LAIQBUN

PHONOLOGY

Laiqbun's phonology consists of 31 phonemes, divided into 25 consonants and 6 vowels.

	Labial	Alveolar	Palatal	Velar	Glottal
Plosive	p b p’	t d t’		k g k’	
Nasal	m	n		ŋ	
Fricative	f v	s z	ʃ ʒ		h
Affricate		t͡s d͡z	t͡ʃ d͡ʒ		
Tapped		ɾ			
Approx.		l			

	Front	Back
Open	i	u ɯ
Mid	e	o
Closed	a	

The possible syllable structure is CX(V)(V)(N), where C is any consonant, V is any vowel, N is any nasal and X is the vowel that marks part of speech. More on the latter later in this document. When two identical vowels are together, they're separated with a glottal stop.

ORTHOGRAPHY

Laiqbun's sounds are written like in the following chart:

a	**e**	**o**	**u**	**ɯ**	**i**	**k**	**k’**	**g**	**t**	**t’**	**d**	**l**	**ɾ**	**m**	**n**
a	e	o	u	y	i	k	kk	g	t	tt	d	l	r	m	n
p	**p’**	**b**	**t͡s**	**d͡z**	**h**	**ŋ**	**t͡ʃ**	**d͡ʒ**	**ʃ**	**ʒ**	**s**	**z**	**f**	**v**	
p	pp	b	c	x	h	q	ch	xh	sh	zh	s	z	f	v	

Laiqbun's alphabet order, inspired by Hangul.

BASIC GRAMMAR

In Laiqbun, all words (excluding particles) are verbs by default, but their part of speech can be changed. For this document, verbs will be referred to as predicates because of the way they work. Part of speech is marked by the first vowel of a syllable, following this table:

Vowel	Part of Speech
a	Noun
e	Predicate
o	Relative clause
u	Syllable connector
y	Particle
i	Content clause

When turning a word into a noun, these are defined as "that which does [predicate]".

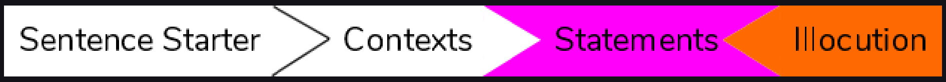
Syllable connectors are simple, they just mean that the current syllable is part of a greater word. It can be expected that the roots of a compound word are in some way related to the meaning they have together, but it isn't always the case.

Particles are a restricted category of words that have a mainly grammatical function. Predicates cannot be turned into particles in a way that makes sense.

The *content/relative clause* vowels will be explained later.

THE SENTENCE

A sentence in Laiqbun will usually look like this:



Statements are the only mandatory part of a sentence, while an illocution is preferred, but not necessary. Everything else is optional and used only in certain cases. We will focus on understanding the mandatory parts first before dwelling on the rest.

A **statement** is any sentence that has a predicate. It doesn't necessarily have its arguments filled, though it usually does. Predicates are VSO, or more precisely, predicates -> arguments.
heo sha maq
I eat food

This sentence uses the word **heo** (x1 consumes x2) as its predicate, and the words **she** (x1 is the speaker) and **meq** (x1 is consumable) in their noun forms as arguments for the predicate.

An **illocution** is used to mark the illocutionary force of the sentence you are saying. There are four illocutions and a nullifier:

Marker	Illocution
dy	Assertive
sym	Interrogative
by	Imperative
ky	Performative
nyq	Illocution nullifier

A **nullified** illocution means that your sentence has no real effort to communicate anything and is, instead, an empty sentence being thrown to the air. It can also be related to the context, without being content in it - sort of like commentary.

The **assertive** illocution is the most commonly used and can be omitted, which only leaves the vagueness of whether you are asserting something or throwing an empty sentence to the air.

An **interrogative** sentence is self-explanatory. Using it in a normal sentence will lead to a yes or no question, more complex questions will be explained later. You answer a yes/no question with sy for yes and ny for no. This is not entirely correct, but will work for now.

An **imperative** sentence is a sentence that tries to get someone to do something, either as a command or a desire.

A **performative** sentence is a sentence that becomes true just by it being said. An example in English could be *"I hereby pronounce you spouses for life"*.

A funnier example could be "I say this".

So, looking back at the previous example, it is clear that it was assertive, so the *dy* illocution can be used.
heo sha maq dy
I eat food.

Now let's look at a statement with no arguments and no illocution marker:
meq

There are two ways to read this. In one, it's just the word "edibles" being said. In the other, it's asserting the satisfaction of the meq predicate. The null illocution marker can be used to mark the former, the assertive one can be used for the latter, but doing either is optional.
meq nyq / meq dy nyq
things that are edible.
meq dy
There are things that are edible.

PREPOSITIONS AND ADVERBS

In Laiqbun, a preposition is made by putting the particle *hy* in front of any word. A Laiqbun preposition means that the statement behind it satisfies the predicate that's being turned into a preposition, with the argument coming after the preposition.
leu sha hy len tam
I sleep in the house.

Here, the word **len** (x1 is in/at/on x2) is used as a preposition. This means that the statement **leu sha** (I sleep) satisfies the predicate **len**, with **tem** (x1 is a house) being its second argument.

If the preposition has more than one argument, only the second is filled. If no argument is filled besides the sentence behind the preposition, it's an adverb.

Another way to do adverbs is marking the word that you want to use as an adverb with the vowel **o**. In this case, the adverb goes right after the predicate that you're modifying. The difference in meaning between them is this: **o**-adverbs apply only to the predicate they're attached to, while **hy**-adverbs modify the whole statement.

RELATIVE AND CONTENT CLAUSES

The main use of the vowel **o** is for relative clauses, with **i** being its counterpart for content clauses. Those vowels replace **e** in the predicate of the relative/content clause. To eliminate some ambiguities, there's the clause closer **gy**.

In a relative clause, **kka** is used to refer to the noun that the clause is describing.
she zhan hoo kka kau gy dy
The person that eats fruits is me

Here, **hoo** is used as the relative clause. **kka**, which in this sentence refers to **zhan** (x1 is a person) is used as its first argument, and **kau** (x1 is a fruit) is used as its second argument. This describes the noun **zhan**. Then, this whole thing is used as the first argument for **she**. For now you won't need to know the uses of **kke** with other vowels.

gy can be omitted if it's before the end of the sentence.

SERIAL PREDICATES

Serial predicates are like predicates, but stacked onto each other and wearing a trench coat, and are a way to shorten sentences that use content clauses. To define them, you put the entire predicate as the last argument of the predicate before it, for example:

sheo: x1 wants x2
tei: x1 builds x2
sheo tei: x1 wants (x1 builds x2)

So, for example, **sheo tei sha tam dy** means "I want to build houses".

Let's take a look at a serial predicate with three predicates stacked onto each other.

teqbue: x1 lies about x2 being the case
teqbue sheo tei: x1 lies about (x1 wants (x1 builds x2)) being the case

teqbue sheo tei sha tam
I lie saying that I want to build houses.

For when a relative/content clause is itself a serial predicate, only the first word is marked with the clause vowel. One example is:

zhan shoo zeao kka lauaq
Person that wants to be aware of the consensus

One can alter how this merging works by using the particle **ly**, this way, "x1" in the predicate you're merging will refer to the slot which the predicate is merging into, and its other slots will be separate.

sheo leo: x1 wants (x1 feels happiness)
sheo ly leo: x1 wants (x2 feels happiness)
sheo ly leo sha tam
I want you to be happy

Pretty cool, right?

TENSE

Time in Laiqbun is actually pretty vague, though most sentences can be assumed to be in the present, this is not necessarily the case; **heo sha dy** can still mean *I ate*. For this, there is the option of using serial predicates, as some of them specifically indicate time relative to the present.

Predicate	Meaning
pe	x1 was the case in the past
teoq	x1 is the case in the present
ceum	x1 will be the case

ceum veq sha
I will be strong.

SENTENCE PREFIX

Now that the basic sentence has been defined, we can now define the two optional parts of the sentence, the prefix and the contexts. Let's start with the most basic one:

A sentence prefix serves as a connector between a sentence and the one right before it, they mark sentence boundaries when the illocution is omitted in the last sentence, and contain information about the relationship between them.

Sentence Prefix	Meaning
zy	Null
gyn	Additional to the last sentence
kyo	But, however, contrary to the last sentence

CONTEXTS

Contexts are where Laiqbun gets interesting. If you are familiar with Toaq's prenexes or toki pona's context marker, these will definitely ring a bell. Contexts mark what the sentence will be about, bound variables and tell the perspective from which the sentence should be seen. There are three kinds of contexts.

PRENEX CONTEXT

The prenex context serves as a way to bound variables. If you say "some apples” twice in a sentence, it's ambiguous if the second instance is the same as the first or not, a prenex serves as a way to bound that, so that all the instances of it refer to those in the prenex. If an instance is different from that of the prenex (or the one mentioned before, in case there is no prenex), that can be marked using **zhyku**. It also marks topical information.

The prenex context is marked with the particle **zhy**.

PERSPECTIVE CONTEXT

A perspective context is usually, but not necessarily, a sentence, and works very similarly to la in toki pona. It's marked with the particle **lyoq**.

FUNCTIONAL CONTEXT

A functional context serves as a way to bound grammatical functions that are usually optional to mark, but are assumed as something else if unmarked. An example of this is numbers in bases other than dozenal. This will be explained in depth later; for now, knowing that it's marked with **tty** is enough.

SUPERCONTEXTS

Supercontexts are variations of a context that mark every sentence that is said until the particle **vyo** is said or until the conversation is over. They're useful to start conversations that will use a specific context throughout the entire conversation or text that follows. A context particle is turned into a supercontext by adding **gu** to it.

With the Laiqbun sentence fully defined, we can finally move on to the real juice.

OFFICIAL PUNCTUATION

In Laiqbun, normal English punctuation can be used, most symbols have the exact same meaning, except for dots and commas, which only mark pause and aren't supposed to have grammatical meaning.

However, if one wants to type in the official romanization a Zhanbun would use, they'd replace commas with **>** and colons with **>>**, not use periods, not use question marks, only capitalize names (see: **PROPER NAMES**) and replace quotation marks with **«** and **»**.

QUOTATIONS

Laiqbun has a way to let you know when a quotation is being opened and closed just by speech: This is done with the particles **kyn** and **lyq**, which open and close quotations, respectively. In formal writings, these are kept together with the angular quotation marks, although it can be dropped for informal writings, since you can read « as **kyn** and » as **lyq**.

PARENTHESES

In Laiqbun, parentheses are also marked with particles, **kye** to open them and **nye** to close them. Similarly, they are said before their respective symbols, but can be omitted in writing as the symbols themselves can be read as the word.

PASSIVE AND MIDDLE VOICE

Laiqbun marks passive voice with the my particle before the predicate. What this does is inverting the x1 and x2 arguments of a predicate, while keeping the meaning intact.

tei zhan tam dy
The person builds a house
my tei tam zhan dy
The house is built by the person

Middle voice might be a strange concept for you, as it's barely present in English and is never explained in classes. Middle voice occurs when, in a sentence, the subject is both the agent and the patient; a close example in English is **the window broke**. In the sentence, it's not said who broke it, the action only says that the window changes to a state of brokenness - it's both executing the action and being affected by it. Laiqbun marks this with the **ty** particle. It merges the arguments x1 and x2.

peaa vauq vauin dy
One breaks the window
ty peaa vauin dy
The window breaks

There is also *ultrapassive* voice, which is a uniquely Laiqbun concept. Its particle is **cy**, which inverts the order of all three arguments of a predicate (so x1 x2 x3 becomes x3 x2 x1). When the predicate has only two arguments, it works the same as regular passive voice, but puts additional emphasis on the second argument (the one that is second in the dictionary definition).

pean hau gai hauqtuq
They give a feather to the bot
cy pean hauqtuq gai hau
The bot is given a feather by them

QUANTIFIERS

Laiqbun has the following quantifiers, they all go behind the object that they describe.

Quantifier	Meaning
chy	Existential quantifier (some of X)
nyoq	General quantifier (X in general)
byiq	Universal quantifier (every single thing that is X)
sya	None (none of X)

SPECTRUMS

As you might be aware, a lot of things in the world work as spectrums and not defined binaries. In fact, you can make a spectrum out of anything! If such wasn't the case, why are there fights on the sandwichness of a hot dog? There is obviously some perceived sandwichness there, but it's clearly not enough to say that it's a full sandwich. I would say that it is in a second degree of sandwichness.

Laiqbun integrates this inherent nature in itself, by giving its speakers the resources to make any spectrum they may perceive, and express something's position in it. This is done with the particles **py** and **fy**.

py serves as a way to make something the axis of a spectrum. Take the word **ben** (x1 has the property of goodness); if you say **py ben**, this will make a predicate that means "x1 is in a x2 degree of goodness", where the second argument is mandatorily a one-digit number.

If you want to set a negative axis, you can do so by doing **py** twice. **py ben py keaq** means "x1 is in a x2 degree of goodness/badness", where negative numbers mean badness and positive numbers mean goodness.

If you want a multi-axis spectrum, you use **fy** to start the next axis. A multi axis predicate is defined as "x1 is in the x2 degree of [1st axis], x3 degree of [2nd axis] ... xn degree of [(n-1)th axis]".

Finally, to refer to all degrees of a spectrum, you say the object of the axis, in noun form. If the spectrum has a negative side and you still wish to use its entirety as a degree of it, you mention both sides, separated by **ry**.

NUMBERS 1

Laiqbun uses dozenal. Why dozenal? Well, it's better than decimal, and although seximal is better, twelve has a certain *aesthetic* that I prefer over that of six. Laiqbun has two number systems, which we'll divide in three sections. Numbers 1 (this section) will explain the casual number system. Numbers 2 will explain the formal number system and Number 3 will explain one of the extended functionalitites of the formal number system. There are twelve predicates for the digits, which are by default framed as quantities:

Predicate	Definition
meo	x1 is in a quantity of 0
heq	x1 is in a quantity of 1
ttea	x1 is in a quantity of 2
rea	x1 is in a quantity of 3
cheuo	x1 is in a quantity of 4
beua	x1 is in a quantity of 5
ceom	x1 is in a quantity of 6
xeo	x1 is in a quantity of 7
gem	x1 is in a quantity of 8
dea	x1 is in a quantity of 9
ppem	x1 is in a quantity of X (ten)
lea	x1 is in a quantity of Y (eleven)

ttea zhan dy
There are two people
ty deao hau rea dy
The three people are romantic partners (of each other).

These can be easily transformed to ordinal predicates by adding the **rua** syllable next to them, such that **heqrua** means **x1 is the first of x2**.

heqrua sha zhan ppoa kka na dy!
I am the first of the people who came here!

However, here's something that might seem unintuitive to most Laiqbun learners until they think about it: Laiqbun numbers are written starting from the smallest place value and ending in the highest. The reason for this will be discussed in Numbers 2, for now, here are two examples:

meottuaxuo zhan hy len na dy
There are 720 people here.
leochuuo hau dy
They are the 4Yth person.

NUMBERS 2

A number in Laiqbun is marked by putting **heiq** before one of the following:

WORD	NUMBER
ku	0
kku	1
gu	2
tu	3
ttu	4
du	5
lu	6
ru	7
mu	8
nu	9
pu	10
ppu	11

Then, **kuyq** for the first power of twelve, **puyq** for the second power, **vuyn** for the third power, **tuyq** for the fourth, and then it's repeated. This is why the digits are read in the reverse order from English, that is, from the smallest to the largest place value - to make it easier to read out loud without having to count how many digits there are first.

Haiqkkukuyqpu
The number X1 (121_{10})

Ordinal numbers are marked with **heiem**; a quantity is marked with **ppeam** and a number that modifies units (see: **UNITS IN LAIQBUN**) is marked with **xe**.

For aesthetic reasons, individual syllables of numbers can be separated. This does not matter, as the vowel **u** already links them together. **Haiq kku kuyq pu** is equally correct.

NUMBERS 3
Now, some people might prefer using other bases instead, maybe just because of context or because they just think a different base is neat. Laiqbun allows for speaking in other bases natively, up to base 7200.

If you paid attention, you might have noticed a pattern in the digits above. Or probably not. But the idea is that there is a way to construct any digit using the Laiqbun sounds, and since all numbers are connected to the root **heiq**, they won't clash with any words.

The pattern is taking all the consonants, and circling through them with only a linking vowel, then, the same is repeated, with an extra vowel, and everytime the process is finished, the added vowel changes. Once that is done, a new vowel is added, and the process is repeated until the nasals are done, in the "q n m" order. It is important to note that the first added vowel can't be **y**. So, for baker's dozenal, the digit twelve would be **ppu**.

When using another base, the words for powers become powers of that base, instead of twelve.

This is where **functional contexts** come in. With a functional context, you just have to put the number for the base you wish to use and use it as the argument for **beiuq** (x1 is the currently used base), and this will mark that the number you said is in a certain base.

PROPER NAMES

Proper Laiqbun names are marked with the predicate **se**. Any word that comes after it is a name. It only takes one word and the word directly after it is completely free from the grammaticalness of the vowels. To close a name, you use **shy**. You can use **xhe** for names that are valid words in Laiqbun - these don't require shy to close them.

se Keitian shy	**sa Keitian shy**
x1 is named Keitian	That which is named Keitian

In Zhanbun, the culture of the Laiqbun, a surname is a short relative clause that describes you. A person can have as many surnames as they wish. In very formal settings, like in work-related letters, all the surnames are used, in relatively formal sentences, like talking to a stranger, only the first surname is used, and in an informal context, only the first name is used.

MODIFIERS

To modify a word, you put a word in predicate form in front of the noun that you intend to describe, this will indicate that the noun being described satisfies the predicate, filling its first argument.

zhan veq

Strong person

This is, in practice, exactly the same as **zhan voq kka gy**. It's an even shorter way to do it. The only difference is that the latter means "person that is strong".

POSSESSIVES AND AUTOMATIC KKE

In Laiqbun, you express possessives via a relative clause using the word **peu** (x1 possesses x2). Exceptions are on pronouns, where there are the words **peushu** (x1 is mine), **peutu** (x1 is yours) and peuhu (x2 is hers/his/theirs). This can be done for all pronouns.

This is where **automatic kke** is introduced. As we saw before, **kke** refers to the object being described by a content clause; and as it turns out, it can be omitted and something can be assumed.

For **predicates with one argument**, **kka** will be assumed, as there's nothing else that can fill it.

For **other predicates**, **kka** will be assumed in the **second** slot. This shortens a lot of relative clauses, including those with **peu** itself.

Now, if you are a toki pona speaker, you might be tempted to use a word as a modifier to mark possession - especially with pronouns. This won't work, as this will mean that the thing *is* whatever you're describing it with, not that there's a relationship between them. **maq she** means "the food that is me", which usually makes no sense, but can.

CONJUNCTIONS

To put more than one predicate in one statement or more than one noun in an argument, you use a conjunction.

Conjunction	Meaning
ry	And
vy	Collective and
hynu	Inclusive or
hylu	Exclusive or
hynubue	Neither
hylubue	Neither or both
rybue	Neither or either one, but not both

These can be used in three places:

In arguments of a predicate, they mark more than one thing as part of the same argument.

Between predicates, they apply the arguments to the predicates as much as possible.

Between sentences, they're just a normal "and" conjunction.

ry works the same way as English's "and" does.
beoq sha ry ta dy
Me and you are girls.

vy marks two things as part of a single entity. Its uses for predicates will be explained later.
beoq sha vy ta dy
Me and you are *a* girl.

hynu works similar to English's "or", it means "either y, x, or both"
heo sha hynu ta sym?
You or I (or both) eat?

hylu is similar to the last one, except it excludes the cases where *both* are true.
heo sha hyku ta sym?
You or I eat? (but not both)

hynubue is literally translated as "neither".
heo sha hynubue ta dy
Neither you or I eat

hylubue marks cases when you refer to **both** or **neither**, nut never to each one separately.
heo sha hylubue ta dy
Either you and I eat, or no one does

rybue is used to refer to **neither** or **either one**, but never to **both**.
heo sha rybue ta dy
You or I or none of us eat (there's at least one of us not eating)

QUESTIONS
As was seen before, to ask questions it's necessary to use the interrogative illocution. This creates a yes/no question. To create other questions, there are only two words to know:

zeai literally means "x1 is what?", and is used to ask what something is when you don't know any of its properties, or know one of its properties but that is too vague to tell which exact thing it refers to. It also serves for reasons ("because what?") and time ("in what time?").

zeaixhuom literally means "x1 does what" and is used to ask what action is a thing doing, rather than what the thing is.
heo ta zaai sym?
What are you eating?

ANSWERS
It was explained earlier that you answer a yes/no question in Laiqbun with sy for yes and ny for no. Well, that wasn't quite exact - what do "yes" and "no" even mean? What about when none of them are correct? Or when there's a point in between? Well, the nature of Laiqbun is hugely spectral, and these aren't

exceptions. This is why "binary questions” is better said as "spectral questions" in Laiqbun.

sy is defined as "the statement(s) that were asked for are true". If someone asks "do you not eat fruits?", and you indeed don't eat fruits, the answer is **sy**, because it's true that you don't. Similarly, **ny** is defined as "the statement(s) that were asked for are false". If you **do** eat fruits, your answer is **ny**.

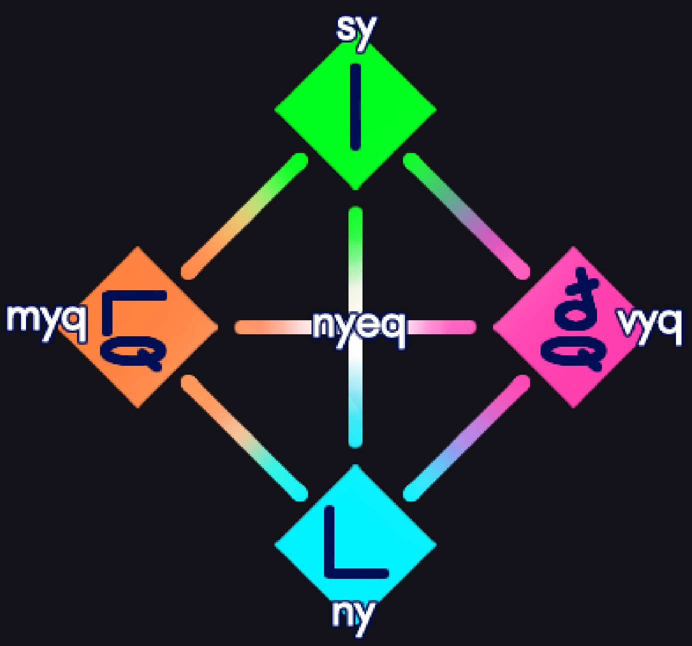
With that out of the way, what if you lack the information to answer the question? Maybe you've never been introduced to the concept of fruits, and as such don't know if you'd eat them or not, maybe you come from another planet, or speak a language which uses a single word for fruits, vegetables and mushrooms and you still don't know which is which very well. For that is the word **myq**, which is for when neither of the binary answers are correct. It's neither true nor false.

What about statements that are self-contradictory? For example, what about **qe saym no dy**, "this statement is not the case". Is it true? Is it false? It definitely can't be neither - we get all the information we need from itself, so what is it? The answer is: this is a superposition of truth and falsehood. If it's false, that makes it true - if it's true, that makes it false. This is **vyq**.

Okay, sure. This is more than the binary logic most readers of this document will be used to. It's still not *spectral*, it's *quaternary*. If you didn't skip the **SPECTRUMS** section, you might expect that you can construct an answer that's located somewhere between two of these, and you'd be correct.

You can still answer with just either of the four logic states and nothing else, but you can construct spectrum answers this way:

First, you put the answer that you want closer to one end of the axis, then you put a digit, and then you put the other end of the spectrum - the lower the digit is, the closer it is to the first state you entered, the larger it is, the closer it is to the second state.



ASSIGNABLE PRONOUNS

Laiqbun allows you to use assignable pronouns, but instead of having a very limited set of them, you can have up to 10664 assignable pronouns, and these can also be assigned to clauses, so you can have a pronoun referring to a sentence.

In Laiqbun, an assignable pronoun is a word that, on itself, means nothing, but that you can assign meaning to in the functional context by using it as the predicate of a sentence, then every time you use it in that sentence (or conversation, assuming you used a supercontext), it will refer to the thing you mentioned.

To construct an assignable pronoun, you just use the root **che** and attach any syllable you wish to it. The syllable must have the vowel **u** as its first vowel, as this will connect it to **che**. You can use up to two syllables. To assign something to them, you use it as a predicate for what you want to assign it to in the **functional context**.

chesui, chepu, and **chetuaoku** are all valid assignable pronouns.
Let's look at an example of them in use:
chekui sa Keitian shy ttygu pei chakui laiqbun

It's quite hard to translate what this sentence means, so let's look at its parts:

chekui sa Keitian shy marks the pronoun **chekui** as referring to "sa Keitian shy", and then uses that pronoun in the sentence "[chekui] creates Laiqbun". In practice, this sentence means "Keitian creates Laiqbun".

Yes, this is very ineffective for short sentences, it would be significantly better to just say **pei sa Keitian shy laiqbun**. It's shorter and doesn't require a new concept; however, for long sentences it can be useful to avoid ambiguity using pronouns, or to make conversations about a topic a lot easier.

UNITS IN LAIQBUN

Units in Laiqbun are based in the Primel metrology - since said metrology fits dozenal better, and the SDN is applicable. This section will explain them as briefly as possible. Skip the two next paragraphs if you already know!

The Systematic Dozenal Nomenclature (SDN) is a nomenclature system for units designed specifically with dozenal in mind. It allows you to create a unit that derives from another very easily. Need a unit that's the meter but multiplied by the second power of twelve? Sure, there you go; Need one that's the hour but multiplied by the twelfth power of thirteen? Weird flex, but okay! Need one that's 1572 over 201ths of a yard? Oddly specific, but gotcha covered!

The Primel metrology is, of the two most common systems of units for Dozenal, the good one. There isn't really a lot to be said about it, and yes, dozenalists argue about these the same way decimalists argue about the imperial system and the metric system. I *already* see that the day I make this document public someone will argue with me about this. That out of the way, let's see how this works in Laiqbun:

Unit	**English Equivalent**	**SI equivalent (decimal)**
ttem	Timel	0.02893518s
leueq	Lenghtel	8.202083mm
veueq	Velocitel	1.020470m/s
heaq	Accelerel	9.79651584m/s^2
heea	Areanel	67.2741mm^2
veouq	Volumel	0.55178ml
meae	Massel	0.5517g
deeq	Densitel	999.972kg/m^3
meom	Momentumel	15.6407g·m/s
qeuaq	Actionel	12.8286g·c^2/s

feouq	Forcel	540.5452dyn
tteaeq	Tensionel	0.6590N/m
ppeeq	Pressurel	80.3495 pascal
heim	Influencel	363.6473 dyn·c
heen	Energel	44.3359mJ
meaen	Massic energel	$0.0803m^2/s^2$
peua	Powerel	1.5322mW
qeiq	Intensitel	$22.7762W/m^2$
reian	Radian	22.7762°
deoan	Dozenal degree (angle)	1.44°
tteem	Temperaturel	$1.9137x10^{-5}K$
deou	Dozenal degree	1.44°C

The most basic predicate for these units is **hei** (x1 measures x2). This is very vague, and can be used with any of these units. You could say something like **hei na ttam ppoamtu** to say "this lasts three timels", or **hei na laueq ppoamlu** to mean "that is six lengthels long".

To use a multiple of an unit, you use the particle **vyuq** after the name of the unit, and then the number in noun form. Unlike normal numbers, for this one you just spell out the digits. Similarly, you use **lyau** for division, **ppyua** for exponentiation and **cya** for inverse exponentiation.

ttam vyuq xanu ppiamlu: Six of an unit that's nine times a timel.
To get a unit that's multiplied by a power of the current base, you use **kkyo** to multiply by a power of the current base and **cyoq** to multiply by an inverse power.

As an aesthetic choice, you can remove the spaces in the unit's name. It won't matter, since the vowels separate the words. **ttamvyuqxanu** is still three separate words.

EMPHASIS IN QUESTIONS

Sometimes, there's a difference between what exactly it is that you're asking about in a yes/no question. "did *you* give him the gift?" as opposed to "did you give him the gift?"

Solving this is pretty simple, by using the particle **xy** right after the thing that you're asking about.

CONDITIONALS

Laiqbun has four types of conditionals. These are divided in two classes, **indicative** and **subjunctive**, which are further divided by **necessity** and **probability**. To understand these, it's necessary to know that a conditional has two parts: **antecedent** and **consequent**. In a sentence like "if it rains, then everything gets wet", "if it rains" is the antecedent and "everything gets wet" is the consequent.

The **indicative** conditionals expresses that the antecedent is true in the current reality, rather than being a hypothetical. Similarly, the **subjunctive** conditionals indicate that the antecedent is a hypothetical, and nothing ensures us that it is true in the current reality (or, that you already know that the hypothetical is false).

Necessity expressess that the consequent is necessarily true if the antecedent is true. It cannot be false if the antecedent is false. Similarly, **possibility** indicates that the consequent *could* be true if the antecedent is true, but if it's false, then there is no way that the consequent is true.

Let's see some examples with a terribly poorly done joke:

Indicative - Necessity (tteun):

Say your friend travels to Paris without knowing that Paris is in France, and then they call you asking if you know what country they're in. Truly, this is a worrisome situation. "If you're in Paris, then you're in France", you respond. This is an **indicative** conditional, as you already know that they are in Paris, so the antecedent is known to be true, and it expresses **necessity**, since if they're in Paris, there is no way that they are not in France. It is simply not possible.

Indicative - Possibility (ppeaq):

Say the same friend travels to some random country without knowing anything about it. Anything. They just don't know where they are going. This friend sure has a problem with travelling. They call you and desperately tell you that everyone around them is speaking Spanish, and ask you what country they are in. "If everyone around you is speaking Spanish, you might be in Spain", you respond. This is an **indicative** conditional, as you already know that the antecedent is true, and it expresses **possibility**, because there are many countries that speak Spanish, and there is no reason to actually believe that they are specifically in Spain. But, if it wasn't true that everyone around them is speaking Spanish, then there is no way that they are in Spain.

Subjunctive - Necessity (leuan):

Say that the same friend calls you one day and tells you that they are travelling. Due to the last two experiences with them travelling that you've had, this annoys you greatly. You sigh, and in an attempt to relieve stress about the situation, you joke by saying "if you don't know anything about where you're going, I'm going to kill you". This conditional is subjunctive, as although you already have many reasons to believe that they don't know anything about where they are going, you don't have anything that confirms that they don't. But if it were to be true, you would kill them - there is no way that in the case that the antecedent is true (though you don't know if it is), the consequent wouldn't be.

Subjunctive - Possibility (geui):
Finally, let's say that the same friend tells you they're tired of getting lost. You tell them "if you were more careful when you travel, maybe this could happen less". This conditional is **subjunctive**, since there's nothing that confirms that they are more careful (and in fact, they are not, but this is a hypothetical), and it expresses **possibility**, since it says that "maybe" the consequent would be true, but there is no certainty about it.

ARGUMENT ORDER

In Laiqbun there are a set of particles that allow you to change word order freely in a sentence, these are **kyu, kkyu** and **gyua** for first, second and third argument respectively - there are no simple predicates with more than three predicates, so no more argument markers are used.

THE FUNCTIONAL CONTEXT

Yet another concept that was previously introduced but never fully explained until later. Truth is, the grammar explained up to that point wasn't enough to help understand what a functional context does - but now, you know that it serves as a way to use bases and define assignable pronouns. What else can it do?

The functional context is, to put it simply, a way the language gives you to reprogram it. It's a screwdriver and puts you in front of a bunch of cables - if you break the logic, it's on your hands now.

Setting the base is just moving the "numbers” cable from 12 to whatever you choose - or disconnecting it altogether and setting it to zero. You choose, really. Setting an assignable pronoun is just adding it a new cable to it and connecting it to something else. It's also the only way in which using a word as a particle can make sense.

You can do stuff like changing word order, setting an assumed illocution, setting an assumed sentence connector, or setting up some properties of verbs. Messing too hard with it may get the language and its understandability to break, but hey. I didn't create that grammar.
To change word order, there are the words **xei** (x1 is an argument), **chea** (x1 is a predicate), **cheaku** (x1 is a serial predicate) and **veuaku** (x1 is an illocution marker), and the particles **kyu, kkyu** and **gyua** which mark first, second and third argument.

So, let's say you want Laiqbun's word order to be like Lojban. That's simple, just use this as the functional context:
dy kyu chea xai tty ...

This makes it so that the illocution marker goes in front, the first argument goes behind the predicate and the rest of the arguments go after it. It can't be entirely like Lojban's grammar, due to limitations in the very base of the language, but I'd say this is quite close.

Or maybe you want to recreate the one Toaq (and Laiqbun) already has. You can do that too:
chea xai dy tty ...

This makes it so that the predicate goes *before* the arguments, but the illocution marker stays at the end, and is assumed to be assertive.

Its main usability is changing what a sentence looks like. It can set up assumed parts of a sentence, even if these don't even make sense - you could make it so that if a sentence has no predicate, it's assumed that the predicate is "see". It's a level of flexibility more meta and more ridiculous than necessary, but it's there in case anyone ever happens to want it.

FORMALITY

In the **PROPER NAMES** section, it was explained that surnames are expressed as relative clauses after the name, and that they're mentioned according to formality. In informal situations only the proper name is used, in formal situations up to the first one is used, and in extremely formal situations, such as work-related letters, all are used. However, the Zhanbun have another way in which formality is associated with their language, and that is in addressing the second person.

Unlike in English, where it's always okay to refer to the person you're talking to with "you", for the Zhanbun it depends on who you're talking to and the context. If you're talking to a stranger, a person of religious importance or someone with higher rank than yours, it's generally considered rude to use the second person pronoun in most sentences - especially in the greeting. Instead, you might want to skip it and leave it to context. A way to greet someone without directly using a second person pronoun is **zeo sha ky** (I hereby greet), or **zeo ky** (hereby there is greeting).

If you ever need to directly address a person in a formal context, you can instead use their rank ro refer to them. Let's say you're talking to a xhaaq - a Zhanbun monk. If you wanted to say "I want to ask you something", you could say **sheo meyqtua sha Xhaaq mayq dy**, (I want to ask The Monk a question).

However, if you can, it's always best to skip it. **sheo meyqxhuom sha mayq dy** (I want to ask a question). If all these alternatives fail, the pronoun **qen** is an alternative.

An even higher level of formality involves skipping both first and second person pronouns when possible - this is not necessary in all sentences, and when all the methods to skip second person fail, first person pronouns can be used. The previous example would be **sheo meyqxhuom dy** (There is wanting to ask a question).

Finally, when a person of higher rank wants to refer to a person of lower rank, the first person pronoun is avoided, and if it's not possible to avoid it the **xeq** pronoun is used.

Do note that these honorifics don't involve age, economical status, or intelligence - these are based on abstracts in the Zhanbun society that are hard to translate to our culture. A famous singer cannot refer to their public as if they (the singer) were on a higher rank, instead, the public is considered to be on a higher rank (unless someone in the public is either famous or powerful), though a person of religious importance would have higher rank than the public despite how known they are, as the Zhanbun religion involves rejecting any given power or status. In most other cases (like a teacher and a student), it's common that both sides of a conversation refer to each other with the same level of formality.

ATTITUDINALS - VY IN PREDICATES

Attitudinals, like in Lojban, express the emotional state. However, they work quite differently - this is, by merging separate predicates as part of the same "entity". If you use, for example, the predicates **cheqkuoiq** and **leo**, and mix them with **vy**, you get something that pretty much means "happily fall in love".

cheqkuoiq vy keuam goau sa Noely shy sa Hakarysha shy dy

Noelle, very angrily, falls in love with Akarsha

Do note that this is only one use of **vy** in predicates - **vy** isn't just an attitudinal marker, it can do a lot of more abstract things. It can mark an action as useful, or merge different actions as part of one single thing.

In the end, **vy** predicates boil down to context.

PREDICATIZERS

A predicatizer in Laiqbun is a word that is used to create predicates related to other predicates. Sure, there's a word for "x1 falls in love with x2", but how do you say "x1 is the action of falling in love"? Or how do you say "x1 is the thing that falls in love"?

For the former, a few words have **-sui** added on to them, which forms a related, important meaning. But sometimes it's not like that, and sometimes the **-sui** word doesn't refer to the action. For that, there's the **ryq** predicatizer.

ryq cheqkuoiq sha

I am the action of falling in love

ben ryq chaqkuoiq

Falling in love is good.

You close the ryq predicatizer with the particle gyq. This is necessary for the following reason: If you put a serial predicate inside of this predicatizer, the whole serial becomes the new predicate. If you, instead, close it at some point then do a serial, the serial part is not inside the predicate. Look at the following example:

ryq laiu tteoq

The action of (importance being small)

ryq laiu gyq tteoq

(The action of importance) being small

It is a subtle difference, but it can be important.

For the latter, there's the predicatizer **leq** (this is both a predicate and a predicatizer!). This one's a bit complex, but it refers to "the thing that kke is filling in the relative clause that describes this word". It's best explained with an example.

leq hoo sha kka gy

x1 is the thing that I eat

Here, **hoo sha kka gy** is just the relative clause that means "that I eat", and leq makes it so that the predicate refers to that thing. **«leq hoo kka maq gy»** would mean "x1 is that which eats food". **kke** can also be omitted to refer to the entire action.

leq hoo sha maq gy

x1 is the action of me eating food

MATHEMATICS

Although it's not *really* necessary since this is a conlang, I really prefer giving this language a way to express mathematics in it, mainly because I know there's people that love math and conlangs and like the idea of putting them together. Expressing math in Laiqbun is pretty simple. There are particles that express some mathematical operations.

Particle	Operator
zyem	Sum
ryeq	Substraction
vyuq	Multiplication
lyau	Division
ppyua	Exponentiation
cya	Root

Equations are considered one single argument, and to equate them to something, you use the **heiaq** (x1 is equal to x2) predicate.

heiaq haiqku zyaq haiqku haiqkkudy
2 + 2 equals 4

Laiqbun also has spoken parentheses that work exactly like the quotative particles. **ppykun** opens them and **ppylun** closes them.

Finally, it's considered valid to append the numbers to the operation particles.

heiaq haiqku zyaq ku haiqkkudy
2 + 2 equals 4

PARTS OF SPEECH FROM A LAIQBUN EYE

No, I'm not going to make you call parts of speech their Laiqbun word like Lojban does.

So, up to this point you've already seen Laiqbun explained under an English perspective - this document is made so that an English speaker can learn it. However, you may notice that there's a strong connection between predicates, modifiers, nouns and prepositions, and that particles are very separate, and what is kke, anyways?

Earlier, it was explained that predicates are how all words except for particles are defined, and that nouns and modifiers are derived from these - in Laiqbun, these all belong to a single group, they all behave the same way and can perform the same functions. This is a **content word**, the ones that carry the semantic meaning. Examples of content words are **she**, **cheai** or **ben**.

The other group is formed by words that don't mean anything semantically, and they only have grammatical functions, these words are **particles**. Examples of particles are **sy**, **ny**, **hya** and **fy**.

However, there's a slight overlap between them - especifically the word **kke**. This word means nothing in itself and its function exists only due to a grammatical need, but when it's functioning, it does carry semantic meaning based on where and how it's being applied - what is this word?

This word is what's called a **content particle**, the result of overlapping content words and particles - its function is entirely grammatical, and its semantics are derived from this function.

Other content particles would be **se** and **xhe**.

Under this understanding, nouns, modifiers and predicates still exist on their own, they're just applied content words (and content particles).

A bit on the sentence structure side, just to give things official names, prepositional phrases are composed of two parts, the **prepositional predicate** and the **prepositional argument**.

PLURALITY

Laiqbun also seeks to include plurality into its grammar, and it does so by having pronouns for systems separately from pronouns for individual people. These pronouns are formed by adding **-vu** to them. If the pronoun is intended to be plural too, **-ru** is added. If **-vu** is added before **-ru**, it indicates the presence of more than one system, if the opposite is the case, it indicates more than one speaker from the same system.

shevuru

x1 is our systems

sheruvu

x1 is us (different alters from the same system)

For **peu** and **cheaq** compounds, -**ruvu** can be shortened to **-ruq**, and **-vuru** can be shortened to **-vuq**.

OTHER MISCELLANEOUS AMBIGUITY

Tteo and Zeq: Consider the following scenario: Three people get lost in the forest. To survive, they all start collecting bugs to eat. In the end, the amount of bugs they eat accumulates to 144. You could say that **the three people ate 144 bugs**, which in Laiqbun is **heo ppaamtu zhen ppaampuyq ken dy**. This sentence, however, has the problem of being read in two possible ways, the intended way, in which the three people individually eat some unspecified amount of bugs that, together, amounts to 144, and another way in which they all eat the 144 bugs, maybe by splitting them. **tteo** and **zeq** solve this. All you have to do is put either at the beginning of the sentence, and this will make matters more specific.

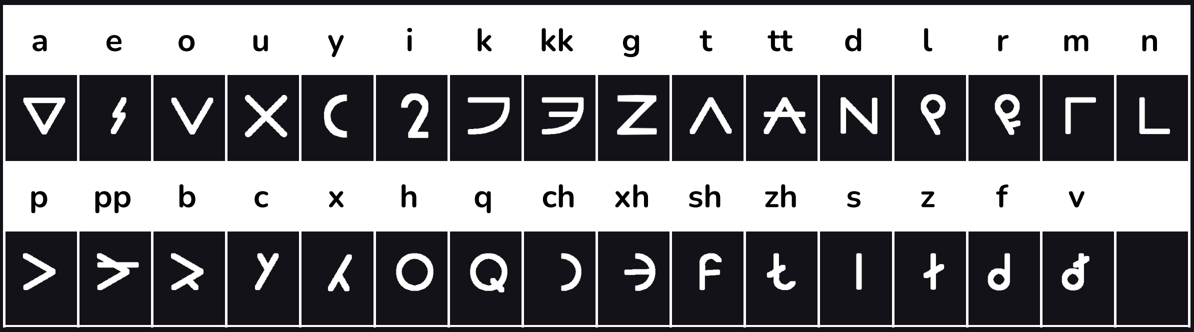
zeq heo ppaamtu zhen ppaampuyq ken dy

Each of the three people, to each of the 144 bugs, apply relationship «the three people consume 144 bugs» (such that the amount eaten by each individual person amounts 144)

RAIAQBUN

Laiqbun's Writing System

For artistic purposes, like writing the Zhanbun culture or making art related to the language, I've made a writing system specifically for Laiqbun. It's inspired by Hangul and I'm certain I was looking at a Cyrillic chart when I was making its symbols.



a	e	o	u	y	i	k	kk	g	t	tt	d	l	r	m	n
p	pp	b	c	x	h	q	ch	xh	sh	zh	s	z	f	v	

If you write only a consonant, **y** is the assumed vowel; this allows you to write most particles as one single character. Final consonants are flattened and written below the syllable block. A vowel is written to the right of the consonant. When there are two vowels, the first one is written under the consonant and the second is written to its right; when there are three, the first vowel is written below the consonant and the last two are written one above the other to the right of it.

The vowels' design is made based on the function they represent when applied to predicates. For **a** I was thinking of a triangle with the three states of matter. **e** is a lightning bolt, representing the energy of an action, **o** is the same as **a** but slightly different. **u** is like when you sew a button to a shirt or to a plushie and you form that cross with thread. **y** has no meaning, like the particles, and **i** is the number 2, because what's more of a modifier than a quantity?

HISTORY OF LAIQBUN

Yeah, because why not, I guess.

Laiqbun was first published on the internet on August 4th 2020, in the Toaq Discord server. It was supposed to be a Toaq ripoff where the only difference was that instead of using tones to mark part of speech, that would be done with vowels. Laiqbun is one of the few languages I ever did a draft of, but that quickly became exhausting, leading me to creating this documentation ten days after. The original Laiqbun draft was VSO and had 8 vowels, although the grammar was quite similar. This was quickly changed to SOV and 7 vowels.

The concept of the Zhanbun had started forming, inspired from Rain World and my own personal beliefs. I kept presenting the language in the *ma pona pi toki pona* server, and eventually a few people became interested in it. Its vibe and the way it works was interesting enough to keep at least one person engaged.

At this point, the grammar was quite unclear, so on August 14th, I decided to start documenting Laiqbun in a more organized way. I had an aesthetic in mind, and asked what kind of colors would fit it on the *ma pona* server. I got my answers, decorated the document and started documenting the language. This is when the language was reduced to six vowels.

I grew convinced to make a server specifically for Laiqbun, so I started making the flag. Its first WIP was presented one day after I created this documentation, and that same day the flag was finished. By this point I had the sketch of the server, and the invite was first given to anyone else one day after that, but I only started giving it out to people on August 21st.

The community slowly grew up from people that I usually interact to - it's not yet a public community, although at the time of writing (August 31st) I'm starting to sketch out a plan to make it into one. The people slowly decided that a spelling reform was needed, and once someone came up with one that I liked and that fulfilled the goals I have in mind for a Laiqbun romanization, this was done.