## 2015. 10. 18. P. u. 21. vasárnap, Lukács apostol és evangélista, 4. hang

Ft. tropár, 4. hang

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

\include "gregorian.ly"

\paper {
  paper-width = 123.0136\mm
  paper-height = 184.6262\mm
  top-margin = 0\mm
  bottom-margin = 0\mm
  left-margin = 0\mm
  right-margin = 2\mm
  ragged-bottom = ##t
  print-page-number = ##f
  page-breaking = #ly:minimal-breaking
  system-system-spacing =
    #'((basic-distance . 12)
       (minimum-distance . 12)
       (padding . 0)
       (stretchability . 0))
%  page-breaking-system-system-spacing = #'((padding . 2))
  markup-system-spacing = #'((padding . 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
f'2 g' a'4 bes' c''\breve \hideNotes c'' \unHideNotes d''4 c''2 bes' a'2 \divisioMaxima
bes'2 g'4 bes'\breve \hideNotes bes' bes' \unHideNotes c''4 \break d''\breve \unHideNotes c''4( bes') a'2 \divisioMaxima
c''\breve \hideNotes c'' c'' c'' c'' \unHideNotes d''4 c''2 bes' a' \divisioMaxima
bes'2 g'4 bes'\breve \break \unHideNotes bes'4 c'' d'' d'' c''4.( bes'4) a'2 \divisioMaxima
c''\breve \unHideNotes d''4 c''2 bes' a' \divisioMaxima
bes'2 g'4 bes' d'' d'' c''2( bes'4) a'2 \divisioMaxima
c''\breve \hideNotes c'' c'' \unHideNotes a'4 bes' a'2 g' f' 
\hideNotes \grace {s16 } \finalis
}

verba = \lyricmode {
"A" "föl" -- "tá" -- "ma" -- \alignAt #-0.57 "dás" \alignAt #-1.165 "örven" -- "de" -- "tes" "hí" -- "rét"
"az" "an" -- \alignAt #-0.453 "gyaltól" \alignAt #-0.744 "megtudván" \alignAt #-1.482 "az" "Úr" \alignAt #-0.95 "tanítvá" -- "nya" -- "i,"
\alignAt #0 "és" \alignAt #-1.482 "az" \alignAt #-1.321 "ősi" \alignAt #-1.247 "átok" \alignAt #-1.098 "elvetéséről" "ér" -- "te" -- "sül" -- "vén,"
"az" "a" -- \alignAt #-0.872 "postoloknak" "di" -- "cse" -- "ked" -- "ve" "mon" -- "dák:"
\alignAt #-0.776 "Legyőze" -- "tett" "a" "ha" -- "lál,"
"föl" -- "tá" -- "madt" "Krisz" -- "tus" "Is" -- "ten,"
\alignAt #-1.078 "ajándékozván" \alignAt #0 "a" \alignAt #-0.888 "világnak" "gaz" -- "dag" "ke" -- "gyel" -- "met."
}


\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  föl- tá- ma- dás örven- de- tes hí- rét
d2 r    m4  f   s0         l   s2  f   m2 

az an- gyaltól megtudván az Úr  tanítvá- nya- i,
f2 r4  f0                   s / l0       s f  m2

és az ősi átok elvetéséről ér- te- sül- vén,
s0                         l4  s2  f    m   

az a- postoloknak   di- cse- ked- ve mon-   dák:
f2 r4 f0          / f   s    l    l  s4. f4 m2  

Legyőze- tett a  ha- lál,
s0       l4   s2 f   m   

föl- tá- madt Krisz- tus Is-   ten,
f2   r4  f    l      l   s2 f4 m2  

ajándékozván a világnak gaz- dag ke- gyel- met.
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>

Ap. tropár, 3. hang

<?xml version="1.0" encoding="utf-16"?>
<item>
  <title>Ap. tropár, 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 . 67.8))
  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'4 a' bes' a' g' a'2 bes'4 bes' a' \divisioMaxima
bes'4 c'' bes' a' g' g' \break g'\breve \hideNotes g' \unHideNotes a'2 g'4 g' fis'2 
\hideNotes \grace {s16 } \finalis
}

verba = \lyricmode {
"Szent" "Lu" -- "kács" "a" -- "pos" -- "tol," "i" -- "mádd" "a" "ke" -- "gyel" -- "mes" "Is" -- "tent,"
"hogy" "a" -- "ján" -- "dé" -- "koz" -- "zon" \alignAt #-0.892 "bűnbocsánatot" \alignAt #0 "a" "mi" "lel" -- "künk" -- "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 = #0.5
      skipBars= ##t
      barAlways = ##t
      lyricMelismaAlignment = #CENTER
    }
    \context {
      \Lyrics
      \override LyricSpace #'minimum-distance = #0.5
      \override LyricText #'font-size = #1.065
      \override LyricText #'font-name = #"TimesNewRoman"
      \override LyricText #'color = #(rgb-color 0 0 0)
      \override LyricHyphen #'color = #(rgb-color 0 0 0)
      \override LyricExtender #'color = #(rgb-color 0 0 0)
    }
  }
}

\header {
tagline = "" 
title = ""
}
</LilyPond>
  <EasyChant>Szent Lu- kács  a-  pos- tol, i-  mádd a  ke- gyel- mes Is- tent,
l,4   si, l, m, fi, si,  l,4  t,  d    t, l,  t,2   d4  d   t,   

hogy a- ján- dé- koz- zon  bűnbocsánatot a mi  lel- künk- nek!
d4   r  d    t,  l,    l, / l,0             t,2 l,4  l,    si,2

</EasyChant>
  <Clef>0</Clef>
  <Key>9</Key>
  <LyricsSpacing>0.5</LyricsSpacing>
  <TimeOption>0</TimeOption>
  <TimeText>N/A (cadenza)</TimeText>
  <StaffSizeOption>2</StaffSizeOption>
  <StaffSizeText>14</StaffSizeText>
  <LyricsSize>1.1</LyricsSize>
  <LyricsFontName>Times New Roman</LyricsFontName>
  <LyricsFontBold>False</LyricsFontBold>
  <LyricsFontItalic>False</LyricsFontItalic>
  <LyricsFontColor>Color [A=255, R=0, G=0, B=0]</LyricsFontColor>
  <LineSpacing>12</LineSpacing>
  <LineThickness>4</LineThickness>
  <Options>0,1,2,4,6,7,8</Options>
  <pages>1</pages>
  <page>1</page>
</item>

Ft. konták, 4. hang

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

\include "gregorian.ly"

\paper {
  paper-width = 123.0136\mm
  paper-height = 184.6262\mm
  top-margin = 0\mm
  bottom-margin = 0\mm
  left-margin = 0\mm
  right-margin = 2\mm
  ragged-bottom = ##t
  print-page-number = ##f
  page-breaking = #ly:minimal-breaking
  system-system-spacing =
    #'((basic-distance . 12)
       (minimum-distance . 12)
       (padding . 0)
       (stretchability . 0))
%  page-breaking-system-system-spacing = #'((padding . 2))
  markup-system-spacing = #'((padding . 85.18))
  top-markup-spacing = #'((basic-distance . 0) (minimum-distance . 0) (padding . 0))
}

#(set-global-staff-size 14)

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

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

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

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

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

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

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

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




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

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

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

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

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

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

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

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

go = {
\grace {s16 } 
}



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

verba = \lyricmode {
"Üd" -- "vö" -- "zí" -- "tőm" "és" "Meg" -- "vál" -- "tóm,"
\alignAt #-0.947 "föltámasztád" \alignAt #0 "a" \alignAt #-0.823 "sír" "bi" -- "lin" -- "cse" -- "i" -- "ből,"
"mint" "Is" -- "ten," "a" "föld" "szü" -- "löt" -- "te" -- "it,"
\alignAt #0 "és" \alignAt #0 "a" \alignAt #-0.726 "pokol" \alignAt #-0.706 "kapu" -- "it" "ösz" -- "sze" -- "tör" -- "ted,"
"s mint" "U" -- "ral" -- "ko" -- "dó,"
\alignAt #-0.855 "harmadna" -- "pon" "föl" -- "tá" -- "mad" -- "tál."
}


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

\header {
tagline = "" 
title = ""
}
</LilyPond>
  <EasyChant>Üd- vö- zí- tőm és Meg- vál- tóm,
s4  f   m   m   f  s2   f    m   

föltámasztád a sír bi- lin- cse- i- ből,
s0                 m4  l    s2   f  m   

mint Is- ten,  a föld szü- löt- te- it,
s4   f   m   / m m    f    s2   f   m  

és a pokol kapu- it   ösz- sze- tör- ted,
s0               m4   l    s2   f    m   

s__mint U-  ral- ko- dó,
s4 f    m f s2   f   m  
/
harmadna- pon   föl- tá- mad- tál.
s0        m4    l    s2  f    m   

</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... 4. hang, konták-dallam

<?xml version="1.0" encoding="utf-16"?>
<item>
  <title>Dicsőség, konták, 8. 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 . 135.4))
  top-markup-spacing = #'((basic-distance . 0) (minimum-distance . 0) (padding . 0))
}

#(set-global-staff-size 14)

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

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

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

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

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

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

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

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




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

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

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

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

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

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

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

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

go = {
\grace {s16 } 
}



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

verba = \lyricmode {
\alignAt #-0.769 "Dicsőség" \alignAt #-1.482 "az" \alignAt #-1.046 "Atyának" \alignAt #0 "és" \alignAt #-0.643 "Fiúnak" "és" "Szent" -- "lé" -- "lek" -- "nek."
}


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

\header {
tagline = "" 
title = ""
}
</LilyPond>
  <EasyChant>Dicsőség az Atyának és Fiúnak és Szent- lé- lek- nek.
s0                            m4 l      s2  f2   m   

</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>

Ap. konták, 4. hang (Melléklet, 17. o.)

<?xml version="1.0" encoding="utf-16"?>
<item>
  <title>Konták, 4. 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 . 4.31))
  top-markup-spacing = #'((basic-distance . 0) (minimum-distance . 0) (padding . 0))
}

#(set-global-staff-size 14)

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

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

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

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

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

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

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

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




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

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

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

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

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

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

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

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

go = {
\grace {s16 } 
}



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

verba = \lyricmode {
"Az" "Is" -- "ten" -- \alignAt #-1.438 "Ige" \alignAt #0 "ta" -- "nít" -- "vá" -- "nya" -- "ként"
\alignAt #-0.705 "Pállal" \alignAt #-1.172 "együtt" \alignAt #-1.482 "az" \alignAt #-1.193 "egész" \alignAt #-0.861 "világot" \alignAt #-0.544 "megvi" -- "lá" -- "go" -- "sí" -- "tot" -- "tad,"
"és" "a" \alignAt #-0.9 "sötétséget" "el" -- "osz" -- "lat" -- "tad"
\alignAt #-0.545 "Krisztus" \alignAt #-0.406 "szent" \alignAt #-1.039 "Evangéliumának" "meg" -- "í" -- "rá" -- "sá" -- "val."
}


\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 Is-  ten- Ige ta- nít- vá- nya- ként
s4 f    m    m0      f4   s2  f    m   

Pállal együtt az egész világot megvi- lá- go- sí- tot- tad,
s0                                    m4  l   s2  f    m   

és a sötétséget el- osz- lat- tad
s4 f m0         f4  s2   f    m  

Krisztus szent Evangéliumának meg- í- rá- sá- val.
s0                            m4   l  s2  f   m   

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

Most és... 4. hang, konták-dallam

<?xml version="1.0" encoding="utf-16"?>
<item>
  <title>Dicsőség, konták, 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 . 32.48))
  top-markup-spacing = #'((basic-distance . 0) (minimum-distance . 0) (padding . 0))
}

#(set-global-staff-size 14)

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

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

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

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

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

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

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

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




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

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

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

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

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

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

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

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

go = {
\grace {s16 } 
}



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

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


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

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

</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, 4. hang

<?xml version="1.0" encoding="utf-16"?>
<item>
  <title>Istenszülői, 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 . 48.18))
  top-markup-spacing = #'((basic-distance . 0) (minimum-distance . 0) (padding . 0))
}

#(set-global-staff-size 14)

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

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

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

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

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

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

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

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




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

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

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

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

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

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

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

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

go = {
\grace {s16 } 
}



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

verba = \lyricmode {
"A" "te" \alignAt #-0.691 "születésed" \alignAt #-1.246 "által" "leg" -- "tisz" -- "tább" "Szűz,"
\alignAt #-0.703 "Joákim" \alignAt #0 "és" \alignAt #-1.072 "Anna" \alignAt #0 "a" \alignAt #-0.691 "magtalan" -- "ság" "szé" -- "gye" -- "né" -- "től,"
"Á" -- "dám" \alignAt #0 "és" \alignAt #-1.168 "Éva" \alignAt #-0.742 "pedig" \alignAt #0 "a" \alignAt #-0.707 "halál" \alignAt #-1.086 "enyészetétől" \alignAt #-0.041 "sza" -- "ba" -- "dul" -- "tak" "meg,"
\alignAt #-1.366 "ezt" \alignAt #-1.107 "ünneplik" \alignAt #0 "a" "te" "né" -- "pe" -- "id" "is,"
"le" -- "té" -- "vén" "az" "ő" -- "si" "bűn" "ter" -- "hét,"
\alignAt #-0.622 "midőn" \alignAt #-1.475 "így" \alignAt #0 "ki" -- "ál" -- "ta" -- "nak" "hoz" -- "zád:"
"Mag" -- "ta" -- \alignAt #-0.947 "lanság" \alignAt #-0.455 "szülte" \alignAt #-1.482 "az" "Is" -- "ten" -- "a" -- "nyát,"
\alignAt #0 "a" \alignAt #-0.088 "mi" \alignAt #-1.363 "éle" -- "tünk" "táp" -- "lá" -- "ló" -- "ját."
}


\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 = #0.5
      skipBars= ##t
      barAlways = ##t
      lyricMelismaAlignment = #CENTER
    }
    \context {
      \Lyrics
      \override LyricSpace #'minimum-distance = #0.5
      \override LyricText #'font-size = #1.065
      \override LyricText #'font-name = #"TimesNewRoman"
      \override LyricText #'color = #(rgb-color 0 0 0)
      \override LyricHyphen #'color = #(rgb-color 0 0 0)
      \override LyricExtender #'color = #(rgb-color 0 0 0)
    }
  }
}

\header {
tagline = "" 
title = ""
}
</LilyPond>
  <EasyChant>A  te születésed által leg- tisz- tább Szűz,
s4 f  m0               f    s2    f    m    

Joákim és Anna a magtalan- ság  szé- gye- né- től,
s0                         m4 / l    s2   f   m   

Á- dám és Éva pedig a halál enyészetétől sza- ba- dul- tak meg,
s4 f   m0                                     f   s2   f   m   
/
ezt ünneplik a te né- pe- id is,
s0             m4 l   s2  f  m  

le- té- vén az ő- si bűn ter- hét,
s4  f   m   m  m  f  s2  f    m   

midőn így ki- ál- ta- nak  hoz- zád:
s0            m4  l   s2 / f    m   

Mag- ta- lanság szülte az Is- ten- a- nyát,
s4   f   m0               f   s2   f  m    

a mi éle- tünk táp-   lá- ló- ját.
s0        m4     l    s2  f   m   

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

Ft. prokimen, 4. hang

<?xml version="1.0" encoding="utf-16"?>
<item>
  <title>Prokimen, 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 . 6.03))
  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'\breve \hideNotes f' f' f' \unHideNotes g'\breve \hideNotes g' \unHideNotes g'2 \divisioMaxima
a'4 bes' c'' c'' a' bes' c'' bes' a'2( g'4 a') bes'2 \break \divisioMaxima
c''4 bes' a'\breve \hideNotes a' \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 #-0.389 "Mily" \alignAt #-0.885 "nagyszerűek" \alignAt #0 "a" \alignAt #0 "te" \alignAt #-0.635 "műveid," \alignAt #0 "U" -- "ram,"
"böl" -- "cses" -- "sé" -- "ged" -- "ben" "al" -- "kot" -- "tál" "min" -- "dent,"
"böl" -- "cses" -- \alignAt #-0.908 "ségedben" \alignAt #-1.141 "alkottál" "min" -- "dent!"
\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>Mily nagyszerűek a te műveid, U- ram,
d0                    r0         r2  

böl- cses- sé- ged- ben al- kot- tál min-    dent, 
m4   f     s   s    m   f   s    f   m2 r4 m f2   
/
böl- cses- ségedben alkottál min-        dent!
s4   f     m0                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: Áldjad, én lelkem, az Urat, Uram, Istenem, igen hatalmas vagy!

(103. zsoltár)

Ap. prokimen, 8. hang

<?xml version="1.0" encoding="utf-16"?>
<item>
  <title>Prokimen, 8. 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 . 48.83))
  top-markup-spacing = #'((basic-distance . 0) (minimum-distance . 0) (padding . 0))
}

#(set-global-staff-size 14)

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

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

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

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

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

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

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

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




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

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

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

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

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

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

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

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

go = {
\grace {s16 } 
}



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

verba = \lyricmode {
\alignAt #-1.11 "Egész" \alignAt #-0.886 "földre" \alignAt #-1.141 "elhatott" \alignAt #-1.482 "az" "ő" "szó" -- "za" -- "tuk,"
"és" "a" "föld" -- "ke" -- "rek" -- "ség" "ha" -- "tá" -- \alignAt #-0.718 "raira" \alignAt #0 "i" -- "gé" -- "jük,"
"a" \alignAt #-0.947 "földkerekség" \alignAt #-0.767 "határa" -- "i" -- "ra" "i" -- "gé" -- "jük."
\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>Egész földre elhatott az ő   szó- za- tuk,
d0                       t,4 r    r   r2  

és a   föld- ke- rek- ség ha- tá- raira i- gé-       jük,
r4 r   m     r   d    t,  d   r   m0       t,2 d4 t, l,2 
/
a    földkerekség határa- i- ra i- gé-   jük.
d4 r m0                   t, 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: Az egek beszélik Isten dicsőségét, és az ő kezeművét hirdeti az égbolt.

(18. zsoltár)

Ft. alleluja, 4. hang

<?xml version="1.0" encoding="utf-16"?>
<item>
  <title>Alleluja, 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 . 84.54))
  top-markup-spacing = #'((basic-distance . 0) (minimum-distance . 0) (padding . 0))
}

#(set-global-staff-size 14)

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

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

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

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

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

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

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

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




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

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

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

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

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

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

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

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

go = {
\grace {s16 } 
}



chant = {
\key f \major
a'4 bes' c'' a'2 bes'( c''4) bes' a'2 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  f   s   m2  f s4 f   m2  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: Indulj meg! Haladj szerencsésen, uralkodj az igazságért, szelídségért és igazságosságért!

Vers: Szeretted az igazságot, és gyűlölted a gonoszságot!

(44. zsoltár)

Ap. alleluja, 8. hang

<?xml version="1.0" encoding="utf-16"?>
<item>
  <title>Alleluja, 8. 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 . 48.83))
  top-markup-spacing = #'((basic-distance . 0) (minimum-distance . 0) (padding . 0))
}

#(set-global-staff-size 14)

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

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

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

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

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

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

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

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




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

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

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

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

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

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

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

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

go = {
\grace {s16 } 
}



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

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


\score {
  \new Staff &lt;&lt;
    #(set-accidental-style 'modern-cautionary)
    \new Voice = "melody" \chant
    \new Lyrics = "one" \lyricsto melody \verba
  &gt;&gt;
  \layout {
    line-thickness = #(/ (ly:output-def-lookup $defaultpaper 'staff-space) 4)
    \context {
      \Staff
      \override BarLine  #'stencil = ##f
      \remove Time_signature_engraver
      \override Stem #'transparent = #hideStems
      \override Beam #'transparent = #hideStems
      \override Flag #'transparent = #hideStems
      \override StaffSymbol #'after-line-breaking = #staff-break-callback
      \override StaffSymbol #'thickness = #(/ (ly:output-def-lookup $defaultpaper 'staff-space) 3.0)
      \override StaffSymbol #'stencil = #staff-print
    }
    \context {
      \Voice
      \override Stem #'length = #0
   \override BreathingSign #'after-line-breaking = #set-breathe
    }
    \context {
      \Score
      \override  RehearsalMark #'self-alignment-X = #LEFT
      \override RehearsalMark #'break-align-symbols = #'(key-signature)
      \override  NonMusicalPaperColumn #'padding = #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: Magasztalják az egek csodatetteidet, Uram, és igazvoltodat a szentek gyülekezetében!

Vers: Isten dicsőséges a szentek tanácsában.

(88. zsoltár)

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