

# OOK Protocols description

## Introduction

This document summarises some findings about decoding ASK / OOK protocols, more specifically for Home Automation usage.

Most of the home automation protocols are derived from the old X10 protocol (see [https://en.wikipedia.org/wiki/X10\\_\(industry\\_standard\)](https://en.wikipedia.org/wiki/X10_(industry_standard))).

For light switch control a common one is known as KAKU (Klick Aan / Klick Uit or Click On Click Off) which is widely deployed in The Netherlands (see <http://www.klikaanklikuit.nl/home/>). Plenty of other manufacturers use the same technique (Home Easy, Chacon, D.I.O, Cogex, etc...).

I have currently decoded 3 variants of these protocols (hereafter referred as Kaku).

- Kaku Old
- Kaku Cogex (see <http://www.superstar.com.tw>)
- Kaku New

Other weather stations applications (typically referred as Oregon Scientific protocol) are using a similar method but are not covered by this document

## Kaku Old

This protocol uses an OOK modulated datagram of 12 symbols.

The datagram is summarised as follow:

Bit order	B A 9 8 7 6 5 4 3 2 1 0
Weight	8 4 2 1 8 4 2 1 8 4 2 1
Symbols	L 1 1 0 U U U U A A A Start

S = SYNC/Start bit	
L = Level bit	0 OFF, 1 ON 1
U = Unit bits	(0)1...(15)16
A = Home Address bit	(0)1...(15)16 or (A...P)

Note that the lowest bit is send first.

The characteristics are the following:

- Use a limited number of hand held controller addresses (aka House address); 16, from A to P
- Use a limited number of controlled units per house address; 16 from 1 to 16
- Do not allow dimming
- Unidirectional (controller to unit)
- No encryption
- No anti-replay

## Kaku Old Home Address

The home address and units are identified by rotary switches.

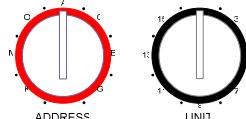


Figure 1: Kaku Old Home Address and Unit switches

## Kaku Old bit pattern

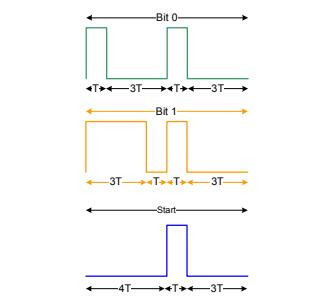


Figure 2: Kaku old bit pattern

# OOK Protocols description

The datagram looks like (see OOK Poor man's monitoring tool vx.y.pdf for tracing):

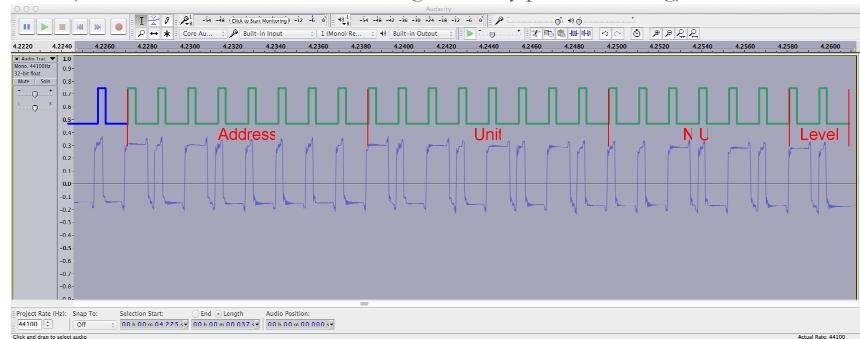


Figure 3: Kaku Old - Datagram

## Kaku Old bit timing

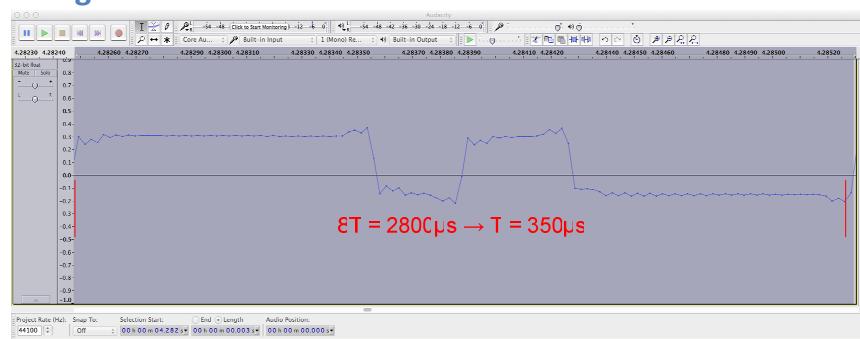


Figure 4: Kaku Old - Bit timing

## Kaku Inter datagram timing

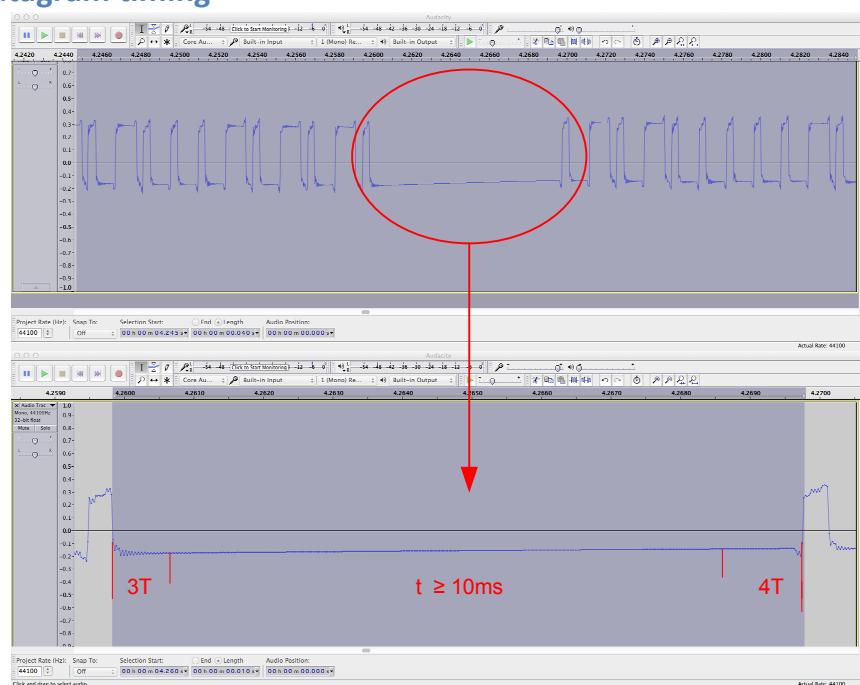


Figure 5: Kaku old - Inter datagram timing

## Kaku Old Timing summary

T  
Inter datagram  
300...450 nominal 375µs  
 $t \geq 10\text{ms}$

# OOK Protocols description

## Datagram timing

$$8T + 12 * 8T = 104T \text{ or } 39.000\mu\text{s}$$

## Kaku Old example

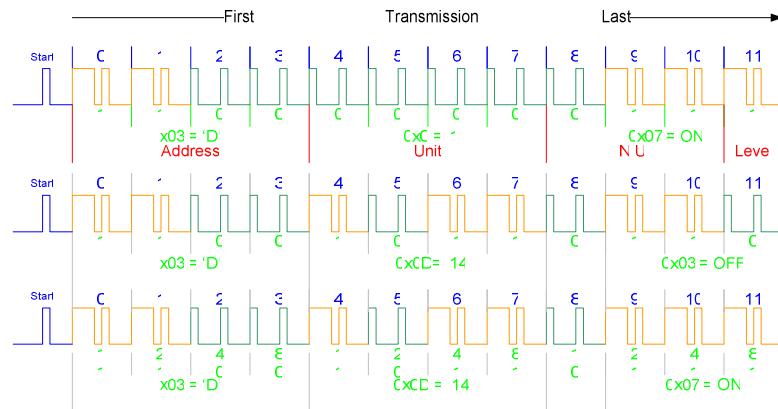


Figure 6: Kaku old timing and data pattern example

## COGEX

This protocol uses an OOK modulated datagram of 12 symbols.

The datagram is summarised as follow:

Bit order	B A 9 8 7 6 5 4 3 2 1 0
Weight	8 4 2 1 8 4 2 1 8 4 2 1
Symbols	L 1 1 U U U U A A A A L Start
S	= SYNC/start bit
L	= Level 0 OFF, 1 ON
U	= Unit bits (0)1... (15)16
A	= Home Address bits (0)1... (15)16

Notes:

Lowest bits are sent first

Timing is quite similar to the old KAKU however the datagram structure is quite different

The characteristics are the following:

- Use a limited number of hand held controller addresses (aka House address); 16, from A to P
- Use a limited number of controlled units per house address; 16 from 1 to 16
- Do not allow dimming
- Unidirectional (controller to unit)
- No encryption
- No anti-replay

# OOK Protocols description

## Cogex Home Address

The home address and units are identified by hard coded jumpers.

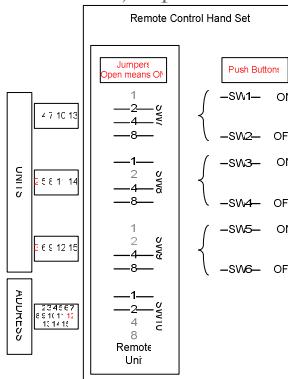


Figure 7: Cogex Home Address and Unit jumpers

## Cogex bit pattern

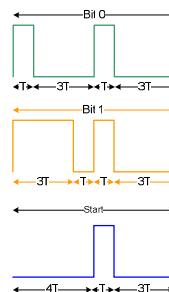


Figure 8: Cogex bit pattern

The datagram looks like (see OOK Poor man's monitoring tool vx.y.pdf for tracing):

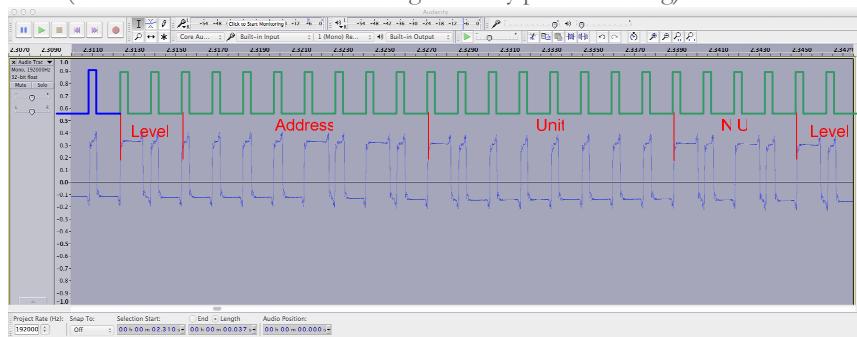


Figure 9: Cogex - Datagram

## Cogex bit timing

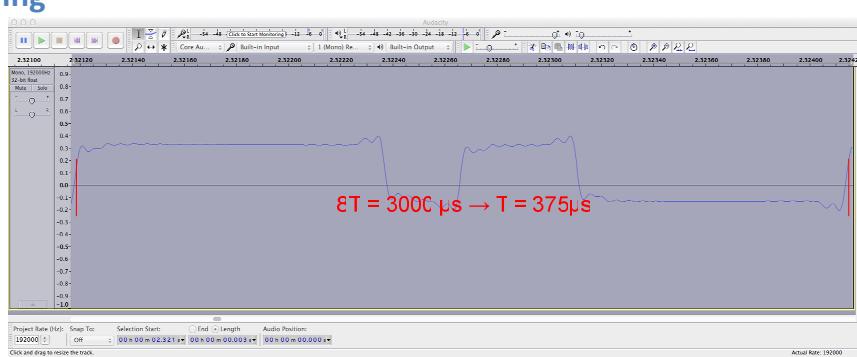


Figure 10: Cogex - Bit timing

# OOK Protocols description

## Cogex Inter datagram timing

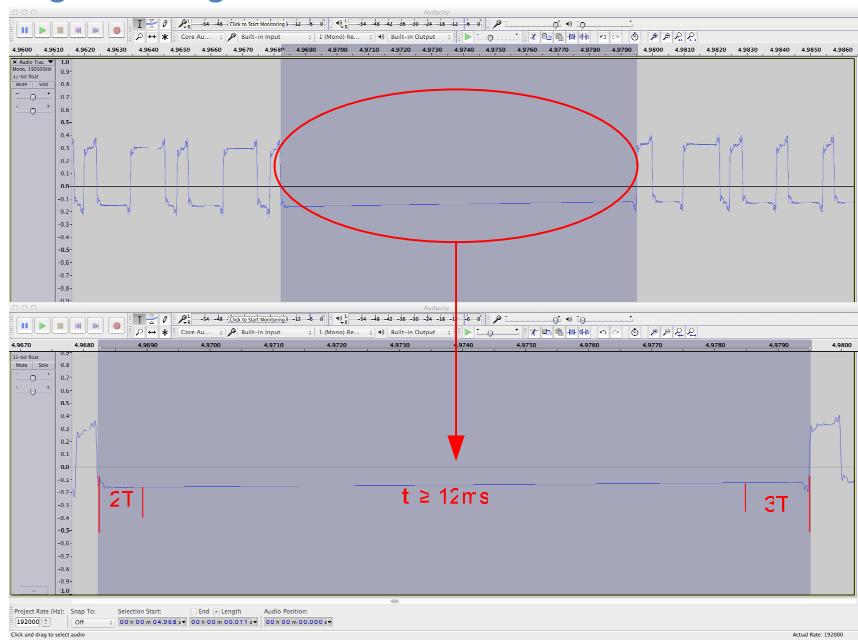


Figure 11: Cogex - Inter datagram timing

## Cogex Timing summary

$T$                               200...500 nominal 350 $\mu\text{s}$   
Inter datagram                >12ms

## Datagram timing

$$8T+12*8T = 104T \text{ or } 36.400\mu\text{s}$$

## Cogex example

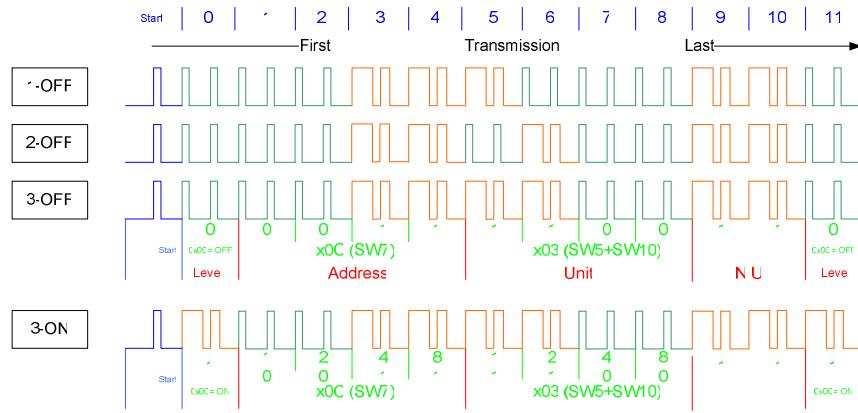


Figure 12: Cogex timing and data pattern example

# OOK Protocols description

## Kaku New

This protocol uses an OOK modulated datagram of 32/36 symbols.

The datagram is summarised as follow:

```
ON/OFF
=====
Bit order      31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
Weight         1  2  4  8   1  2  4  8  1  2  4  8  1  2  4  8  1  2  4  8  1  2  4  8  1  2  4  8  1  2  4  8  1  2
Symbols        Stop U  U  U  U  L  G  A  A  A  A  A  A  A  A  A  A  A  A  A  A  A  A  A  A  A  A  A  A  A  Start
           !    0    !    !    !    0 (LSB) !    1    !    2    !    3    !    4    !    5    !6(MSB) !
DIM
===
Bit order      35 34 33 32 31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
Weight         1  2  4  8  1  2  4  8   1  2  4  8  1  2  4  8  1  2  4  8  1  2  4  8  1  2  4  8  1  2  4  8  1  2
Symbols        Stop 1  1  1  1  U  U  U  U  d  G  A  A  A  A  A  A  A  A  A  A  A  A  A  A  A  A  A  A  A  Start
           !    0    !    0    !    !    !    0 (LSB) !    1    !    2    !    3    !    4    !    5    !6(MSB) !

S = SYNC/Start/Stop bit
L/d = Level (or remote switch state) bit: ON = 1, OFF = 0 ; 'L' symbol is replaced by 'd' symbol if a Dim level is specified
U = Unit           bits: 0...15; Practical handset values are 1...16
A = Address (or handset) bits: 0...2^26=67.108.863; value of handset addresses are defined by Manufacturer
G = Group bit     bit: Group = 1, no Group = 0
l = Dim Level for Dim option bits: 1...15; 0 means no Dim
```

The characteristics are the following:

- Large number of hand held controller addresses (aka House address); 16, from A to P
- Use a limited number of controlled units per house address; 16 from 1 to 16 (0 to 15)
- Units supports multiple controllers (up to 4)
- Allow dimming
- Unidirectional (controller to unit)
- No encryption
- No anti-replay

Note that some units support both old and new Kaku protocols

## Kaku New Home Address

Each controller should have a unique House address which is determined by the vendor.

Locally administrated addresses are possible but care should be taken to avoid conflict with other houses!

Cloning of existing address in the same house (no anti-replay is implemented). This is convenient to simulate handset command via wireless applications.

House code addresses are 26 bits long for a decimal value comprise 1 to 67.108.863.

The easiest way to decode an exiting address for cloning is by using a RX433 receiver and the *NewRemoteSwitch ShowReceivedCode* sketch (see SAW Devices and OOK vx.y.pdf and

<https://bitbucket.org/fuzzillogic/433mhzforarduino/wiki/Home>).

A second way is to decode the trace using the “OOK Poor mans’ monitor tool”, see figure below.

A third way is to create your own code and force the units to learn the address via transmission (see *NewRemoteSwitch LearnCode* sketch).

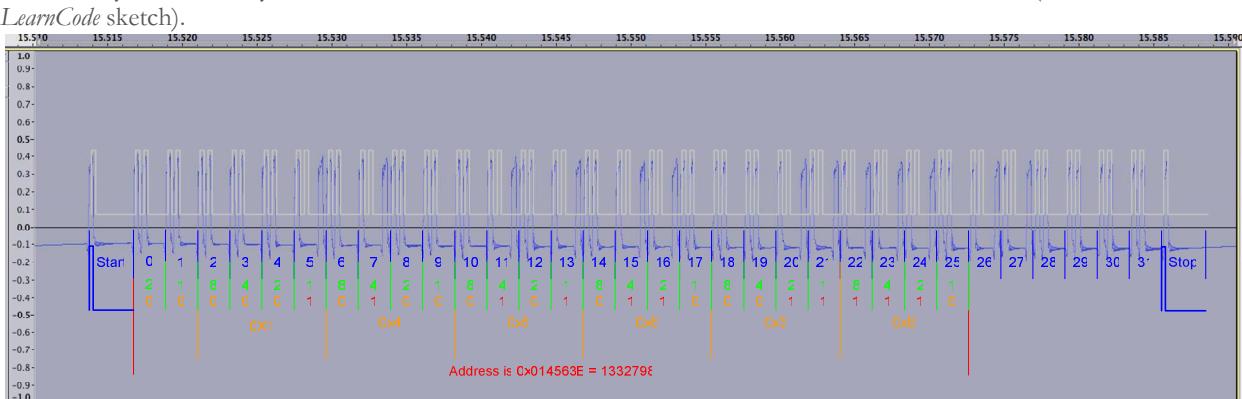


Figure 13: Kaku new - Home Address

# OOK Protocols description

## Kaku New bit pattern

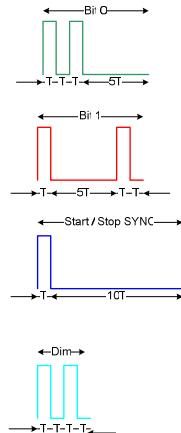


Figure 14: Kaku New bit pattern

The datagram looks like (see OOK Poor man's monitoring tool vx.y.pdf for tracing):

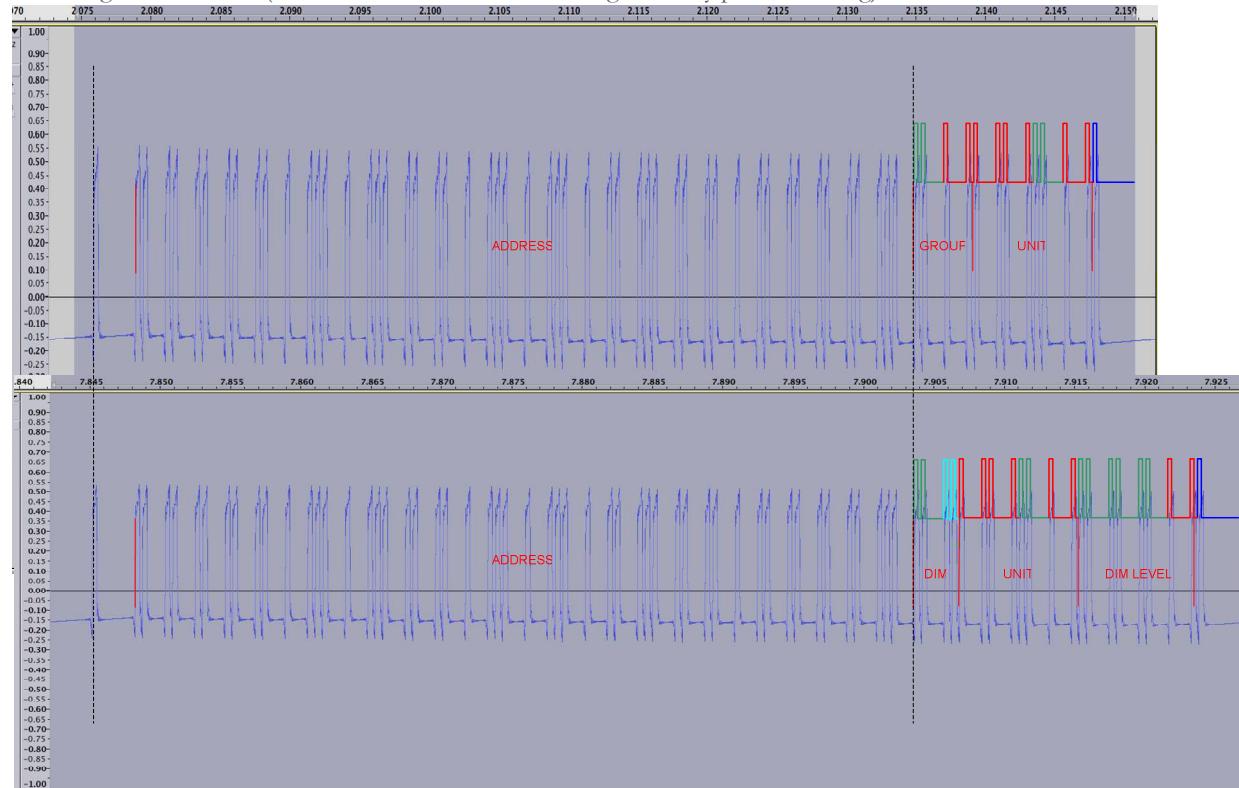


Figure 15: Kaku New – Datagram (ON/OFF) and DIMM

# OOK Protocols description

## Kaku New bit timing

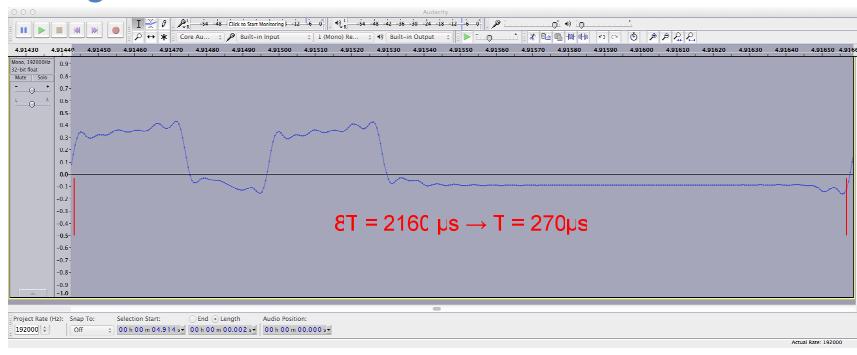


Figure 16: Kaku New - Bit timing

## Kaku New Inter datagram timing

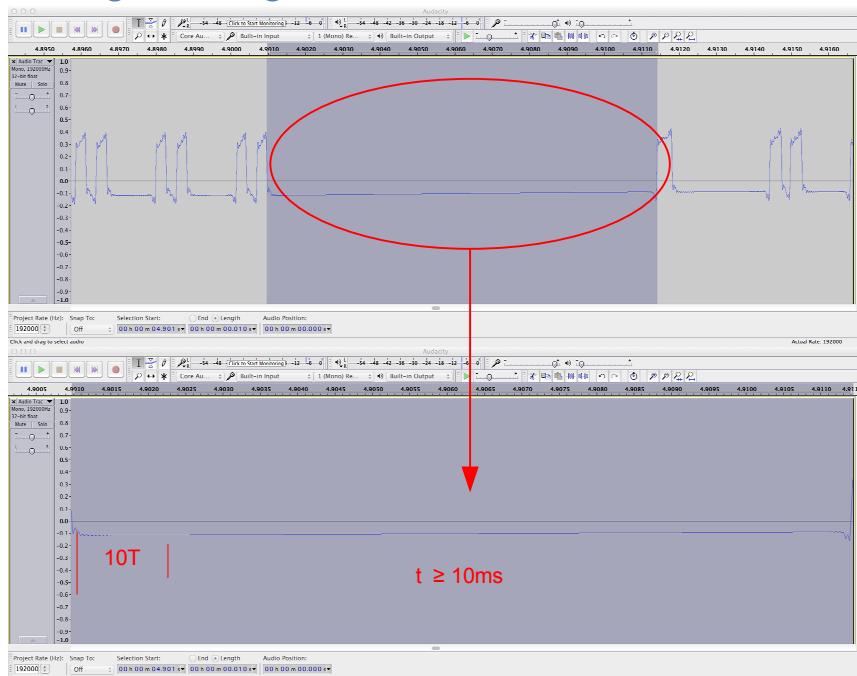


Figure 17: Kaku New - Inter datagram timing

## Kaku New Timing summary

T	200...300 nominal 250μs
Inter datagram	$>10ms$

### Datagram timing (ON/OFF)

$$11T + 32 * 8T = 267T \text{ or } 66.750\mu s \sim 67ms$$

### Datagram timing (Dimmer)

$$11T + 36 * 8T = 299T \text{ or } 74.750\mu s \sim 75ms$$

# OOK Protocols description

## Kaku New example

