

Plover Is Great

Hi,

Congratulations on discovering [Plover](#) and joining the [mailing list](#)! For over a century, the amazing speed and ergonomics of stenographic typing was locked up by patents and trade secrets. If it hadn't been, we would all undoubtedly use stenotype keyboards instead of QWERTY. As it is, stenotype keyboards still cost thousands of dollars, the software to use them costs thousands as well, and the lessons to learn them cost thousands more.

That's over now. With the advent of Plover, it's possible to use low-tech QWERTY keyboards as stenotype keyboards (or any compatible professional steno keyboard), double or triple your typing speed, and perhaps mitigate some of the health issues associated with typing on QWERTY keyboards.

I got interested in finding a chorded typing system a few years ago, before Plover existed, and gave up discouraged. Years later, when a friend mentioned Plover, I got interested again in a hurry. I contacted Mirabai, and she generously agreed to give me Plover lessons in exchange for documentation. This is the result.

My goal from the start has been to make Plover easy. It's not a complex system. But it's also not entirely intuitive. As I discovered myself, it's possible to get a little lost at first, and follow some wrong paths before finding the right one. With these lessons, I've tried to start simple, and to stay simple. Each lesson introduces a few new ideas, and builds on the ideas of previous lessons. If I've done a good job, you should never feel lost, and you should never feel that the topic is so big that you'll never reach the end of it.

The best way to approach these tutorials is step by step. Learn the rules in one lesson, practice the exercises until you're good at them, and only then move on to the next lesson.

I strongly welcome feedback and suggestions. If you have a problem with any of the lessons, chances are someone else will too. Contact me at zacharyb@gmail.com with your feedback. Be sure to include "Plover Docs" in the subject line. In the email body you should give a link to the particular lesson you're talking about, and tell me as much as you can about the problems you found. If you have ideas about how I might fix those problems, tell me that too.

Finally, for a list of Plover-related study aids, see <https://github.com/openstenoproject/plover/wiki/Learning-Stenography>.

Be well,
Zack Brown

Typographic Conventions

Some conventions are used to make it easier to know what you're looking at when you read these docs. Each convention is explained below:

- **ALLCAPS REVERSE COLORS:** physical keys and full keystrokes on the Plover keyboard.

- **FORWARD/SLASH/BETWEEN/STROKES**: a series of strokes to type, one after the other, on the Plover keyboard.
- **HYPHEN-ATION**: in ambiguous cases, a hyphen indicates that a physical key is located on the left or the right side of the keyboard. For example, the word '**edited**' is stroked **ED/EUT/-D**. Like the forward slash, the hyphen is just an aid to reading, and isn't typed when using Plover. It is, however, included in entries in the dictionary file. When defining dictionary definitions by hand, you should be sure to include the hyphen when appropriate. If you only use Plover's native dictionary editor to make definitions, the hyphens will be added automatically.
- '**single quote boldface**': English letters, words, and sentences to be output by Plover or typed in QWERTY. For example, '**cat**' is typed **KAT** in Plover. The hard '**c**' is represented by a **K** on the keyboard.
- "double quotes": English text that's intended to convey spelling in particular, or that is not intended to be typed. For example, the '**a**' sound in '**bait**' uses the "ai" vowel cluster.
- **Green monospace**: a command given at the command-line, or its output, or the contents of the dictionary file. For example, **grep**.

Maintain This Site

If you'd like to be added as an editor to the Learn Plover site, email me about it at zacharyb@gmail.com and I'll send you an invite.

Lesson 1: Fingers And Keys

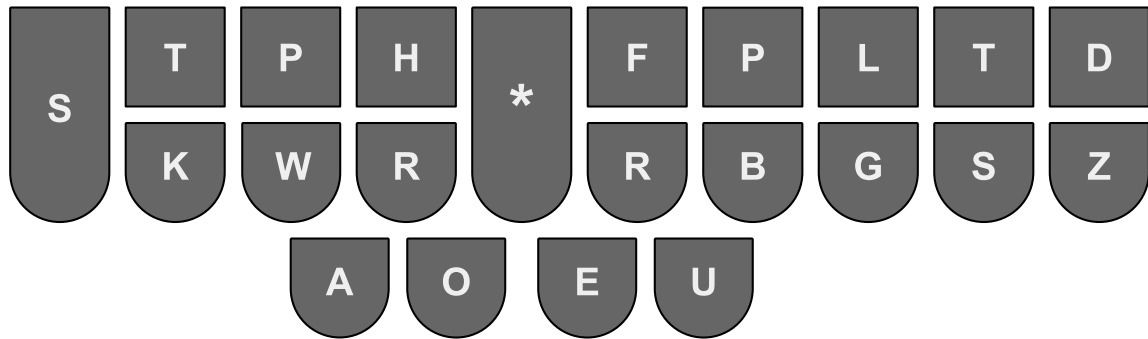
Contents

- 1 The Keyboard
 - 1.1 Rules For Short Vowel Sounds
 - 1.2 What About Long Vowels And Diphthongs?
- 2 Stroking Your First Words
 - 2.1 Exercise: One-Syllable Words
 - 2.2 Considerations
 - 2.3 Exercise: Consonant Clusters In Single-Stroke Words
 - 2.4 Considerations
- 3 Where Are We?
- 4 Solutions To Exercises
 - 4.1 Solution: Single-Syllable Words
 - 4.2 Solution: Consonant Clusters In Single-Stroke Words

The Keyboard

A nice place to start learning Plover is with its most obvious feature - the keyboard. What is this thing? Some letters seem to be missing, others seem to be duplicated. It looks weird.

Fig. 1: The Steno Keyboard

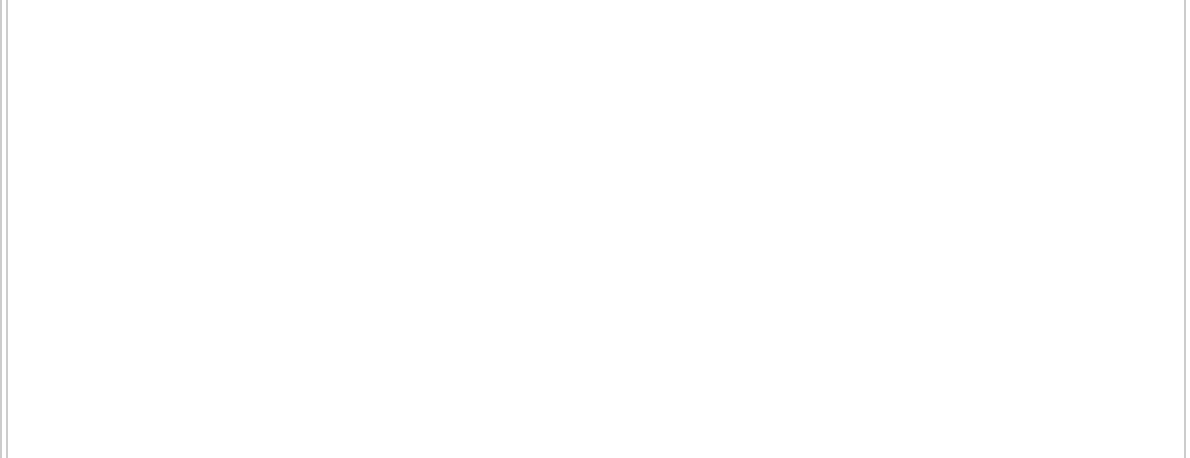


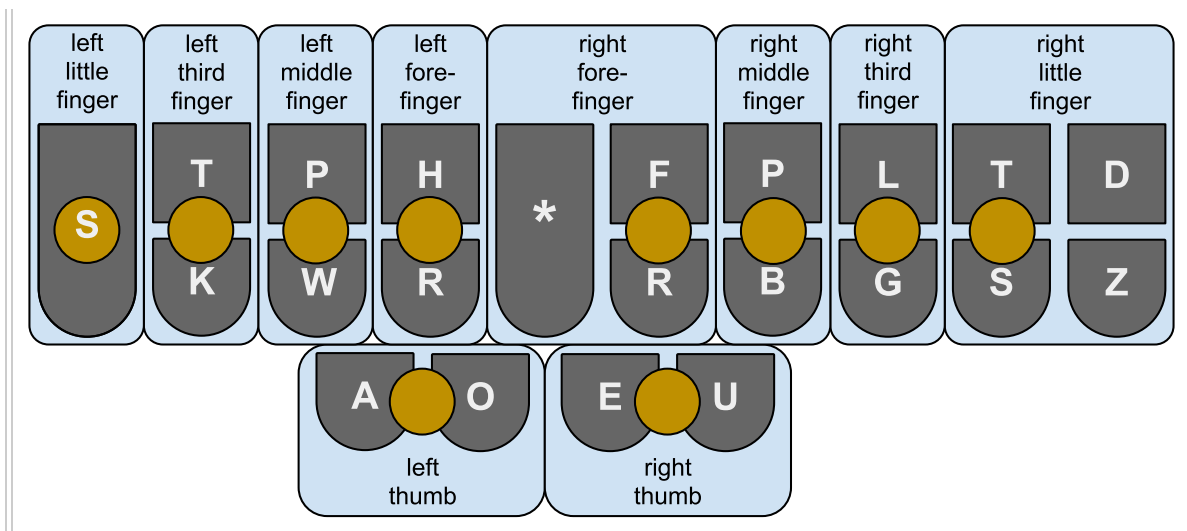
The most immediately useful thing to know is that all the consonant keys map to their corresponding consonant sounds. Plover - and all steno systems - express words primarily as groups of sounds rather than groups of letters of the alphabet. So right away you now know that either of the two **P** keys would be used to type the 'p' sound, either of the two **R** keys would be used to type the 'r' sound, and so on. That's already a lot of consonant sounds available for typing, without having to memorize anything. The current lesson will focus on these consonants and the four somewhat more complicated vowel keys; the rest will be covered in later lessons.

Before anything else, you should learn the proper keyboard fingering. If you've ever studied touch-typing on the QWERTY keyboard, you probably spent quite a bit of time getting comfortable with the home position and with the proper ways to stretch your fingers to reach all the keys. On a steno keyboard, none of that complexity exists, and you can learn proper fingering in about one minute.

In the following diagram, the blue boxes contain the keys associated with each finger, and the tan circles represent the home position for the finger responsible for the keys in that blue box.

Fig. 2: Fingering And Home Position





As you can see, with the exception of the left little finger, each finger rests at the border between two keys; and that finger is used to press either (or both) of those keys. The right little finger has the additional responsibility of the **D** and **Z** keys, and the right forefinger is responsible for the asterisk. The thumbs rest on their sides, rather than on their pads. But you can easily see that unlike QWERTY keyboards, none of your fingers will ever move very far from their home positions. This helps with the blazingly fast typing speeds.

You're almost ready to start typing words, but there's a detail or two regarding the **F** and **G** keys, and we need to go over the vowel keys as well.

- **F**: In addition to the 'f' sound, this key is also used to indicate the 'v' sound (and sometimes 's', as you'll see in [Lesson 2: Steno Order](#)).
- **G**: In addition to the hard 'g' sound used in words like 'hug' and 'rag', this key may also be used to indicate the 'ing' sound, in words like 'using' and 'holding'.

Rules For Short Vowel Sounds

If you're not sure about the difference between long and short vowels, or what diphthongs are, here's a useful [article on long and short vowels](#) and an [article on diphthongs](#). The rest of this lesson, and future lessons, all assume you have a clear understanding of those things.

Rule 1: If the vowel letter in the written word appears without another vowel letter next to it, then it doesn't matter which short vowel sound it makes. The Plover key labeled with that letter is *always* used to stroke the word.

So 'not' is stroked with **O** because it's spelled with "o", and 'tar' is stroked with **A** because it's spelled with "a". Likewise, 'pert' uses **E** and 'purse' uses **U**.

Rule 2: If the vowel sound in the written word is spelled with two or more consecutive vowel letters, then it doesn't matter what spelling is used. The Plover key whose short vowel sound matches the sound in the word is *always* used to stroke the word.

So 'heart' is stroked with an **A** because it uses a short 'a' sound; 'roar' is stroked with **O** because it uses a short 'o' sound; 'head' is stroked with **E** because it uses a short 'e' sound;

'**tough**' is stroked with a **U** because it uses a short 'u' sound. Short vowel sounds like the 'a' in '**bat**' and the 'u' in '**put**' follow the same rule.

What About Long Vowels And Diphthongs?

A long vowel sounds kind of like saying the name of the letter. For example, the word '**bake**' uses a long 'a' sound, the word '**poke**' uses a long 'o' sound, the word '**teeth**' uses a long 'e' sound, and the word '**few**' uses a long 'u' sound. The 'oo' sound in words like '**glue**' and '**crew**' is also considered a long 'u' sound in Plover.

Diphthongs are where the sound of one vowel transitions to the sound of another, as in the '**ow**' sound in '**down**' and the '**oi**' sound in '**toil**'; or where two vowel sounds are merged together, as in the '**aw**' sound in '**dawn**'.

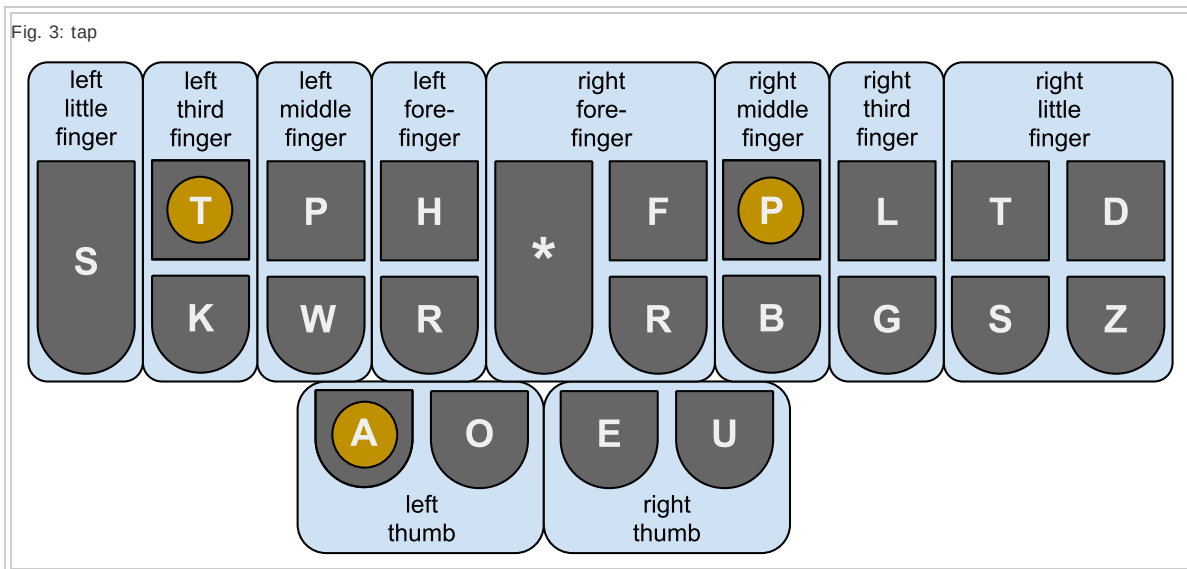
Plover handles long vowels and diphthongs by stroking two or more vowel keys together. See [3A: Vowels](#) for the full explanation.

If you're still a little fuzzy on the vowel sounds, don't worry. The exercises below will help.

Stroking Your First Words

Now you're ready to start typing words. Open a text editor, put your steno keyboard (or your converted QWERTY keyboard) in front of you, and activate the Plover software. Once you're set up, do the following steps:

- Rest your left third finger on the **T** key on the left half of the keyboard
- Rest your left thumb on the **A** key
- Rest your right middle finger on the **P** key on the right half of the keyboard



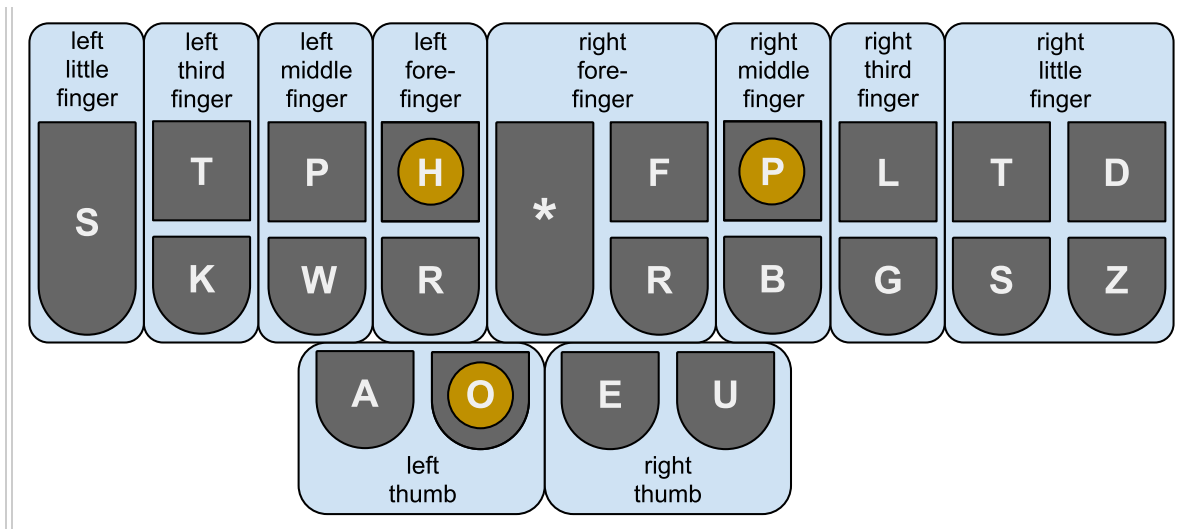
When your fingers are in position, press them all down together, and release them. Out comes the word **'tap'**! You've just tapped your first word in steno!

Notice that it doesn't really matter that all the keys go down absolutely simultaneously. The only thing Plover cares about is that there's one moment in time when all three keys are down together. If one goes down a little before the others, or comes up later than the others, Plover still does the right thing.

Try another word:

- Rest your left forefinger on the **H** key on the left half of the keyboard
- Rest your left thumb on the **O** key
- Rest your right middle finger on the **P** key on the right half of the keyboard

Fig. 4: hop



When your fingers are in position, press them all down together, and release them. You just stroked **'hop'**! Now hop for joy - you just tapped your second word in steno!

One important detail to take note of is the fact that it matters which side of the keyboard contains the consonant sound you want. In the case of **'tap'** and **'hop'**, you used the **P** from the right side of the keyboard rather than the left. This is because the **P** was needed at the tail end of the syllable rather than at the start of it. A more rigorous explanation of this can be found in [Lesson 2: Steno Order](#). For now, just remember that the start of a syllable is keyed on the left side of the keyboard, and the end of a syllable is keyed on the right. The vowels are keyed toward the middle of the keyboard, just as they usually occur in the middle of a syllable.

Think of a few more one-syllable words that use the consonant and vowel sounds described above, and try stroking them on your own. But don't just press keys at random - if you press keys that don't sound out a real word, it's possible Plover will output one of its many *briefs* (special time-saving keystrokes, covered in [Lesson 4: Common Briefs](#)); and that won't help you learn the keyboard. Just take your time, think of words that use the sounds represented on the keyboard, and stroke them in, just as you did with **'tap'** and **'hop'**.

Exercise: One-Syllable Words

It's a bit too soon for whole sentences, but try to figure out the chords for these words, and stroke them out on the keyboard. Practice until you can type each word without looking at the keyboard or reviewing the meanings of the vowel keys. Note that there's a little surprise in this exercise, that I'll explain afterwards:

- 'sap' 'sag' 'sat' 'sass' 'sad' 'sop' 'sob' 'sell' 'set' 'says'
- 'tar' 'tap' 'tab' 'tag' 'tad' 'tour' 'top' 'toll' 'tell' 'tough' 'tub' 'tug'
- 'car' 'cap' 'cab' 'cat' 'cad' 'core' 'cop' 'cog' 'cot' 'cod' 'keg'
'cuff' 'cur' 'cup' 'cub' 'cull' 'cut' 'cuss'
- 'pal' 'pat' 'pass'
'pad' 'pour' 'poll' 'pot' 'pod' 'pep' 'peg' 'pet' 'puff' 'pup' 'pub' 'pull' 'pug' 'put' 'pus'
- 'war' 'wag' 'wad' 'was' 'wore' 'web' 'well' 'wet' 'wed'
- 'half' 'hag' 'hat' 'had' 'has' 'hop' 'hog' 'hot' 'her' 'hell' 'head' 'huff' 'hub' 'hull' 'hug'
'hut'

- 'rap' 'rag' 'rat' 'roar' 'rob' 'roll' 'rot' 'rod' 'red' 'rough' 'rub' 'rug' 'rut'

Try the same words in a different order:

- 'half' 'tough' 'cuff' 'puff' 'huff' 'rough'
- 'tar' 'car' 'war' 'tour' 'core' 'pour' 'wore' 'roar' 'her' 'cur'
- 'sap' 'tap' 'cap' 'rap' 'sop' 'top' 'cop' 'hop' 'pep' 'cup' 'pup'
- 'tab' 'cab' 'sob' 'rob' 'web' 'tub' 'cub' 'pub' 'rub'
- 'pal' 'toll' 'poll' 'roll' 'sell' 'tell' 'well' 'hell' 'cull' 'pull' 'hull'
- 'sag' 'tag' 'wag' 'hag' 'rag' 'cog' 'hog' 'keg' 'peg' 'tug' 'pug' 'hug' 'rug'
- 'sat' 'cat' 'pat' 'hat' 'rat' 'cot' 'pot' 'hot' 'rot' 'set' 'pet' 'wet' 'cut' 'put' 'hut' 'rut'
- 'sass' 'pass' 'cuss' 'pus'
- 'sad' 'tad' 'cad' 'pad' 'wad' 'had' 'cod' 'pod' 'wed' 'head' 'red'
- 'has' 'says' 'was'

There are some [interactive drills that correspond to the 'Learn Plover' exercises](#). Try the drills for [one syllable words](#).

Although these exercises are for single words as you make progress feel to try other practice methods which include words in context. A [list of resources](#) is available on the Plover Wiki.

Considerations

Were you able to stroke all those words without looking any of them up? You might have had trouble with '**says**', if you tried to stroke it **SES**. The correct stroke is **SEZ**, because the '**s**' at the end of the word is really pronounced like a '**z**'.

You probably had no trouble stroking the word '**was**', whether you conceived of it phonetically as **WAS**, **WAZ**, or **WUZ**. Regardless of which one you tried, you guessed right. The Plover dictionary file defines each of those as '**was**'. Multiple dictionary entries for the same word is quite common with Plover and other steno theories. It helps reduce the impact of misstrokes. Of course, this approach only works if there are no similar-sounding words that need those strokes. But Plover tries to anticipate as many different reasonable ways of stroking a word as possible.

If you made any mistakes at all in this exercise, you probably noticed some unexpected output. These would take two possible forms. Either you saw an all-caps representation of your keypresses, such as '**HOL**' or '**HOB**', or you saw words that you didn't know you were typing, such as '**possible**' or '**suggest**'. The all-caps output just means that Plover has no dictionary entry for the chord you pressed; while the unexpected words are called briefs - they save keystrokes by letting you type fewer strokes (or just one) to get a word that might otherwise require several strokes pressed in sequence. Briefs are an important topic in Plover, but we'll cover them more in [Lesson 4: Common Briefs](#), and [Lesson 9: Designing Briefs](#).

Sometimes it's possible to run into trouble without making any mistakes. The two sets of words were the same in this exercise, but just had a different order. If you typed the second set as listed, however, you probably noticed that '**pad**' and '**wad**', when stroked one after the other, produced the odd-looking word '**Paduaed**'. Even odder, if you look at the Plover dictionary file, '**Paduaed**' isn't even listed. What's going on?

(Update: As of Plover 3.0.0 although **PAD/WA** does lead to 'Padua' **PAD/WAD** produces 'pad' and 'wad'. The discussion is still the same even though that doesn't occur.)

This is really a sneak peak into future lessons. The main thing that happened is that you encountered a word boundary error. Since Plover inserts all spaces for you, it has to try to be smart about where to put the spaces. And since some words are typed with more than one stroke, Plover sometimes can't tell what you want it to do. Whenever that happens, it's called a word boundary error. In this case, Plover saw the pair of strokes **PAD/WA** and thought you meant the city of 'Padua'. But Plover also saw that you added a **D** to the final stroke, to get **PAD/WAD**, which Plover interpreted as adding the past tense '-ed' suffix to the word. Imagine saying something like 'Dude! That guy got totally Paduaed!'

For more on these topics, see [Appendix: Word Boundaries](#), and [Lesson 5: Prefix And Suffix Strokes](#).

However, since all the lessons up to Lesson 5 deal with single stroke words, I can tell you a quick work-around if you encounter any more word boundary errors with single stroke words. First, use the * key to undo Plover's unexpected output. Then type each word again, with the stroke **S-P** between them, to make sure Plover knows to break the words there. So in this case, if you see 'Paduaed', type * twice to get rid of both strokes, then type **PAD** then **S-P** then **WAD** to get 'pad wad' as you expect. For other ways to get around word boundary errors, see [Lesson 6: Prefix/Suffix Alternatives](#).

Exercise: Consonant Clusters In Single-Stroke Words

The previous exercise involved mostly three- and four-letter words. But it's sometimes possible to cluster consonants together into a single chord, to produce a longer word, or even a multi-syllable word, with that one stroke. For example, the word 'scrabble' can be stroked **SKRABL**. The three consonant sounds 's', 'k', and 'r' at the start of that word are also available on the left side of the keyboard as the **S**, **K**, and **R** keys. The next sound, 'a', also fits in after that. The final two sounds, 'b' and 'l', can be stroked with the **B** and **L** keys on the right side of the keyboard. And that's the whole word. All the sounds in the word 'scrabble' were able to fit into a single stroke. Try to stroke the following examples of similar words. Don't look at the keyboard any more than you have to:

- 'course' 'cover' 'hover' 'rabble' 'refer' 'rebel' 'robbed' 'rubbed' 'rubble' 'straps'
'strapped' 'trouble' 'troubles' 'waft' 'webbed'

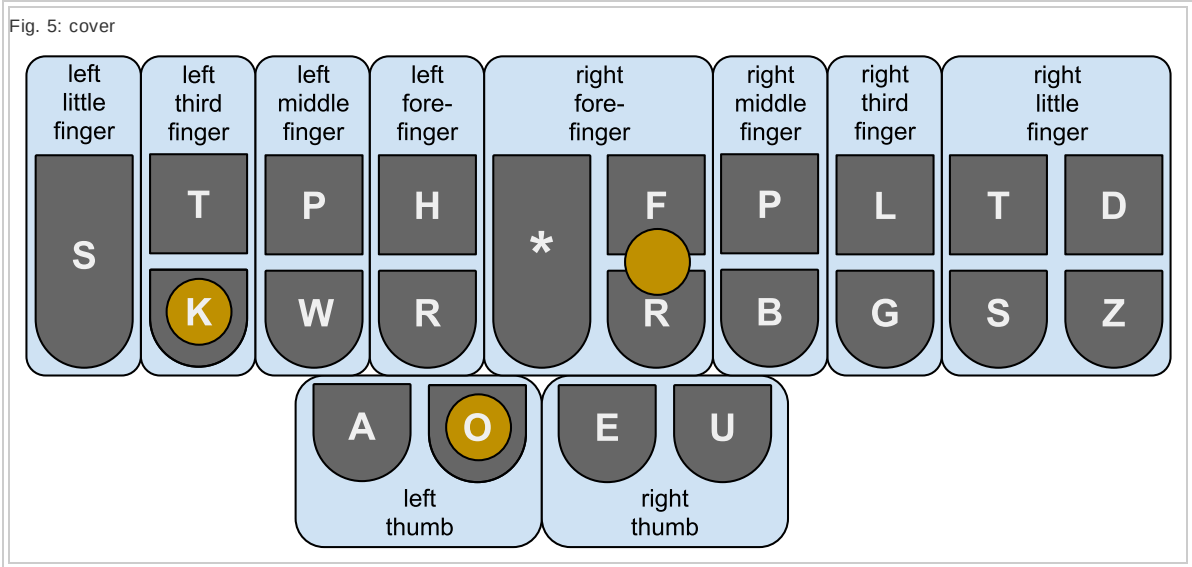
Now look at the Plover keyboard and try to find more words that could be typed in a single stroke but that use more than one consonant before or after the vowel sound. Take your time, and stroke the words you discover. See if Plover outputs the word you expected, or something else. Don't get discouraged to find that your chords don't always make the words you expect - part of the point of the exercise is to show that there's something significant we haven't covered yet, that's coming up in the next lesson.

Try the interactive drills for [consonant clusters](#).

Considerations

Did you have any trouble figuring out the chords for the words given? They all were pretty phonetic. But you may have had a problem if you tried to stroke 'strapped' as **STRAPT**. The correct stroke is **STRAPD**.

It also may not have been obvious what fingering to use when stroking 'cover', 'hover', and 'refer'. The **F** and the **R** keys on the right side of the keyboard are both required by each of these words, but only your right index finger is assigned to those keys. You may have been tempted to press one key with your forefinger, and the other with another finger. But the correct way is to use the forefinger alone, pressing down at the border between the two keys.



In the case of the right little finger, it's possible that you might need to press the **T**, **S**, **D**, and **Z** keys all at once. In that case, the correct way is to use the little finger alone, pressing down at the adjacent corners of all four keys.

Back to this exercise. What happened when you tried making your own longer words? Did it work as you expected? You may have found that some of the chords you came up with didn't produce the words you expected, or even may not have produced *any* word, but just the raw Plover keystroke. If you tried to write the word 'trust', for example, you may have used the stroke **TRUST**, and discovered that this didn't work. Likewise if you tried to stroke 'tassle' as **TASL**.

This leads us to the all-important [Lesson 2: Steno Order](#).

Where Are We?

This lesson was long, but look at all the words you can type already!

By the end of this lesson, you should be able to:

- draw a correctly labeled picture of the Plover keyboard without looking at the actual keyboard
- identify all the consonant sounds represented by the different keys
- understand the rules governing how to stroke short vowel sounds
- correctly stroke some single-syllable words on your keyboard, typing earlier consonant sounds on the left side of the keyboard, and later consonant sounds on the right

If you're still fuzzy on some of the above, you should repeat the parts of this lesson that gave you trouble. Later lessons assume that you've mastered the material covered in this one.

Solutions To Exercises

Solution: Single-Syllable Words

- 'had': HAD
- 'half': HAF
- 'hag': HAG
- 'has': HAZ or HAS
- 'hat': HAT
- 'head': HED
- 'hell': HEL
- 'her': HER
- 'hog': HOG
- 'hop': HOP
- 'not': NOT
- 'hub': HUB
- 'huff': HUF
- 'hug': HUG
- 'hull': HUL
- 'hut': HUT
- 'cab': KAB
- 'cad': KAD
- 'cap': KAP
- 'car': KAR
- 'cat': KAT
- 'keg': KEG
- 'cod': KOD
- 'cog': KOG
- 'cop': KOP
- 'core': KOR
- 'cot': KOT
- 'cub': KUB
- 'cuff': KUF
- 'cull': KUL
- 'cup': KUP
- 'cur': KUR
- 'cuss': KUS
- 'cut': KUT
- 'pad': PAD
- 'pal': PAL
- 'pass': PAS
- 'pat': PAT

- 'peg': PEG
- 'pep': PEP
- 'pet': PET
- 'pod': POD
- 'poll': POL
- 'pour': POR
- 'pot': POT
- 'pub': PUB
- 'puff': PUF
- 'pug': PUG
- 'pull': PULL
- 'pup': PUP
- 'pus': PUS
- 'put': PUT
- 'rag': RAG
- 'rap': RAP
- 'rat': RAT
- 'red': RED
- 'rob': ROB
- 'rod': ROD
- 'roll': ROLL
- 'roar': ROR
- 'rot': ROT
- 'rub': RUB
- 'rough': RUF
- 'rug': RUG
- 'rut': RUT
- 'sad': SAD
- 'sag': SAG
- 'sap': SAP
- 'sass': SAS
- 'sat': SAT
- 'sell': SEL
- 'set': SET
- 'says': SEZ
- 'sob': SOB
- 'sop': SOP
- 'tab': TAB
- 'tad': TAD
- 'tag': TAG
- 'tap': TAP
- 'tar': TAR
- 'tell': TEL

- 'toll': TOL
- 'top': TOP
- 'tour': TOR
- 'tub': TUB
- 'tough': TUF
- 'tug': TUG
- 'wad': WAD
- 'wag': WAG
- 'war': WAR
- 'was': WUZ, WAS, or WAZ
- 'web': WEB
- 'wed': WED
- 'well': WEL
- 'wet': WET
- 'wore': WOR

Solution: Consonant Clusters In Single-Stroke Words

- 'course': KORS
- 'cover': KOFR
- 'hover': HOFR
- 'rabble': RABL
- 'refer': REFR
- 'rebel': REBL
- 'robbed': ROBD
- 'rubbed': RUBD
- 'rubble': RUBL
- 'straps': STRAPS
- 'strapped': STRAPD
- 'trouble': TRUBL
- 'troubles': TRUBLS
- 'waft': WAFT
- 'webbed': WEBD

Lesson 2: Steno Order

Contents

- 1 Single-Stroke Versus Multi-Stroke Words
- 2 Steno Order
- 3 Where's The TRUFT?
 - 3.1 Exercise: Where's The TRUFT?
 - 3.2 Considerations
- 4 Dropping Unstressed Vowels
 - 4.1 Exercise: Dropping Unstressed Vowels

4.2 Considerations

5 Inversion

5.1 Exercise: Inversion

5.2 Considerations

6 Solutions To Exercises

6.1 Solution: Where's The TRUFT?

6.2 Solution: Dropping Unstressed Vowels

6.3 Solution: Inversion

Single-Stroke Versus Multi-Stroke Words

One of the most important things to know about Plover is that virtually all the rules - really almost everything you need to explicitly *learn* - applies only to single strokes of the keyboard. In the current lesson, and in the next bunch of lessons, you'll learn chords for all the different sounds, and how to type lots of words using just a single stroke. In almost all cases, single-stroke words are one or maybe two syllables. If you didn't know better, you might start to wonder when we were going to get to bigger words like '**encyclopedia**' and '**rambunctious**' - words that would require more than a single stroke to produce phonetically.

In fact, each stroke of a multi-stroke word obeys all the rules laid down in the lessons for typing single-stroke words. By learning how to type single-stroke words, you're also learning how to type multi-stroke words.

What might still seem unclear, however, is how to correctly tell what *part* of a longer word goes in the 1st stroke, the 2nd stroke, and so on. When experts and advocates talk about steno theories, they often say that longer words are split up syllabically. But that's an oversimplification. As you'll see in the current lesson, many multi-syllable words can be typed with just a single stroke. Clearly all steno theories obtain much of their speed by finding ways to fit multiple syllables into single strokes - and Plover is the same.

Plover - and all steno theories - solve the question of multi-stroke words in a very elegant way: **As long as you follow the rules for constructing single strokes, you can - and *should* - divide a longer word into a series of strokes that feels most natural and intuitive to you.**

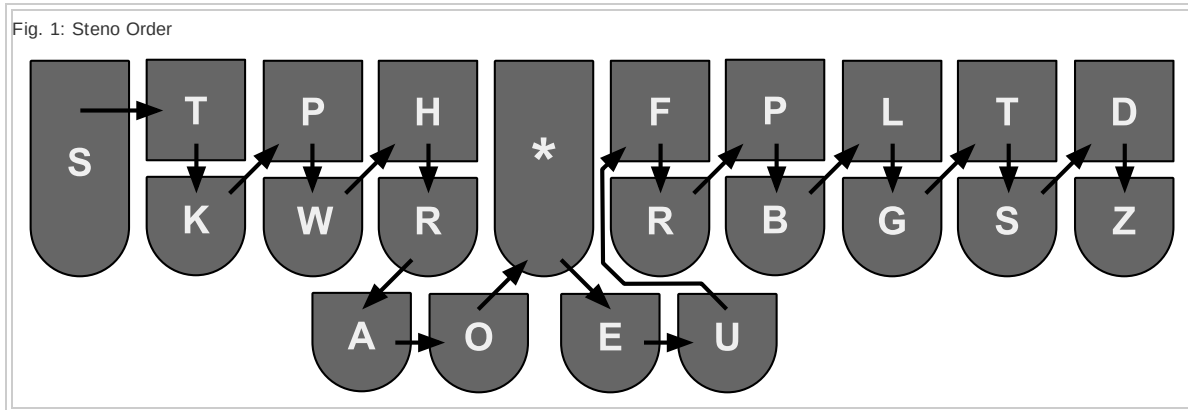
The default Plover dictionary provides many alternative entries for multi-stroke words, and the chances are high that any given multi-stroke word already has an entry that matches what feels right to you. But for the multi-stroke words that don't, **the recommended and correct approach is to *create a new dictionary entry with your preferred set of strokes*, rather than try to memorize any of the existing entries for that word.**

This is all just by way of explaining why most of the upcoming lessons cover only single-stroke words. The reason is that single-strokes *are* Plover theory. A single stroke may be used to type a whole word, or it may be used to type part of a bigger word. But Plover theory itself is about typing that single stroke.

Having said that, let's look more deeply at what is actually meant by a single stroke, and at how much of a word we can fit into it.

Steno Order

In [Lesson 1: Fingers And Keys](#), I mentioned that chord sounds are constructed from left to right on the keyboard. But there is actually a formal order to this, called "steno order":



Written out as a single string of characters, this looks like **STKPWHRAO*EUFRLPBLGTSDZ**. Memorize this string of characters until it comes absolutely naturally to you. It's the cornerstone of the entire system. [Lars Doucet](#) has come up with this funny mnemonic to help learn steno order:

"Should The King Please Wear His Red And Orange STARry Elephant Underwear For Royal Purposes Before Lunch Gets Totally Super Devoured (by) Zombies?"

If you try to see the chord for a word within the steno order string, you'll find that all the examples in [Lesson 1: Fingers And Keys](#) can be done entirely by reading from left to right. Consider a few examples:

- 'pad': **STKPWHRAO*EUFRLPBLGTSDZ**
- 'roll': **STKPWHRAO*EUFRLPBLGTSDZ**
- 'tub': **STKPWHRAO*EUFRLPBLGTSDZ**
- 'straps': **STKPWHRAO*EUFRLPBLGTSDZ**

Any word that fits into steno order can be typed in a single stroke. That's why memorizing steno order is so important. The faster you can call it to mind in a useful way, the faster you'll be able to decide on the appropriate stroke for a word.

But there are also several standard ways of getting a word into a single stroke when it seems like it shouldn't fit. The rest of this lesson covers these standard ways.

Where's The TRUFT?

In [Lesson 1: Fingers And Keys](#), recall that the word 'trust' didn't chord the way we expected. That's because if we try to sound it out in steno order, we get **STKPWHRAO*EUFRPBLGTSDZ**. Note that the **T** and the **S** are out of order. And when we try to sound out 'tassle' in steno order, we get **STKPWHRAO*EUFRPBLGTSDZ**, where the **L** and the **S** are out of order.

At the top of Lesson 1, I mentioned that in addition to being used for 'f' and 'v' sounds, the **F** key was also sometimes used for 's' sounds. But it's only now, with an understanding of steno order, that the reason for this becomes clear. After all, there's already an **S** key on the right half of the keyboard - why would we need another on that half as well?

With a grasp of steno order, it's easy to explain: the **S** key on the right side of the keyboard is very far to the right - near the end of steno order. This placement is convenient because plurals and possessives tend to end in 's'. But a lot of words, like 'trust', 'rasp', and 'tassle', have consonant clusters where an 's' sound is followed by another consonant. If you try to make chords for these words using the regular **S** key, they violate steno order:

- 'trust': STKPWHRAO*EUFRPBLGTSDZ (incorrect)
- 'rasp': STKPWHRAO*EUFRPBLGTSDZ (incorrect)
- 'tassle': STKPWHRAO*EUFRPBLGTSDZ (incorrect)

The above chords don't produce the words you want. But the use of 's' in this sort of consonant cluster occurs so frequently in English that Plover has a special solution for it: if you use the ability of the **F** key to represent the 's' sound, the words no longer violate steno order:

- 'trust': STKPWHRAO*EUFRPBLGTSDZ
- 'rasp': STKPWHRAO*EUFRPBLGTSDZ
- 'tassle': STKPWHRAO*EUFRPBLGTSDZ

It may look weird at first, but you get used to it.

You may be wondering what happens when you encounter a conflict - words like 'cost' and 'coughed', or 'rust' and 'roughed', where either the 'f' or the 's' sounds would produce legitimate English words. How would **KOFT** and **RUFT** be interpreted by Plover? Would the **F** key result in an 'f' sound or an 's' sound?

First of all, there are relatively few of these cases. For the ones that do exist, Plover has a variety of 'disambiguation' techniques to define different chords for each conflicting word. Disambiguation is an important topic in Plover, and it's covered in a later tutorial. For now, focus on the keyboard, the details of steno order, and the ways to combine sounds into chords that make words.

Exercise: Where's The TRUFT?

Use your knowledge of steno order and the **F** key to construct chords for these words:

- 'past' 'castle' 'stressed' 'pressed' 'passed' 'test' 'tussle' 'crossed'

Practice these until you can do them easily, and don't find it too unnatural to hit the **F** key to get an 's' sound.

Try the interactive drills for ['Where's The TRUFT'](#).

Considerations

Fluency with the **F** key is important. A lot of words combine 's' with another consonant sound. The need to get around steno order in this way - placing the 's' earlier in the chord - is so common in English that most steno theories, including Plover, give the **F** key this special power.

Dropping Unstressed Vowels

In English, vowel sounds can either be stressed or unstressed. See the [Wikipedia article on linguistic stress](#).

When stroking multi-syllable words in Plover, unstressed vowels don't need to be included in a stroke. They can just be dropped. This is very useful because steno order only allows for a single vowel sound in any one stroke. Up till now we've only typed single-syllable words. By dropping unstressed vowels in multi-syllable words, it becomes possible to type many such words in just a single stroke.

For example, the two-syllable word **'tablet'** can be typed in a single stroke, as **TABL**T. Without the unstressed **'e'** sound, the rest of the consonants have no trouble fitting into steno order. Likewise, **'rugged'** is stroked with the single chord RUGD.

Exercise: Dropping Unstressed Vowels

Try to figure out the chords for the following words. They all have at least two syllables, but can be chorded with just one stroke:

- **'several'** **'suppress'** **'averages'** **'tablet'** **'tepid'** **'superb'** **'scaffold'** **'scarlet'**
'starlet' **'started'** **'ruffled'** **'scuffled'** **'corrupted'** **'spotted'** **'horrible'** **'effort'**

Try the interactive drills for '[Dropping Unstressed Vowels](#)'.

Considerations

Not everyone has a conscious understanding of whether a vowel is stressed or unstressed. It may not help that some words like **'rebel'** can have different meanings depending on which syllable has the stress. It also may not help that certain words like **'umbrella'** are pronounced **'umbrella'** by some people, and **'umbrella'** by others.

In fact, those few special cases can be solved by adding appropriate dictionary entries to match the different pronunciations.

It's also important to remember that the rule about dropping unstressed vowel sounds applies *only* to multi-syllable words. Don't bother trying to figure out if the vowel sound in a word like **'me'** is stressed or unstressed, because you should always include the vowel sounds for single-syllable words, regardless of stress.

In steno a good understanding of stressed and unstressed vowels in multi-syllable words is important. If you're having trouble with it, just keep reading articles about it online, and talking to people. If you know anyone who loves sonnets and old poetry, ask them to help you with vowel stress. Regardless of whether you consciously understand it or not, vowel stress *is* a part of English, and you use it every time you speak. You should be able to understand it in time.

Inversion

In steno, a word is allowed to violate steno order if it meets *all* of the following conditions:

1. it saves a stroke
2. it only happens once in a word
3. it only inverts two sounds that are next to each other in the word
4. it does not conflict with other entries in the dictionary

A good example of inversion is the word '**portal**'. Without the rules in this lesson, you'd have to stroke something like **POR** and **TAL**, taking two strokes to type the word.

First of all, the "a" in '**portal**' is really an unstressed vowel, so you can drop it, and end up with '**portl**'. But if you try and put that in steno order, you get **PORLT**, where the **L** and the **T** are out of order. But thanks to the inversion rule, you don't have to worry about that. You can correctly stroke '**portal**' as **PORLT**, because the inversion reduces the number of needed strokes from two to one, only happens once, only inverts the neighboring sounds 'l' and 't', and has no conflicts in the dictionary.

If there were another word '**porlet**' that meant something in English, that word would have a more natural claim to the stroke **PORLT**, and so '**portal**' would have to use a different stroke, or perhaps even the two strokes **POR** and **TAL**.

Exercise: Inversion

- 'edit' 'elves' 'twelve' 'credit' 'portal'

Try the interactive drills for '[Inversion](#)'.

Considerations

Did you notice that the inversion rule could also have come to our rescue when we tried to type '**trust**' as **TRUTS**? Why would we need to overload the **F** key to mean 's' in addition to meaning 'f' and 'v'? Why not just stroke **TRUTS** and be done with it?

The reason is that the 's' sound appears so frequently before other consonants in English that it's better to give it its own rule than to heap it onto the inversion rule. Since inversion is only allowed once in a given word, this means that you can use **F** for an 's' sound, *and* use inversion in the same word. For example '**worst**' is stroked as **WOFRT**. In that case, the **F** is used for 's', and the **F** and **R** use the inversion rule to fit the whole word into a single stroke.

Where Are We?

By the end of this lesson, you should be able to:

- figure out steno order with your eyes closed, even if it takes a minute
- likewise, figure out steno order backwards
- use the F key to make the 's' sound in a word
- use your ability to drop unstressed vowels to express some multi-syllable words in a single stroke

- use inversion to chord some words that would otherwise not fit into steno order

Solutions To Exercises

Solution: Where's The TRUFT?

- 'past': PAFT
- 'castle': KAFL
- 'stressed': STREFD
- 'pressed': PREFD
- 'passed': PAFD
- 'test': TEFT
- 'tussle': TUFL
- 'crossed': KROFD

Solution: Dropping Unstressed Vowels

- 'several': SEFRL
- 'suppress': SPRES
- 'averages': AFRGS
- 'tablet': TABLT
- 'tepid': TEPD
- 'superb': SPERB
- 'scaffold': SKAFLD
- 'scarlet': SKARLT
- 'starlet': STARLT
- 'started': STARTD
- 'ruffled': RUFLD
- 'scuffled': SKUFLD
- 'corrupted': KRUPTD
- 'spotted': SPOTD
- 'horrible': HORBL
- 'effort': EFRT

Solution: Inversion

- 'edit': ETD
- 'elves': EFLS
- 'twelve': TWEFL
- 'credit': KRETD
- 'portal': PORLT

Lesson 3: English Sounds

This lesson is divided into several parts, each covering some of the remaining phonemes used in English, as well as a few additional chords used for common compound clusters. When you reach the end of [Lesson 3: English Sounds](#), you should be able to type lots of single-stroke words.

[Lesson 3: English Sounds](#) >

3A: Vowels

- [3A-1: The Fifth Vowel Key](#)
- [3A-2: Long Vowels](#)
- [3A-3: Diphthongs](#)
- [3A-4: Vowel Disambiguators](#)

[Lesson 3: English Sounds](#) > [3A: Vowels](#) >

3A-1: The Fifth Vowel Key

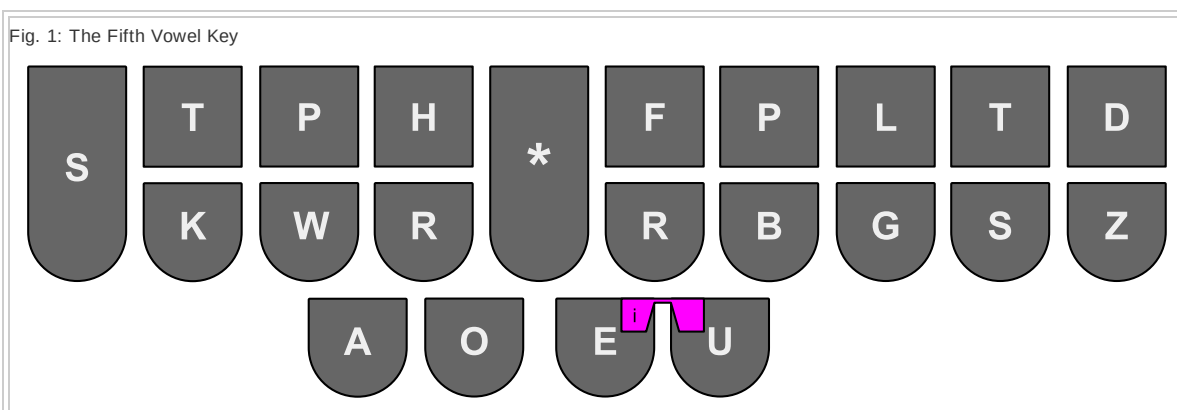
Contents

- 1 The Fifth Vowel Key
 - [1.1 Exercise: The Fifth Vowel Key](#)
 - [1.2 Considerations](#)
- 2 Exercise Solutions
 - [2.1 Solution: The Fifth Vowel Key](#)

The Fifth Vowel Key

Everyone who has English as their first language grows up learning the five vowel letters "a", "e", "i", "o", and "u", and how one of them has to be present in (almost) every word, or else it's not a word. In [Lesson 1: Fingers And Keys](#), however, we saw that Plover only has individual keys for four of those vowels. But what about "i"?

In fact, the short 'i' sound is chorded **EU**. Plover, and all steno theories, define special chords to indicate different phonetic sounds. The current lesson, and many other lessons here, are all about defining chords and assigning them to English sounds. Note that the keys used in a chord may not be intuitively related to the sound. The letter "i" is a case in point. The 'i' sound uses the **EU** chord, not because the sounds of 'e' and 'u' somehow combine to make the 'i' sound, but because Plover and other steno theories believe that that spot is the best place on the keyboard to represent the 'i' sound.



The **EU** chord really belongs in [Lesson 1: Fingers And Keys](#), with the **A**, **O**, **E**, and **U** keys. The same set of rules govern the **EU** chord as govern those others:

- When the desired word is spelled with the letter "i" without a neighboring vowel, and where the sound is a short vowel sound, **EU** is always used regardless of which short vowel sound the word has.
- When the desired word is spelled without an "i", or with "i" and another vowel together, to make a short 'i' sound, **EU** is used.
- The **EU** chord is *never* used alone to express a long 'i' sound, as in 'mile'.

If you look back to [Lesson 1: Fingers And Keys](#), you'll see that these rules match perfectly with those for **A**, **O**, **E**, and **U**. You can really think of **EU** as the fifth vowel key.

When we get to [Lesson 5: Prefix And Suffix Strokes](#), you'll see that **EU** is also sometimes used for the long 'e' sound.

Exercise: The Fifth Vowel Key

Try stroking these words, using the consonants covered in [Lesson 1: Fingers And Keys](#), and the rules covered in [Lesson 2: Steno Order](#). Practice until you can type them without looking at the explanations above.

- 'still', 'rig', 'hit', 'sip'
- 'sir', 'skirt'
- 'crypt', 'syrup'
- 'pig' 'rift' 'scribble' 'rid' 'river'
- 'hid' 'wilt' 'wig' 'wit' 'spill'

Try stroking any others you can think of. Do you get the words you expect?

Try the interactive drills for '[The Fifth Vowel Key](#)'.

Considerations

It may seem weird to type words using keys that don't seem to sound like the words. But this is one of the fundamental elements of Plover and all steno. Try to think of the keypresses as meaning the sounds that you want them to mean, rather than the letters printed on the keys. Ultimately you won't be looking at those letters anyway; you'll be looking up at the screen.

Exercise Solutions

Solution: The Fifth Vowel Key

- 'still': STEUL
- 'rig': REUG
- 'hit': HEUT
- 'sip': SEUP
- 'sir': SEUR
- 'skirt': SKEURT
- 'syrup': SEURP
- 'tryst': TREUST
- 'pig': PEUG
- 'rift': REUFT
- 'scribble': SKREUBL
- 'rid': REUD
- 'river': REUFR
- 'hid': HEUD
- 'wilt': WEULT
- 'wig': WEUG
- 'wit': WEUT
- 'spill': SPEUL

[Lesson 3: English Sounds > 3A: Vowels >](#)

3A-2: Long Vowels

Contents

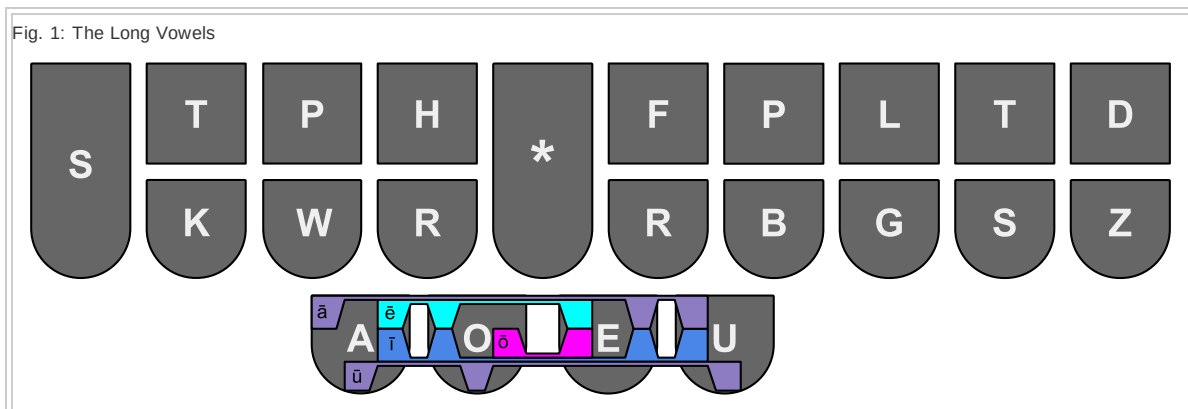
- 1 Long Vowel Chords
 - 1.1 Exercise: Long Vowel Chords
- 2 Exercise Solutions
 - 2.1 Solution: Long Vowel Chords

Long Vowel Chords

As in the case of **EU**, the four vowel keys **A**, **O**, **E**, and **U** can be combined into chords to indicate other vowel sounds. But again, it's important to remember that this isn't about combining the *sounds* of 'a', 'o', 'e', and 'u'. We're simply constructing unique key combinations, and *assigning* vowel sounds to them.

Here are all the long vowel sounds in English:

- **AEU**: This is used for the 'a' sound in words like 'grape' and 'saint'
- **AOE**: This is used for the 'ee' sound in words like 'seen' and 'preach'
- **AOEU**: This is used for the 'i' sound in words like 'guise' and 'spite'
- **OE**: This is used for the 'oh' sound in words like 'boat' and 'grown'
- **AOU**: This is used for two types of sounds:
 - The 'oo' sound in words like 'glue', *but not* when spelled "oo"
 - The 'ew' sound in words like 'few'



Exercise: Long Vowel Chords

Try stroking these words, using the consonants covered in [Lesson 1: Fingers And Keys](#), and the rules covered in [Lesson 2: Steno Order](#). Practice until you can type them without looking at the explanations above.

- 'aids' 'ace' 'ate' 'able' 'ape' 'raid' 'raise' 'rail' 'rate' 'pay' 'paid' 'pace' 'tape' 'spray' 'praise'
- 'weaver' 'trees' 'eel' 'eat' 'evil' 'ear' 'heat' 'heap' 'wield' 'weird' 'peer' 'priest' 'tree' 'tweeze' 'tweed' 'seat' 'cease' 'seed' 'seize' 'secrete'
- 'ire' 'ill' 'ice' 'rife' 'ripe' 'right' 'height' 'wild' 'pipe' 'pride' 'prize' 'kite' 'type' 'spite'
- 'hope' 'spore' 'post' 'sold' 'prose' 'ode' 'oat' 'over' 'robe' 'rope' 'roar' 'rove' 'host' 'wove' 'wrote' 'pole' 'pose' 'cope' 'coat' 'code' 'crow' 'told'
- 'cube' 'use' 'rude' 'rule' 'pure' 'prude' 'Proust' 'cure' 'cruel' 'crude' 'cruise' 'truce' 'truth' 'spew' 'skew'

Try the interactive drills for '[Long Vowel Chords](#)'.

Exercise Solutions

Solution: Long Vowel Chords

'ay':

- 'aids': AEUDZ
- 'ace': AEUS
- 'ate': AEUT
- 'able': AEUBL
- 'ape': AEUP
- 'raid': RAEUD
- 'raise': RAEUZ
- 'rail': RAEUL
- 'rate': RAEUT
- 'race': RAEUS
- 'pay': PAEU
- 'paid': PAEUD
- 'pace': PAEUS
- 'tape': TAEUP
- 'spray': SPRAEU
- 'praise': PRAEUZ

'ee':

- 'weaver': WAOEFR
- 'trees': TRAUES
- 'eel': AOEL
- 'eat': AOET
- 'evil': AOEFL
- 'ear': AOER
- 'heat': HAOET
- 'heap': HAOEP
- 'wield': WAOELD
- 'weird': WAOERD
- 'peer': PAOER
- 'priest': PRAOEFT
- 'tree': TRAOE
- 'tweeze': TWAOEZ
- 'tweed': TWAOED
- 'seat': SAOET
- 'cease': SAOES
- 'seed': SAOED
- 'seize': SAOEZ

- 'secrete': SKRAOET

'aye':

- 'ire': AOEUR
- 'ill': AOEUL
- 'ice': AOEUS
- 'rifle': RAOEUF
- 'ripe': RAOEUP
- 'right': RAOEUT
- 'height': HAOEUT
- 'wild': WAOEULD
- 'pipe': PAOEUP
- 'pride': PRAOEUD
- 'prize': PRAOEUZ
- 'kite': KAOEUT
- 'type': TAOEUP
- 'spite': SPAOEUT

'o':

- 'hope': HOEP
- 'spore': SPOER
- 'post': POEFT
- 'sold': SOELD
- 'prose': PROES
- 'ode': OED
- 'oat': OET
- 'over': OEFR
- 'robe': ROEB
- 'rope': ROEP
- 'roar': ROER
- 'rove': ROEF
- 'host': HOEFT
- 'wove': WOEF
- 'wrote': WROET
- 'pole': POEL
- 'pose': POES
- 'cope': KOEP
- 'coat': KOET
- 'code': KOED
- 'crow': KROE
- 'told': TOELD

'ew' and 'oo' (but not spelled "oo"):

- 'cube': KAOUB
- 'use': AOUZ
- 'rude': RAOUD
- 'rule': RAOUL
- 'pure': PAOUR
- 'prude': PRAOUD
- 'Proust': PRAOUFT
- 'cure': KAOUR
- 'cute': KAOUT
- 'crew': KRAOU
- 'cruel': KRAOUL
- 'crude': KRAOUD
- 'cruise': KRAOUS
- 'truce': TRAOUS
- 'truth': TRAOUT
- 'spew': SPAOU
- 'skew': SKAOU

[Lesson 3: English Sounds > 3A: Vowels >](#)

3A-3: Diphthongs

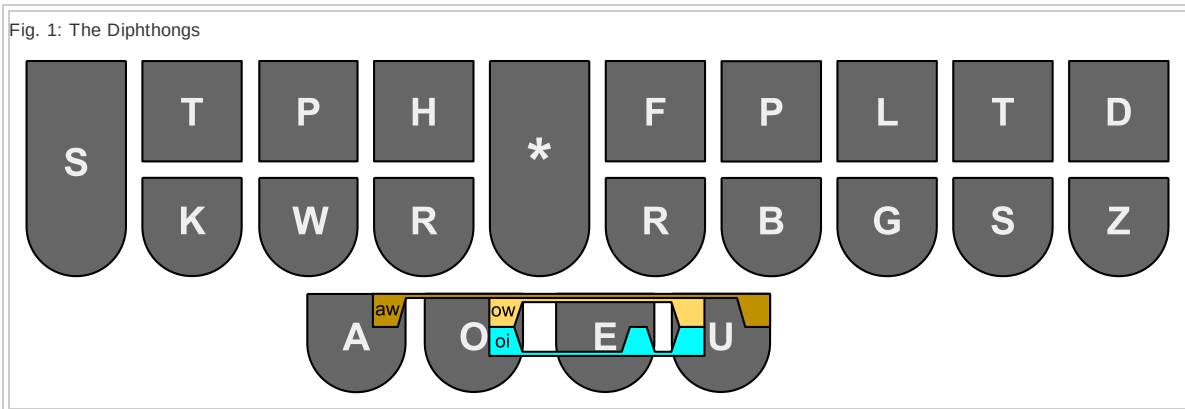
Contents

- 1 Diphthong Chords
- 2 Here are the English diphthongs:
 - 2.1 Exercise: Diphthong Chords
- 3 Solutions
 - 3.1 Solution: Diphthong Chords

Diphthong Chords

Here are the English diphthongs:

- **AU**: This is used for the 'aw' sound in words like 'bought', 'tawny', and 'faun'
- **OU**: This is used for the 'ow' sound in words like 'down' and 'mound'
- **OEU**: This is used for the 'oi' sound in words like 'toil' and 'ploy'



Exercise: Diphthong Chords

- 'all' 'awful' 'raw' 'call' 'caught' 'crawl' 'sprawl' 'scald' 'straw' 'halt' 'hall' 'wall'
- 'out' 'how' 'howl' 'house' 'pout' 'power' 'prowl' 'tower' 'spouse' 'sprout' 'scour'
- 'soy' 'oil' 'coil' 'toil' 'soil'

Try the interactive drills for ['Diphthong Chords'](#).

Solutions

Solution: Diphthong Chords

'aw':

- 'all': AUL
- 'awful': AUFL
- 'raw': RAU
- 'call': KAUL
- 'caught': KAUT
- 'crawl': KRAUL
- 'sprawl': SPRAUL
- 'scald': SKAULD
- 'straw': STRAU
- 'halt': HAULT
- 'hall': HAUL
- 'wall': WAUL

'ow':

- 'out': OUT
- 'how': HOU
- 'howl': HOUL
- 'house': HOUS
- 'pout': POUT
- 'power': POUR
- 'prowl': PROUL
- 'tower': TOUR
- 'spouse': SPOUS
- 'sprout': SPROUT
- 'scour': SKOUR

'oi':

- 'soy': SOEU
- 'oil': OEUL
- 'coil': KOEUL
- 'toil': TOEUL
- 'soil': SOEUL



Contents

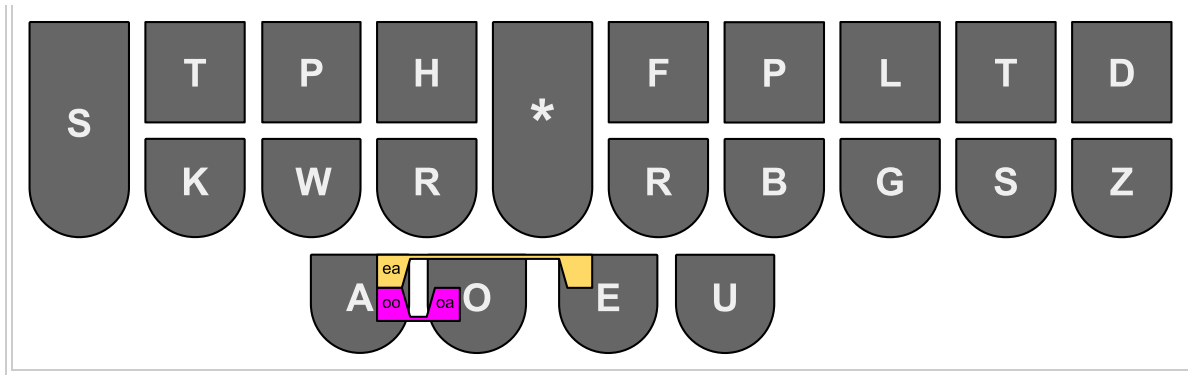
- 1** Vowel Disambiguator Chords
 - 1.1** Exercise: Vowel Disambiguator Chords
- 2** Solutions
 - 2.1** Solution: Vowel Disambiguator Chords

Vowel Disambiguator Chords

English spelling is very useful for creating homophones, which causes problems for all steno theories. Among Plover's various techniques for disambiguating words, it includes two very useful vowel chords, **AE** and **AO**. Unlike other vowel chords, these indicate a certain spelling as well as a certain sound:

- **AE**: This is used to disambiguate word conflicts where one of the words is spelled with "ea".
 - - The word '**pair**' is chorded **PAEUR** using the **AEU** chord, while the word '**pear**' is spelled with "ea" and so is disambiguated **PAER**.
 - The word '**beet**' is chorded **PWAOET** using the **AOE** chord, while the word '**beat**' is spelled with "ea" and so is disambiguated **PWAET**
 - The word '**bare**' is chorded **PWAEUR** using the **AEU** chord, while the word '**bear**' is spelled with "ea" and so is disambiguated **PWAER**
 - If neither word is spelled with an "ea", but one of them has both an "a" and an "e", the word with the "a" and "e" gets the **AE** chord:
 - - The word '**gait**' is chorded **TKPWAEUT** using the **AEU** chord, while the word '**gate**' has an "a" and an "e" and so is disambiguated **TKPWAET**
 - The word '**ail**' is chorded **AEUL** using the **AEU** chord, while the word '**ale**' has an "a" and an "e" and so is disambiguated **AEL**
- **AO**: This is used for two types of words:
 - - Words that are spelled with "oo", like '**book**' and '**floor**'
 - Words that have the same 'o' sound defined by **O** or **OE**, but that are spelled with "oa".
 - - The word '**horse**' is chorded **HORS**, while the word '**hoarse**' is spelled with an "oa" and so is disambiguated **HAORS**.
 - The word '**rode**' is chorded **ROED**, while the word '**road**' is spelled with an "oa" and so is disambiguated **RAOD**.

Fig. 1: Vowel Disambiguators



Exercise: Vowel Disambiguator Chords

- 'wheel' 'wheal' 'read' 'reed' 'reel' 'real' 'heel' 'heal' 'hear' 'here' 'ware' 'wear' 'pea' 'pee' 'peace' 'piece' 'tee' 'tea' 'sea' 'see'
- 'tail' 'tale' 'sale' 'sail' 'stare' 'stair' 'waist' 'waste'
- 'hood' 'rude' 'pool' 'crew' 'soot' 'truce' 'school' 'ruse'
- 'road' 'rode' 'roar' 'toad' 'soar' 'sore'

Try the interactive drills for ['Vowel Disambiguator Chords'](#).

Solutions

Solution: Vowel Disambiguator Chords

- 'wheel': WHAOEL
- 'wheal': WHAEL
- 'reed': RAOED
- 'read': RAED
- 'reel': RAOEL
- 'real': RAEL
- 'heel': HAOEL
- 'heal': HAEL
- 'here': HAOER
- 'hear': HAER
- 'ware': WAEUR
- 'wear': WAER
- 'pee': PAOE
- 'pea': PAE
- 'piece': PAOES
- 'peace': PAES
- 'tee': TAOE
- 'tea': TAE
- 'see': SAOE
- 'sea': SAE
- 'tail': TAEUL
- 'tale': TAEL
- 'sail': SAEUL
- 'sale': SAEL
- 'stair': STAEUR
- 'stare': STAER
- 'waist': WAEUFT
- 'waste': WAEFT
- 'hood': HAOD
- 'rude': RAOUD
- 'pool': PAOL
- 'crew': CRAOU
- 'soot': SAOT
- 'truce': TRAOUS
- 'school': SKAOL
- 'ruse': RAOUZ
- 'road': RAOD
- 'rode': ROED

- 'roar': RAOR
- 'toad': TAOD
- 'soar': SAOR
- 'sore': SOR

[Lesson 3: English Sounds >](#)

3B: k, d, f, l, g, b, z, and v

Contents

- 1** The Missing Keys
 - 1.1** Exercise: The Missing Keys
- 2** Exercise Solutions
 - 2.1** Solution: The Missing Keys

The Missing Keys

Some keys, like **S**, **T**, and **P**, are found on both sides of the keyboard. But keys like **K**, **D**, and **F** are only found on one side. The current lesson evens up this state of affairs, and relies on an idea you've already encountered in [Lesson 3A: Vowels](#) - that of assigning a particular sound to a set of keys.

On the left side of the keyboard:

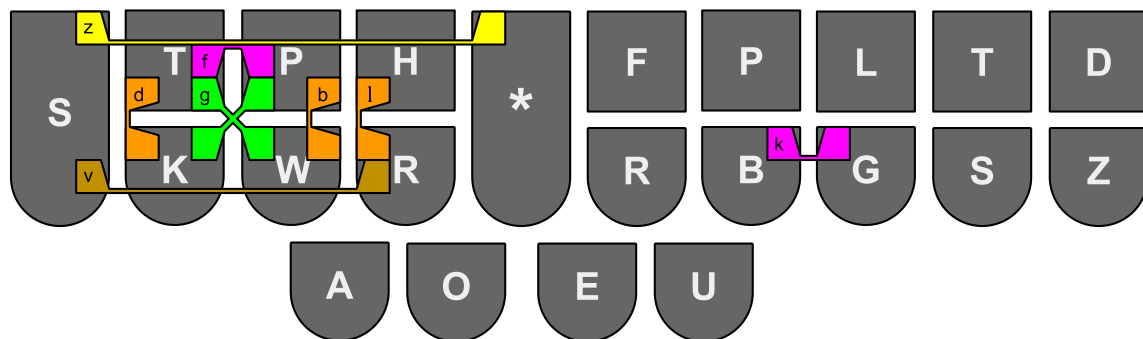
- **TK-**: this is used for the 'd' sound
- **TP-**: this is used for the 'f' sound
- **HR-**: this is used for the 'l' sound
- **TKPW-**: this is used for the hard 'g' sound, as in 'gulp' and 'gulf'
- **PW-**: this is used for the 'b' sound
- **S*-**: this is used for the 'z' sound
- **SR-**: this is used for the 'v' sound

On the right side of the keyboard:

- **-BG**: this is used for the 'k' sound

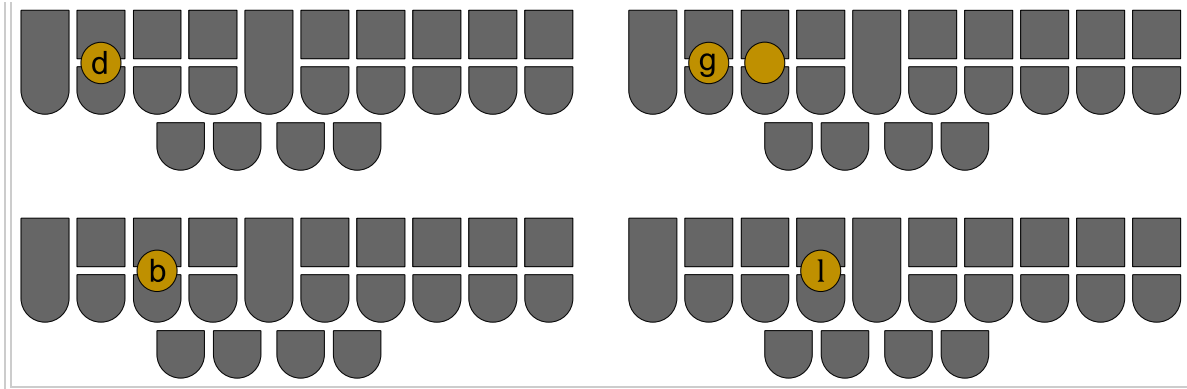
Note that there are no 'h' or 'w' sounds represented on the right side of the keyboard, even though there are keys for those sounds on the left. This is because English doesn't really have any 'h' or 'w' sounds at the end of syllables. Even words that seem to have a 'w' sound at the end, like 'how', can be expressed easily with just a vowel, as in **HOU**.

Fig. 1: The Missing Keys



Note that the 'd', 'g', 'b', and 'l' sounds each use single fingers to press two keys:

Fig. 2: Fingering For d, g, b, And l



Using a single finger to press two vertical keys is standard in Plover and in all steno systems. Never press two horizontal keys with the same finger, except for the right index finger pressing *F, *R or *FR, or the right little finger pressing TD or SZ. Or the thumbs pressing vowel keys.

Now you can type all of these sounds on both sides of the keyboard. For example, 'kiss' is stroked as **KEUS**, while 'sick' is stroked as **SEUBG**. And 'fed' is stroked as **TPED**, while 'deaf' is stroked as **TKEF**. Note that even though they use sounds represented by multiple keys, these words still fit perfectly in steno order:

- 'kiss': **KEUS**: STKPWHRAO*EUFRPBLGTSZ
- 'sick': **SEUBG**: STKPWHRAO*EUFRPBLGTSZ
- 'fed': **TPED**: STKPWHRAO*EUFRPBLGTSZ
- 'deaf': **TKEF**: STKPWHRAO*EUFRPBLGTSZ

Exercise: The Missing Keys

- 'due' 'duffer' 'deferral' 'devil' 'double' 'drug' 'depress' 'desire' 'dessert' 'destroyed'
- 'feral' 'ford' 'for' 'phrase' 'fierce' 'fable' 'feeble' 'sphere' 'fries'
- 'leader' 'lace' 'letter' 'lust' 'lovers'
- 'glad' 'glare' 'glides' 'give' 'get' 'group' 'guest' 'guide' 'gravel' 'cigarette' 'goblet'
- 'bored' 'board' 'bruise' 'buyer' 'bobble' 'brutal'
- 'zest' 'zap' 'zag'
- 'vile' 'vase' 'virus'
- 'eke' 'rockets' 'correct' 'quake' 'task'

Try the interactive drills on [The Missing Keys](#).

Exercise Solutions

Solution: The Missing Keys

- 'due': TKAOU
- 'duffer': TKUFR
- 'deferral': TKEFRL
- 'devil': TKEFL
- 'double': TKUBL
- 'debt': TKET
- 'dabble': TKABL
- 'drug': TKRUG
- 'depress': TKPRES
- 'desire': STKAOEUR
- 'dessert': STKERT
- 'destroyed': STKROEUD
- 'feral': TPERL
- 'ford': TPORD
- 'for': TPOR
- 'phrase': TPRAEUZ
- 'fierce': TPAOERS
- 'fable': TPAEUBL
- 'feeble': TPAOEBL
- 'sphere': STPAOER
- 'fries': TPRAOEUS
- 'leader': HRAOERD
- 'lace': HRAEUS
- 'letter': HRERT
- 'lust': HRUFT
- 'lovers': HROFRS
- 'leaders': HRAOERDZ
- 'glad': TKPWHRAD
- 'glare': TKPWHRAEUR
- 'glides': TKPWHRAOEUDZ
- 'give': TKPWEUF
- 'get': TKPWET
- 'group': TKPWRAOUP
- 'guest': TKPWEFT
- 'guide': TKPWAOEUD
- 'gravel': TKPWRAFL
- 'cigarette': STKPWRET
- 'goblet': TKPWBLT

- 'bored': PWORD
- 'board': PWAORD
- 'bruise': PWRAOUZ
- 'buyer': PWAOEUR
- 'bobble': PWOBL
- 'brutal': PWRAOULT
- 'zest': S*EFT
- 'zap': SA*P
- 'zag': SA*G
- 'vile': SRAOEUL
- 'vase': SRAEUS
- 'virus': SRAOEURS
- 'eke': AOEBG
- 'rockets': ROBGTS
- 'correct': KREBGT
- 'quake': KWAEUBG
- 'task': TAFBG

[Lesson 3: English Sounds >](#)

3C: n, m, j, And y

Contents

- 1 The Remaining Single Letters
 - 1.1 Exercise: The Remaining Single Letters
 - 1.2 Exercise: Including Sounds From Previous Lessons
- 2 Exercise Solutions
 - 2.1 Solution: The Remaining Single Letters
 - 2.2 Solution: Including Sounds From Previous Lessons

The Remaining Single Letters

There are four more consonant sounds that we typically express with just a single letter. The 'n', 'm', and 'j' sounds, and the consonant form of 'y'. The 'y' sound is only found at the start of a given word or syllable. If it appears later on, it's expressed as a vowel. For that reason, its Plover chord is only on the left side of the keyboard.

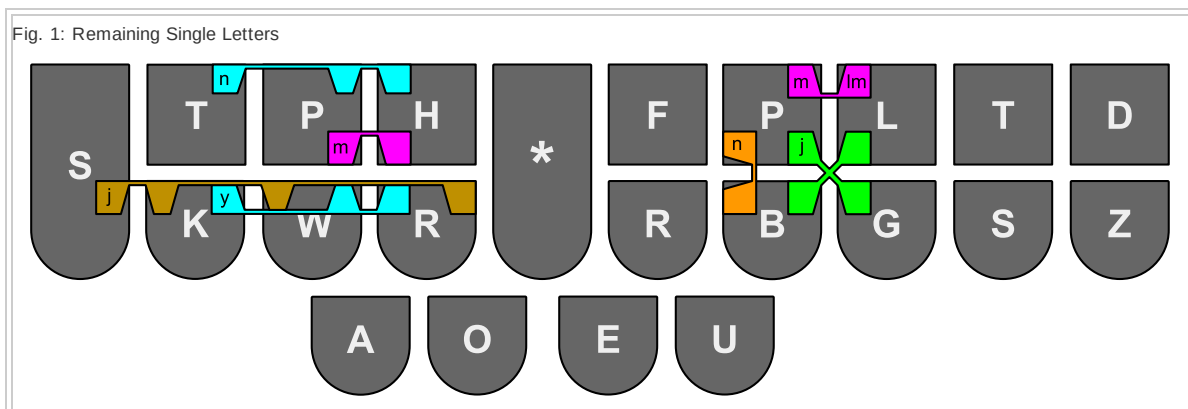
On the left side of the keyboard:

- **TPH-**: This is used for the 'n' sound.
- **PH-**: This is used for the 'm' sound.
- **SKWR-**: This is used for the 'j' sound.
- **KWR-**: This is used for the 'y' sound.

On the right side of the keyboard:

- **-PB**: This is used for the 'n' sound.
- **-PL**: This is used for two related sounds:
 - the 'm' sound.
 - the 'lm' sound as in 'realm', 'palm', and 'calm'.
- **-PBLG**: This is used for the 'j' sound.

Depending on which side of the word a sound falls, you would chord it using one or the other of the above forms. For example, 'man' is chorded **PHAPB**; 'hedge' is chorded **HEPBLG**; and 'joy' is chorded **SKWROEU**.



Exercise: The Remaining Single Letters

Try chording the following words:

- 'nag' 'nap' 'nab' 'nut' 'never' 'nestle' 'nod' 'nest' 'nerd'
- 'pent' 'parent' 'went' 'earns' 'rant' 'hunt' 'hand' 'panel' 'stun'
- 'must' 'muffle' 'maggot' 'mallet' 'smuggle' 'morals'
- 'arm' 'rum' 'harm' 'tempt' 'term'

- 'calmed' 'palm' 'qualms'
- 'jut' 'jug' 'just' 'jest' 'jets' 'job' 'jostle' 'jazz' 'jagged'
- 'urge' 'edge' 'average' 'purge' 'trudge' 'storage'
- 'yard' 'yet' 'yurt'

Try the interactive drills on ['The Remaining Missing Letters'](#).

Exercise: Including Sounds From Previous Lessons

These words may be a bit more difficult. They include all the sounds learned so far:

- 'noun' 'inhibit' 'nudge' 'notes' 'knack' 'enacts' 'neck' 'known' 'knock' 'gnome'
'noise' 'novice' 'named' 'neural' 'snide' 'announce'
- 'loin' 'donor' 'winner' 'dinner' 'learned' 'lend' 'allowance' 'flaunt' 'deference'
'different' 'dance' 'diner' 'demand' 'grunt' 'grant' 'gleans' 'severance' 'cement'
'design'
- 'mound' 'mourn' 'maim' 'matter' 'commit' 'commend' 'smudge' 'smuggle' 'semester'
- 'forms' 'primed' 'serum' 'time' 'hermit' 'maim' 'plumb' 'dream' 'gym' 'germ' 'jam'
'blame' 'bottom' 'grammar'
- 'balm' 'psalm'
- 'judge' 'journal' 'join' 'joyful' 'jam' 'gerunds'
- 'forge' 'budgets' 'average' 'leverage' 'merge' 'beige' 'carriage' 'fidget' 'frigid' 'digit'
'gadget' 'garage' 'grudge' 'turgid'
- 'year' 'yearn' 'yolk'

Try the interactive drills on ['Including Sounds From Previous Lessons'](#).

Exercise Solutions

Solution: The Remaining Single Letters

'n-':

- 'nag': TPHAG
- 'nap': TPHAP
- 'nab': TPHAB
- 'nut': TPHUT
- 'nub': TPHUB
- 'never': TPHEFR
- 'nestle': TPHEFL
- 'nod': TPHOD
- 'nest': TPHEFT
- 'nerd': TIPHERD

'-n':

- 'pent': PEPBT
- 'parent': PARPBT
- 'went': WEPBT
- 'earns': ERPBS
- 'rant': RAPBT
- 'hunt': HUPBT
- 'hand': HAPBD
- 'panel': PAPBL
- 'stun': STUPB

'm-':

- 'must': PHUFT
- 'muffle': PHUFL
- 'maggot': PHAGT
- 'mallet': PHALT
- 'smuggle': SPHULG
- 'morals': PHORLS

'-m':

- 'arm': ARPL
- 'rum': RUPL
- 'harm': HARPL
- 'tempt': TEPLT
- 'term': TERPL

'-lm'

- 'calmed': KAPLD
- 'palm': PAPL
- 'qualms': KWAPLS

'j-':

- 'jut': SKWRUT
- 'jug': SKWRUG
- 'just': SKWRUFT
- 'jest': SKWREFT
- 'jets': SKWRETS
- 'job': SKWROB
- 'jostle': SKWROFL
- 'jazz': SKWRAZ
- 'jagged': SKWRAGD

'-j':

- 'urge': URPBLG
- 'edge': EPBLG
- 'average': AFRPBLG
- 'purge': PURPBLG
- 'porridge': PORPBLG
- 'trudge': TRUPBLG
- 'storage': STORPBLG

'y-':

- 'yard': KWRARD
- 'yet': KWRET
- 'yurt': KWRURT

Solution: Including Sounds From Previous Lessons

'n-':

- 'noun': TPHOUPB
- 'inhibit': TPHEUBT
- 'nudge': TPHUPBLG
- 'notes': TPHOETS
- 'knack': TPHABG
- 'enacts': TPHABGTS
- 'neck': TPHEBG
- 'known': TPHOEPB
- 'knock': TPHOBG

- 'gnome': TPHOEPL
- 'noise': TPHOEUS
- 'novice': TPHOFS
- 'named': TPHAEUPLD
- 'neural': TPHAOURL
- 'snide': STPHAOEUD
- 'announce': TPHOUPBS

'-n':

- 'loin': HROEUPB
- 'donor': TKOERP
- 'winner': WEURPB
- 'dinner': TKEURPB
- 'learned': HRERPBD
- 'lend': HREPBD
- 'allowance': HROUPBS
- 'flaunt': TPHRAUPBT
- 'deference': TKEFRPBS
- 'different': TKEUFRPBT
- 'dance': TKAPBS
- 'diner': TKAOEURPB
- 'demand': TKPHAPBD
- 'grunt': TKPWRUPBT
- 'grant': TKPWRAPBT
- 'gleans': TKPWHRAOEPBS
- 'severance': SEFRPBS
- 'cement': SPHEPBT
- 'design': STKAOEUPB

'-m':

- 'mound': PHOUPBD
- 'mourn': PHOURPB
- 'maim': PHAEUPL
- 'matter': PHAEURT
- 'commit': KPHEUT
- 'commend': KPHEPBD
- 'smudge': SPHUPBLG
- 'smuggle': SPHULG
- 'semester': SPHEFRT

'-m':

- 'forms': TPORPLS
- 'primed': PRAOEUPLD

- 'rhymes': RAOEUPLS
- 'serum': SAOERPL
- 'time': TAOEUPL
- 'hermit': HERPLT
- 'maim': PHAEUPL
- 'plumb': PHRUPL
- 'dream': TKRAOEPL
- 'gym': SKWREUPL
- 'germ': SKWRERPL
- 'jam': SKWRAPL
- 'blame': PWHRAEUPL
- 'bottom': PWOPLT
- 'grammar': TKPWRARPL

'-lm'

- 'balm': PWAUPL
- 'psalm': SAUPL

'j-':

- 'judge': SKWRUPBLG
- 'journal': SKWRURPBL
- 'join': SKWROEUPB
- 'joyful': SKWROEUFL
- 'jam': SKWRAPL
- 'gerunds': SKWRERPBDZ

'-j':

- 'forge': TPORPBLG
- 'budgets': PWUPBLGTS
- 'average': AFRPBLG
- 'leverage': HREFRPBLG
- 'merge': PHERPBLG
- 'beige': PWAEUPBLG
- 'carriage': KAEURPBLG
- 'fidget': TPEUPBLGT
- 'frigid': TPREUPBLGD
- 'digit': TKEUPBLGT
- 'gadget': TKPWAPBLGT
- 'garage': TKPWRAPBLG
- 'grudge': TKPWRUPBLG
- 'turgid': TURPBLGD

'y-':

- 'year': KWRAOER
- 'yearn': KWREERP
- 'yolk': KWROEBG

[Lesson 3: English Sounds >](#)

Fingerspelling!

Contents

- 1 How To Spell Words Out
- 2 The Missing Letters

How To Spell Words Out

The previous lessons have introduced many different sounds, all of which correspond to letters of the alphabet. Plover mostly uses these to create chords for typing whole words. But you can also use them if you just want to spell a word out letter by letter. The trick is to only use the left side of the keyboard except for vowels, and add the * key to each chord.

For example, to spell out the word '**plover**', you would stroke **P***, then **HR***, then **O*** then **SR***, then ***E**, then **R***.

To capitalize any letter, add the **-P** key on the right side of the keyboard.

To spell out '**Plover**', you would stroke **P*P**, then **HR***, then **O***, then **SR***, then ***E**, then **R***. And to spell out '**VIP**', you would stroke **SR*P**, then ***EUP**, then **P*P**.

The Missing Letters

Some letters haven't been covered yet, because other phonemes include their sounds. For example, the letter "c" only makes the sounds of 's' and 'k', so it doesn't have its own regular chord.

Here is how to fingerspell the remaining few English letters:

- "c": **KR*** (the **KR** chord is also used to disambiguate homophonic words that are spelled with 'c'; for example, 'cite' is stroked **KRAOEUT** because 'sight' already has the **SAOEUT** stroke, and 'site' has the **SAO*EUT** stroke).
- "C": **KR*P**
- "q": **KW***
- "Q": **KW*P**
- "x": **KP***
- "X": **KP*P**

You might think the letter "z" would be fingerspelled **S***, since this is how the 'z' sound is stroked in full words. But the letter "s" already uses the **S*** stroke in fingerspelling, so we also need a chord for the letter "z".

- "z": **STKPW***
- "Z": **STKPW*P**

As you'll see, fingerspelling is very useful when creating your own dictionary entries. You may also use fingerspelling for proper names like '**Villa Villekulla**' and other things that you don't plan to use enough times to warrant their own dictionary entry. You should practice fingerspelling until proper names aren't too much of an inconvenience to type.

Try the interactive drills for [fingerspelling](#).

One can also use fingerspelling in conjunction with chording. For example, to do '**Brexit**' one could fingerspell '**B**', fingerspell '**r**', use **TK-LS** to remove a space, and then chord '**exit**'.

One could also add proper nouns or other words to the dictionary from within Plover or by editing a [JSON](#) file (see [Appendix: The Dictionary Format](#)). See also [this blog post](#) which describes a method to speed up adding words to the dictionary.

[Lesson 3: English Sounds >](#)

3D: th, ch, sh, And ng

Contents

1 Digraphs

[1.1 Exercise: Digraphs](#)

[1.2 Exercise: Including Sounds From Previous Lessons](#)

2 Solutions To Exercises

[2.1 Solution: Digraphs](#)

[2.2 Solution: Including Sounds From Previous Lessons](#)

Digraphs

Digraphs are consonants that combine in English to make a single distinct sound:

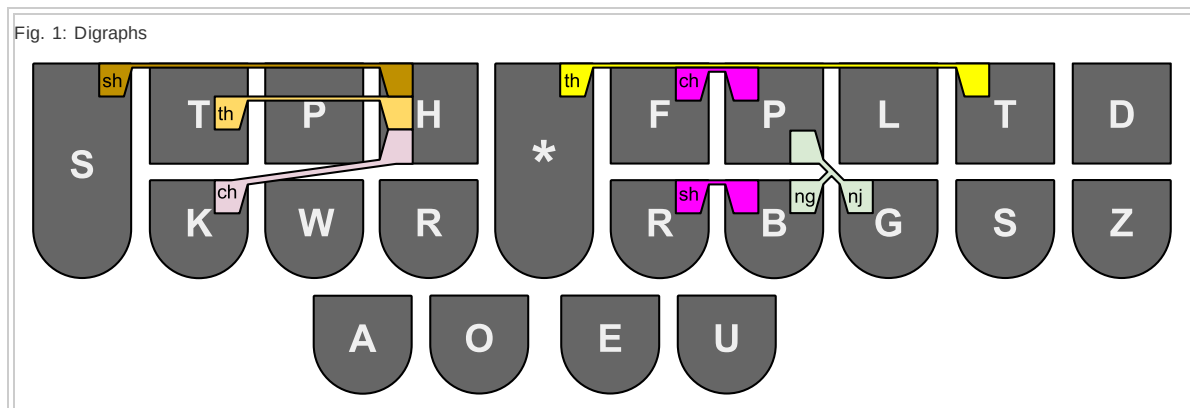
On the left side of the keyboard:

- **TH-**: This is used for the '**th**' sound.
- **KH-**: This is used for the '**ch**' sound.
- **SH-**: This is used for the '**sh**' sound.

On the right side of the keyboard:

- ***T**: This is used for the '**th**' sound.
- **-FP**: This is used for the '**ch**' sound.
- **-RB**: This is used for the '**sh**' sound.
- **-PBG**: This is used for the '**ng**' sound and the '**nj**' sound.

Notice that there is no way to produce the '**ng**' or '**nj**' sounds on the left side of the keyboard. This is because in English, virtually no words begin with those sounds. And any few that you may encounter are easily dealt with via briefs.



Exercise: Digraphs

Try chording the following words:

- 'thefts' 'thud' 'thus' 'thug'
- 'hath' 'earth' 'oath' 'health' 'wealth' 'worth' 'path' 'troth'
- 'chess' 'chest' 'chart' 'chat' 'chop' 'chore' 'chaff'
- 'touch' 'etch' 'ratchet' 'hutch' 'hatch' 'watch' 'patch' 'catch' 'crutch' 'such' 'sketch' 'stretch' 'retch'
- 'shell' 'shuffled' 'shall'
- 'ash' 'rush' 'rash' 'hush' 'hash' 'wash' 'push' 'posh' 'crush' 'crash' 'trash' 'squash' 'stash'
- 'anger' 'storing' 'rung' 'rang' 'prong' 'tongue' 'twang' 'song' 'stung' 'strong'
- 'sponge' 'orange'

Try the interactive drills on ['Digraphs'](#).

Exercise: Including Sounds From Previous Lessons

These words use all the sounds you've learned so far:

- 'thing' 'thence' 'them' 'thumb' 'thrill' 'throng' 'thrash'
- 'seethe' 'method' 'math' 'birth' 'breath' 'fifth' 'death' 'sleuth' 'blithe' 'growth'
- 'choose' 'chasm' 'chuck' 'check' 'churn' 'cherub' 'chin' 'channel' 'chant' 'chance' 'chive' 'charm'
- 'bleach' 'much' 'latch' 'leech' 'match' 'botch' 'fetch' 'ditch' 'glitch' 'vouch' 'slouch' 'smooch' 'splotch'
- 'shim' 'slime' 'shrewd' 'shrine' 'shuck' 'shark' 'shock' 'sheesh' 'shrivel' 'sugar'
- 'lash' 'mesh' 'mash' 'plush' 'bush' 'brush' 'fish' 'fresh' 'flush' 'flesh' 'flash' 'dash' 'delish' 'gosh' 'gash' 'shush' 'slash' 'smush' 'slosh' 'splash' 'squish' 'Irish'
- 'anger' 'finger' 'dung' 'lung' 'ping' 'pong' 'among' 'bring' 'young' 'fang' 'flung' 'gang' 'belong'
- 'change' 'range' 'hinge' 'lounge' 'plunge' 'cringe' 'tinge' 'fringe' 'derange' 'grunge' 'syringe'

Try the interactive drills on ['Including Sounds From Previous Lessons'](#).

Solutions To Exercises

Solution: Digraphs

'th-':

- 'thefts': THEFTS
- 'thud': THUD
- 'thus': THUS
- 'thug': THUG

'-th':

- 'hath': HA*T
- 'earth': *ERT
- 'oath': O*ET
- 'health': H*ELT
- 'wealth': W*ELT
- 'worth': WO*RT
- 'path': PA*T
- 'troth': TRO*T

'ch-':

- 'chess': KHES
- 'chest': KHEFT
- 'chart': KHART
- 'chat': KHAT
- 'chop': KHOP
- 'chore': KHOR
- 'chaff': KHAF

'-ch':

- 'touch': TUF~~P~~
- 'etch': EF~~P~~
- 'ratchet': RAF~~P~~T
- 'hutch': HUF~~P~~
- 'hatch': HAF~~P~~
- 'watch': WAF~~P~~
- 'patch': PAF~~P~~
- 'catch': KAF~~P~~
- 'crutch': KRUF~~P~~
- 'such': SUF~~P~~
- 'sketch': SKEF~~P~~
- 'stretch': STREF~~P~~

- 'retch': REFP

'sh-':

- 'shell': SHEL
- 'shuffled': SHUFLD
- 'shall': SHAL
- 'shore': SHOR
- 'shovels': SHOFLS

'-sh':

- 'ash': ARB
- 'rush': RURB
- 'rash': RARB
- 'hush': HURB
- 'hash': HARB
- 'wash': WARB
- 'push': PURB
- 'posh': PORB
- 'crush': KRURB
- 'crash': KRARB
- 'trash': TRARB
- 'squash': SKWARB
- 'stash': STARB

'-ng':

- 'anger': ARPBG
- 'storing': STORPBG
- 'rung': RUPBG
- 'rang': RAPBG
- 'prong': PROPBG
- 'tongue': TUPBG
- 'twang': TWAPBG
- 'song': SOPBG
- 'stung': STUPBG
- 'strong': STROPBG

'-nj'

- 'sponge': SPOPBG
- 'orange': ORPBG

Solution: Including Sounds From Previous Lessons

'th-':

- 'thing': THEUPBG
- 'thence': THEPBS
- 'them': THEPL
- 'thumb': THUPL
- 'thrill': THREUL
- 'throng': THROPBG
- 'thrash': THRARB

'-th':

- 'seethe': SAO*ET
- 'method': PH*ETD
- 'math': PHA*T
- 'birth': PW*EURT
- 'breath': PWR*ET
- 'fifth': TP*EUFT
- 'death': TK*ET
- 'sleuth': SHRAO*UT
- 'blithe': PWHRAO*EUT
- 'growth': TKPWHRO*ET

'ch-':

- 'choose': KHAOS
- 'chasm': KHAFPL
- 'chuck': KHUBG
- 'check': KHEBG
- 'churn': KHURPB
- 'cherub': KHERB
- 'chin': KHEUPB
- 'channel': KHAPBL
- 'chant': KHAPBT
- 'chance': KHAPBS
- 'chive': KHAEUF
- 'charm': KHARPL

'-ch':

- 'bleach': PWHRAOEFP
- 'much': PHUFP
- 'latch': HRAFP
- 'leech': HRAOEFP
- 'match': PHAFP
- 'botch': PWOFPP
- 'fetch': TPEFP
- 'ditch': TKEUFP

- 'glitch': TKPWHREUFP
- 'vouch': SROUFP
- 'slouch': SHROUFP
- 'smooch': SPHAOFP
- 'splotch': SPHROFP

'sh-':

- 'shim': SHEUPM
- 'slime': SHRAOEUPL
- 'shrewd': SHRAOUD
- 'shrine': SHRAOEUPB
- 'shush': SHURB
- 'shuck': SHUBG
- 'shark': SHARBG
- 'shock': SHOBG
- 'sheesh': SHAOERB
- 'shrivel': SHREUFL
- 'sugar': SHAOURG

'-sh':

- 'lash': HRARB
- 'mesh': PHERB
- 'mash': PHARB
- 'plush': PHRURB
- 'bush': PWURB
- 'brush': PWRURB
- 'blush': PWHRURB
- 'fish': TPEURB
- 'fresh': TPRERB
- 'flush': TPHRURB
- 'flesh': TPHRERB
- 'flash': TPHRARB
- 'dash': TKARB
- 'delish': TKHREURB
- 'gosh': TKPWORB
- 'gash': TKPWARB
- 'shush': SHURB
- 'slash': SHRARB
- 'smush': SPHURB
- 'slosh': SHRORB
- 'splash': SPHRARB
- 'squish': SKWEURB
- 'Irish': AOEURB

'-ng':

- **'anger': ARPBG**
- **'finger': TPEURPBG**
- **'dung': TKUPBG**
- **'lung': HRUPBG**
- **'ping': PEUPBG**
- **'pong': POPBG**
- **'among': PHOPBG**
- **'bring': PWREUPBG**
- **'young': KWRUPBG**
- **'fang': TPAPBG**
- **'flung': TPHRUPBG**
- **'gang': TKPWAPBG**
- **'belong': PWHROPBG**

'-nj'

- **'change': KHAEUPBG**
- **'range': RAEUPBG**
- **'hinge': HEUPBG**
- **'lounge': HROUPBG**
- **'plunge': PHRUPBG**
- **'cringe': KREUPBG**
- **'tinge': TEUPBG**
- **'fringe': TPREUPBG**
- **'derange': TKRAEUPBG**
- **'grunge': TKPWRUPBG**
- **'syringe': SREUPBG**

[Lesson 3: English Sounds >](#)

3E: Compound Clusters

Contents

- 1 Beyond The Phoneme**
- 2 Common Compound Clusters**
 - 2.1 Exercise: Common Compound Clusters**
 - 2.2 Considerations**
 - 2.3 Exercise: Including Sounds From Previous Lessons**
 - 2.4 Considerations**
- 3 Exercise Solutions**
 - 3.1 Solution: Common Compound Clusters**
 - 3.2 Solution: Including Sounds From Previous Lessons**

Beyond The Phoneme

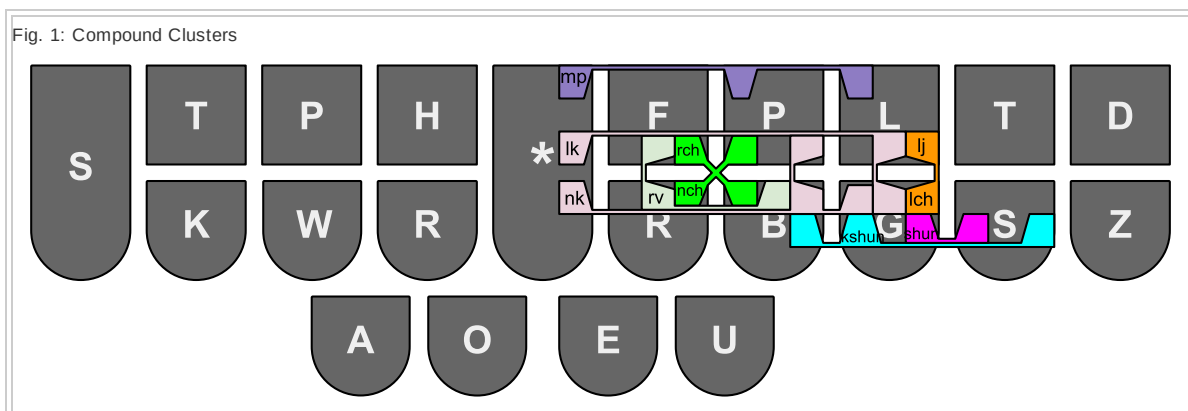
Previous lessons have identified keyboard chords that Plover assigns to particular phonemes. Using those chords, you can sound out all English words, even if it takes more than one stroke to do so.

Inevitably, some English words simply resist steno order because of the way they group sounds; for example, the '**mp**' sound in '**bump**', or the '**rv**' sound in '**swerve**'. To provide single stroke dictionary entries for many of these words, Plover defines a set of chords specifically to deal with those groupings.

Common Compound Clusters

These are the most common compound clusters defined in Plover:

- ***PL**: the 'mp' cluster, as in 'bump', 'camp' and 'limp'
- **-FRB**: the 'rv' cluster, as in 'carve', 'swerve', and 'nerve'
- **-LG**: this could be used for either of two clusters:
 -
 - 'lch', as in 'mulch', 'gulch', and 'squelch'
 - 'lj', as in 'divulge', 'bulge', and 'bilge'
- ***LG**: the 'lk' cluster, as in 'elk', 'ilk', 'hulk', and 'milk'
- ***PBG**: the 'nk' cluster, as in 'mink', 'honk', and 'prank'
- **-GS**: the 'shun' cluster, as in 'lesion', 'ocean', 'ration', and 'cushion'
- **-BGS**: the 'kshun' cluster, as in 'direction', 'selection', and 'seduction'
- **-FRPB**: this could be used for either of two clusters:
 -
 - 'rch', as in 'march', 'lurch', 'perch', and 'search'
 - 'nch', as in 'finch', 'clench', 'branch', and 'hunch'



Exercise: Common Compound Clusters

For this exercise, first try to figure out how you might stroke each word *without* using the chords defined in this lesson. Then stroke the words using the chords defined here. It should become clear why these chords are so useful.

- 'hemp' 'trump' 'rump' 'romp' 'ramp' 'pump' 'camp' 'cramp' 'tamp' 'pomp'
- 'curve' 'carve' 'serve' 'swerve' 'starve'
- 'squelch'
- 'hulk' 'calc' 'sulk' 'talc'
- 'rank' 'honk' 'wonk' 'prank' 'crank' 'tank'
- 'session' 'option' 'ration' 'portion' 'passion' 'cushion' 'caption' 'suppression'
- 'section' 'correction' 'suction'
- 'arch' 'ranch' 'hunch' 'porch' 'crunch' 'quench' 'torch' 'trench' 'stench' 'starch'

Try the interactive drills on '[Common Compound Clusters](#)'.

Considerations

You may have had trouble with '**suppression**' and '**correction**' if you didn't remember to drop the unstressed vowels.

Did you try to figure out how to stroke each word without its special chord? Did you notice that each chord solved a particular problem that otherwise had no obvious solution?

Exercise: Including Sounds From Previous Lessons

- 'limp' 'blimp' 'chomp' 'clamp' 'damp' 'slump' 'shrimp' 'jump'
- 'nerve' 'verve' 'marvel'
- 'village' 'mulch' 'bulge' 'belch' 'bilge' 'gulch' 'pillage'
- 'ilk' 'milk' 'bulk' 'silk' 'bilk'
- 'wink' 'mink' 'plank' 'brink' 'blink' 'blank' 'flank' 'flunk' 'dank' 'drink' 'gunk' 'junk' 'link' 'chunk'
- 'lesion' 'provision' 'fusion' 'lotion' 'operation' 'mission' 'motion' 'pollution'
- 'election' 'auction' 'correction' 'collection' 'fraction' 'friction' 'depiction' 'selection' 'seduction'
- 'finch' 'clench' 'branch' 'march' 'lurch' 'lynch' 'birch' 'brunch' 'church' 'drench'

Try the interactive drills on '[Including Sounds From Previous Lessons](#)'.

Considerations

Did you notice that '**village**' and '**pillage**' each had to drop an unstressed vowel in order to bring the 'l' and 'j' together to form the 'lj' sound? And did you notice the inversion rule used in '**provision**'?

Exercise Solutions

Solution: Common Compound Clusters

'mp'

- 'hemp': H*EPL (otherwise HEPL/-P)
- 'trump': TR*UPL (otherwise TRUPL/-P)
- 'rump': R*UPL (otherwise RUPL/-P)
- 'romp': RO*PL (otherwise ROPL/-P)
- 'ramp': RA*PL (otherwise RAPL/-P)
- 'pump': P*UPL (otherwise PUPL/-P)
- 'camp': KA*PL (otherwise KAPL/-P)
- 'cramp': KRA*PL (otherwise KRAPL/-P)
- 'tamp': TA*PL (otherwise TAPL/-P)
- 'pomp': PO*PL (otherwise POPL/-P)

'rv'

- 'curve': KUFRB (otherwise KUR/-F)
- 'carve': KAFRB (otherwise KAR/-F)
- 'serve': SEFRB (otherwise SER/-F)
- 'swerve': SWEFRB (otherwise SWER/-F)
- 'starve': STAFRB (otherwise STAR/-F)

'lch', 'lj'

- 'squelch': SKWELG

'lk'

- 'hulk': H*ULG (otherwise HUL/-BG)
- 'calc': KA*LG
- 'sulk': S*ULG (otherwise SUL/-BG)
- 'talc': TA*LG (otherwise TAL/-BG)

'nk'

- 'rank': RA*PBG (otherwise RAPB/-BG)
- 'honk': HO*PBG (otherwise HOPB/-BG)
- 'wonk': WO*PBG
- 'prank': PRA*PBG (otherwise PRAPB/-BG)
- 'crank': KRA*PBG (otherwise KRAPB/-BG)
- 'tank': TA*PBG (otherwise TAPB/-BG)

'shun'

- 'session': SEGS (otherwise SERB/OPB)
- 'option': OPGS
- 'ration': RAGS
- 'portion': PORGS
- 'passion': PAGS
- 'cushion': KUGS
- 'caption': KAPGS
- 'suppression': SPREGS

'kshun'

- 'section': SEBGS
- 'correction': KREBGS
- 'suction': SUBGS (otherwise SUBG/SHUPB)

'rch', 'nch'

- 'arch': AFRPB
- 'ranch': RAFRPB
- 'hunch': HUFRPB
- 'porch': POFRPB
- 'crunch': KRUFRPB
- 'quench': KWEFRPB
- 'torch': TOFRPB
- 'trench': TREFRPB
- 'stench': STEFRPB
- 'starch': STAFRPB

Solution: Including Sounds From Previous Lessons

'mp'

- 'limp': HR*EUPL (otherwise HREUPL/-P)
- 'blimp': PWHR*EUPL (otherwise PWHREUPL/-P)
- 'chomp': KHO*PL
- 'clamp': KHRA*PL (otherwise KHRAPL/-P)
- 'damp': TKA*PL (otherwise TKAPL/-P)
- 'slump': SHR*UPL (otherwise SHRUPL/-P)
- 'shrimp': SHR*EUPL (otherwise SHREUPL/-P)
- 'jump': SKWR*UPL (otherwise SKRWUPL/-P)

'rv'

- 'nerve': TPHEFRB (otherwise TIPHER/*F)
- 'verve': SREFRB
- 'marvel': PHAFRBL (otherwise PHAR/SREL)

'lch', 'lj'

- **'village': SREULG** (otherwise **SREUL/APBLG**)
- **'mulch': PHULG**
- **'bulge': PWULG** (otherwise **PWUL/-PBLG**)
- **'belch': PWELG**
- **'bilge': PWEULG** (otherwise **PWEUL/-PBLG**)
- **'gulch': TKPWULG**
- **'pillage': PEULG** (otherwise **PEUL/APBLG**)

'lk'

- **'ilk': *EULG** (otherwise **EUL/-BG**)
- **'milk': PH*EULG** (otherwise **PHEUL/-BG**)
- **'bulk': PW*ULG** (otherwise **PWUL/-BG**)
- **'silk': S*EULG** (otherwise **SEUL/-BG**)
- **'bilk': PW*EULG** (otherwise **PWEUL/-BG**)

'nk'

- **'wink': W*EUPBG**
- **'mink': PH*EUPBG**
- **'plank': PHRA*PBG**
- **'brink': PWR*EUPBG**
- **'blink': PWHR*EUPBG**
- **'blank': PWHRA*PBG**
- **'flank': TPHRA*PBG**
- **'flunk': TPHR*UPBG**
- **'dank': TKA*PBG**
- **'drink': TKR*EUPBG**
- **'gunk': TKPW*UPBG**
- **'junk': SKWR*UPBG**
- **'link': HR*EUPBG**
- **'chunk': KH*UPBG**

'shun'

- **'lesion': HRAOEGS** (otherwise **HRAOR/SHUPB**)
- **'provision': PREUFGS**
- **'fusion': TPAOUGS**
- **'lotion': HROEGS**
- **'operation': PRAEUGS**
- **'mission': PHEUGS**
- **'motion': PHOEGS**
- **'pollution': PHRAOUGS**

'kshun'

- 'election': HREBGS
- 'auction': AUBGS
- 'correction': KREBGS
- 'collection': KHREBGS
- 'fraction': TPRABGS
- 'friction': TPREUBGS
- 'depiction': TKPEUBGS
- 'selection': SHREBGS
- 'seduction': STKUBGS

'rch', 'nch'

- 'finch': TPEUFRPB
- 'clench': KHREFRPB
- 'branch': PWRAFRPB
- 'march': PHAFRPB
- 'lurch': HRUFRPB
- 'lynch': HREUFRPB
- 'birch': PWEUFRPB
- 'brunch': PWRUFRPB
- 'church': KHUFRPB
- 'drench': TKREFRPB

Lesson 4: Common Briefs

Contents

- 1 What Is A Brief?
- 2 Brief Culture
- 3 Stroke Abundance
- 4 Brief Solitude
- 5 100 Common Briefs
- 6 Studying Briefs

What Is A Brief?

A brief is any stroke that doesn't phonetically match its target word. For example, **'the'** is stroked **-T**, and **'of'** is stroked **F**. Unlike phonetic strokes, briefs must be memorized because they don't follow the predictable patterns of Plover theory. But most of the best briefs try to be memorable in some way or another. Very rarely is a brief entirely distinct from the word it represents.

Why make a brief? Memorization is work, and to some extent arbitrary. And lots of words that have brief forms in the dictionary have phonetic forms as well. If it's possible to stroke a word without knowing any more than Plover's general phonetic rules, then why bother having a brief? What's the value of it?

There are a few answers to this. One reason is that the vast majority of briefs let you type a word in a single stroke that would otherwise require two or more. For example, the brief stroke for **'development'** is **SREPLT**.

Another reason why briefs are good is that many of them exist in order to disambiguate homophones. For example, **'site'** is stroked **SAO*EUT** to disambiguate it from **'sight'**, which uses the phonetic **SAOEUT** stroke. And **'cite'** is stroked **KRAOEUT** to disambiguate it from both of those.

Some words occur so frequently that even though a single stroke phonetic form may exist, the word is given a brief that uses even fewer keys. For example, **'the'** could have used the single stroke **THE**. But it appears so often in English (one out of every 14 words, according to Peter Norvig), that Plover gives it the even easier stroke **-T**. And the word **'they'** could have used the single stroke **THAEU**, but it also appears so often (one out of every 303 words) that Plover gives it the stroke **THE**, which as we've seen has been conveniently vacated by **'the'**. As you'll discover when you start creating your own briefs, these very short strokes are a hot commodity, because they are so easy (and therefore faster) to type.

Brief Culture

Professional stenographers often find themselves transcribing testimony or reportage about highly specialized topics, and so there has emerged a tradition of constructing new briefs on the fly, to accommodate those changing contexts. For example, someone transcribing medical language might use **AOEG** as a brief for '**electroencephalogram**', which would otherwise require six or seven strokes to type. Then when they're done with the job, the professional stores the brief in a special medical dictionary, for the next time they're hired to do medical transcription.

The culture of brief-making is something to be embraced. As you get more fluent with Plover you'll undoubtedly discover that you type some words more than others, and you'll wish you could type those words faster. Maybe it's time for a brief! That's a very important impulse in stenographic typing. Briefs increase typing speed, help avoid frustration, and can be *fun*! [Lesson 9: Designing Briefs](#) presents guidelines for constructing more memorable briefs.

Stroke Abundance

Many words have multiple entries in Plover's default dictionary. Consider all the ways you might stroke **'relationship'**:

- **R*EUP**
- **RAOE/HRAEUBGS/SHEUP**
- **RAOE/HRAEUGS/SHEUP**
- **RAOELGS/SHEUP**
- **RAOELT/SHEUP**
- **RE/HRAEUGS/SH-P**
- **RE/HRAEUGS/SHEUP**
- **REL/SH-P**
- **RELG/SH-P**
- **RELGS/-S/SH-P**
- **RELGS/SHEUP**

There are several reasons why the dictionary file might have so many entries for a single word. If there is more than one way to pronounce the word, or more than one way to envision its syllabification, you might want a phonetic entry for each such way, as in the case of **RAOE/HRAEUGS/SHEUP** and **RE/HRAEUGS/SHEUP**.

Another reason for more dictionary entries is to auto-correct your misstrokes. Misstrokes are very interesting. They occur when you accidentally press the wrong key during a stroke. For example, **RAOE/HRAEUBGS/SHEUP** uses the **BGS** chord for the **'kshun'** sound, instead of just **BS** for **'shun'**. The word is supposed to be **'relationship'**, but the misstroke makes it sound more like **'relayktionship'**. To correct this, the Plover dictionary file includes the incorrect **RAOE/HRAEUBGS/SHEUP** stroke as another entry for **'relationship'**. It's not phonetically correct, and doesn't even make a great brief, but it does mean that if you accidentally hit the wrong key, Plover will correct the mistake automatically and transparently.

Briefs are another reason for multiple dictionary entries. Learning Plover's general phonetic theory allows you to get started with Plover, without having to commit to massive amounts of memorization. But as you start typing faster and faster, you'll discover certain multi-stroke words that you type over and over again, and you may finally want to learn the brief form of those words. Or if you discover that no brief exists, or if you don't find the existing brief forms memorable, you may want to invent one for yourself, using the techniques described in [Lesson 9: Designing Briefs](#).

Brief Solitude

In some cases you'll find the very opposite of stroke abundance - a word that has only one entry in the dictionary file, and where that entry is a brief form. For example, 'if' exists only as the brief stroke **TP**.

Unlike '**relationship**', which can be stroked phonetically without memorization, there is no gradual approach available for words that exist only in brief form. In order to type them, you need to memorize them as briefs from the very beginning. And, as you'll see in the current lesson, many of the most commonly used words in English fall into this category. That's why up until now the exercises have focused on practice words instead of practice sentences. It's very difficult to type *any* complete sentence without using a brief.

100 Common Briefs

Peter Norvig has published [various linguistic statistics](#), and he includes a link to his [list of the most-used words in English](#).

Among the top thousand or so entries, I culled the following list. Most of them are examples of 'brief solitude', but some are not. Here they are, roughly in order of frequency of use. Notice that common words are not included here if they have phonetic entries in the dictionary. For example, the word 'and' can be stroked phonetically as **APBD** (though you should really use the much easier-to-stroke brief form **SKP**).

'the': -T	'you're': UR	'very': SRE	'somebody': S-B	'describe': SKREU
'of': F	'you've': UF	'many': PHAEPB	'somehow': SPHOU	'described': SKREUD
'to': TO	'were': WR	'even': AOEPB	'someone': SPH-PB	'anything': TPHEUG
'in': TPH	'can': K	'where': W-R	'someplace': SPHRAEUS	'continue': T-PB
'a': AEU	'there': THR	'through': THRU	'something': S-G	'continued': T-PBD
'is': S-	'been': PWPB	'being': -BG	'sometimes': STAOEUPLS	'beginning': TKPWEUPBG
'that': THA	'if': TP	'because': PWAUS	'somewhere': SWR or SW-R	'understand': URPBD
'with': W	'would': WO	'before': PW-F	'question': KWE	'understanding': URPBGD
'be': B	'who': WHO	'upon': POPB	'almost': HR-PL	'today': TOED
'by': PWEU	'other': OER	'without': WOUT	'interest': T-R or TR	'opinion': P-PB
'he': E	'what': WHA	'another': TPHOER	'ever': -FR	'becomes': PW-BGS
'I': EU	'only': OEPBL	'against': TKPWEPPBS	'became': PWAEUPL	'yes': KWRE
'this': TH	'do': TKO	'every': EFR	'probably': PROBL	'idea': KWR-D
'are': R-	'new': TPHU	'within': W-PB	'include': KHRU	'ideas': KWR-DZ
'which': WEU	'about': PW	'example': KP-PL	'includes': KHRUS	'actually': TWAEUL
'have': SR	'two': TWO	'others': OERS	'included': KHRUD	'move': PHOF
'they': THE	'any': TPHEU	'therefore': THR-FR	'including': KHRUG	'ask': SK
'you': U	'could': KO	'having': SR-G	'amount': APLT	'unless': TPH-LS
'you'd': UD	'after': AF	'become': PW-BG	'receive': SAOEF	'easy': EZ
'you'll': UL	'said': SAEUD	'whether': WHR	'received': SAOEFD	'otherwise': O*ERZ

Don't be alarmed by the quantity. Just think of the current lesson as a reference, that you can return to as needed. Here's the list again, in alphabetical order:

'a': AEU	'being': -BG	'in': TPH	'received': SAOEFD	'understanding': URPBGD
'about': PW	'by': PWEU	'include': KHRU	'said': SAEUD	'unless': TPH-LS
'actually': TWAEUL	'can': K	'included': KHRUD	'somebody': S-B	'upon': POPB
'after': AF	'continue': T-PB	'including': KHRUG	'somehow': SPHOU	'very': SRE
'against': TKPWEPPBS	'continued': T-PBD	'includes': KHRUS	'someone': SPH-PB	'were': WR
'almost': HR-PL	'could': KO	'interest': T-R or TR	'someplace': SPHRAEUS	'what': WHA
'amount': APLT	'describe': SKREU	'I': EU	'something': S-G	'where': W-R
'another': TPHOER	'described': SKREUD	'is': S-	'sometimes': STAOEUPLS	'whether': WHR
'any': TPHEU	'do': TKO	'many': PHAEPB	'somewhere': SWR or SW-R	'which': WEU
'anything': TPHEUG	'easy': EZ	'move': PHOF	'that': THA	'who': WHO
'are': R-	'even': AOEPB	'new': TPHU	'there': THR	'with': W
'ask': SK	'ever': -FR	'of': F	'therefore': THR-FR	'within': W-PB
'be': B	'every': EFR	'only': OEPBL	'the': -T	'without': WOUT
'became': PWAEUPL	'example': KP-PL	'opinion': P-PB	'they': THE	'would': WO
'because': PWAUS	'have': SR	'other': OER	'this': TH	'yes': KWRE
'become': PW-BG	'having': SR-G	'others': OERS	'through': THRU	'you': U
'becomes': PW-BGS	'he': E	'otherwise': O*ERZ	'to': TO	'you'd': UD
'been': PWPB	'idea': KWR-D	'probably': PROBL	'today': TOED	'you'll': UL
'before': PW-F	'ideas': KWR-DZ	'question': KWE	'two': TWO	'you're': UR
'beginning': TKPWEUPBG	'if': TP	'receive': SAOEF	'understand': URPBD	'you've': UF

Try the interactive drills for briefs [1-20](#), [21-40](#), [41-60](#), [61-80](#), and [81-100](#).

Studying Briefs

Some briefs, like many of the above, are a prerequisite to using Plover in a productive way. If you try to stroke a word phonetically, only to discover that the stroke doesn't work, it's possible you've discovered another case of 'brief solitude'. Don't shy away from these. Instead, search for the word in your dictionary file. If you find it only in brief form, write it down someplace handy, and practice it.

It's possible you may discover that the word you're looking for has no entry at all; or that it *does* exist in the dictionary in phonetic form, but that the phonetic form doesn't agree with your intuition of what it should be. In that case, the proper thing to do is often to make a new dictionary entry that matches your intuition.

Lesson 5: Prefix And Suffix Strokes

Contents

- 1 Multi-Stroke Words
- 2 What Are Prefix And Suffix Strokes?
- 3 When To Use Them
- 4 Some Starter Strokes
 - 4.1 Prefixes
 - 4.2 Suffixes
- 5 Looking Up Prefix And Suffix Strokes

Multi-Stroke Words

Hey! You've been typing words as single strokes, now you can learn how to do it with several! Here's how. In fact, a lot of words in Plover's dictionary file use more than one stroke to type. So, you'll press a set of keys for the beginning sounds of a word, then release them, and press another set of keys for the next sounds in the word, and so on. Once you've pressed the correct series of strokes for all the sounds of the word, out pops the desired word.

Everything you've learned for single-stroke words applies to multi-stroke words as well. The phoneme use the same chords; you can drop unstressed vowels, do inversions, and so on. So there is no new primary knowledge to learn for multi-stroke words. However, there are certain problems that can crop up just by virtue of the stroking process itself; and the solutions to these problems are covered in this and related sections. For starters, you really need to learn about prefix and suffix strokes.

What Are Prefix And Suffix Strokes?

Plover supports multi-stroke words, which are very important, but they do introduce the possibility of multi-stroke homophonic conflicts, as described in [Appendix: Word Boundaries](#).

Prefix and suffix strokes are strokes that typically sound similar to the start or end of the word they are part of, but that are constructed in such a way that they would never be found at the other end of a word. So, a prefix stroke would never correspond to the final stroke of any word in the dictionary file, and a suffix stroke would never correspond to the starting stroke of any word in the dictionary file. This allows Plover to know for certain that you intend a given stroke to be really and truly the *start* or *end* of a word, so Plover doesn't have to use its internal logic to make that choice for you.

Prefix and suffix strokes most often correspond to parts of words that are generally considered prefixes and suffixes in the English language. Word-parts like '**em-**', '**en-**', '**re-**', '**-ed**', '**-ing**', and '**-ment**'. But Plover isn't a grammarian - any word-part that's used a lot at the start or end of a word may have a prefix or suffix stroke associated with it, even if it isn't technically a prefix or suffix in English. You can also define your own prefix and suffix strokes, just the way you can define any other Plover dictionary entry.

Consider the '**-y**' suffix stroke **KWREU**. As you can see, it's an odd little key cluster that *in*exactly resembles the associated English suffix. But by using it, Plover is able to unfailingly recognize **PUP/KWREU** as the word '**puppy**', regardless of any surrounding strokes.

Or consider the '**over-**' prefix stroke **AUFR**. It doesn't sound phonetically quite right either, but with it, Plover is able to recognize **AUFR/KOPL** as the word '**overcome**'. If you had used the more phonetic strokes **OEFR/KOPL**, Plover would have output the two words '**over come**'.

Many suffixes are distinguished from word parts by adding the *****; the word '**on**' is **OPB**, while the '**-on**' suffix as used in a word like '**mutton**', is **O*PB**. Many prefixes are distinguished from word parts by altering the vowels; the '**on-**' prefix as used in a word like '**onramp**', is **AUPB**. But these are not universal rules. As you learn more prefix and suffix strokes, you may recognize more of the patterns behind them. If you're ever unsure about how to make a prefix or suffix, consult the dictionary file.

When To Use Them

You should use prefix and suffix strokes whenever they're available. But there are a great many prefix and suffix strokes defined in the dictionary file. You don't need to learn all of them right away. Here is a good rule of thumb:

Whenever you encounter a word boundary error, learn the prefix or suffix stroke that would have avoided that error.

That's probably the easiest way to proceed. If you follow that guideline, you'll learn the prefix and suffix strokes that match your own writing style and vocabulary. Over time, you'll find yourself less and less often needing to look up a new prefix or suffix stroke.

It can be inconvenient to stop what you're doing and learn a new stroke. [Lesson 6: Prefix/Suffix Alternatives](#) describes ways to avoid that interruption. But you should regard those techniques as the wrong way to type with Plover. You'll never type fast that way. Use those techniques as an expedient when you don't know the right prefix or suffix stroke for a situation, but always go back and learn the proper prefix or suffix stroke afterwards.

Some Starter Strokes

For the current lesson, I googled a list of [common prefixes and suffixes](#). According to the page, they account for 97% of all cases. These seem like a good starting point for learning Plover:

Prefixes

- 'un-': UPB
- 're-': RE
- 'in-': EUPB
- 'im-': EUPL
- 'ir-': EUR
- 'dis-': TKEUS
- 'en-': EPB
- 'em-': EPL
- 'non-': TPHAUPB
- 'over-': AUFR
- 'mis-': PHEUZ
- 'sub-': SAUB
- 'pre-': PRE
- 'inter-': EUPBT
- 'fore-': TPOER
- 'de-': TKE
- 'trans-': TRAPBS
- 'super-': SAOUP
- 'semi-': SPHEU
- 'anti-': A*EUPBT
- 'mid-': PHEUD
- 'under-': UPBD

Some examples:

- 'underwear': UPBD/WAER
- 'misdeed': PHEUZ/TKAOED
- 'embalm': EPL/PWAUPL
- 'disease': TKEUZ/AOEZ
- 'retreat': RE/TRAOET

Suffixes

- '-s': -S
- '-ed': D
- '-ing': G
- '-ly': HREU

- '-er': *ER
- '-or': O*R
- '-tion': SH*UPB
- '-ation': AEUGS
- '-ion': KWROPB
- '-able': BL
- '-al': A*L
- '-ial': KWRAL
- '-ness': -PBS

Some examples:

- 'happily': HAP/HREU
- 'cider': SAOEUD/*ER
- 'relation': REL/AEUGS

Looking Up Prefix And Suffix Strokes

To learn new prefix and suffix strokes, you'll need to look them up in your dictionary file. Try to make this as easy a process as you can.

- Learn the location of your dictionary file on disk.
- Learn the format of dictionary entries, so you can read and understand them.

If you're on a UNIX system, a simple `grep -E` command will display all prefix and suffix entries for you.

Prefix strokes:

```
$ grep -E "\{\w\w*\^\}" dict.json
"*EFR": "{every^}",
"*ERBGS": "{extra^}",
"*EURL": "{im^}",
"*FR": "{ever^}",
"A": "{a^}",
"A*EPBT": "{ante^}",
"A*ER": "{aero^}",
"A*F": "{after^}",
"A*UT": "{auto^}",
"A*UT/SKWR0": "{auto^}",
...
```

Suffix strokes:

```
$ grep -E "\{^\w\w*\}" dict.json
"*BG": "{^k}",
"*EB": "{^en}",
"*EFBG": "{^esque}",
"*EPB": "{^en}",
"*EPB/TPELD": "{^enfeld}",
"*ER": "{^er}",
"*ERS": "{^ers}",
"*ES": "{^ess}",
"*ET": "{^eth}",
"*EUFPL": "{^ism}",
...
```

As you can see, there are many, many prefix and suffix strokes in the dictionary file. Don't try to learn them all at once, just make sure you know how to find them when you need them.

Lesson 6: Prefix/Suffix Alternatives

Contents

- 1 When Not To Use Prefix And Suffix Strokes
- 2 Fingerspelling
- 3 Manually Inserting Spaces
- 4 Avoiding Spaces With The "No Space" Stroke

When Not To Use Prefix And Suffix Strokes

It's a trick question - you should really *always* use [prefix and suffix strokes](#) when they'd help type a word. The techniques described in this lesson are not true alternatives. Instead they should be your way of recognizing an opportunity to learn a new prefix or suffix stroke. If you find yourself doing *any* of the techniques described in this lesson as a way of getting around a word boundary error, view it as identifying a bit of skill that you need to work on; and make a note of the particular word in question, so you can look up the appropriate prefix or suffix stroke when you get a chance.

Fingerspelling

A guaranteed way to get the word you want - though very slow - is to spell a word out by hand. Fingerspelling is an important skill in Plover, and you should really master it. If you've gotten this far in the tutorials without taking the time to get good at fingerspelling, you should [go back and get good at it now](#). There are plenty of names and other special words that legitimately call for fingerspelling. Here are some examples of fingerspelling to avoid prefix and suffix strokes:

- 'puppy': P* *U P* P* KWR*
- 'overcome': O* SR* *E R* KR* O* PH* *E
- 'fire place': TP* *EU R* *E S-P P* HR* A* KR* *E
- 'light engage': HR* EU* TKPW* H* T* S-P *E TPH* TKPW* A* TKPW* *E

Manually Inserting Spaces

It you want to type a phrase like **'fire place'**, but Plover insists on producing the single word **'fireplace'**, there is a relatively straightforward solution. In the fingerspelling example above, note the **S-P** stroke, to add a space between the words **'fire'** and **'place'**. The same technique forces Plover to insert a space between two words that you strike whole. So the phrase **'fire place'** can be stroked **TPAOEUR S-P PHRAEUS**.

If you realize the problem only after you've stroked **TPAOEUR/PHRAEUS** and gotten **'fireplace'**, use the ***** key to undo the most recent stroke. This leaves you with **'fire'**. Then stroke **S-P PHRAEUS** to end up with **'fire place'**.

The same technique works with **'light engage'**, which can be stroked **HRAOEUT S-P EPB/TKPWAEUPBLG**.

Avoiding Spaces With The "No Space" Stroke

If you want to type the compound word '**overcome**', but Plover insists on producing the phrase '**over come**', there is a relatively straightforward solution. To prevent a space being added between words, use the **TK-LS** stroke. To type '**overcome**', use the stroke **OEFR TK-LS KOPL**. This works because both '**over**' and '**come**' are distinct words in the Plover dictionary.

Words like '**puppy**' are a different case because they are not compound words. Even though '**pup**' is a whole word that may be stroked on its own, there is no stroke for a whole word '**py**' in Plover. So, after stroking **PUP TK-LS**, you'd be at a loss for the next stroke, unless you went back to fingerspelling. At that point you probably should have just fingerspelled the whole word to begin with.

The **TK-LS** technique also doesn't work to get '**light engage**'. If you stroke **HRAOEUT EPB TK-LS TKPWAEUPBLG**, you only end up with '**lightengage**'. Or if you try combining **TK-LS** and **S-P** to get **HRAOEUT S-P EPB TK-LS TKPWAEUPBLG**, you do end up with '**light engage**', but only because **EPB** is the prefix stroke for '**en-**', and so it produces the correct letters even though there's no whole word '**en**'. At that point you're basically using prefix strokes anyway, and might as well [really use them](#).

Lesson 7: Orthography Rules And Suffix Keys

Contents

- [1 Orthography Rules](#)
- [2 Orthography Rules and Suffix Keys](#)

Orthography Rules

Sometimes, Plover may give you a pleasant surprise when it writes what you intend, even without the requisite strokes in the dictionary file. How does this work?

The Plover software uses "orthography rules" for some translations. Let's look at an example: "cherries". Plover's dictionary contains an entry for one cherry, "KHER/REU": "cherry". To write cherry, you'd stroke **KHER/REU**. At the time of writing, Plover's dictionary has no entry for "cherries". If you tried to write **KHER/REU-S** using the suffix **-S**, what would you expect? If Plover didn't use orthography rules, **KHER/REU-S** would produce the incorrect spelling, "cherrys". As Plover does use orthography rules, you'll find that Plover automatically translates **KHER/REU-S** to "cherries", even though there's no dictionary entry for it.

You mostly don't have to think about English orthography rules because Plover takes care of it for you! Plover also uses an American English word list to check against for possible translations. Plover supports the following English orthography rules.

+ 'ly': 'artistic' + 'ly' = 'artistically'

Adjectives ending in 'ic' can sometimes be made into adverbs by adding 'ally'. For example, 'artistic' (stroked **AR/TEUS/TEUBG**) can become 'artistically' (stroked **AR/TEUS/TEUBG/HREU**), and 'logic' (stroked **HROPBLG/EUBG**) can become 'logically' (stroked **HROPBLG/EUBG/HREU**).

'te' + 'ry': 'statute' + 'ry' = 'statutory'.

The adjective suffix 'ory' can be appended to various nouns and sometimes verbs to make an adjective form by replacing the 'te' with 'ory'. For example, 'statute' (stroked **STAUT**) can become 'statutory' (stroked **STAUT/REU**). The noun suffix 'ory' can also be added to words by replacing 'te' with 'ory'. For example, 'ambulate' (stroked **APL/HRAEUT**) can become 'ambulatory' (stroked **APL/HRAEUT/REU**).

't' + 'cy': 'frequent' + 'cy' = 'frequency'

Adjectives ending in 't' can sometimes be made into nouns by replacing the 't' with a 'cy'. For example, 'frequent' (stroked **TPREBG**) can become 'frequency' (stroked **TPREBG/SEU**), and 'regent' (stroked **RE/SKWREPBT**) can become 'regency' (stroked **RE/SKWREPBT/SEU**). This rule also applies to adjectives ending in 'te' that can be made into nouns by replacing the 'te' with 'cy'. For example, 'adequate' (stroked **RE/SKWREPBT/SEU**) can become 'adequacy'.

+ 's': 'establish' + 's' = 'establishes'

This rule can address "sibilant pluralization", such as replacing 's' with 'es'. For example, 'establish' (stroked **STAEB**) can become 'establishes' (stroked **STAEB/S**). This rule can also address soft 'ch' pluralization. For example, 'speech' (stroked **SPAOEFP**) can become 'speeches' (stroked **SPAOEFP/S**). Finally, this rule can address a consonant plus 'y' pluralization. For example, 'cherry' (stroked **KHER/REU**) can become 'cherries' (stroked **KHER/REU/S**).

± 'y': 'die' + 'ing' = 'dying'

This rule can address adding 'ing'. For example, 'die' (stroked **TKAOEU**) can become 'dying' (stroked **TKAOEU/-G**). This rule can also address adding 'ist'. For example, 'metallurgy' (stroked **PHET/HRURPBLG**) can become 'metallurgist' (stroked **PHET/HRURPBLG/EUFT**). Finally, this rule can address adding 'ful'. For example, 'beauty' (stroked **PWAOUT**) can become 'beautiful' (stroked **PWAOUT/-FL**).

±'e': 'narrate' + 'ing' = 'narrating'

This rule can address adding 'en' to words ending in 'e'. For example, 'write' (stroked **WREU**) can become 'written' (stroked **WREU/*EPB**). This rule can also address adding 'e' to a word already ending in 'ee'. For example, 'free' (stroked **TPRAOE**) can become 'freed' (stroked **TPRAOE/-D**). Finally, this rule can replace a silent 'e'. For example, 'narrate' (stroked **TPHAR/RAEUT**) can become 'narrating' (stroked **TPHAR/RAEUT/-G**).

consonant doubling: 'defer' + 'ed' = 'deferred'

Sometimes, consonants are doubled when adding certain endings, such as 'ed', 'ing', 'er', or 'est'. For example, 'defer' (stroked **TKEFR**) can become 'deferred' (stroked **TKEFR/-D**), 'sit' (stroked **SEUT**) can become 'sitting' (stroked **SEUT/-G**), 'rub' (stroked **RUB**) can become 'rubber' (stroked **RUB/*ER**), and 'big' (stroked **PWEUG**) can become 'biggest' (stroked **PWEUG/EFT**).

Orthography Rules and Suffix Keys

Additionally, Plover has special behavior for these suffix keys: **-Z**, **-D**, **-S**, **-G**. Let's look at another example: "settled". At the time of writing, Plover's dictionary contains an entry for the present tense verb of settle, "**SELT**": "settle". It has no entry for the outline **SELTD**. And yet, you can stroke **SELTD** and it will produce the word, "settled". When one of the suffix keys is tucked into an existing outline, it's automatically appended. It's as if you had written **SELT/D**. Given **D** translates to the suffix '-ed', you might expect this to result in the incorrect spelling, "settleded". Once again, the orthography rules save us, producing the correct spelling, "settled".

In rare cases, Plover's use of orthography rules might produce a different result to what you're expecting. Without a dictionary entry to delete, you may need to add a new entry to override the automatic behavior.

Lesson 8: Non-Letters

Contents

- [1 Punctuation Strokes](#)
- [2 Useful Strokes](#)

Punctuation Strokes

So far you've learned how to stroke a vast number of words, but without punctuation it's still more of a parlor trick than a useful skill. There are a lot of punctuation marks in Plover. To get started, just learn the ones you find yourself needing:

- `: **KH-FG** (Example: 'files = `bin/lis`' is stroked **KWA*LS KH-FG OEU PWEUPB OEU HR* S* KH-FG**)
- ^: **KR-RT** (Example: 'x^5 is equal to x^2 times x^3.' is stroked **KP* KR-RT #A S KWAL TO KP* KR-RT #T- TAOEUPLS KP* KR-RT #P-**)
- ~: **T*LD**. The tilde with a space after it. (Example: 'Yaaayy~ baby!' is stroked **KWR* A* A* A* KWR* KWR* T*LD PWAEUB/KWREU SKHRAPL**)
- <: **AEPBGT** (Example: 'x<10' is stroked **KP* AEPBGT #SO**)
- =: **KWA*LS** (Example: 'x=10' is stroked **KP* KWA*LS #SO**)
- >: **A*EPBGT** (Example: 'x>10' is stroked **KP* A*EPBGT #SO**)
- |: **PAO*EUP**. The vertical bar, with no space after it. (Example: 'Is -R | less' is stroked **HR* S* S-P H-PB R*P S-P PAO*EUP S-P HRES**)
- _: **R*UPBD**. An underscore character, with no space after it. (Example: 'camelCase vs. under_scores` is stroked **KAPL/EL KPA* KAEUS SR* S* P-P S-P TPH-P R*UPBD SKORS**)
- -: **H-PB**. A single hyphen, with no space after it. (Example: 'It was a bit-coin moment' is stroked **EUT WAS AEU PWEUT H-PB KOEUPB PHOEPLT**)
- --: **TK-RB**. An em-dash (i.e. two hypens), with a space after it. (Example: 'Wait -- I'm going too' is stroked **WAEUT TK-RB AOEUPL TKPWOG TAO**)
- ,: **KW-BG**. A comma, with a space after it. (Example: 'Be good, and be kind' is stroked **-B TKPWAOD KW-BG APBD -B KAOEUPBD**)
- ;: **SKWR*RBGS**. A semicolon, with a space after it, and the next word uncapitalized. (Example: 'Be good; be kind' is stroked **-B TKPWAOD SKWR*RBGS KAOEUPBD**)
- ': **STPH-FPLT**. A colon, with a space after it, and the next word is capitalized. (Example: 'Plover: The winnah' is stroked **PHRO*FR STPH-FPLT W* EU* TPH* TPH* A* H***)
- ::: **KHR-PB**. A colon, with no space after it, and the next word is uncapitalized. Primarily used for specifying clock time. (Example: '24:00' may be stroked **#TH KHR-PB #0D**)
- !: **SKHRAPL**. An exclamation point, with a space after it, and the next word is capitalized. (Example: 'Why? Because!' is stroked **KWR H-F BWAUS SKHRAPL**)
- ?: **H-F**. A question mark, with a space after it, and the next word is capitalized. (Example: 'Why? Why not?' is stroked **KWR H-F KWR PHOT H-F**)
- /: **OEU** (Example: 'x/y=10' is stroked **KP* OEU KWR* KWA*LS #SO**)
- .: **TP-PL**. A regular period, with a space after it, and the next word is capitalized. (Example: 'There it is. Better' is stroked **THR EUT S TP-PL PWET/ER**)
- .: **P-P**. A period used as a decimal point, with no space after it, and the next word uncapitalized. (Example: 'It's.here.for.you' is stroked **T-S P-P HAOER P-P TPOR P-P U P-P**)

- '...' **SKWR-RBGS**. Ellipses, with a space after it, and the next word is capitalized. (Example: 'No... Just no...' is stroked **TPHO SKWR-RBGS SKWRUFT TPHO SKWR-RBGS**)
- "'": **A*E**. Open single quote, with no space after it.
- '"': **AE**. Close single quote, with a space after it. (Example: 'Just say 'hi' and see what happens!' is stroked **SKWRUFT SAEU A*E HEU AE APBD SAOE WHA HAPS SKHRAPL**)
- '""': **KW-GS**. Open double quote, with no space after it.
- '"""': **KR-GS**. Close double quote, with a space after it. (Example: 'I think "love" can be hard to say.' is stroked **EU THEU KW-GS HROF KR-GS K -B HARD TO SAEU TP-PL**)
- '(': **PREPB**
- ')': **PR*EPB** (Example: 'I (love you) want some milk.' is stroked **EU PREPB HROF U PR*EPB WAPBT SOPL PH*EULG TP-PL**)
- '[': **PWR-BGT**
- ']': **PWR*BGT** (Example: 'I luv [sic] you.' is stroked **EU HR* *U SR* PWR-BGT S*EUBG PWR*BGT U TP-PL**)
- '{': **TPR-BGT**
- '}': **TPR*BGT** (Example: 'if(x) { print "hello, world" }' is stroked **TP PREPB KP* PR*EPB TPR-BGT PREUPBT KW-GS HEL/HRO KW-BG WORLD KR-GS TPR*BGT**)
- '@': **KWRAT** (Example: 'zacharyb@gmail.com` is stroked **STKPW* A* KR* H* A* R* KWR* PW* KWRAT TKPW* PH* A* EU* HR* P- P KR* O* PH***)
- '\$': **TK-PL** (Example: 'I got the \$\$\$\$. You got the stuff?' is stroked **EU TKPWOT -T TK-PL TK-PL TK-PL TP-PL U TKPWOT -T STUF H-F**)
- '*': **STA*R** (Example: 'Love you babe :-* :-*' is stroked **HRUF U PWAEUB KHR-PB H-PB STA*R S-P KHR-PB H-PB STA*R**)
- '\': **SPWHRAERB** (Example: 'C:\' is stroked **KR*P KHR-PB SPWHRAERB**)
- '&': **SP-PBD** (Example 'U R my <3 & soul' is stroked **U*P S-P R*P PHAOEU AEPBGT #P- S-P SP-PBD SOUL**)
- '#': **HAERB** (Example: 'You're my #1 guy!' is stroked **UR PHAOEU HAERB #S TKPWAOEU SKHRAPL**)
- '%': **PERS** (Example: 'What % is that? 5%?' is stroked **WHA PERS S THA H-F #A PERS H-F**)
- '+': **PHR*US** (Example: 'y=x+10` is stroked **KWR* KWA*LS KP* PHR*US #SO**)

If your dictionary file is missing any punctuation marks, add a dictionary entry for that mark, using the definitions above. See [Appendix: The Dictionary Format](#) for instructions on how to do that.

Useful Strokes

- Carriage Return: **R-R**
- New paragraph. Insert two carriage returns and capitalize the next word: **SKWRAURBGS**
- Undo stroke: *****
- Letter-by-letter backspace: **PW-FP**
- Insert a space: **S-P**
- Insert a space and capitalize the next word: **KPA**
- Do not insert a space before the next word, and don't capitalize the next word: **TK-LS**
- Do not insert a space before the next word, and capitalize the next word: **KPA***

Plover takes care of a lot of spacing and capitalization, but it can't detect when you want a new paragraph. For that, you still need to a carriage return. If you're using a converted Qwerty keyboard with Plover, you may wonder why Plover bothers providing a whole keystroke for the carriage return, when the original carriage return key is right there on your keyboard. Why learn something new when the old thing is so easily accessible? The reason for this will become clearer and clearer as your typing speed increases. Keeping your fingers at or near their home position is very important, both for typing speed and hopefully for reducing repetitive strain injuries.

Even more convenient is Plover's ability to map keystrokes to just about anything you need. The **SKWRAURBGS** chord isn't intended to sound like anything in particular - it's just a pretty easy stroke to remember. It creates a blank line with two carriage returns, and capitalizes the first word of the new paragraph.

The ***** key is heavily used as a disambiguator in Plover. When used alone, however, the ***** key undoes the most recent keystroke. If you press it repeatedly, you can remove up to a dozen or so keystrokes this way. If you're in the middle of typing a multi-stroke word and accidentally press the wrong chord, the ***** key will remove the unintended stroke, and still remember where you are in the multi-stroke word.

As with the carriage return, the original backspace key is not used in Plover, but the **PW-FP** stroke accomplishes the same behavior, without the fingers having to leave the home position. And even though Plover outputs complete words, there are still occasions when deleting individual characters is useful.

Plover inserts spaces automatically between words and around punctuation as needed; but you may still need to insert spaces yourself from time to time. In particular, if you're dealing with a word boundary error, inserting a space with **S-P** is an appropriate way to force a word division where you want it. For example, the word '**sunlight**' is stroked **SUPB/HRAOEUT**. But what if you want to type the two words '**sun light**' separately? The word '**sun**' is stroked **SUPB**, and the word '**light**' is stroked **HRAOEUT**. If you type them one after the other, Plover will think you mean '**sunlight**'. To make sure that doesn't happen, you should manually add a space between the words. So '**sun light**' is stroked **SUPB S-P HRAOEUT**.

Another use for **S-P** is if you're fingerspelling something that you doubt you'll use often enough to warrant adding its own dictionary entry. In that case, you may want to use **S-P** to include spaces in what you fingerspell.

KPA* is used for [camelcase](#), in other words for a series of words that have no spaces between them, but where each word in the series must be capitalized. For instance, if you wanted to write 'YouTube', it would be **KPA U KPA* TAOUB**. The initial **KPA** ensures there is a space before 'YouTube' and that the 'Y' will be capitalized, while the **KPA*** ensures that the 'T' will be capitalized with *no* preceding space.

Or, if you're writing software, a variable named 'palindromeWordCount' would be stroked **PAL/*EUPB/TKROEPL KPA* WORD KPA* KOUPBT**.

Lesson 9: Numbers

Contents

- 1 The Number Bar
- 2 One And Two Digit Numbers
- 3 Longer Numbers
- 4 Hundreds
- 5 Hundreds Of Dollars
- 6 O'Clock

The Number Bar

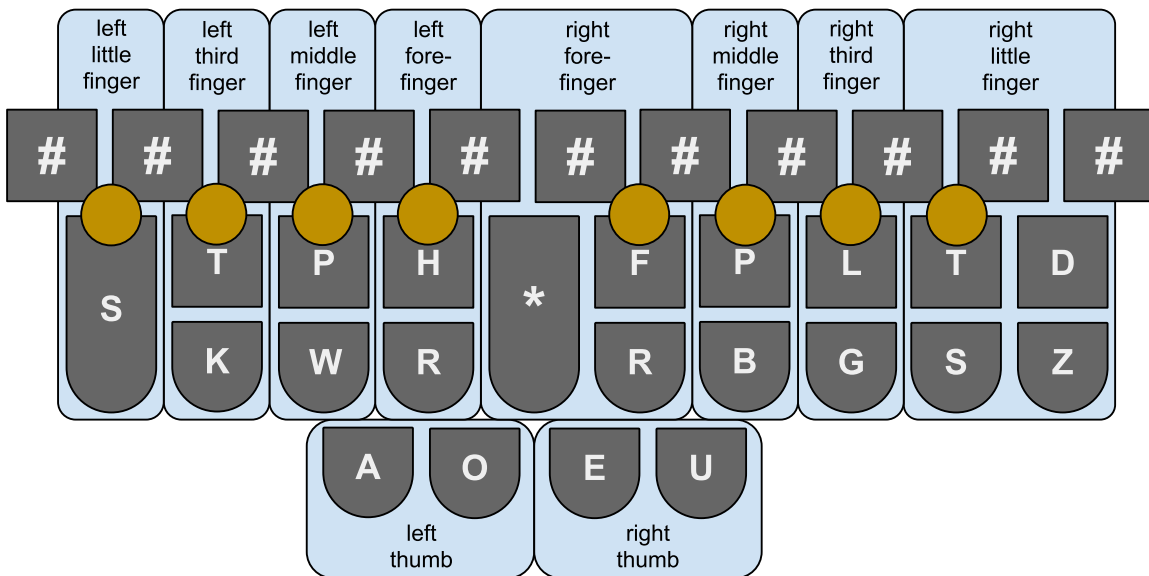
Traditional stenotype keyboards have a single long key at the top, running from left to right, called the number bar. If you've converted a Qwerty keyboard for use with Plover, then pressing any one of the number keys is the same as pressing the steno number bar. It doesn't matter which one you press; they all mean the same thing. Specifically they mean that any other keys in the stroke must be interpreted as producing not English text, but Arabic numerals.

Fig. 1: The Number Bar



For all keys aside from the number bar, all the same fingers remain responsible for the same keys, as covered in the fingering chart from [Lesson 1: Fingers And Keys](#). However, any finger that presses a key in the top row should also simultaneously press the number bar:

Fig. 2: Number Bar Fingering



When pressing more than one key in the top row, only one of your fingers needs to press the number bar - though pressing it with more than one won't harm anything.

If no key in the top row is pressed (for example for **#A** or **#O** as described below), the left middle finger is responsible for pressing the number bar.

One And Two Digit Numbers

These are the strokes to type each of the ten digits:

- '1': #S
- '2': #T-
- '3': #P-
- '4': #H
- '5': #A
- '0': #O
- '6': #F
- '7': #-P
- '8': #L
- '9': #-T

Notice that the '0' is right in the middle, in order to match up with the **O** key. With that exception, also notice that the list of numbers is in steno order: **STKPWHRAO*EUFRPBLGTSDZ**.

To type a two digit number, stroke two single digit numbers together:

- '25': #TA
- '07': #OP
- '49': #HT
- etc.

In their unmodified form, two digit numbers always match the steno order of the keys used to stroke them. To invert that order, add the **EU** chord:

- '52': #TAEU
- '70': #OEUP
- '94': #HEUT
- etc.

To have the second digit be the same as the first, use the single digit that you want, but add the **D** key to the stroke:

- '11': #SD
- '22': #T-D
- '33': #P-D
- '44': #HD
- '55': #AD
- '00': #OD
- '66': #FD
- '77': #-PD
- '88': #LD

- '99': #-TD

Longer Numbers

When you stroke one number after another, Plover does not add a space between them. It's possible to type a long number by stroking all the parts of that number in turn:

- '1068': #SO/#FL
- '700039': #OEUP/#OD/#PT
- '81254583': #LS/#TA/#HA/#PEUL
- etc.

Sequences of digits that are all in steno order may be typed all at once. For example, '25689' may be stroked as #TAFLT. However, when stroking more than two digits this way, the **D**, **EU**, and other modifiers don't work.

There are typically many different ways to stroke a long number. For example, '1068' may also be stroked as #S#O/#F/#L, or #SOFL. It's up to you.

Hundreds

If you want to add a double zero to a single digit number, you could use the **#OD** stroke after the number, for example stroking **'100'** as **#S/#OD**, or you could include the **Z** key in your stroke.

- **'100': #SZ**
- **'200': #T-Z**
- **'300': #P-Z**
- **'400': #HZ**
- **'500': #AZ**
- **'600': #FZ**
- **'700': #-PZ**
- **'800': #LZ**
- **'900': #-TZ**

Hundreds Of Dollars

To prefix a single digit number with the '\$' character and add a double zero after it, add the **DZ** chord to the number stroke:

- '\$100': #SDZ
- '\$200': #T-DZ
- '\$300': #P-DZ
- '\$400': #HDZ
- '\$500': #ADZ
- '\$600': #FDZ
- '\$700': #-PDZ
- '\$800': #LDZ
- '\$900': #-TDZ

O'Clock

To express a one or two digit number as clock time, add either **K** or **BG** to the stroke:

- '1:00': #SK or #SBG
- '2:00': #TK or #TBG
- '3:00': #KP or #PBG
- '4:00': #KH or #HBG
- '5:00': #KA or #ABG
- '6:00': #KF or #FBG
- '7:00': #K-P or #-PBG
- '8:00': #KL or #BLG
- '9:00': #KT or #BGT
- '10:00': #SKO or #SOBG
- '11:00': #SKD or #SBGD
- '12:00': #STK or #STBG
- '13:00': #SKP- or #SP-BG
- '14:00': #SKH or #SHBG
- '15:00': #SKA or #SABG
- '16:00': #SKF or #SFBG
- '17:00': #SK-P or #S-PBG
- '18:00': #SKL or #SBLG
- '19:00': #SKT or #SBGT
- '20:00': #TKO or #TOBG
- '21:00': #STKEU or #STEUBG
- '22:00': #TKD or #TBGD
- '23:00': #TKP- or #T-PBG
- '24:00': #TKH or #THBG

Lesson 10: Designing Briefs

Contents

- 1 Introduction
- 2 Check First
- 3 Brief Community
- 4 Make It Easy
- 5 Avoid Words And Word Parts
- 6 The Essence Of The Word
- 7 The Uncommon Syllable
- 8 First And Last Word Parts
- 9 Distinctive Parts Of Phrases
- 10 Multiple Inversions
- 11 Drop Stressed Vowels
- 12 Drop More Sounds
- 13 The Asterisk Is Everyone's Friend
- 14 Single Keys For Word Prefixes And Suffixes
- 15 Overlap Chords

16 Single Keys For Related Sounds
17 The OEU Wildcard

Introduction

Plover, and all steno systems, assume that users regularly update their dictionary files with personalized briefs, commands, and other entries. You don't have to do it. The default dictionary may be sufficient for your needs. You may be comfortable typing multi-stroke words, and fingerspelling names, and so on. Or you may simply dislike the concept of briefs, and prefer to avoid too much memorization.

Designing briefs is not for everyone, but it is an integral part of high speed typing. The default Plover dictionary is the result of years of refinement in a professional setting, and so you can learn to type extremely fast with it. It includes well-designed single-stroke briefs for many words. But there will always be the opportunity to increase that efficiency. Any time you find yourself fingerspelling something, or typing a multi-stroke word, that's an opportunity to see if there's a corresponding brief in the dictionary file, or create one yourself. Any time you find yourself using the same phrase many times in a given project, that's an opportunity for a brief. And if you use keystroke-centric software like emacs or vim, or if you do a lot of software programming, those all include many opportunities to create briefs.

The main thing to remember when creating your own briefs is that they should somehow resemble the target word. Most of the suggestions in this lesson have to do with designing a brief that sounds like its word. Internalize that principle and you'll ultimately end up with a clean library of reliable briefs. Ignore that principle and you'll start to run into difficulties remembering the briefs you've created, or avoiding conflicts between them.

Take your time designing briefs, especially when you're just getting started. Give each one some real thought. Try different approaches, and see if one of them in particular stands out as an obviously best choice. Or, if several seem equally good, you can add them all. There's no rule against adding more than one brief for the same word.

Or if you're creating briefs for non-words, like various programming constructs, try to design patterns that make sense to you, and that will support a corpus of related briefs.

Keep track of the briefs you design yourself. It's possible that you'll realize you want to take a different approach with some of them; and if that happens it'll be good to be able to confidently find them all in the dictionary file.

Check First

Before coming up with your own brief, check the dictionary file to see if one exists already for the target word. Try to make this something you can do easily at any time. Learn the location of the dictionary file on disk. Familiarize yourself with the dictionary entry format. If you're on a UNIX system, a simple `grep` command will find all the entries for a given word. So for example, if you use the word 'daughter' a lot, you might want to save a stroke from **TKAU/TER**, and make a brief. Before designing your own brief, check the dictionary file:

```
$ grep \"daughter\" dict.json
\"TKAU/TER\": \"daughter\",
\"TKAUG/TER\": \"daughter\",
\"TKAUGT/ER\": \"daughter\",
\"TKAUR\": \"daughter\",
\"TKAUT/ER\": \"daughter\",
\"TKAUT/TER\": \"daughter\",
```

It looks like **TKAUR** has already been defined. In that case, there's no more work to do, unless you want to add another brief for that word.

Likewise, if you come up with what you think is the perfect brief for a word, don't forget to see if that stroke is already used in the dictionary file. For example, if you think **PWHRAOF** would make a great brief for '**blowfish**', check the dictionary to see if that chord is already in use:

```
$ grep PWHRAOF /home/zbrown/.config/plover/dict.json
\"PWHRAOF\": \"bulletproof\",
\"PWHRAOFP\": \"brooch\",
```

It looks like **PWHRAOF** is already the brief for '**bulletproof**'. In that case, you can decide which of the two words has a better claim to the brief. If you expect to use '**blowfish**' a lot more than '**bulletproof**', you could assign '**blowfish**' to **PWHRAOF** instead.

Brief Community

When designing briefs, especially while you are just starting out, take advantage of your steno friends. Ask for opinions from the folks on the [Plover mailing list](#). Maybe some of them will see a reason why the brief you're considering might cause problems for you later. Once you get the hang of it, your briefs will tend to be well-formed each time. But when you're starting out, the freedom to use *any* stroke for a brief can lead to trouble. It may not be obvious that there are good briefs and bad briefs, and that bad briefs are bad for a reason. The Plover community may help you figure out briefs that you'll like for a longer time.

Make It Easy

If a brief is hard to type, you might as well just use the phonetic form. Whenever you construct a brief, give some thought to the finger movements it requires, and to whether you'll find it comfortable to type.

Equally important is to make your briefs easy to *remember*. This means that whenever possible, you should try to follow some sort of memorable general pattern. The rest of this tutorial explains some simple principles that may help you design easy, memorable briefs.

Avoid Words And Word Parts

Don't choose a stroke that's already part of a multi-stroke word. For example, you might think **SKOP** would make a good brief for something, but it's already used by a number of words, including 'endoscopic': **EPBD/SKOP/EUBG**, 'kaleidoscopic': **KHRAOEUD/SKOP/EUBG**, and many others. If you use **SKOP** for a brief, you risk word boundary errors with all those other words.

One little brief might not seem like a big deal, but they add up. As you develop more and more briefs (which you will), the problem will become more and more visible to you until you find yourself having to perform special contortions for many words - memorizing suboptimal ways to type certain words and word combinations, just to avoid boundary errors. To truly fix the problem, you'd have to go back and expunge all the briefs that you had worked so hard to create and memorize. Bad scene.

The Essence Of The Word

It's easier to remember briefs that sound a bit like the word they represent. For example, 'extraordinary' is stroked **KPRAORD**. The brief sounds like 'ksrord', which may be a reminder of the full word.

The Uncommon Syllable

If one of the syllables of the target word is distinctive, you could try using it as the brief for that word. For example, 'English' is stroked **TKPWHREURB**, which sounds like 'glish'. This may seem to violate the rule about avoiding words and word parts; but really it's OK if the word parts are very rare. Check before using them. In the case of **TKPWHREURB**, the only dictionary entry that uses that stroke is the word 'English' itself. There are no conflicts at all.

First And Last Word Parts

Sometimes the start and end of a word make a good brief. So, '**something**' can use the brief **SG**, and '**mechanism**' can use the brief **PHEUFPL** (which sounds like '**mism**').

Distinctive Parts Of Phrases

You can make briefs for phrases. Sometimes it can be useful to make a brief that incorporates the signature portions of each word in a given phrase. So for example, '**in other words**' can be stroked **TPHOERDZ**, which sounds like '**nords**'. And '**in addition**' can be stroked **TPHAGS**, which sounds like '**nashun**'

Multiple Inversions

According to Plover theory, you can only do one inversion per word. But sometimes you can squeeze a word into a single stroke if you use more than one inversion to do it. For example, 'designer' can be stroked **STKAOEURPB**.

Drop Stressed Vowels

Plover theory says you can drop unstressed vowels, but sometimes that's not enough to fit a word into a single stroke. Sometimes dropping a stressed vowel will manage it. For example, 'affect' can be stroked **AFBGT**. This avoids a conflict with 'effect' which is stroked canonically as **TPEBGT**. As another example, the word 'collar' can be stroked **KHRAR**. Without dropping the stressed vowel, it would require two strokes, either **KOL/AR** or **KOL/HRAR**.

Drop More Sounds

But sometimes you can squeeze a word into a single stroke by dropping a sound or two that still leaves the word very recognizable. For example, '**extraneous**' can be stroked **KPRAEUPBS** if you drop the 't' in addition to the unstressed vowels. Otherwise it's a three stroke word.

The Asterisk Is Everyone's Friend

The * key has many uses. It can be typed on its own to undo previous strokes. In Plover theory the **S*** chord represents the 'z' sound.

In terms of constructing briefs, sometimes you can design a good brief, but it's already taken. In which case, maybe adding the * key will produce a stroke that's not yet in the dictionary file. For example, the word '**herb**' is stroked **HERB**, but the capitalized proper name '**Herb**' can be stroked **H*ERB**.

Single Keys For Word Prefixes And Suffixes

In some cases, you could squeeze a word into a single stroke by replacing its prefix with a single key. A good example is chording the '**com-**' prefix as the **K** key. For example, you can stroke '**compound**' as **KPOUPBD**, '**compress**' as **KPRES**, and '**combat**' as **KPWAT**.

Another good example is chording the '**ex-**' prefix as the **X** key. For example, you can stroke '**exam**' as **KPAPL**, and '**excited**' as **KPAOEUTD**.

Another good example is chording the '**-ly**' suffix as **L** (although that could be thought of as simply dropping the unstressed '**y**' sound). For example, '**highly**' can be stroked **HAEUL**, and '**slowly**' can be stroked **SHROEL**.

The same technique could be used for other keys and prefixes.

Overlap Chords

Sometimes the keys you need for a particular sound have already been used by a different sound in the word. For example, the word '**dimple**' needs both the '**mp**' sound and the '**l**' sound at the end of the word. But the '**mp**' sound needs the ***PL** keys, which means the **L** key isn't available for the '**l**' sound. So, just use the **L** key anyway, and stroke **TK*EUPL**.

As another example, '**shmeat**' needs both the '**sh**' sound and the '**m**' sound at the start of the word. But the '**sh**' sound needs the **SH** keys, while the '**m**' sound needs the **PH** keys. Both sounds need the **H** key, which would violate Plover theory. So, just use it anyway, and stroke **SPHAO*ER**. (also note that the ***** key is used to disambiguate '**shmeat**' from '**smear**', which is stroked **SPHAOER**).

Single Keys For Related Sounds

The 'v' sound on the left side of the keyboard uses the **SR** keys, which are very popular letters. Sometimes you can squeeze a 'v' word into a single stroke by using the **W** key instead. For example, '**diverse**' is stroked **TKWERS**, '**divorce**' is stroked **TKWORS**, and '**diversion**' is stroked **TKWERGS**.

The OEU Wildcard

Since 'oi' is the least common English diphthong, you can sometimes use its Plover chord **OEU** in a brief, without conflicting with other words. So taking one or a couple significant parts of a word, and joining them with **OEU**, might make a good brief. For example, '**carbon dioxide**' could be thought of as being exemplified by its 'k' and 'x' sounds. Then by joining these with **OEU**, you get the brief for '**carbon dioxide**', which is **KOEUBGS**.

Appendix: Word Boundaries

Contents

- 1 The Multi-Stroke Dilemma
- 2 The Natural Resolution
- 3 Taking Control

The Multi-Stroke Dilemma

If Plover only supported single-stroke words, then it would be obvious where each word ended and the next began. But with multi-stroke words, Plover has to decide whether a given stroke is a whole word or just a part of a larger word. It needs to make this decision partly because it wants to output the correct word, and partly because Plover inserts spaces automatically between words, and needs to get that right as well.

Plover (and all steno systems) must make this choice for each stroke you type. This introduces a whole new class of homophones into the mix. If Plover only supported single stroke words, it would only need to disambiguate between individual words like '**beat**' and '**beet**'. By supporting multi-stroke words, Plover also has to deal with homophonic word sequences like '**light engage**' and '**lighten gauge**'.

The Natural Resolution

Plover can't intuitively know which homophonic word sequence you intend with a series of strokes. But it does have a consistent and predictable set of internal rules that results in a clean selection for every sequence of strokes:

1. **Plover will never *split* a word it has already output. For example, once Plover has output the word 'lighten', a later stroke can never cause Plover to back up and split that word into 'light' and another word beginning with 'en-'.**
2. **Plover will always engulf a whole word it has already output into a longer word, if available.** For example, if you stroke **TPAOEUR**, Plover will immediately output the word 'fire'. If you then stroke **PHRAEUS**, Plover will remove the word 'fire', and engulf that stroke into the new word 'fireplace' instead.
3. Plover will always choose the longest available series of strokes that match a dictionary entry. This is really just another way to say (2) above. For example, if you type a series of 10 strokes, you might see Plover output 10 different words. But if, on the 11th stroke, Plover sees a matching 11-stroke entry in your dictionary file, it will backspace over all 10 of those previous words, and replace them with the word that matches that 11-stroke entry.

Taking Control

Plover supports various techniques for controlling which multi-stroke homophone it should output. The most preferred option by far is described in [Lesson 5: Prefix And Suffix Strokes](#). But there are [other techniques](#) that, while perhaps more convenient for the beginner, are slow and should not be considered a real solution to the problem.

Appendix: The Dictionary Format

Contents	
1	The Basics
2	Introduction To Commands
3	Extended {^} Commands
4	The {&} Glue Command
5	Raw {#} Keyboard Inputs
5.1	Available Key Names
5.2	Key Sequences
5.3	Holding Shift Keys
5.4	Nesting For Arbitrary Combinations
6	The {Plover:} Control Commands
6.1	Add Translation
6.2	Suspend
6.3	Resume
6.4	Toggle
6.5	Configure
6.6	Focus
6.7	Quit

This appendix draws heavily from [Josh Lifton's 2010 Plover Guide](#).

The Basics

The Plover dictionary file is in [JSON](#) format, essentially a bunch of key/value pairs. Each pair is separated from the next by a comma, and the key and the value are each in double quotes, and separated from each other by a colon. And the entire dictionary is enclosed in a single pair of curly braces:

```
{
  "KEYSTROKE": "output",
  "KEYSTROKE2": "output2",
  "KEYSTROKE3": "output3"
}
```

Note that the last entry in the dictionary file has no trailing comma.

In the keystroke portion of each entry, a hyphen is used to disambiguate which side of the keyboard has a particular key:

```
"SK-D": "asked",
"SKW-D": "asked",
"SK-R": "asker",
"SK-RS": "askers",
```

Note that the hyphen does not indicate multiple keystrokes. When stroking **SK-D**, you would press the **S**, **K**, and **D** keys at the same time.

To indicate multiple strokes, the keystroke portion of each entry uses a forward slash:

```
"AS/KAPBS": "askance",
"AS/SKAPBS": "askance",
"A/SK-D": "asked",
"A/SKAOU": "askew",
"AS/KAOU": "askew",
"AS/SKAOU": "askew",
"SK/-G": "asking",
```

Some entries have lots of strokes:

```
"HREBG/TR0/EPB/SEF/HRO/TKPWRAPL": "electroencephalogram",
```

An entry can output more than just a single word. In the output portion of the entry, use blank spaces to create strings of words to output:

```
"TP-FS": "all of a sudden",
"TPHUFBR": "area under the curve",
"SPHABGT": "as a matter of fact",
"HRAEURPBLG": "ladies and gentlemen of the jury",
```

Aside from spaces and letters, you can use numerals, and any of `.'!-$%:./()&=>?*'` in the output portion of an entry.

```
"AOEU/KWRAOE": "i.e.",
"UD": "you'd",
"AUP/TO/TKAEUT": "up-to-date",
"TK-PL": "quot;",
"TPEUF/PERS": "5%",
"KHRPB": ":",
"ETS/ETS/ETS": "et cetera, et cetera, et cetera",
"HA*F": "1/2",
"TPHAUBL": "(inaudible)",
"KW-RBGS/TPHA*": "Q&A",
"KW-L": "=",
"TKPWR*PB": ">",
"STPHU": "you?",
"STA*R": "*",
```

Note that the above characters aren't special in any way. They are simply output by Plover, exactly like other entries.

To use the double quote "" character in the output portion of an entry, you need to precede it with a backslash \:

```
"KWOT": "\",
```

The above entry isn't in the dictionary file because it's not very useful. As it is, Plover treats it like any other word, and inserts spaces before and after. So if you stroke **SAEU/KWOT/HEU/KWOT/TO/HEUPL**, Plover will produce **'say " hi " to him'** instead of **'say "hi" to him'**.

Introduction To Commands

To get special behaviors from certain characters, you must enclose them in curly braces within the output portion of an entry. For example, '{.}' tells Plover to output a period followed by a space, and then capitalize the first letter of whatever word the user types next:

```
"TP-PL": "{.} ",
```

Likewise, '{?}' does the same with the question mark, and '{!}' does the same with an exclamation mark:

```
"H-F": "{?} ",  
"SKHRAPL": "{!} ",
```

These characters wrapped in curly braces do not have to take up the whole output by themselves. They can be combined with regular text and other curly braced characters:

```
"-RBGS/TOS/STPH": "{,}does it{?} ",
```

The above entry attaches a comma to the previous word, then adds a space, and the words "does it", then adds a question mark, then adds a space and forces Plover to capitalize the next word. For example, stroke **EUT/TKOPBT/HURT/-RBGS/TOS/STPH** to produce '**it doesn't hurt, does it?**'

Note that any blank spaces surrounding the curly braces are ignored. So, the above entry produces the exact same output as:

```
"-RBGS/TOS/STPH": " {,} does it {?} ",
```

If you don't necessarily want to add punctuation, but you still want Plover to capitalize the first letter of whatever the user types next, put '{-|}' at the end of the output portion of the entry:

```
"TKR*FPLT": "Dr.{-|} ",
```

In the above entry, it's not the '.' outside of curly braces, but the '{-|}', that causes the first letter of the next word to be capitalized. So to get '**Dr. King**', you could stroke **TKR*FPLT/KEUPBG**.

Notice that in the above example, there's a space between the '**Dr.**' and the '**King**'. This has nothing to do with either the '.' or the '{-|}'. This is simply Plover inserting a space before each next word, as usual.

Just as the '{-|}' command forces the first letter of the next word to be capitalized, so the '{>}' command forced the first letter of the next word to be lowercase:

```
"KEYSTROKE": "{>} ",
```

Normally, stroking **KO*EUPBG** produces '**King**'. But if you stroke the above entry first, and then stroke **KO*EUPBG**, it produces '**king**'.

It's possible to suppress the space between words, using '{^}'. Consider if the 'king' entry used this:

```
"KEUPBG": "{^}king",
```

Now stroking **TKR*FPLT/KEUPBG** would produce '**Dr.King**' instead of '**Dr. King**'. The '{-|}' from the previous entry causes '**King**' to be capitalized, while the '{^}' in this entry suppresses the space between '**Dr.**' and '**King**'.

You could get a similar effect by moving the '{^}' from the start of the '**king**' entry to the end of the '**Dr.**' entry.

```
"KEUPBG": "king",  
"TKR*FPLT": "Dr.{-|}{^}",
```

Now **TKR*FPLT/KEUPBG** will still produce '**Dr.King**' with no space between '**Dr.**' and '**King**'. But the same effect will now be achieved for other doctors as well. So, **TKR*FPLT/HOUS** produces '**Dr.House**', and **TKR*FPLT/WHO** produces '**Dr.Who**'.

If you like this space-suppressing effect in general, but just not for a specific case, you can force Plover to output a space, using '{ }' (a blank space wrapped between curly braces).

```
"KEUPBG": "{ }king",  
"TKR*FPLT": "Dr.{-|}{^}",
```

Now, **TKR*FPLT/HOUS** will still produce '**Dr.House**' and **TKR*FPLT/WHO** will still produce '**Dr.Who**', but **TKR*FPLT/KEUPBG** will produce '**Dr. King**', with a space between '**Dr.**' and '**King**'.

Notice that the second stroke overrode the first. The **TKR*FPLT** stroke instructs Plover to suppress the following space, but then the **KEUPBG** stroke tells Plover to add it back in again.

Extended {^} Commands

In the previous section, we saw how '{^}' could be used to suppress the space between a word and its neighbor in either direction. In fact, this is the foundation of a very powerful feature of Plover - the ability to support [prefix and suffix strokes](#).

```
"K*EUL": "{kilo^}",
```

In the above entry, the '^' is on the right, and suppresses the space between the 'kilo' and the next word. It has the effect of turning the 'kilo' into a prefix that you can prepend onto any word. So **K*EUL/KEUPBG** produces 'kiloking', **K*EUL/HOUS** produces 'kilohouse', and **K*EUL/WHO** produces 'kilowho'.

Or consider the opposite case:

```
"-G": "{^ing}"
```

In the above entry, the '^' is on the left, and suppresses the space between the 'ing' and the previous word. It has the effect of turning the 'ing' into a suffix that you can append onto any word. So **KEUPBG/-G** produces 'kinging', and **HOUS/-G** produces 'housing', and **WHO/-G** produces 'whoing'.

But wait! There was something unexpected in one of those examples. Why didn't **HOUS/-G** produce 'houseing' instead of 'housing'? How did Plover know to drop the final 'e' before adding the 'ing'?

In fact, Plover recognizes four special suffixes - '-ed', '-ing', '-er', and '-s':

```
"-D": "{^ed}",  
"-G": "{^ing}",  
"*ER": "{^er}"  
"-S": "{^s}",
```

These four suffix strokes have special logic in the Plover engine itself that has nothing to do with the dictionary file, and that cause them to do the right thing when used at the end of most words. So **HEUT/-G** produces 'hitting' with a doubled 't', while **KAOEP/-G** produces 'keeping' with only a single 'p'. The Plover engine understands the English rules for adding these suffixes to words.

Other suffix strokes simply append their suffix to the existing word as stroked.

```
"KRAES": "{^cracy}",
```

The above suffix has no logic internal to the Plover engine, and simply appends 'cracy' to whatever word came before. So **KEUPBG/-KRAES** produces 'kingcracy', **HOUS/-KRAES** produces 'housecracy', and **WHO/-KRAES** produces 'whocracy'.

Prefixes and suffixes can be combined without any root word:

```
"EUD/KWRO": "{idio^}",  
"KRAES": "{^cracy}",
```

The above two entries can be stroked together as **EUD/KWRO/KRAES** to produce 'idiocracy'.

It's also possible to use the '^' at both ends of an entry, to suppress the space on both sides:

```
"K*UPL": "{^-cum-^}",
```

Using the above entry, it's possible to stroke **PHURD/K*UPL/S-DZ** to produce 'murder-cum-suicide'.

The '^' is a general-purpose command. It can be used wherever you'd like to suppress a previous or subsequent word break:

```
"SKWR-RBGS": "{^...}{ }{-|}",
```

The above entry produces the ellipses, followed by a space, and capitalizing the following word. It can be used when the ellipses are intended to end a sentence. So you could stroke **HE/WAS/TKPWOPB/SKWR-RBGS/WE/HREFT/-T/TPHELGT/TKAEU** to produce 'He was gone... We left the next day.'

An example of a case where word breaks must be routinely altered is when using quotes.

```
"KR-GS": "{^~|\""},  
"KW-GS": "{~|\"^}",
```

These are the dictionary entries for the closing quote and the opening quote. For example, stroke **SAEU/KW-GS/HEU/KR-GS/TO/HEUPL** to produce 'say "hi" to him'.

The '~|' tells Plover to maintain capitalization after the quote. If one does **KPA/KW-GS/HEL/HROE** the result is '"Hello'; however, if one were to do **KPA/S-P/HEL/HROE** the result would be ' hello' as the **KPA** capitalization is no longer in effect.

Sometimes you want punctuation on the inside of the quote, sometimes on the outside. Personally, I put punctuation on the inside when quoting human speech, and outside when quoting computer speech. Plover supports both approaches:

```
"KR-GS/TP-PL": "{^\""}{^}.{^ ^}{-|}",  
"TP-PL/KR-GS": "{^}.{^\""}{^ ^}{-|}",
```

The first entry suppresses a space after the previous word, then adds the double quote character, then suppresses the space after that and adds the period character, then adds a space, and capitalizes the first letter of whatever word comes next. So for example, stroke **AOUS/KW-GS/TKPW*/R*/E*/P*/KR-GS/TP-PL/EUT/WORBGS** to produce 'use "grep". It works'.

The second entry does the same, but outputs the period before the double quote character. So for example, stroke **SAEU/KW-GS/HEU/TP-PL/KR-GS/TKPWO/OPB** to produce 'say "hi." Go on'.

The {&} Glue Command

The '&' command is similar to the '^' command. It always suppresses spaces on both the left and the right sides of the output; but only for neighboring strokes that are also '&' glue commands. This is the foundation of Plover's fingerspelling support.

```
"P*": "{&p}",
"*EU": "{&i}",
"T*": "{&t}",
```

These are the dictionary entries for the fingerspelled letters 'p', 'i', and 't'. To fingerspell the word 'pit', you could stroke **P*/*EU/T***. To type 'snake pit city' you could stroke **STPHAEUBG/P*/*EU/T*/STEU**.

Notice that 'snake' and 'city' are not joined with 'pit' into 'snakepitcity', because the entries for 'snake' and 'city' don't contain the '&' glue command, while the strokes for the letters 'p', 'i', and 't' do:

```
"STPHAEUBG": "snake",
"STEU": "city",
```

Also notice that the '&' character has no special power in the dictionary file, outside of curly braces:

```
"HED/PH-PBD/SHOULD/ERS": "Head & Shoulders",
```

In the above entry, the '&' in the output portion of the entry is just a regular ampersand. It only becomes a glue command inside curly braces. So when you stroke **HED/PH-PBD/SHOULD/ERS**, you get the exact output 'Head & Shoulders'.

Raw {#} Keyboard Inputs

A lot of computer software takes keyboard command sequences, and often these sequences use keys like `CTRL`, `ESC`, `SHIFT`, `NumLock`, `F7`, and so on. Plover allows you to define strokes that simulate pressing any key on the keyboard, in any simultaneous combination and in any sequential order. This feature can give you tremendous power over your computer, but it also comes with dangerous caveats. As we've seen, Plover will unhesitatingly guess at the meaning of a stroke, then back up and [replace that stroke with another](#). If your software can't tolerate commands that back up and change, then you'll need to make sure none of the strokes for these commands have any conflicting matches in the dictionary file. And since the dictionary file changes over time as you refine it, you'll need to make sure no conflicts emerge over time.

Before explaining how to encode raw keyboard input, it may be useful to see the full list of available key names. These correspond exactly to the traditional names of key codes that a keyboard can send into to the computer.

Available Key Names

The following is the full set of keys that may be referenced with the '{#}' command in Plover (note that the `Super_L` and `Super_R` keys correspond to the command keys in OSX):

0	BackSpace	ediaeresis	F29	igrave	KP_Next	Multi_key	plus	sectio
1	bar	Ediaeresis	F3	Igrave	KP_Page_Down	MultipleCandidate	plusminus	Select
2	Begin	egrave	F30	Insert	KP_Page_Up	multiply	PreviousCandidate	semic
3	braceleft	Egrave	F31	j	KP_Prior	n	Print	Shift_
4	braceright	Eisu_Shift	F32	J	KP_Right	N	Prior	Shift_
5	bracketleft	Eisu_Toggle	F33	k	KP_Separator	Next	q	Shift_
6	bracketright	End	F34	K	KP_Space	nobreakspace	Q	Single
7	Break	equal	F35	Kana_Lock	KP_Subtract	notsign	question	slash
8	brokenbar	Escape	F4	Kana_Shift	KP_Tab	ntilde	questiondown	space
9	c	eth	F5	Kanji	KP_Up	Ntilde	quotedbl	sshar
a	C	Eth	F6	Katakana	l	numbersign	quoteleft	sterlin
A	Cancel	ETH	F7	KP_0	L	Num_Lock	quoteright	Super
aaacute	Caps_Lock	exclam	F8	KP_1	L1	o	r	Super
Aacute	cedilla	exclamdown	F9	KP_2	L10	O	R	Sys_R
acircumflex	Cedilla	Execute	Find	KP_3	L2	oacute	R1	t
Acircumflex	cent	f	g	KP_4	L3	Oacute	R10	T
acute	Clear	F	G	KP_5	L4	ocircumflex	R11	Tab
adiaeresis	colon	F1	grave	KP_6	L5	Ocircumflex	R12	thom
Adiaeresis	comma	F10	greater	KP_7	L6	odiaeresis	R13	threeq
ae	Control_L	F11	guillemotleft	KP_8	L7	Odiaeresis	R14	threes
AE	Control_R	F12	h	KP_9	L8	ograve	R15	Touro
agrave	copyright	F13	H	KP_Add	L9	onehalf	R2	twosu
Agrave	currency	F14	Hankaku	KP_Begin	Left	onequarter	R3	u
Alt_L	d	F15	Henkan	KP_Decimal	less	onesuperior	R4	U
Alt_R	D	F16	Henkan_Mode	KP_Delete	Linefeed	ordfeminine	R5	uacut
ampersand	dedilla	F17	Hiragana_Katakana	KP_Divide	m	oslash	R6	ucircu
apostrophe	degree	F18	Home	KP_Down	M	otilde	R7	udiaer
aring	Delete	F19	Hyper_L	KP_End	macron	p	R8	ugrave
Aring	diaeresis	F2	Hyper_R	KP_Enter	Mae_Koho	P	R9	Ugrav
asciicircum	division	F20	hyphen	KP_Equal	masculine	Page_Down	Redo	under
asciitilde	dollar	F21	i	KP_F1	Massyo	Page_Up	registered	Undo
asterisk	Down	F22	l	KP_F2	Menu	paragraph	Return	Up
at	e	F23	iacute	KP_F3	Meta_L	parenleft	Right	v
atilde	E	F24	iacute	KP_F4	Meta_R	parenright	Romaji	V
Atilde	eacute	F25	icircumflex	KP_Home	minus	Pause	s	w
b	Eacute	F26	Icircumflex	KP_Insert	Mode_switch	percent	S	W
B	ecircumflex	F27	idiaeresis	KP_Left	mu	period	script_switch	x
backslash	Ecircumflex	F28	Idiaeresis	KP_Multiply	Muhanan	periodcentered	Scroll_Lock	X

Key Sequences

To define a dictionary entry that will send a series of keystrokes to your computer, list all the keys within the '{#}' command, separated by spaces:

```
"KEYSTROKE": "{#w h i l e parenleft parenright Return braceleft Return space space space space Return braceright Return Up Up End}",
```

The above entry will produce:**while()**

```
{  
  -  
}
```

In the above output, the cursor is placed at the line between the curly braces, at the far right.

Any '{#}' command may be interspersed with other commands:

```
"KEYSTROKE": "{^} {#h i space t h e} {#r e}",
```

The above entry will prevent Plover from adding a space after the previous word, and will then output **'hi there'**. Note that the word **'there'** is not output as **'the re'**. Plover does not automatically insert spaces between two '{#}' commands.

Holding Shift Keys

To define an entry that uses one of the shift keys, first include the name the shift key, then in parentheses put the names of the sequence of keys to press and release:

```
"KEYSTROKE": "{#Shift_L(h a p p y)",
```

The above entry will produce **'HAPPy'**. It was as if you first held down the left shift key, then pressed and released the "h", "a", and "p" keys in succession; then released the left shift key, and finally pressed and released the "p" and "y" keys in succession.

You may intersperse parenthesized shift-groups within any '{#}' command:

```
"KEYSTROKE": "{#Control_L(f) Shift_L(t)}",
```

The above entry is as if you

1. held down the left CTRL key
2. pressed and released the "f" key
3. released the CTRL key
4. held down the left shift key
5. pressed and released the "t" key
6. released the left shift key

Nesting For Arbitrary Combinations

It is also possible to nest parenthesized shift-groups, to simulate holding down more than one shift key at a time:

```
"KEYSTROKE": "{#Control_L(Alt_L(Delete))}",
```

The above entry might reboot some computers. It is as if you first held down the left CTRL key, then *also* held down the left ALT key, and then (with both of those keys held down) pressed and released the delete key.

The {Plover:} Control Commands

You can give commands to the Plover software to turn it on, off, add a new dictionary entry, and a few other things. In the following examples, replace "STROKE" with the actual key combinations you want to associate with each command.

Defaults are set for some of these commands in [commands.json](#).

Add Translation

The `ADD_TRANSLATION` command pops up Plover's dictionary editing window. The introduction of this command is what brought Plover across the threshold and made it professionally usable.

```
"TKUPT": "{PLOVER:ADD_TRANSLATION}",
```

Notice that the `TKUPT` stroke is short for "dictionary update". It's the recommended stroke for this command.

The `ADD_TRANSLATION` command pops up a window that has input fields to add a new dictionary stroke and its desired output. Note that you can tab between the input and output fields using the `Tab` command (either via a Plover stroke or pressing the Tab key on the QWERTY keyboard). You may also accept the entry using the `Enter` command (also either via a Plover stroke or by pressing the Enter key on the QWERTY keyboard). Or, instead of pressing Enter, you may abort the new translation and close the pop-up interface using the `Escape` command (also either via a Plover stroke or by pressing the Esc key on the QWERTY keyboard).

Suspend

The `SUSPEND` command causes Plover to stop interpreting keystrokes as steno inputs and return to normal QWERTY mode.

```
"PHROF": "{PLOVER:SUSPEND}",
```

Resume

The `RESUME` command causes Plover to leave QWERTY mode and interpret keystrokes as steno inputs. Note that the keystroke to invoke the `RESUME` command does work when the keyboard is in QWERTY mode. That and the `TOGGLE` command are the only strokes that do.

```
"PHROPB": "{PLOVER:RESUME}",
```

Toggle

The `TOGGLE` command causes plover to enter QWERTY mode if it's in steno mode, and enter steno mode if it's in QWERTY mode. Note that Plover correctly interprets the keystroke to invoke the `TOGGLE` command regardless of which mode Plover is in at the time.

```
"PHROLG": "{PLOVER:TOGGLE}",
```

Configure

The `CONFIGURE` command pops up Plover's configuration window.

```
"KEYSTROKE5": "{PLOVER:CONFIGURE}",
```

Focus

The `FOCUS` command raises the Plover window and puts the cursor focus inside it; or if the window is minimized, Plover de-minimizes it and then raises it.

```
"KEYSTROKE6": "{PLOVER:FOCUS}",
```

Quit

The `QUIT` command causes the Plover process to terminate. The software exits entirely, leaving the keyboard in QWERTY mode.

```
"KEYSTROKE7": "{PLOVER:QUIT}",
```

Appendix: Cheat Sheet

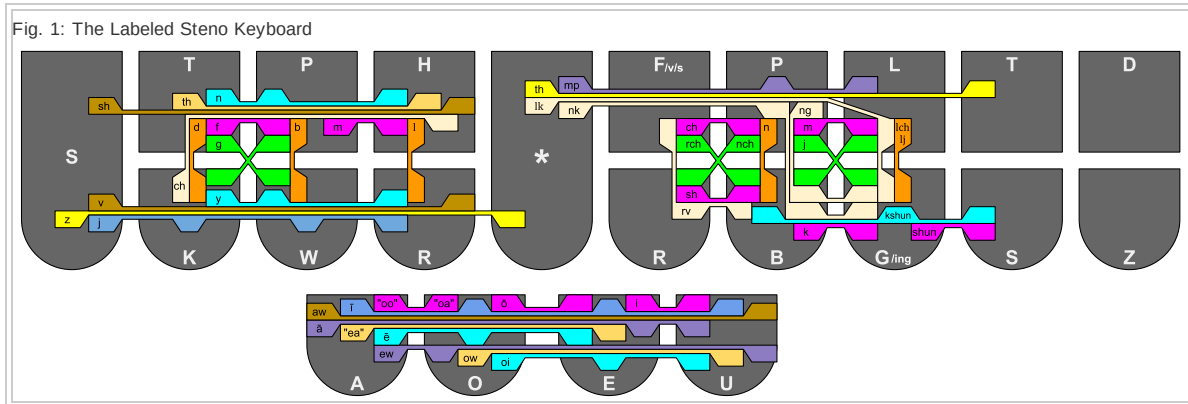
Contents

- 1 Phonetic Strokes
- 2 Fingering
- 3 Fingerspelling
- 4 Steno Order
- 5 Briefs
- 6 Prefix And Suffix Strokes
- 7 Non-Letters
- 8 Numbers

Check out [Mirabai's Plover cheat sheet](#) as well.

Phonetic Strokes

These are the keys and chords for all standard sounds in Plover. For example, to type 'gone', use the stroke **TKPWOPB**.

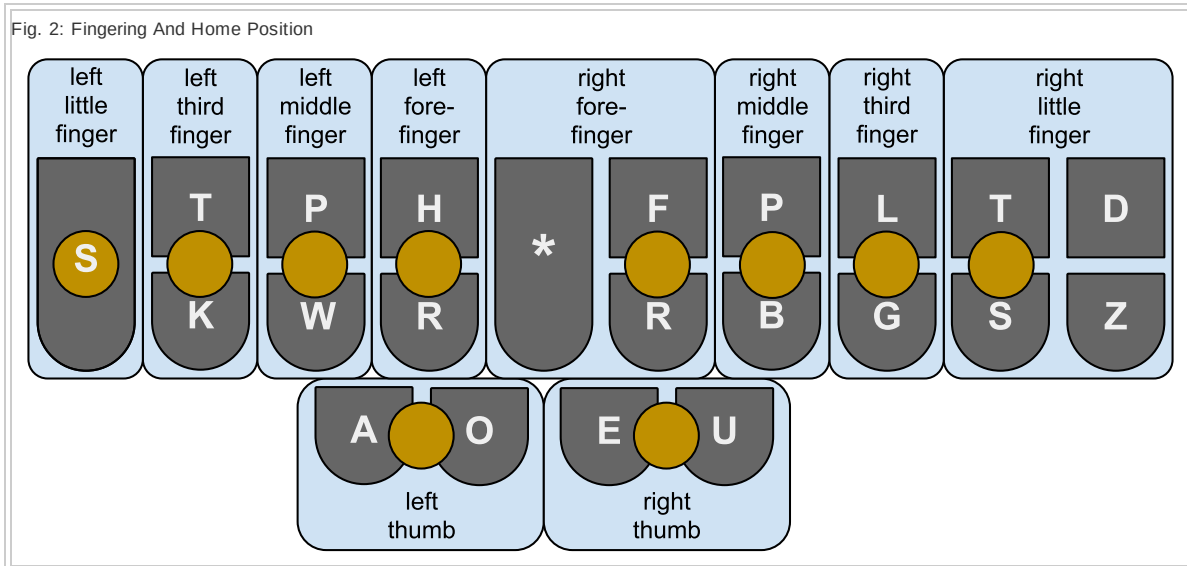


Rules for short vowels using single vowel keys and the **EU** chord:

1. If the vowel letter in the written word appears without another vowel letter next to it, then it doesn't matter which short vowel sound it makes. The Plover key labeled with that letter is *always* used to stroke the word.
2. If the vowel sound in the written word is spelled with two or more consecutive vowel letters, then it doesn't matter what spelling is used. The Plover key whose short vowel sound matches the sound in the word is *always* used to stroke the word.

Fingering

With three exceptions, each finger is responsible for the two keys above and below, and each thumb is responsible for the two keys to the left and right:



Fingerspelling

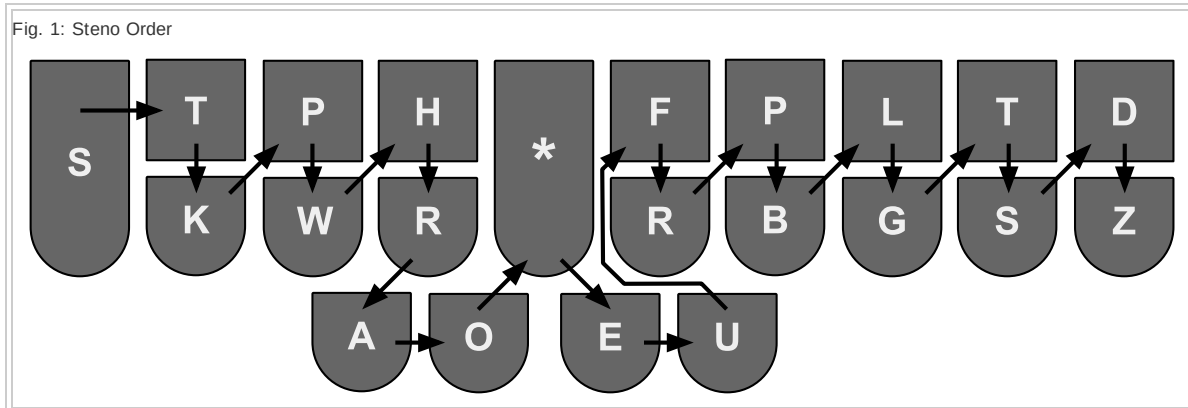
To type a single letter, use the key or chord for that letter on the left side of the keyboard, along with the asterisk. For example, the 'b' sound is stroked **PW**, and the letter "b" is stroked **PW***. To get an uppercase letter, add the **P** key on the right side of the keyboard. The uppercase "B" is stroked **PW*P**.

Here is how to fingerspell the remaining English letters:

- "c": **KR***
- "C": **KR*P**
- "q": **KW***
- "Q": **KW*P**
- "x": **KP***
- "X": **KP*P**
- "z": **STKPW***
- "Z": **STKPW*P**

Steno Order

If the sounds for a word can fit into steno order, then you can type that word with one stroke:



This is equivalent to the string **STKPWHRAO*EUF RPBLGTSDZ** which you should memorize.

Additional ways to try to fit a word into steno order:

- drop unstressed vowel sounds. For example, 'tablet' is stroked **TABL T**
- allow two neighboring sounds to invert steno order, but only once in a word. For example, 'portal' is stroked **PORLT**.

Briefs

Briefs are dictionary entry that violate Plover theory in order to use fewer strokes. These are some of the most commonly used briefs, most of which don't have any phonetic alternative in the dictionary file:

'the': -T	'you're': UR	'very': SRE	'somebody': S-B	'describe': SKREU
'of': F	'you've': UF	'many': PHAEPB	'somehow': SPHOU	'described': SKREUD
'to': TO	'were': WR	'even': AOEPB	'someone': SPH-PB	'anything': TPHEUG
'in': TPH	'can': K	'where': W-R	'someplace': SPHRAEUS	'continue': T-PB
'a': AEU	'there': THR	'through': THRU	'something': S-G	'continued': T-PBD
'is': S-	'been': PWPB	'being': -BG	'sometimes': STAOEUPLS	'beginning': TKPWEUPBG
'that': THA	'if': TP	'because': PWAUS	'somewhere': SWR or SW-R	'understand': URPBD
'with': W	'would': WO	'before': PW-F	'question': KWE	'understanding': URPBGD
'be': B	'who': WHO	'upon': POPB	'almost': HR-PL	'today': TOED
'by': PWEU	'other': OER	'without': WOUT	'interest': T-R or TR	'opinion': P-PB
'he': E	'what': WHA	'another': TPHOER	'ever': -FR	'becomes': PW-BGS
'I': EU	'only': OEPBL	'against': TKPWEPBS	'became': PWAEUPL	'yes': KWRE
'this': TH	'do': TKO	'every': EFR	'probably': PROBL	'idea': KWR-D
'are': R-	'new': TPHU	'within': W-PB	'include': KHRU	'ideas': KWR-DZ
'which': WEU	'about': PW	'example': KP-PL	'includes': KHRUS	'actually': TWAEUL
'have': SR	'two': TWO	'others': OERS	'included': KHRUD	'move': PHOF
'they': THE	'any': TPHEU	'therefore': THR-FR	'including': KHRUG	'ask': SK
'you': U	'could': KO	'having': SR-G	'amount': APLT	'unless': TPH-LS
'you'd': UD	'after': AF	'become': PW-BG	'receive': SAOEF	'easy': EZ
'you'll': UL	'said': SAEUD	'whether': WHR	'received': SAOEFD	'otherwise': O*ERZ

In alphabetical order:

'a': AEU	'being': -BG	'in': TPH	'received': SAOEFD	'understanding': URPBGD
'about': PW	'by': PWEU	'include': KHRU	'said': SAEUD	'understands': URPBDZ
'actually': TWAEUL	'can': K	'included': KHRUD	'somebody': S-B	'unless': TPH-LS
'after': AF	'continue': T-PB	'including': KHRUG	'somehow': SPHOU	'very': SRE
'against': TKPWEPBS	'continued': T-PBD	'includes': KHRUS	'someone': SPH-PB	'were': WR
'almost': HR-PL	'could': KO	'interest': T-R or TR	'someplace': SPHRAEUS	'what': WHA
'amount': APLT	'describe': SKREU	'I': EU	'something': S-G	'where': W-R
'another': TPHOER	'described': SKREUD	'is': S-	'sometimes': STAOEUPLS	'whether': WHR
'any': TPHEU	'do': TKO	'many': PHAEPB	'somewhere': SWR or SW-R	'which': WEU
'anything': TPHEUG	'easy': EZ	'move': PHOF	'that': THA	'who': WHO
'are': R-	'even': AOEPB	'new': TPHU	'there': THR	'with': W
'ask': SK	'ever': -FR	'of': F	'therefore': THR-FR	'within': W-PB
'be': B	'every': EFR	'only': OEPBL	'the': -T	'without': WOUT
'became': PWAEUPL	'example': KP-PL	'opinion': P-PB	'they': THE	'would': WO
'because': PWAUS	'have': SR	'other': OER	'this': TH	'yes': KWRE
'become': PW-BG	'having': SR-G	'others': OERS	'through': THRU	'you': U
'becomes': PW-BGS	'he': E	'otherwise': O*ERZ	'to': TO	'you'd': UD
'been': PWPB	'idea': KWR-D	'probably': PROBL	'today': TOED	'you'll': UL
'before': PW-F	'ideas': KWR-DZ	'question': KWE	'two': TWO	'you're': UR
'beginning': TKPWEUPBG	'if': TP	'receive': SAOEF	'understand': URPBD	'you've': UF

Designing Briefs:

- They should resemble the word somewhat
- Check the dictionary file to see if a brief already exists - no sense reinventing the wheel
- Ask for advice from other steno people. Take your time and get it right
- Briefs should be easy to type and easy to remember
- Avoid strokes that are already used for other words, or for one of the strokes of a multi-stroke word. Reusing the same stroke could lead to word boundary errors
- Sometimes the essence of the word makes a good brief. For example, '**extraordinary**' is stroked **KPRAORD**.
- Sometimes a word has an unusual syllable that can be used as the stroke for the whole word. For example, '**English**' is stroked **TKPWHREURB**, which sounds like '**glish**'.
- Sometimes the whole middle of a word can be skipped, and you can just make a stroke for the first and last parts. For example, '**mechanism**' can use the brief **PHEUFPL**, which sounds like '**mism**'.
- Sometimes to stroke a common phrase, you can use the distinctive parts of each word. For example, '**in other words**' can be stroked **TPHOERDZ**, which sounds like '**nords**'.

- Sometimes you can make a brief that uses more than a single inversion. For example, '**designer**' can be stroked **STKAOEURPB**.
- Sometimes you can make a brief that drops stressed vowels as well as (or instead of) unstressed vowels. For example, the word '**collar**' can be stroked **KHRAR**.
- Sometimes you can make a brief by dropping additional sounds aside from vowels. For example, '**extraneous**' can be stroked **KPRAEUPBS** if you drop the 't' in addition to the unstressed vowels.
- Sometimes if you find a great brief for a word, but it's already used in a dictionary entry, you could add the asterisk and get a brief that's not in use. For example, the word '**herb**' is stroked **HERB**, but the capitalized proper name '**Herb**' can be stroked **H*ERB**.
- Sometimes there are single keys that can stand in for a prefix or suffix. A good example is chording the '**com-**' prefix as the **K** key. For example, you can stroke '**compound**' as **KPOUPBD**, '**compress**' as **KPRES**, and '**combat**' as **KPWAT**. Another good example is chording the '**ex-**' prefix as the **X** key. For example, you can stroke '**exam**' as **KPAPL**, and '**excited**' as **KPAOEUTD**. Another good example is chording the '**-ly**' suffix as **L** (although that could be thought of as simply dropping the unstressed 'y' sound). For example, '**highly**' can be stroked **HAOEUL**, and '**slowly**' can be stroked **SHROEL**. The same technique could be used for other keys and prefixes.
- Sometimes you can use the same keys to mean multiple things in a stroke. For example, the word '**dimple**' needs both the '**mp**' sound and the '**l**' sound at the end of the word. But the '**mp**' sound needs the ***PL** keys, which means the **L** key isn't available for the '**l**' sound. So, just use the **L** key anyway, and stroke **TK*EUPL**. As another example, '**shmeat**' needs both the '**sh**' sound and the '**m**' sound at the start of the word. But the '**sh**' sound needs the **SH** keys, while the '**m**' sound needs the **PH** keys. Both sounds need the **H** key, which would violate Plover theory. So, just use it anyway, and stroke **SPHAO*ER**. (also note that the ***** key is used to disambiguate '**shmeat**' from '**smear**', which is stroked **SPHAOER**).
- Sometimes you can use the **W** key for '**v**' sounds. The '**v**' sound on the left side of the keyboard uses the **SR** keys, which are very popular letters. Sometimes you can squeeze a '**v**' word into a single stroke by using the **W** key instead. For example, '**diverse**' is stroked **TKWERS**, '**divorce**' is stroked **TKWORS**, and '**diversion**' is stroked **TKWERGS**.
- Sometimes you can substitute **OEU** for the vowel sound. Since '**oi**' is the least common English diphthong, you can sometimes use its Plover chord **OEU** in a brief, without conflicting with other words. So taking one or a couple significant parts of a word, and joining them with **OEU**, might make a good brief. For example, '**carbon dioxide**' could be thought of as being exemplified by its '**k**' and '**x**' sounds. Then by joining these with **OEU**, you get the brief for '**carbon dioxide**', which is **KOEUBGS**.

Prefix And Suffix Strokes

Prefix and suffix strokes prevent Plover from getting confused about where to break words. Learn the ones you need, and gradually build your prefix/suffix knowledge in order to increase your typing speed.

Prefixes:

- 'un-': UPB
- 're-': RE
- 'in-': EUPB
- 'im-': EUPL
- 'ir-': EUR
- 'dis-': TKEUS
- 'en-': EPB
- 'em-': EPL
- 'non-': TPHAUPB
- 'over-': AUFR
- 'mis-': PHEUZ
- 'sub-': SAUB
- 'pre-': PRE
- 'inter-': EUPBT
- 'fore-': TPOER
- 'de-': TKE
- 'trans-': TRAPBS
- 'super-': SAOUP
- 'semi-': SPHEU
- 'anti-': A*EUPBT
- 'mid-': PHEUD
- 'under-': UPBD

Some examples:

- 'underwear': UPBD/WAER
- 'misdeed': PHEUZ/TKAOED
- 'embalm': EPL/PWAUPL
- 'disease': TKEUZ/AOEZ
- 'retreat': RE/TRAOET

Suffixes:

- '-s': -S
- '-ed': D
- '-ing': G
- '-ly': HREU

- '-er': *ER
- '-or': O*R
- '-tion': SH*UPB
- '-ation': AEUGS
- '-ion': KWROPB
- '-able': BL
- '-al': A*L
- '-ial': KWRAL
- '-ness': -PBS

Some examples:

- 'happily': HAP/HREU
- 'cider': SAOEUD/*ER
- 'relation': REL/AEUGS

Prefix/Suffix alternatives are slow and don't address all cases. Don't rely on them. Learn the appropriate prefix or suffix stroke when you can. But here they are anyway:

- **S-P**: insert a space after the current word. This prevents it from being included in a multi-stroke word
- **TK-LS**: prevent a space after the current word. This forces Plover to output compound words

Non-Letters

If any of these are missing from your dictionary file, add an entry for them:

- `: **KH-FG** (Example: 'files = `bin/lis`' is stroked **KWA*LS KH-FG OEU PWEUPB OEU HR* S* KH-FG**)
- ^: **KR-RT** (Example: 'x^5 is equal to x^2 times x^3.' is stroked **KP* KR-RT #A S KWAL TO KP* KR-RT #T- TAOEUPLS KP* KR-RT #P-**)
- ~: **T*LD**. The tilde with, a space after it. (Example: 'Yaaay~ baby!' is stroked **KWR* A* A* A* KWR* KWR* T*LD PWAEUB/KWREU SKHRAPL**)
- <: **AEPBGT** (Example: 'x<10' is stroked **KP* AEPBGT #SO**)
- =: **KWA*LS** (Example: 'x=10' is stroked **KP* KWA*LS #SO**)
- >: **A*EPBGT** (Example: 'x>10' is stroked **KP* A*EPBGT #SO**)
- |: **PAO*EUP**. The vertical bar, with no space after it. (Example: 'Is -R | less' is stroked **HR* S* S-P H-PB R*P S-P PAO*EUP S-P HRES**)
- _: **R*UPBD**. An underscore character, with no space after it. (Example: 'camelCase vs. under_scores` is stroked **KAPL/EL KPA* KAEUS SR* S* P-P S-P TPH-P R*UPBD SKORS**)
- -: **H-PB**. A single hyphen, with no space after it. (Example: 'It was a bit-coin moment' is stroked **EUT WAS AEU PWEUT H-PB KOEUPB PHOEPLT**)
- --: **TK-RB**. An em-dash (i.e. two hypens), with a space after it. (Example: 'Wait -- I'm going too' is stroked **WAEUT TK-RB AOEUPL TKPWOG TAO**)
- ,: **KW-BG**. A comma, with a space after it. (Example: 'Be good, and be kind' is stroked **-B TKPWAOD KW-BG APBD -B KAOEUPBD**)
- ;: **SKWR*RBGS**. A semicolon, with a space after it, and the next word uncapitalized. (Example: 'Be good; be kind' is stroked **-B TKPWAOD SKWR*RBGS KAOEUPBD**)
- ': **STPH-FPLT**. A colon, with a space after it, and the next word is capitalized. (Example: 'Plover: The winnah' is stroked **PHRO*FR STPH-FPLT W* EU* TPH* TPH* A* H***)
- ': **KHR-PB**. A colon, with no space after it, and the next word is uncapitalized. Primarily used for specifying clock time. (Example: '24:00' may be stroked **#TH KHR-PB #0D**)
- !: **SKHRAPL**. An exclamation point, with a space after it, and the next word is capitalized. (Example: 'Why? Because!' is stroked **KWR H-F BWAUS SKHRAPL**)
- ? : **H-F**. A question mark, with a space after it, and the next word is capitalized. (Example: 'Why? Why not?' is stroked **KWR H-F KWR PHOT H-F**)
- /: **OEU** (Example: 'x/y=10' is stroked **KP* OEU KWR* KWA*LS #SO**)
- .: **TP-PL**. A regular period, with a space after it, and the next word is capitalized. (Example: 'There it is. Better' is stroked **THR EUT S TP-PL PWET/ER**)
- .: **P-P**. A period used as a decimal point, with no space after it, and the next word uncapitalized. (Example: 'It's.here.for.you' is stroked **T-S P-P HAOER P-P TPOR P-P U P-P**)

- '...' **SKWR-RBGS**. Ellipses, with a space after it, and the next word is capitalized. (Example: 'No... Just no...' is stroked **TPHO SKWR-RBGS SKWRUFT TPHO SKWR-RBGS**)
- "'": **A*E**. Open single quote, with no space after it.
- '"': **AE**. Close single quote, with a space after it. (Example: 'Just say 'hi' and see what happens!' is stroked **SKWRUFT SAEU A*E HEU AE APBD SAOE WHA HAPS SKHRAPL**)
- '""': **KW-GS**. Open double quote, with no space after it.
- '"""': **KR-GS**. Close double quote, with a space after it. (Example: 'I think "love" can be hard to say.' is stroked **EU THEU KW-GS HROF KR-GS K -B HARD TO SAEU TP-PL**)
- '(': **PREPB**
- ')': **PR*EPB** (Example: 'I (love you) want some milk.' is stroked **EU PREPB HROF U PR*EPB WAPBT SOPL PH*EULG TP-PL**)
- '[': **PWR-BGT**
- ']': **PWR*BGT** (Example: 'I luv [sic] you.' is stroked **EU HR* *U SR* PWR-BGT S*EUBG PWR*BGT U TP-PL**)
- '{': **TPR-BGT**
- '}': **TPR*BGT** (Example: 'if(x) { print "hello, world" }' is stroked **TP PREPB KP* PR*EPB TPR-BGT PREUPBT KW-GS HEL/HRO KW-BG WORLD KR-GS TPR*BGT**)
- '@': **KWRAT** (Example: 'zacharyb@gmail.com` is stroked **STKPW* A* KR* H* A* R* KWR* PW* KWRAT TKPW* PH* A* EU* HR* P- P KR* O* PH***)
- '\$': **TK-PL** (Example: 'I got the \$\$\$\$. You got the stuff?' is stroked **EU TKPWOT -T TK-PL TK-PL TK-PL TP-PL U TKPWOT -T STUF H-F**)
- '*': **STA*R** (Example: 'Love you babe :-* :-*' is stroked **HRUF U PWAEUB KHR-PB H-PB STA*R S-P KHR-PB H-PB STA*R**)
- '\': **SPWHRAERB** (Example: 'C:\' is stroked **KR*P KHR-PB SPWHRAERB**)
- '&': **SP-PBD** (Example 'U R my <3 & soul' is stroked **U*P S-P R*P PHAOEU AEPBGT #P- S-P SP-PBD SOUL**)
- '#': **HAERB** (Example: 'You're my #1 guy!' is stroked **UR PHAOEU HAERB #S TKPWAOEU SKHRAPL**)
- '%': **PERS** (Example: 'What % is that? 5%?' is stroked **WHA PERS S THA H-F #A PERS H-F**)
- '+': **PHR*US** (Example: 'y=x+10` is stroked **KWR* KWA*LS KP* PHR*US #SO**)

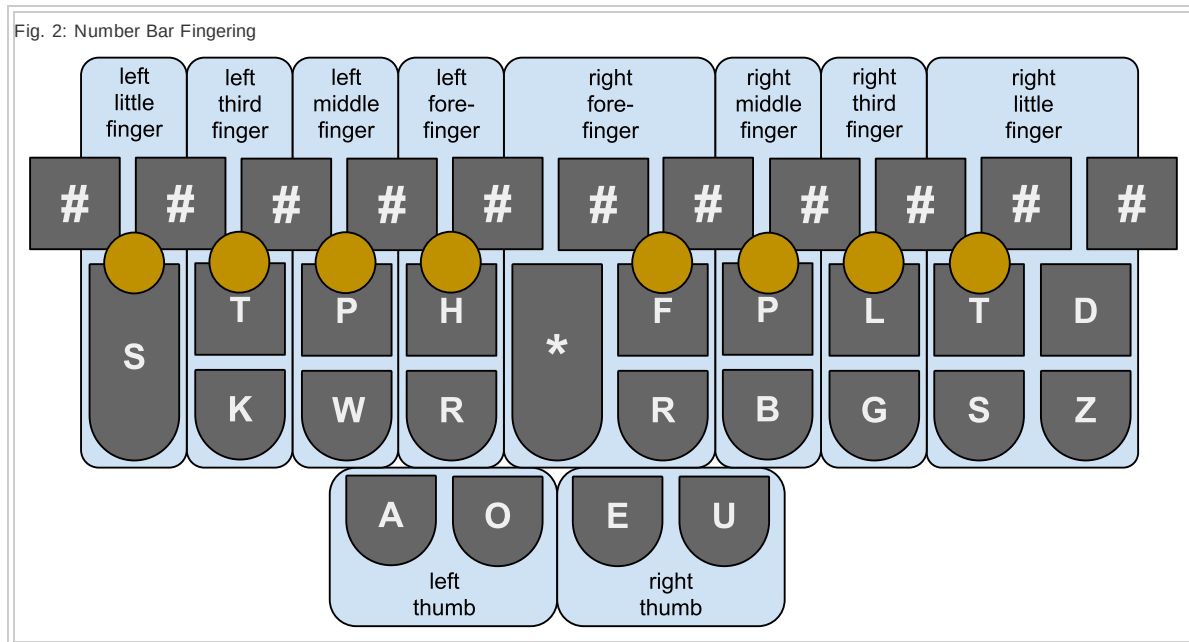
Useful strokes:

- Carriage Return: **R-R**
- New paragraph. Insert two carriage returns and capitalize the next word: **SKWRAURBGS**
- Undo stroke: *****
- Letter-by-letter backspace: **PW-FP**
- Insert a space: **S-P**
- Insert a space and capitalize the next word: **KPA**

- Do not insert a space before the next word, and don't capitalize the next word: **TK-LS**
- Do not insert a space before the next word, and capitalize the next word: **KPA***

Numbers

Keyboard fingering is different when typing numbers:



These are the strokes to type each of the ten digits:

- '1': #S
- '2': #T-
- '3': #P-
- '4': #H
- '5': #A
- '0': #O
- '6': #F
- '7': #-P
- '8': #L
- '9': #-T

Notice that the '0' is right in the middle, in order to match up with the **O** key. With that exception, also notice that the list of numbers is in steno order: **STKPWHRAO*EUFRPBLGTSZ**.

To type a two digit number, stroke two single digit numbers together:

- '25': #TA
- '07': #OP
- '49': #HT
- etc.

In their unmodified form, two digit numbers always match the steno order of the keys used to stroke them. To invert that order, add the **EU** chord:

- '52': #TAEU
- '70': #OEUP
- '94': #HEUT
- etc.

To have the second digit be the same as the first, use the single digit that you want, but add the **D** key to the stroke:

- '11': #SD
- '22': #T-D
- '33': #P-D
- '44': #HD
- '55': #AD
- '00': #OD
- '66': #FD
- '77': #-PD
- '88': #LD
- '99': #-TD

When you stroke one number after another, Plover does not add a space between them. It's possible to type a long number by stroking all the parts of that number in turn:

- '1068': #SO/#FL
- '700039': #OEUP/#OD/#PT
- '81254583': #LS/#TA/#HA/#PEUL
- etc.

Sequences of digits that are all in steno order may be typed all at once. For example, '25689' may be stroked as #TAF~~L~~T. However, when stroking more than two digits this way, the **D**, **EU**, and other modifiers don't work.

There are typically many different ways to stroke a long number. For example, '1068' may also be stroked as #S/#O/#F/#L, or #SOFL. It's up to you.

If you want to add a double zero to a single digit number, you could use the #OD stroke after the number, for example stroking '100' as #S/#OD, or you could include the **Z** key in your stroke.

- '100': #SZ
- '200': #T-Z
- '300': #P-Z
- '400': #HZ
- '500': #AZ
- '600': #FZ
- '700': #-PZ

- '800': #LZ
- '900': #-TZ

To prefix a single digit number with the '\$' character and add a double zero after it, add the **DZ** chord to the number stroke:

- '\$100': #SDZ
- '\$200': #T-DZ
- '\$300': #P-DZ
- '\$400': #HDZ
- '\$500': #ADZ
- '\$600': #FDZ
- '\$700': #-PDZ
- '\$800': #LDZ
- '\$900': #-TDZ

To express a one or two digit number as clock time, add either **K** or **BG** to the stroke:

- '1:00': #SK or #SBG
- '2:00': #TK or #TBG
- '3:00': #KP or #PBG
- '4:00': #KH or #HBG
- '5:00': #KA or #ABG
- '6:00': #KF or #FBG
- '7:00': #K-P or #-PBG
- '8:00': #KL or #BLG
- '9:00': #KT or #BGT
- '10:00': #SKO or #SOBG
- '11:00': #SKD or #SBGD
- '12:00': #STK or #STBG
- '13:00': #SKP- or #SP-BG
- '14:00': #SKH or #SHBG
- '15:00': #SKA or #SABG
- '16:00': #SKF or #SFBG
- '17:00': #SK-P or #S-PBG
- '18:00': #SKL or #SBLG
- '19:00': #SKT or #SBGT
- '20:00': #TKO or #TOBG
- '21:00': #STKEU or #STEUBG
- '22:00': #TKD or #TBGD
- '23:00': #TKP- or #T-PBG
- '24:00': #TKH or #THBG

Glossary

Brief: *n.* A dictionary entry that expresses a word, phrase, or other sequence of characters using fewer keystrokes than the particular steno theory requires. Typically, steno users develop and maintain a large corpus of briefs, to suit their most commonly used words and phrases.

Chord: *n.* A group of keyboard keys that mean something when stroked together. A chord typically indicates a phoneme, group of phonemes, word, or phrase. A single stroke typically includes some number of chords and individual keys pressed simultaneously.

Dictionary: *n.* A text file containing words and their corresponding steno keyboard combinations.

Fingerspell: *v.* To type letter-by-letter using Plover strokes.

Mirabai Knight: A professional stenotypist; founder of the Plover open source project.

Misstroke: *n.* This occurs when you press the wrong key or the wrong chord. Misstrokes are often included as dictionary entries for the intended word. This way, even when you type a word wrong, Plover outputs it correctly.

N-Key Rollover: A feature of some keyboards that let you press any number of keys at the same time, and have the computer operating system receive a correct set of keypresses from the keyboard. When using Plover with a qwerty keyboard, Plover only works if the keyboard supports n-key rollover.

NKRO: see **N-Key Rollover**.

Outline: *n.* A dictionary entry. Typically used to distinguish between multiple entries for the same word. "I use several different outlines to cover misstrokes."

Plover: *n.* An open source stenographic typing system that interprets chorded keystrokes from n-key rollover qwerty keyboards and performs a dictionary lookup to determine the output of each stroke. Typing speeds of 250 words per minute have been demonstrated using Plover.

Stroke: *n.* A group of keyboard keys that are typed simultaneously to produce output in the form of a word, part of a word, or group of words. A stroke is typically made up of some number of chords.

v. To press and release a group of keyboard keys in order to produce a word or phrase.

Theory: *n.* Specific rules for spelling or sounding out words on a steno keyboard. If you understand a particular steno theory, then you can figure out the keypresses to type most or all words in the dictionary.

Ward Stone Ireland: The inventor of stenotype. Ireland's keyboard and accompanying theory are still the basis of modern stenotype systems.

Kinglet

1	Introduction
1.1	Try Kinglet
1.2	The Basics
2	The Nine-Key Alphabet
3	The Six-Key Alphabet
4	The Thumb Alphabet
4.1	Thumb Spaces
5	Fingering
5.1	QWERTY
5.2	Termination Conflicts
6	Capitalization, Punctuation, And Numbers
6.1	The Cap Chord
7	Dictionary Lookup
8	Cheat Sheet

NOTE: Kinglet was an attempt to create a spelling-based steno system that would have a theoretical speed equal to Plover, i.e. about 1 word per stroke of the keyboard. It does succeed in this. However, this success comes at the cost of a high degree of finger contortion and accuracy in each individual finger. While I attempted to reduce these issues as much as possible, they do remain significant problems in using the system. I therefore have to regard kinglet as a theoretical success, but a practical failure. A success because two robots with equal dexterity would type at the same speed using either kinglet or plover; and a failure because humans are not robots, and the requirements for actual use make kinglet too difficult for humans.

I also regard kinglet as strong evidence that no spelling-based steno system can be both speed-competitive and dexterity-competitive with plover. Please view this as a challenge, and prove me wrong by either proposing modifications to kinglet, or by designing your own system that accomplishes these things.

Introduction

Kinglet is a stenotype system in which chords are used to directly spell out words instead of relying on phonetics, mnemonics, or word lookup tables. With Kinglet, you type out the exact characters you want to see, typically much faster than you would with QWERTY.

An exact speed comparison with Plover would be difficult because the two systems treat words differently. Plover supports briefs that can match words of any size, while Kinglet requires spelling out each word individually. However, on average, and taking account of word frequency statistics for the English language, both systems output roughly one word per stroke. Because of that, they should end up being approximately the same speed, with Plover being faster for certain things (legal and medical transcriptions), and Kinglet being faster for others (names and other unusual strings of text).

One advantage that Kinglet has over traditional stenotype systems is the relative ease of learning. To get good at Plover, you need to memorize hundreds of briefs, prefix, and suffix strokes. With Kinglet, each finger-pair has a simple chart of 48 chords (144 altogether), a few rules regarding fingering and chord ordering, and that's it.

Kinglet is also useful as a programmer's stenotype system because it produces exactly the output you want. Traditional steno systems frequently back up and replace existing output with something else, which makes it hard to give commands to tools like vim and emacs.

The difficulty with Kinglet - which traditional steno systems don't suffer from - is the fingering. Kinglet's chords are more complex than Plover's, and the fingers need to cultivate a higher degree of dexterity and accuracy in order to type fast and avoid misstrokes. With Plover, the bulk of one's study involves memorization. With Kinglet, it involves practicing fingering techniques.

Try Kinglet

I designed Kinglet with help and advice from the Plover community; and Benoit Pierre has implemented a working version, available at github. It should work on either Linux or Windows. A Windows build is available at <https://drive.google.com/file/d/0B9Je8TDYCh1xSVRjVmQ0RUF0ejA>. Please send feedback and bug reports to the Plover mailing list, at ploversteno@googlegroups.com.

To retrieve and run the Kinglet code under Linux, give the following commands:

```
$ git clone https://github.com/benoit-pierre/plover.git
$ cd plover
$ git checkout kinglet
$ ./launch.py
```

You may need to install any missing python dependencies that cause `launch.py` to give an error.

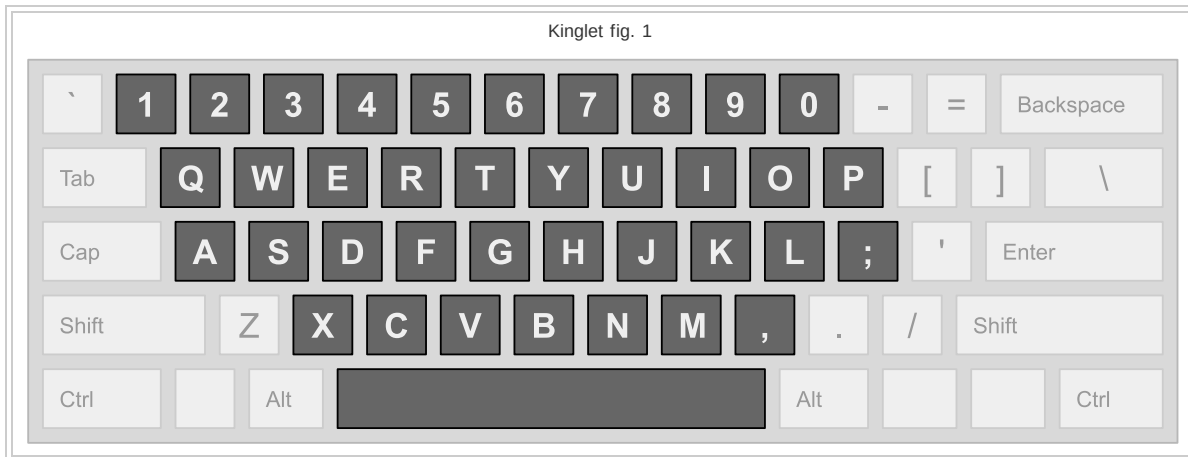
You'll also need an n-key rollover keyboard, of the same sort that [work with Plover](#).

The Basics

With Kinglet, the QWERTY aspect of the keyboard still works - if you hit a single key in Kinglet, you'll get the output of that key, just as you'd get on a normal keyboard. So there's no need to switch modes the way there is in Plover. In fact, Kinglet *relies* on the QWERTY behavior for numbers and some punctuation. This isn't so much worse than traditional steno systems, which require spending a whole stroke on each punctuation mark.

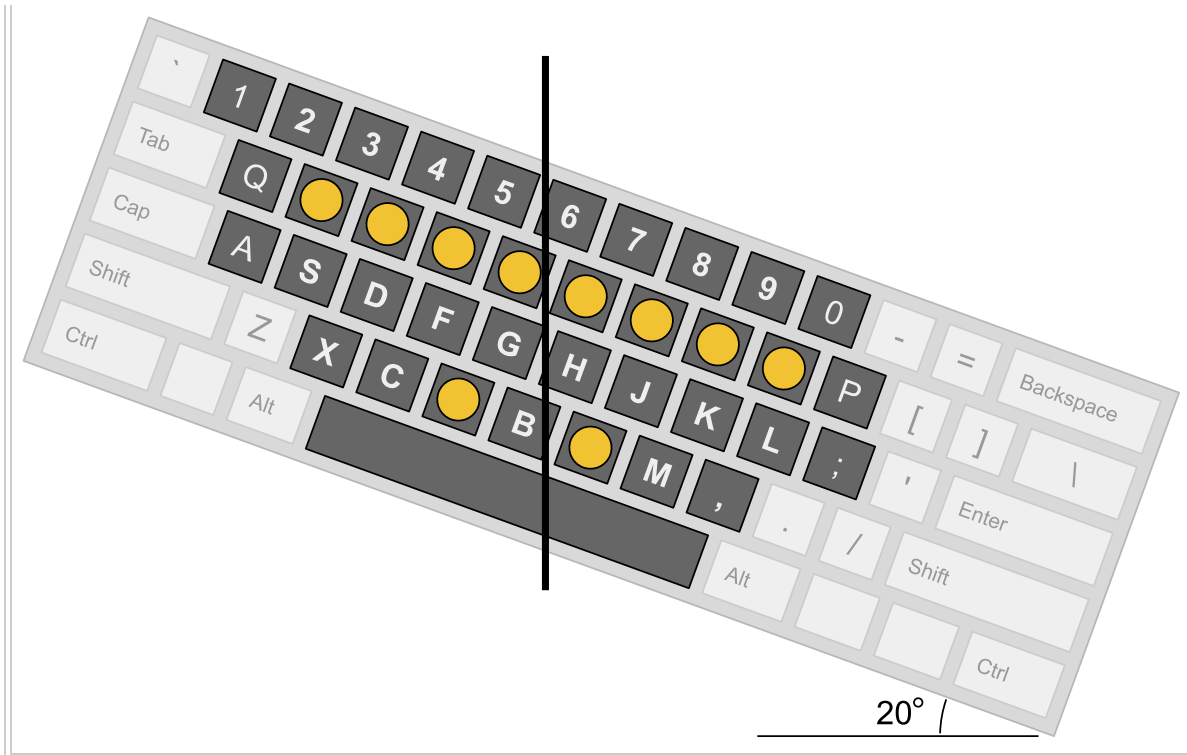
When typing chords in Kinglet, each pair of fingers - which I'll get specific about in a moment - can type a single letter of the alphabet, certain two-letter combinations, a selection of punctuation, the caps shift, or a space. So, with four pairs of fingers and one pair of thumbs, you can type a bit more than five characters per stroke, on average.

Kinglet uses the following keys on the QWERTY keyboard:

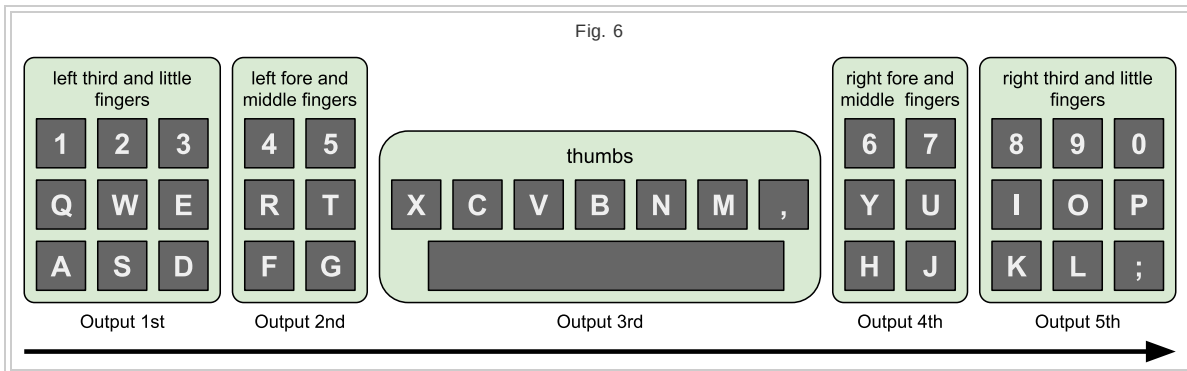


Until Kinglet has a suitable hardware keyboard, try keeping the keyboard at a 20° angle to keep the chord fingering more vertical. As with QWERTY and Plover, each finger has a home position. Keep them near their home positions so that typing becomes more a question of muscle memory than of feeling around for the right key:





Kinglet theory splits the keyboard down into five sets of keys, where each set maps to an assigned finger-pair. As with Plover, the chords on the left produce the first output, and the chords on the right produce the last.



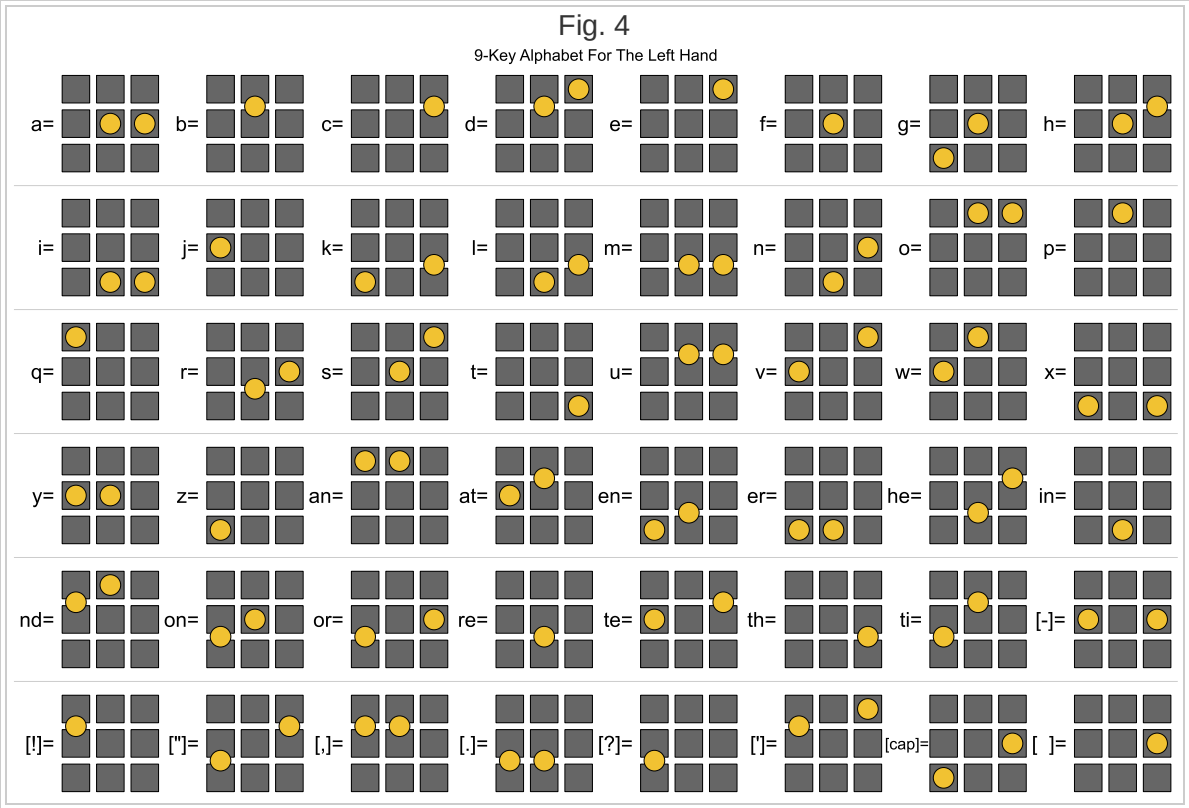
When just starting out, if you find it easier, you can leave out the third and little fingers of each hand, and just stroke chords with the rest. That way for any short chords that you want to type (and there will be plenty), you can use those fingers and not the smaller, weaker ones.

Kinglet does have an equivalent string of characters that corresponds with Plover's concept of 'steno order', but you typically won't need to know it. The string is constructed from Kinglet's finger-pair order, like this: **1QA2WS3ED4RF5TG []XCVBNM,6YH7UJ8IK9OL0P;**. Unlike Plover, there's no need for you to memorize it. The only reason to know the text string itself is if you want to write down Kinglet strokes for tutorials and whatnot.

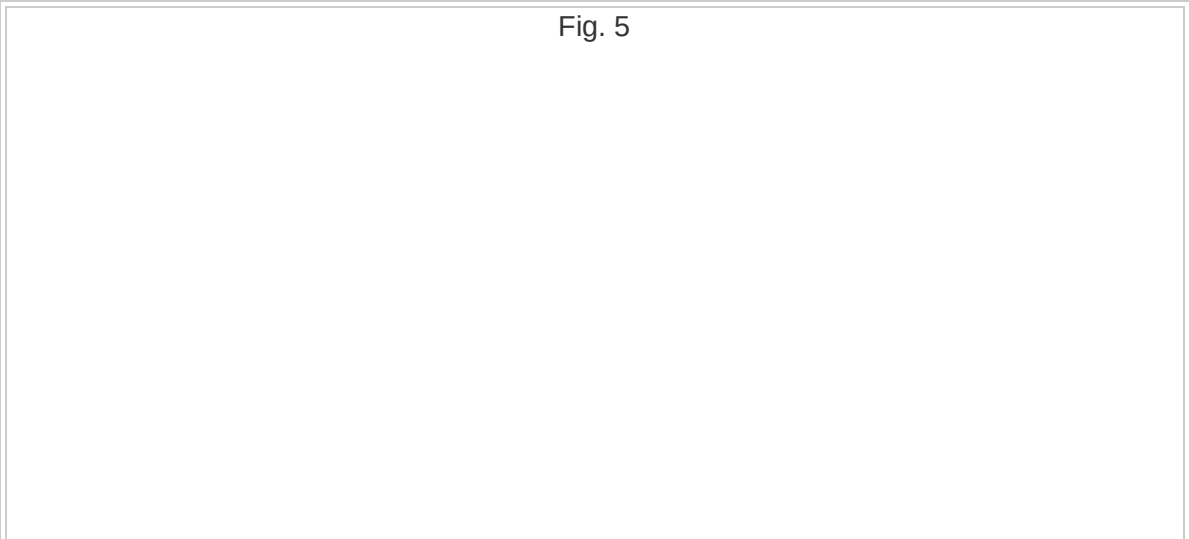
The Nine-Key Alphabet

The nine-key alphabet is a set of chords used by the third and little fingers on both hands to produce any single letter, space, a few common letter-pairs, and the most common punctuation marks (shown here inside square braces).

The nine-key alphabet for the left hand:



The nine-key alphabet for the right hand is the mirror image of the chords for the left:

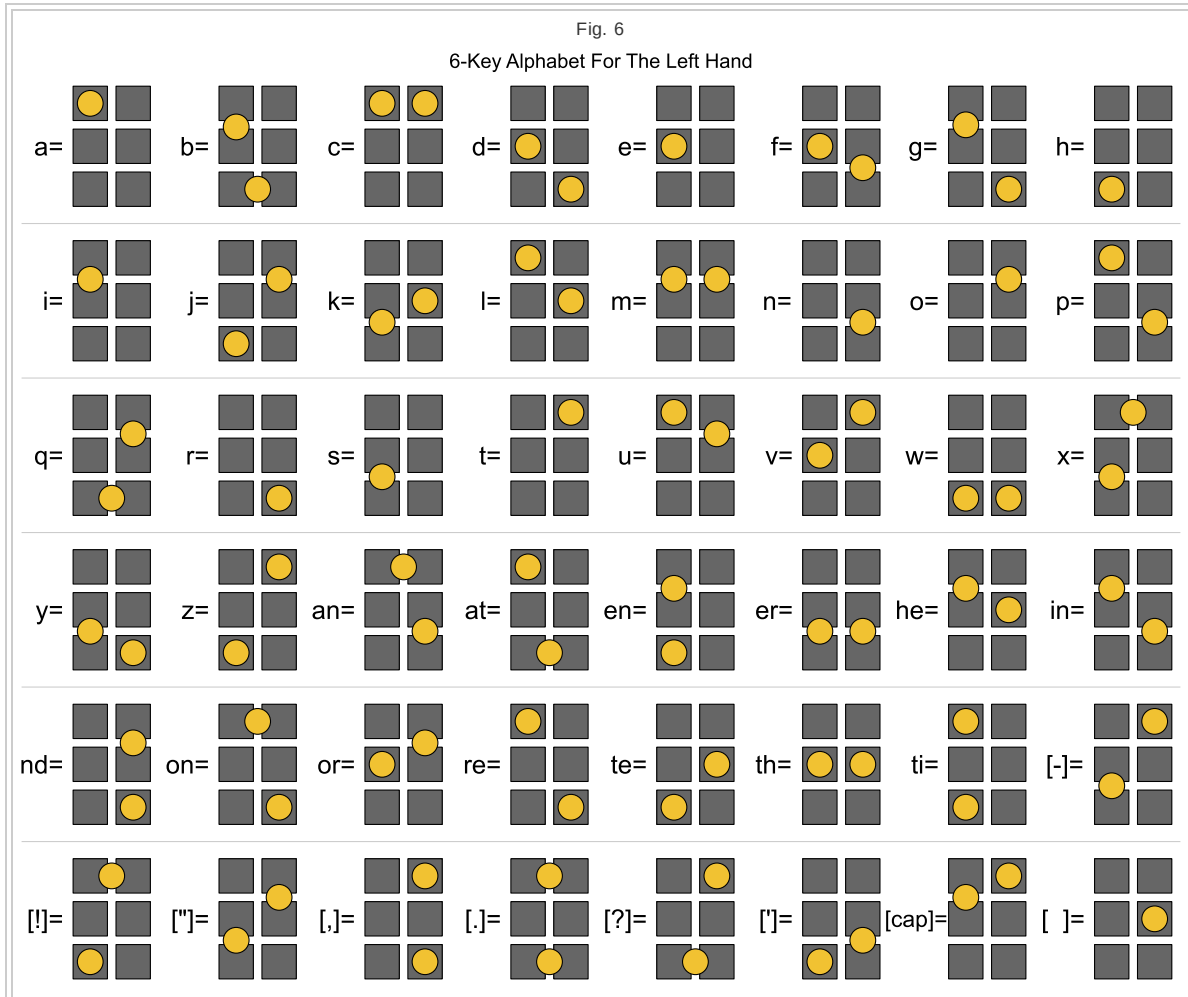


9-Key Alphabet For The Right Hand

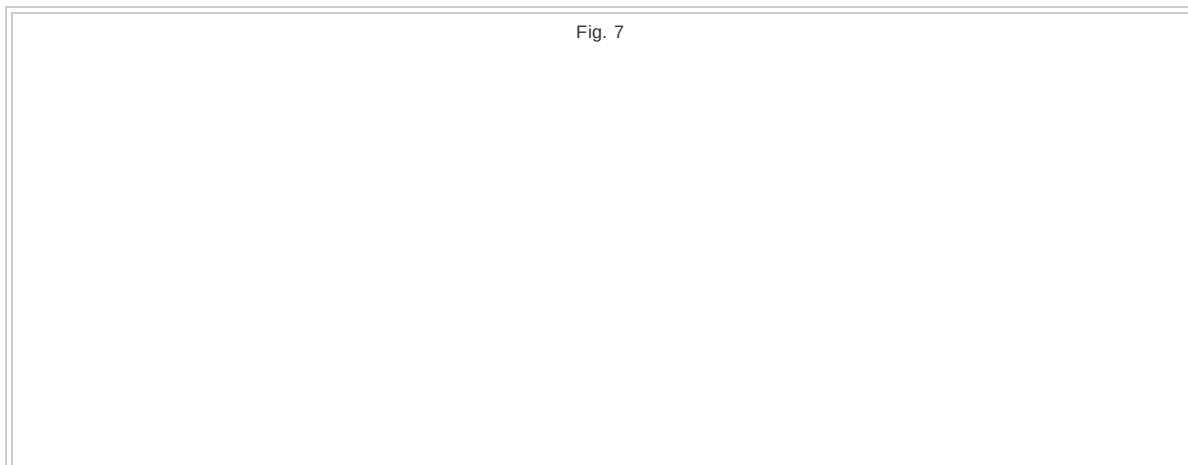
a=	b=	c=	d=	e=	f=	g=	h=
i=	j=	k=	l=	m=	n=	o=	p=
q=	r=	s=	t=	u=	v=	w=	x=
y=	z=	an=	at=	en=	er=	he=	in=
nd=	on=	or=	re=	te=	th=	ti=	[-]=
[!]=	["]=	[,]=	[.]=	[?]=	[']]=	[cap]=	[]=

The Six-Key Alphabet

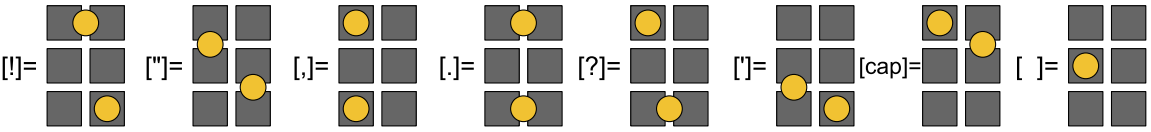
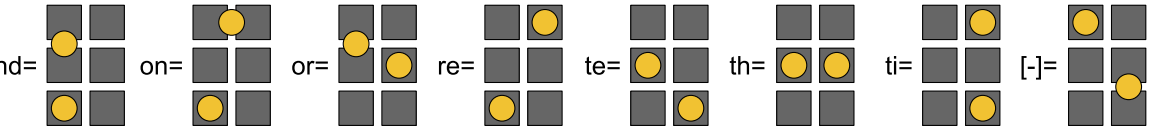
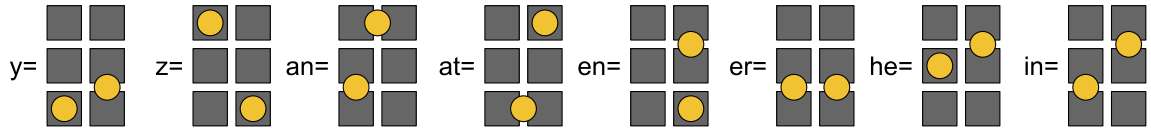
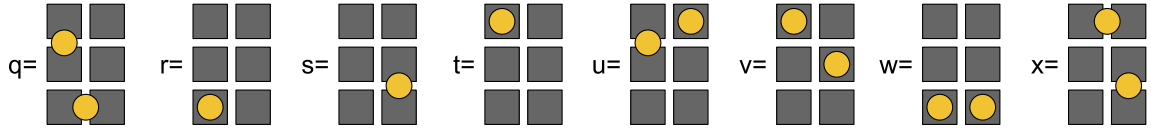
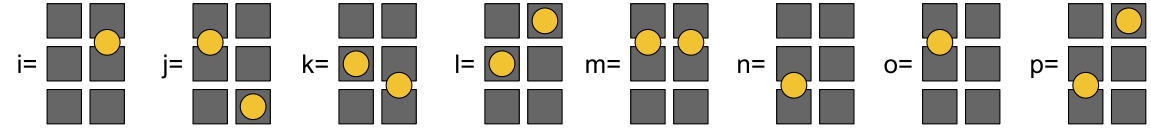
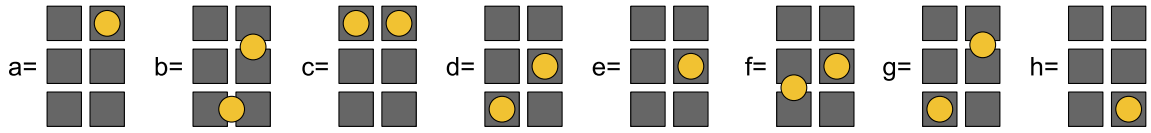
The six-key alphabet is the set of chords for the first and second fingers on either hand:



The six-key alphabet for the right hand is a mirror image of the chords for the left:

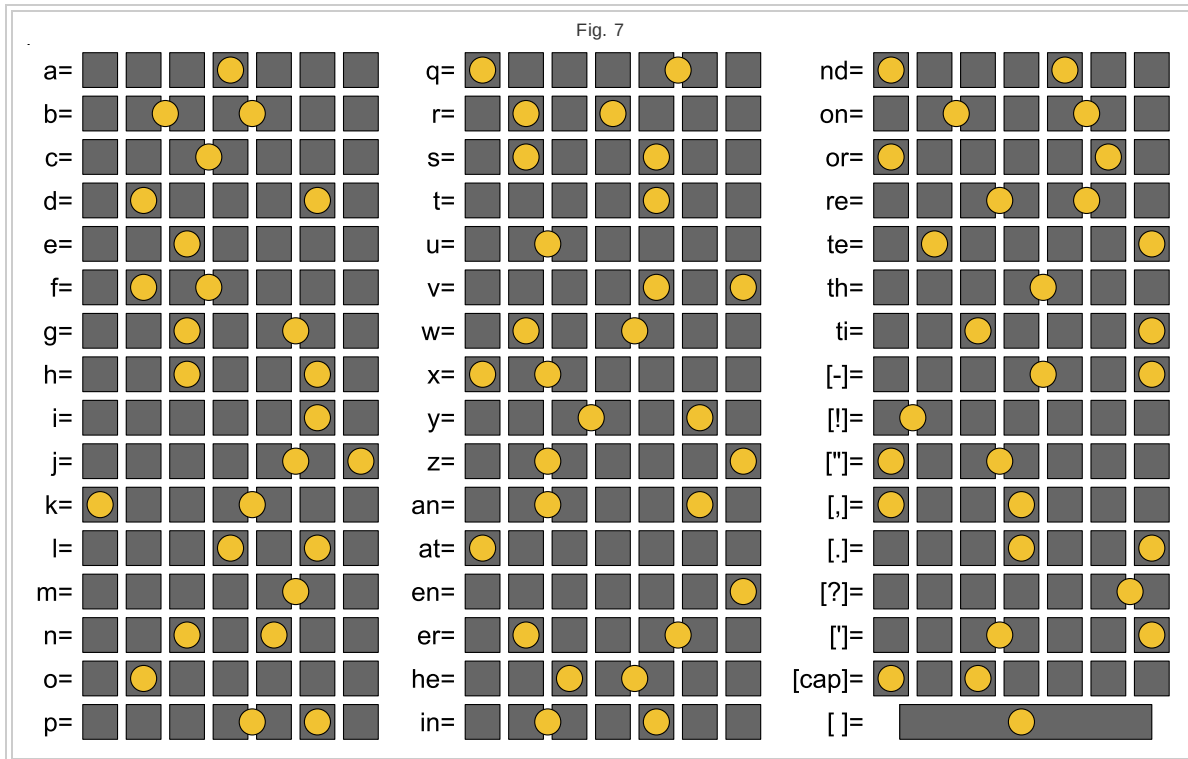


6-Key Alphabet For The Right Hand



The Thumb Alphabet

The thumbs are given their own set of chords in Kinglet, partly because they're not able to produce all the chords in the six-key alphabet, and partly because different chords are easier for the thumbs than for the fingers, and the most-used output characters should always be assigned to the easiest chords:



Notice that there is no empty chord for any finger-pair. This is because if a finger-pair does not strike a chord, then that pair produces no output in that stroke, and those two fingers are simply ignored by Kinglet.

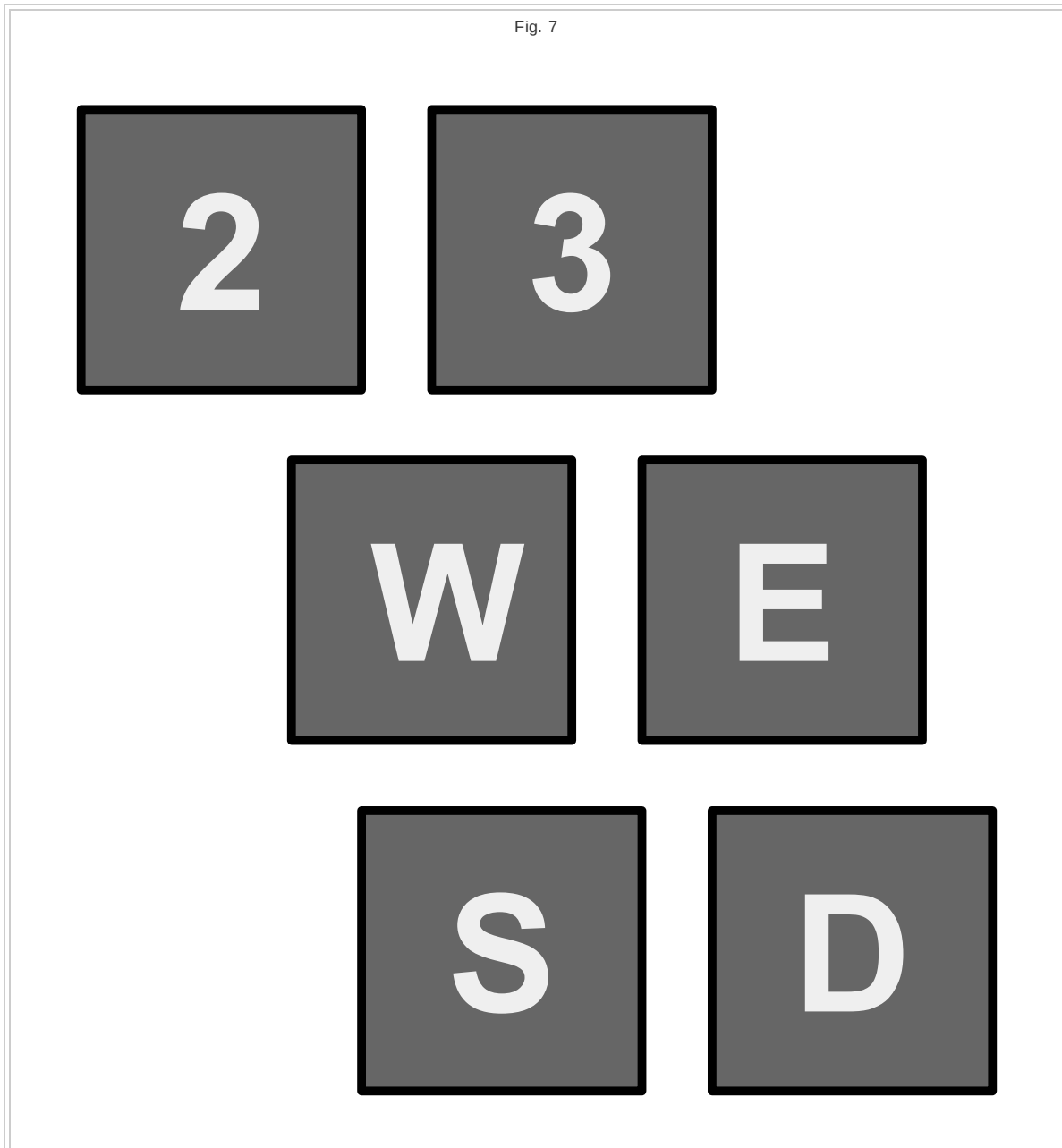
Thumb Spaces

The space character is produced by the using the spacebar. When the thumbs use it by itself, it produces just a space. When the thumbs add the spacebar to any other chord, a space character precedes the output of that chord.

So to type 'e', the thumbs would use **V**. But to type ' e' (a space with an "e" after it), the thumbs would use **[]V**.

Fingering

Since thumb positioning is obvious, let's address only the rest of the fingers. These rules take account of the actual key positions on the keyboard, not the abstract key columns shown in the chord charts:



Here are the fingering rules:

1. **Earlier rules take precedence over others**

2. **Rules apply to finger pairs, not to all neighboring fingers** - the third and little fingers are a pair; the fore and middle fingers are a pair; the middle and third fingers are not.
3. **Don't cross fingers** - a finger that is biologically to the right of another, should never take a position that is physically to the left of that finger.
4. **Keep outer fingers below inner fingers** - the left little finger should be below the left third finger; the right little finger should be below the right third finger; the right fore finger should be below the right middle finger; the left fore finger should be below the left middle finger.

QWERTY

The behavior of the keys is slightly different when typing QWERTY:

- With normal QWERTY keyboards, the symbol appears on screen as soon as you press the key. With Kinglet, the symbol appears as soon as you *release* the key. Kinglet assumes that when you press a key, it could be part of a chord. It's only after you release it that Kinglet can be sure you intended to strike only that one key.
- Autorepeat is disabled. This is again because Kinglet has to wait until you release the key, to know if it was part of a chord.

Termination Conflicts

Sometimes you'll come to the end of a thought, and the final output will only require a stroke that has a single key in it. This produces a conflict because Kinglet has to assume that any single key strokes are intended to be interpreted as QWERTY typing. For example, if you try to stroke **'stripe'** as **W35CB7U9/3**, you'll produce **'strip3'** because Kinglet interprets the **3** as a single keystroke on the QWERTY layout.

There are several ways to deal with this:

- **Preferred approach:** If you find you've arrived at a single-key stroke, just leave it out. So to get **'stripe'**, just stroke **'strip'** without the final 'e', as **W35CB7U9**. Then when you're ready to type more, finish the word in the first part of the following strokes, e.g. **'e and'** as **W3TCVMHU**.
- If that technique doesn't appeal, you can just type the QWERTY character you need. So **'stripe'** would be stroked as **3TCVMHU/e**.
- You could also avoid single-key strokes by migrating a character from the current stroke onto the next. So you could stroke **'stripe'** as **'stri'** and **'pe'**, with **W35CB7U/2R**.
- If you know you're in the middle of a sentence, you could also just add a space to the single-key stroke, to give it another key. Then you could stroke **'stripe'** (with a space) as **W35CB7U9/3T**.

The reason the first option is preferable is because it maximizes your typing speed. All the other options involve rethinking your chord selection. Only the first option lets you get the biggest output from a stroke, without having to change what that stroke was going to be.

Note that some examples in this text have a termination conflict. The assumption is that you'll resolve them using one of the techniques described above. Termination conflicts are not things to be avoided; they're just part of typing.

Capitalization, Punctuation, And Numbers

For all numbers, and for any punctuation not included in the six-key alphabet, use QWERTY. With Kinglet, you always have the option of using the QWERTY keyboard without chords, to type the single characters printed on each key.

The Cap Chord

The above chord chart identifies one chord as "cap". It behaves as a cross between SHIFT and CAPSLOCK.

- If CAPSLOCK is already on, the "cap" chord turns it off, and subsequent letters are output in lowercase.
- If CAPSLOCK is off, using the "cap" chord causes the next single letter to be uppercase, and subsequent letters to return to lowercase.
- - 'I'm Zack.': **AE4RVB,6Y7UI/AEF5B67IK;/QAWS**
 - 'camelCaseVar': **3E4NMUIKL/AE45BUJ8/AER5BH**
- If CAPSLOCK is off, using the "cap" chord by two neighboring finger-pairs turns CAPSLOCK on, and subsequent letters will be output in uppercase.
- - 'Kinglet's
reDONKulous': **AERFTCVNH7UIKL/35VB,UJI/WS4R5XVHUOP;/AED4R5CVY78
9/2W3ERF**

Dictionary Lookup

Here's a Perl script I wrote (see the attachment below), that will take a bit of text as input, and produce a sequence of Kinglet strokes as output. This is how I created the examples for this document. The script also tells how many characters per stroke and strokes per word you're getting. Kinglet needs to average at most one stroke per word, or at least 6.1 characters per stroke, in order to be competitive with traditional steno.

Here are some sample uses. To get **'hello'**:

```
$ ./kinglet.pl hello
WS3E4TBM6Y
chars per stroke: 6.25
strokes per word: 1
```

To get **'where have all the flowers gone?'**

```
$ ./kinglet.pl "where have all the flowers gone?"
Q24RTVBNMY8I0/WER5VYI0/SED4T[ ]BNUI/W4TCHJL;/W3TVNM6H78/QA
chars per stroke: 6.15
strokes per word: 1
```

To indicate the "cap" chord, just use capital letters as you would normally. So to get **'I prefer 'ck' for Zack, and 'ch' for Zachary. Is that weird?'**, you'd do this:

```
$ ./kinglet.pl "I prefer 'ck' for Zack, and 'ch' for Zachary. Is that weird?"
AE4R[ ]BNMH70/ASTVB,67IK;/1Q3TCVB6YUI/AEF5B67IK;/1Q2WTCVMHUI/1Q345VMYHJI/WR5T[
]XV6JIO/3EFBHOP/QAWSTXV7U80/ERTXY9P/34RCBHUP;/
chars per stroke: 5.82
strokes per word: 1
```

You can get **'it's friggin' NUTSO dude!'** like this:

```
./kinglet.pl "it's friggin' NUTSO dude!"
SD5VB,UJI/WGMH7U0;/SFTG[ ]XV67UIL/2W3E5CN6YI;/ERGCVHU8/1Q
chars per stroke: 5.38
strokes per word: 1.5
```

You can get **'Oblique anastomosis effected between spatulated upper ureter and renal pelvis.'** like this:

```
$ ./kinglet.pl "Oblique anastomosis effected between spatulated upper ureter and
renal pelvis."
AE5TCVBNY7KL/145TVY90/WERFN6YIKOL/23RFMUJI/3RTGCVBU8I/Q3ERG[
]CVBNUK/Q2R,Y80/24FGCVY790P/3RG[
]CVYH79/ASTCVH78IP/WSETCVMHUI/WSTGBY7I/2RBM6UKL/W34F5G
chars per stroke: 5.9
strokes per word: 1.4
```

Or try sonnet #43:

\$./kinglet.pl "When most I wink then do mine eyes best see, For all the day they view things unrespected, But when I sleep, in dreams they look on thee, And darkly bright, are bright in dark directed. Then thou whose shadow shadows doth make bright How would thy shadow's form, form happy show, To the clear day with thy much clearer light, When to unseeing eyes thy shade shines so! How would (I say) mine eyes be blessed made, By looking on thee in the living day, When in dead night thy fair imperfect shade, Through heavy sleep on sightless eyes doth stay! All days are nights to see till I see thee, And nights bright days when dreams do show thee me."

AEFGVBNYHI/WSED5TCN6I/AE4R[
]CBNYH7UIK;/ERT,Y890/23TNMYH7U8/ERVBMU80/E4RFGVUJK/ERFVU900P/E4R5CVB6YUI/WE4TBM6YIK/
3TCM70P/ERTVHUJI/Q34RVHJI/ED4RTGVNMUJI/2W3ETGVBNUJ9/345C,HU900P/E4R5CVB6Y7K/EFGBV
NYHI/AE4R[]CNY78/34TGXBVL/
ERGVBNM7IKOL/W3TBNUOP/E4TC6YIK;/E45G[]BNU8/1Q2WTXV6YH7890/ERGBHIK;/SEDRFG[
]CVBNHKL/AFW6HI/WE4G[]CVBNHKL/
AFWNYL/ERGBHIK;/ERGMH78I/Q3ERGB,YI;/ED4RF[
]BN6Y8I90/EFVGM6Y80/3TCNJIO/2W35TCBNY80/W3E4CM6Y9P/W3TCM6YIK/
E4R5TBYUJ8/E4RFGCB7U0;/W3E5[]XVJ89/Q2TCBN6Y8I90/SEDRG[
]BNHUJI/W3FBHU89/Q2FTGCNYO/QAE4R5TXBYO/QAE4R5T[]VM79/
2RFG[]CNJ89/Q25G[]XV689/ERTVY8I/SEDRBHI/2W34VBM9P/SDRT[
]BNHUJI/WSED45TVBJI/3E4TV70L/WSETBM7U0;/W3E5XBYI;/
Q24RTVNYK/23TCVYH80/3RCVNH7UI/3RFGVUJI/EDRFG[]CNJIO/2W3R[]CNJL/3RF[
]CN6Y0P/E4R5VM6Y9P/EFGC6Y7IKL/2W3T/[]/
AE4R[]CN70P//[]/E4R5TCVNUI/3RFGVUJI/2WR[
]CVBNY78/W3RFVHUI/WSED4CMU900P/E4R5CVBNHUJI/SED5TCYUJL/AWTCVNM6YIK/
3R[]CVNYIK/3TBM7U8P/S4RG[]CM70P/1Q2WTXVHJ8IOL/SETCVNY890/34CMYIL/SD4RGVM6I/EDRFG[
]CVB7KL/WSETM6Y7U9/ASRTGV67K/
ERFVM7890/35G[]XVYUIOL/2345TVNMJI/WS3E4N,HUJI/W34TVU9/E45G[
]CN7U0;/W3E5BMU80/W3TVHUJ8/W3TCM6YIK/ERFN70P/
1QTXV7IKL/SEDTCM70P/W3TBH7I/SE4RVNMJK/W3TN6YI/W3RVY90P;/SED4T[
]XV7UI/W3RVYIK/3RXBYI;/
12RG[
]VN7U0;/W3E5CNY90/WSE4RVNMJK/ERGBHUJ80/EFGBVNYHI/2W34GB6Y7U80/ERGCY80/W3E5TCBNYIK/3
R[]NMUOLP;/
chars per stroke: 5.76
strokes per word: 0.94

Cheat Sheet

Cheat Sheet

The cheat sheet is organized into several sections:

- 9-Key Alphabet For The Left Hand:** A grid of 24 letter combinations (a-z) for the left hand, each with a yellow dot indicating the finger used.
- Thumb Alphabet:** A vertical column of 26 letter combinations (a-z) for the thumb, each with a yellow dot indicating the finger used.
- 6-Key Alphabet For The Right Hand:** A grid of 24 letter combinations (a-z) for the right hand, each with a yellow dot indicating the finger used.
- 9-Key Alphabet For The Left Hand (repeated):** A second grid of 24 letter combinations for the left hand, similar to the first section.

At the bottom, there is a list of special characters and symbols, each with a yellow dot indicating the finger used:

- [!]=
- [@]=
- [^]=
- [&]=
- [*=]
- [^]=
- [?]=
- [cap]=
- []=

Jackdaw

Contents

- 1 Introduction
- 2 The Basic Principles
- 3 Jackdaw on a Conventional Keyboard
 - 3.1 The Missing Consonants
 - 3.1.1 Left Hand
 - 3.1.2 Right Hand
 - 3.2 Further Consonant Combinations
 - 3.2.1 Left Hand opening with A
 - 3.2.2 Other Left Hand Combinations
 - 3.2.3 Right Hand
 - 3.3 Vowels
 - 3.4 Other Rules

Introduction

An orthographic theory of shorthand is based on spelling rather than phonetics. The advantage is that you can (in theory) write any word you know how to spell at least as fast as you can type it, because you do not need a dictionary to convert it from phonetic to written form. This is a sketch for an orthographic theory that might perhaps be implemented on a [SOFTHRUF](#) or a [Planck](#), [Atreus](#) or other small keyboard that allows easy chording. It is intended to be a less specialised tool than Plover, easier to learn and to use as a general personal keyboard, while hopefully having more significant advantages than a simple change to key layout like Dvorak or Colemak.

Please be aware that this is still only a concept; there is no software or hardware implementation currently available. Feel free to develop and share your own version in the spirit of Open Source.

Update: *Vissale NEANG now has working firmware for the Ergodox keyboard - available [here](#).*

I will add versions for other keyboards as I hear of them.

The Basic Principles

As with Plover, words are stroked (ideally) a syllable at a time. Once you know the basic chords for all the letters it is possible to type anything, but common key sequences can be combined in a single stroke. If you are not sure of how a stroke is formed, it is *always* possible to type it in sequential parts. The left hand types the leading consonants, the 'onset'; the thumbs type the vowels, and the right hand types the concluding consonants, the 'coda'. To reduce the number of keys and the finger-movement required, some letters or combinations of letters are represented by a chord. The stroke is separated into four parts, the onset the vowels, the coda and the inflection (both of the latter controlled by the right hand). Each of these is interpreted independently, and their output combined in that order.

Jackdaw on a Conventional Keyboard



This is inspired by a patent for a ['syllabic typewriter'](#) filed by Dothan Shelton in 1976, adapted for a standard keyboard.

In general, the white keys (control keys, numbers and punctuation) work as they do in QWERTY, if any control key other than Shift (Ctrl, Alt, Win or Alt Gr) is used then the keys with which they are combined have their usual values - i.e. Ctrl-Z, X, C, V, etc. are all where you would expect them to be,

There are eight green consonant keys for each hand (giving up to 255 possibilities each for onset and coda), five blue vowel keys and nine yellow modifiers, taking advantage of the fact that with a staggered keyboard you can press three keys with one digit, but cannot easily press four.

'Steno order' is left to right. As can be seen, it is possible to stroke many words without knowing any chords at all. A is the most common initial letter apart from T, and it makes it possible to stroke such common words as AGAIN, ABOVE and ALONG in one. The left little finger operates A and S, and the right one, T and S. Shift only capitalises the initial character of a chord. If you want all the characters in UPPERCASE you must use Caps Lock.

If the common punctuations ?, , ; or : are combined in a stroke they are placed at the end, and a space automatically added. In the case of . and ? , the next word has an uppercase initial without having to use Shift.

The Missing Consonants

Left Hand

B=CTWH	D=CT	F=CTH	G=SCT
J=TWN	K=TWH	L=NR	M=WN
P=CW	Q=TNR	V=TN	X=STW
Y=HN	Z=CN		

Right Hand

B=GC	D=NLG or CHS	F=GCH	K=GT
M=NGH	P=LC	S=NL or S	
V=NH	X=LGH	Z=LH	

Further Consonant Combinations

Like single letter keys, consonant chords may be combined in the one stroke. For example, BL = CTWHNR

You can get a long way with just these, but Shelton adds some common and not-so-common combinations that cannot be produced so simply:

Left Hand opening with A

ABB=ASCTWH			
ACC=ACHN	ACCL=ACHNR	ACCR=ACHR	ACKN=ATWHN
ADD=ACTW	ADV=ACTN		
AFF=ACTHN	AFFR=ASCTHR	AFT=ASCTHN	
AGG=ASCTW			
ALL=AWNR			
AMM=ACWN			
ANN=ASN			
APP=ASCWN	APPL=ASCWNR	APPR=ASCWR	
ARR=ASR			
ASS=ASCN			
ATT=ATW			

Again, these can be combined with other keys to form a longer chord, e.g. ABBR = ASCTHWNR.

Other Left Hand Combinations

BY=CTWHN			
DEL=CTNR	DEM=CTWN	DER=CTWR	DEV=CTN
JER=TWNR			
RH=HR			
SER=SR	SERV=STWNR		

Right Hand

BL=LGC	
CK=GCT	

```
GN= RNG
CKL= LGCT
NCES= NCS
SP= NLC
TH= HT TION= NCT TCH= CHT
VES= VS
WN= RNH WL - RLH
ZES= LHS
```

Vowels

A minimum of four keys are needed to represent the vowels, allowing the the ten most common vowel pairs to be produced in one stroke, because here are eleven possible combinations of more than one key and we need one of those for the missing single vowel. Fortunately the top ten vowel pairs account for nearly 90% of all vowel pairs in typical text, so perhaps that is good enough. A little logic will provide some more.

Following Plover tradition, they could be coded thus:

```
AI=AEU AU=AU
EA=AE EE=AOE
I=EU IE=OE IO=AO
OI=OEU OO=AOU OU=OU
```

In this case it is possible to use more keys. so no such coding is needed. Vowel keys stroked together will always appear in the same left-to right sequence (EOAUJ). This sequence is chosen so that the most common vowel pairs are in order, and those that are not are (IO, UA) are often in distinct syllables. The control keys E and Y can also help, as we shall see later.

The Modifiers

The ***** modifier may be operated by either thumb. On its own, it deletes the last stroke, as in Plover. As part of a stroke, it doubles the vowel if there only one. If there are more than one, it reverses the order of the last two, eg PARANOIA could be stroked **CWARIANOAI***. The only exceptions are **O*UI** which produces the more useful IOU, and **A*UI** which produces UOU. EUA is also found in no English word that I know of, but I can't think of a useful substitute,

The **'** modifier is normally operated by the right thumb. On its own it produces an apostrophe or (with Shift) an @. It is used to modify the output of the right hand keys, signifying for example either a contraction e.g. N'T = **'NT**, a doubling of the first consonant -DDL = **'RNLG**. or possibly a multi-syllable suffix or the closing part of a phrase.

In a similar way, the **|** modifier is normally operated by the left pinkie, produces a \ (or a | with Shift), and is used to modify the output of the onset (typically for the first part of a two-part phrase).

The **E** is operated by the right little finger, and the **Y** by the third finger. They append those letters to the coda, E preceding Y if both are present together with at least one right hand consonant, as in HONEY. Otherwise E follows Y as in EYE or RYE. If Y follows a right hand consonant and is followed by another stroke for a continuation of the word (does not have a space or a punctuation in the same stroke), then it changes to an I. Thus, BUSINESS is stroked as BUSY|NESS. but the Y in PAY|MENT does not change. If you really want a Y after a consonant in the middle of a word, combine it with a left hand consonant: BU|SY|NESS.

ING, **ER**, **ED** and **S** on their own, produce the punctuation they normally would (ING does nothing). As part of a stroke, they add their inflection to the coda. More than one may be present in that order, and may follow either E or Y. If S follows Y and another right hand consonant, it produces IES..

Other Rules

TNR for Q always adds a U, if followed by a vowel in the same stroke.