt #-0.872 "dicsőítő" \alignAt #-1.306 "éne" -- "ket" "zen" -- "ge" -- "nek,"
"mert" "az" "Is" -- "ten" -- "szü" -- "lő" \alignAt #-1.192 "esede" -- "zik" "é" -- "ret" -- "tünk"
"az" \alignAt #-1.101 "örökkéva" -- "ló" "Is" -- "ten" -- "hez."
}


\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>A    Szűz ma je- len van az Úr haj- lé- ká-  ban,
r4 m f2   m4 r   d   m   s  s  r    s2  f4 m r2  

és   a  meny- nye- i ka- rok- kal   láthatatlanul imádkozik éret- tünk Is- ten- hez.
r4 m f2 m4    r    d m   s    s   / s0                            r    s2  f4 m r2  

Az   an- gya- lok a   fő- papokkal le- bo- rul- nak,
r4 m f2  m4   r   d / m   s0       r4  s2  f4 m r2  

az   a- pos- to- lok a prófétákkal dicsőítő éne- ket zen- ge-  nek,
r4 m f2 m4   r   d   m s0                        r / s2   f4 m r2  

mert az Is- ten- szü- lő esede- zik é- ret- tünk
r4 m f2 m4  r    d    m  s0     r   s2 f4 m r2  

az      örökkéva- ló   Is- ten-     hez.
r4 m f2 s0        r4 / s2  f4 m f s r2  

</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>1</pages>
  <page>1</page>
</item>

Ft. prokimen, 3. hang

<?xml version="1.0" encoding="utf-16"?>
<item>
  <title>Prokimen, 3. 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 . 53.89))
  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' a' \unHideNotes g'4 a' g'2 \divisioMaxima
f'4 g' a'\breve \hideNotes a' a' \unHideNotes a'4 a' g'4. f' \break g'4 g' a' g'( a') bes'2 \divisioMaxima
a'4 bes' c''\breve \unHideNotes bes'4 a'\breve \unHideNotes g'4.( fis'4 g' a') e'2 \finalis
\override Lyrics.LyricText.transparent = ##t f'\breve \hideNotes  \unHideNotes 
\hideNotes \grace {s16 } \finalis
}

verba = \lyricmode {
\alignAt #-1.055 "Énekeljetek" \alignAt #0 "a" \alignAt #-0.088 "mi" \alignAt #-1.108 "Istenünknek," \alignAt #-1.306 "éne" -- "kel" -- "je" -- "tek!"
"É" -- "ne" -- \alignAt #-0.82 "keljetek" \alignAt #0 "a" \alignAt #-0.088 "mi" "Ki" -- "rá" -- "lyunk" -- "nak," "é" -- "ne" -- "kel" -- "je" -- "tek,"
"a" "mi" \alignAt #-0.791 "Királyunk" -- "nak" \alignAt #-1.163 "énekel" -- "je" -- "tek!"
\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 = #0.7
      skipBars= ##t
      barAlways = ##t
      lyricMelismaAlignment = #CENTER
    }
    \context {
      \Lyrics
      \override LyricSpace #'minimum-distance = #0.7
      \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>Énekeljetek a mi Istenünknek, éne- kel- je- tek!
m0                                 r4   m   r2  

É- ne- keljetek a mi Ki- rá- lyunk- nak, é-  ne- kel- je- tek,
d4 r   m0            m   m   r4.    d  / r4  r   m    r m f2  

a  mi Királyunk- nak énekel- je-         tek!
m4 f  s0         f   m0      r4. di4 r m t,2 ||

Recit
\"override Lyrics.LyricText.transparent = ##t" d0</EasyChant>
  <Clef>0</Clef>
  <Key>8</Key>
  <LyricsSpacing>0.7</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: Minden nemzetek, tapsoljatok, harsogjon az Istennek ujjongásotok hangja!

(46. zsoltár)

Prokimen, főpap, 7. hang

<?xml version="1.0" encoding="utf-16"?>
<item>
  <title>Prokimen, 7. 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 . 75.93))
  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' g'\breve \unHideNotes a'4 bes' a'2( g'4 f') e'2( d') \divisioMaxima \divisioMaxima
c'4( d') e'( d') c'( d') e'2 e' \divisioMaxima \divisioMaxima \divisioMaxima \divisioMaxima
g'4( fis' g' a') g'2( f'?) e'4( f') g'2( a') g'4.( f'4 e' f') d'2 \finalis
\override Lyrics.LyricText.transparent = ##t f'\breve \hideNotes  \unHideNotes 
\hideNotes \grace {s16 } \finalis
}

verba = \lyricmode {
"Az" "i" -- "gaz" \alignAt #-1.085 "örvendezni" "fog" "az" "Úr" -- "ban,"
"és" "ben" -- "ne" "bíz" -- "ni,"
"ben" -- "ne," "ben" -- "ne" "bíz" -- "ni!"
\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>Az     i- gaz örvendezni fog az Úr-     ban,  
l,4 t, d2 t,  r0         m4  f  m2 r4 d t,2 l,

és     ben-  ne      bíz- ni,
s,4 l, t, l, s, l,   t,2  t, 

ben-       ne,    ben-  ne   bíz-        ni!
r4 di r m  r2 d?  t,4 d r2 m r4. d4 t, d l,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: Hallgasd meg, Isten, hozzád esdő könyögésem szavát!

(63. zsoltár)

Ft. alleluja, 3. hang

<?xml version="1.0" encoding="utf-16"?>
<item>
  <title>Alleluja, 3. 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 . 56.47))
  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 a' g' f' g'2 a'4 a' g'4.( a'4 bes'2) \divisioMaxima \divisioMaxima \divisioMaxima
c''4( bes' a') a' g'4.( fis'4 g' a') e'2 \finalis
\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 {
    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, al- le- lu- ja,      
m4  m   r   d   r2  m4  m   r4. m4 f2

al-    le- lu-         ja.
s4 f m m   r4. di4 r m t,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: Te vagy, Uram, én reményem, soha meg ne szégyenüljek!

Vers: Légy oltalmazó Istenem és menekvésem, hogy megments engem!

(30. zsoltár)

Alleluja, főpap, 8. hang

<?xml version="1.0" encoding="utf-16"?>
<item>
  <title>Alleluja, 8. 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 . 71.2))
  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 = #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, 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>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: Jertek, örvendezzünk az Úrnak, ujjongjunk üdvözítő Istenünknek!

(94. zsoltár)

Áldozási vers: Dicsérjétek... Egész földre...